Overlapping Community Detection and Filtration of Low-Confidence Protein Interactions towards Protein Complex Identification from PPI Networks

Thesis submitted in partial fulfilment of the requirements
for the award of the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Madhusudan Paul

Under the supervision of

Dr. Ashish Anand

Department of Computer Science and Engineering

Indian Institute of Technology Guwahati

Guwahati - 781039 Assam India

JULY, 2020

Dedicated to

Our Nation

For her rich tradition

Acknowledgements

Firstly, I would like to express my sincere gratitude to my supervisor Dr. Ashish Anand. He has taught me how to critically think, analyze, and present an idea, both consciously and unconsciously. I appreciate all his contributions of time and ideas towards my thesis work. The joy and enthusiasm he has for his research were motivational for me, even during tough times in the Ph.D. pursuit.

Besides my supervisor, I would like to thank the rest of my doctoral committee: Dr. V. Vijaya Saradhi, Dr. S. Ranbir Singh, Dr. B. Anand, for their insightful comments and suggestions. I have learned a lot from them. Particularly, I am very grateful to Dr. V. Vijaya Saradhi. He suggested and provided the initial idea regarding the third contribution of my thesis. I also wish to thank Prof. P. K. Das and Dr. Kusum K. Singh for their valuable suggestions.

I should acknowledge Sri Rishav Anand for the initial work he made in the first contribution of the thesis. I need to thank Sri Saptarshi Pyne for executing some scripts in the server regarding the third contribution of the thesis. Further, I want to thank all the faculty members and staffs of the CSE department, IITG, for their assistance. A special gratitude goes to my friends at IITG for many meaningful and meaningless discussions.

I am grateful to my father, Sri Bikash Chandra Paul, and my mother, Smt. Ratna Paul, for their constant support. At the end, I wish to express my appreciation to my wife, Smt. Chandrima Paul, for her love, support, and understanding. I also wish to say thanks to our little son, Samarpan, and I am sorry that I might not have given him enough time when he needed the most due to my thesis work.

July 2020 Madhusudan Paul

Declaration

I certify that

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor.
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis can be considered plagiarism to the best of my knowledge and understanding and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor is not in a position to check for any possible instance of plagiarism within this submitted work.

July 2020 Madhusudan Paul

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

Dr. Ashish Anand
Associate Professor
Email : anand.ashish@iitg.ac.in

Certificate

This is to certify that this thesis entitled "**Overlapping Community Detection and Filtration of Low-Confidence Protein Interactions towards Protein Complex Identification from PPI Networks**" submitted by **Madhusudan Paul**, in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy, to the Indian Institute of Technology Guwahati, Assam, India, is a record of the bonafide research work carried out by him under my guidance and supervision at the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India. To the best of my knowledge, no part of the work reported in this thesis has been presented for the award of any degree at any other institution.

Date: July 2020
Place: Guwahati

Dr. Ashish Anand
(Thesis Supervisor)

Abstract

Proteins perform almost all cellular functions in living systems. Majority of the proteins do not perform their functions in isolation, rather a group of proteins, commonly known as a *protein complex*, physically interact together to accomplish a biological function. Therefore identifying protein complexes is essential to realize the principles of cellular organization. However, experimental detection of protein complexes is not adequate due to the limitation of current high-throughput experimental techniques. From the past few decades, a large volume of *protein-protein interaction* (PPI) data has become available due to high-throughput technologies. Thus computational detection of protein complexes from PPI data is considered as useful complements to the experimental techniques. Generally, PPI data is modeled as an undirected graph, where proteins are represented as nodes and interactions between proteins as edges of the graph. Such graphs are commonly referred to as *PPI networks*. This work explores some of the challenges in detecting protein complexes from PPI networks.

The first challenge this work deals with is the *detection of overlapping communities in PPI networks*. The problem of detecting *protein complexes* from PPI networks are usually mapped to the problem of detecting *communities* from complex networks, particularly *overlapping communities* as the majority of the proteins perform multiple functions. The task of detecting communities in complex networks is computationally challenging as the majority of the problems related to determining the structural properties of graphs are often NP-hard. In our first work, we propose *Extended Greedy Clique Expansion (EGCE)* as an overlapping community detection algorithm (CDA). EGCE improves upon an existing overlapping CDA, namely the *Greedy Clique Expansion (GCE)*. GCE and other methods generally fail to assign a node into multiple communities if the node belongs to more than three communities. The empirical results on state-of-the-art synthetic benchmark datasets show that the EGCE outperforms other methods when overlapping nodes belong to more than three communities. We also experiment on real datasets and observe good performance. The proposed extension is generic and may be used to extend any existing algorithms for detecting overlapping communities.

The second challenge this work deals with is the *presence of false interactions* in PPI data due to experimental limitations. These false positives have a negative effect on any downstream analysis of PPI data, including the identification of protein complexes. Therefore in our second work, we explore the removal of false positives in PPI networks. *Gene ontology (GO)*, a taxonomy of biological terms to represent the properties of gene products and their

relations, can be utilized as a complementary resource for scoring confidence of PPIs in a manner that false positive interactions get a low-confidence score. The ontology-based *semantic similarity measure* is a quantitative function that measures the closeness between two terms based upon their meaning over a set of terms organized as ontology. Hence GO-based similarity measures can be used to score confidence of PPIs, and low-confidence PPIs are highly likely to be false positives. We introduce a new family of similarity measures: *Relative Depth Specificity (RDS)*, *Relative Node-based Specificity (RNS)*, and *Relative Edge-based Specificity (RES)*, by redefining GO term *specificity*. We show that all the three measures, particularly RNS and RES, are quite effective to distinguish true PPIs from false positives than the existing alternatives.

Although GO has been effectively utilized in many genomics applications, it is being evolved regularly with the addition, deletion, and merging of terms. Hence this should affect any method of utilizing GO. However, to the best of our knowledge, there is no such study that evaluates the impact of the continuous evolution of GO on the similarity measures. In our third contribution, we systematically study the effect of GO evolution on similarity measures for the task of scoring confidence of PPIs. For that, we consider nine state-of-the-art similarity measures and nine different Bioconductor versions of GO. Our analysis indicates that the overall impact is minor. However, considering only the most affected proteins during the updates, different measures get affected differently. We observe that RES, one of the proposed similarity measures, exhibits the promising robustness over the GO evolution.

Finally, in our fourth contribution, we study the impact of the filtration of low-confidence PPIs on computational identification of protein complexes. The confidence of PPIs is obtained using different GO-based similarity measures, including the proposed similarity measures, namely, RDS, RNS, and RES. We consider five state-of-the-art representative CDAs. The empirical results indicate that low-confidence interactions have a high negative impact on the performance of CDAs. Each CDA significantly improves its performance after elimination of low-confidence PPIs. We also observe that the rate of improvement and rate of elimination of low-confidence PPIs are highly proportional.

Contents

List of Figures

List of Tables

List of Symbols

Symbol	Description
G	Graph
V	Set of vertices/nodes
E	Set of edges
C	Community
$\mathcal{C}$	Set of communities
k	Clique size, average degree of a node in LFR benchmark
F_C	Fitness function of community C
d_{in}^C	Twice the number of intra-community edges of C
d_{out}^C	Number of inter-community edges of C
ϵ	Community distance
$IP_{u,C}$	Interaction Probability of a node $u \notin C$ with respect to C
$N(C)$	Immediate neighboring nodes of a community C
N	Number of nodes, number of terms in an ontology
k_{max}	Maximum degree of a node
C_{min}	Minimum size of community C
C_{max}	Maximum size of community C
O_n	Number of overlapping nodes
O_m	Number of memberships of the overlapping nodes
τ_1	Exponent of power-law distribution of degrees
τ_2	Exponent of power-law distribution of community sizes
μ	Mixing parameter
$p(t)$	Probability or frequency of occurrence of term t in an ontology
$IC(t)$	Information content of a term t
M	Total number of annotations in a particular gene ontology
$Des(t)$	Set of descendants of term t
$Ant(T)$	Set of annotations to the set of terms T
$H(X)$	Entropy of the random variable X

List of Abbreviations

Abbreviation	Expansion
PPI	Protein-Protein Interaction
Y2H	Yeast Two-Hybrid
TAP	Tandem Affinity Purification
CDA	Community/Complex Detection Algorithm
GCE	Greedy Clique Expansion
EGCE	Extended Greedy Clique Expansion
IP	Interaction Probability
NMI	Normalized Mutual Information
GO	Gene Ontology
BP	Biological Process
CC	Cellular Component
MF	Molecular Function
DAG	Directed Acyclic Graph
SSM	Semantic Similarity Measure
BMA	Best-Match Average
RDS	Relative Depth Specificity
RNS	Relative Node-based Specificity
RES	Relative Edge-based Specificity
IC	Information Content
ROC	Receiver Operating Characteristic
AUC	Area Under the ROC Curve
TPR	True Positive Rate
FPR	False Positive Rate
DP	Discriminating Power
Pfam	Protein Family

Chapter 1

Introduction

Nature organizes the components of living systems into a modular and hierarchical manner. The functionality and organizational principles of higher levels are intelligible. However, the complexity rises exponentially towards lower levels of the hierarchy. There have been continuous efforts to realize living systems at the molecular level of detail. The elementary structural and functional units of a living system are biological cells, which are the smallest units of life. The four essential families of small organic molecules - sugars, fatty acids, amino acids, and nucleotides and corresponding macromolecules (polysaccharides/carbohydrates, lipids, proteins, and nucleic acids, respectively, formed by linking into long chains) constitute the majority of the cell mass. A collection of cells (of similar type and function) makes a tissue, many tissues constitute an organ, and finally, several organs make an organism. At each level, the components interact among themselves to sustain the dynamics of the system. Therefore to realize the essence of life, system-level study (systems biology) has become essential. Network (graph) is a frequently used mathematical model to represent a complex system, where the nodes (vertices) are the components of the system, and the edges are the interactions between the components. By analyzing the properties of the network, we can generate hypotheses that may describe the possible behaviors of the system. This study focuses on proteins and interactions among them.

Proteins perform the majority of the cellular functions. Multiple proteins group together through physical interactions (bind) to accomplish a biological function. Such a functional group of proteins is commonly known as a *protein complex*. Further, a protein may participate in multiple biological functions and hence may belong to multiple complexes. A small change of a single protein may cause a substantial impact on the overall cellular functionality due to cascading effects. In fact, several diseases (e.g., Huntington's disease [1]) are the consequence of a little modification of a single protein that affects the functionality of its interacting partners. Thus the protein complex identification is one of the important

milestones in our endeavor in understanding the principles of cellular organizations.

Direct detection of protein complexes from any experimental technique is highly limited. Tandem Affinity Purification with Mass Spectrometry (TAP-MS) [2] is the widely used experimental technique for detecting complexes. However, it has the following limitations - the majority of transient low-affinity complexes are not detected due to its multiple washing and purification steps; complexes that are not present under the given in vitro conditions (artificial conditions outside the cell) have less chance to get detected [3]. Therefore computational approaches for detecting complexes are considered as useful complements to the experimental techniques.

From the last few decades, a large volume of *protein-protein interaction (PPI)* data has become available due to high-throughput technologies (e.g., yeast two-hybrid (Y2H) system [4]). A PPI may be temporal and spatial since the physical interaction between two proteins depends on certain conditions of the particular cell. Most computational methods model PPI data as an undirected graph, where proteins are represented as nodes and interactions between proteins as edges. Such graphs are commonly known as *PPI networks*. However, depending on the approach used to generate the network, an edge may not always indicate a direct physical interaction, but also represent correlated expression in the cell, similar genomics context, similar function and so on. PPI networks play an important role in the system-level understanding of cellular processes and are frequently used to detect complexes.

Besides protein complexes, two other closely related constructs - *functional modules* and *signaling pathways* are found in the literature. Although it is difficult to differentiate among them, the following distinction given by Spirin and Mirny [5] is widely accepted. The PPIs in a complex occur simultaneously at the same cellular location, while the PPIs in a functional module may not necessarily take place at the same time and location. However, most of the computational approaches do not distinguish between them, since (generally) PPI data do not have temporal and spatial information. Additional domain information such as gene expression data, and gene ontology (GO), need to be incorporated for distinguishing complexes and functional modules [6]. Finally, in a signaling pathway, the PPIs occur in an ordered succession to accomplish a signal transduction process. Certainly, the PPIs in a signaling pathway do not need to take place at the same location and time. Hence many authors consider signaling pathways as a specific kind of functional modules.

The rest of the chapter is organized in the following manner. In section 1.1, we briefly discuss the contributions made in the thesis. Section 1.2 presents a brief survey of literature on the existing computational techniques to detect protein complexes. Finally, the outline of the thesis is given in section 1.3.

1.1 Contributions made in the thesis

Contribution 1: This work focuses on some of the challenges faced by computational methods in detecting protein complexes from PPI networks. The first challenge this work deals with is the *detection of overlapping communities in PPI networks*. The problem of detecting protein complexes from PPI networks can be mapped to the problem of detecting communities from complex networks. We focus on overlapping communities, as many proteins are part of multiple complexes. The majority of the problems related to determining the structural properties of graphs are often NP-hard in nature [7]. Therefore from the computational viewpoint, detecting communities in complex networks is a computationally challenging task. Although there are algorithms for detecting overlapping communities, the majority of these are unable to detect highly overlapping communities properly. The performance of these algorithms falls sharply when overlapping nodes belong to more than three communities. In our first work, we propose *Extended Greedy Clique Expansion (EGCE)* as a highly overlapping community detection algorithm, which is an extension of the existing overlapping community detection algorithm, namely *Greedy Clique Expansion (GCE)* [8]. We evaluate EGCE with other state-of-the-art overlapping community detection algorithms using both synthetic benchmark (LFR benchmark [9]) and real datasets (PPI networks and protein complexes).

Contribution 2: The second challenge this work deals with is the *presence of false interactions* in PPI networks due to experimental limitations. PPI data involve a significant fraction of false positive interactions [10]. These false positives have a negative impact on any downstream analysis of PPI networks, including complex detection. Therefore, in our second work, we try to remove false positives in PPI networks. There have been continuous efforts to utilize complementary resources for scoring confidence of PPIs in a manner that false positive interactions get a low-confidence score. *Gene ontology (GO)* [11], a taxonomy of biological terms to represent the properties of gene products and their relations, has been widely used for this purpose. Ontology-based *semantic similarity* is a measure of closeness between two terms based upon their meaning over a set of terms organized as ontology. GO-based similarity measures can be used to score confidence of PPIs and PPIs with low similarity scores may be treated as false positives. We introduce a new set of similarity measures: *Relative Depth Specificity (RDS)*, *Relative Node-based Specificity (RNS)*, and *Relative Edge-based Specificity (RES)*, by redefining GO term *specificity* to distinguish true PPIs from false positives effectively.

Contribution 3: Although GO has been effectively utilized in many genomics applications including predicting protein functions [12, 13, 14], predicting and validating PPIs [15, 16, 17], analyzing pathways [18], etc., it is being updated regularly with the addition,

deletion, and merging of terms. Consequently, this should affect any method of utilizing GO. However, we could not find a study that evaluates the impact of the continuous evolution of GO on the performance of similarity measures. In our third contribution, we systematically *study the effect of GO evolution on various similarity measures* used for scoring confidence of PPIs. In the evaluation, we use nine state-of-the-art similarity measures, including our proposed ones and nine different Bioconductor versions of GO.

Contribution 4: The performance of any complex detection method gets adversely affected by the false positive interactions present in PPI networks. Although some studies indicate that filtering PPI networks improve the performance of complex detection techniques, we could not find a systematic study on the effect of PPI filtering using GO. Therefore in our fourth contribution, we *systematically study the impact of low-confidence PPIs on computational identification of protein complexes using GO-based semantic similarity measures*. We evaluate the performance of complex detection algorithms based on two reference PPI networks and two ground truth complex datasets of the yeast model organism before and after the filtration of low-confidence interactions (likely to be false positives). Confidence scores of PPIs are obtained using nine GO-based similarity measures, including our proposed ones. We consider five state-of-the-art representative complex detection algorithms, including our proposed one in the evaluation.

1.2 Related work

In this section, we briefly discuss different computational approaches for detecting protein complexes and functional modules. In literature, the existing methods are classified in different ways. Pizzuti *et al.* [19] classified the existing (graph clustering) methods into the following types of algorithmic approaches: 1) Local neighborhood Density search (LD), 2) Cost-based Local search (CL), 3) Flow Simulation (FS), 4) Statistical-based Measures (SM), and 5) Population-based Stochastic search (PS).

1.2.1 Local neighborhood Density search (LD)

The hypothesis behind the LD approach is that protein complexes form dense structures in PPI networks. Hence it tries to identify dense subgraphs in PPI networks. It uses a local density measure to maximize the densities of the subgraphs. MCODE [20], CFinder [21], and ClusterONE [22] are three widely used LD-based algorithms. Other LD-based algorithms include DPClus [23], SWEMODE [24], DECAFF [25], PINCoC [26], PCP [27], GENA [28], and SEGC [29]. Majority of these algorithms requires to set values of several parameters.

These parameters highly influence the number and resolution of detected complexes. Although many complexes have dense structures in PPI networks, there are complexes which are not. Hence these complexes will not be detected by the LD approach [30].

1.2.2 Cost-based Local search (CL)

In the CL-based approach, the PPI network is partitioned into connected subgraphs to extract modules from the network. It uses a local cost function that guides the search towards the best partition. This approach may be utilized to detect functional modules if the cost function is formulated appropriately. RNSC [31], Qcut [32], and ModuLand [33] are three noted algorithms under this approach.

1.2.3 Flow Simulation (FS)

The spread of information on an interaction network is simulated in FS-based approach. This might be an effective approach to detect functional modules. The notion of a random walk is commonly used to simulate the flow. MCL [34] and RRW [35] are two popular algorithms that use the random walk. IFB [36] and STM [37] are two other FS-based algorithms that utilize biological knowledge to spread information among proteins in PPI networks. Recently, Zhang *et al.* [38] proposed an FS-based method to detect complexes from weighted PPI networks by utilizing gene expression analysis. It utilizes MCL followed by a novel co-expression analysis method to measure complexes for further filtration. However, the FS-based methods are prone to detect dense structures as the flow tends to be accumulated in dense regions.

1.2.4 Statistical-based Measures (SM)

SM-based methods utilize some statistical measures to cluster the proteins. Samantha and Liang [39] proposed an SM-based method that keeps two proteins in the same cluster if the number of common interaction partners is greater than the expected. It uses the notion of p-value to measure the statistical significance of forming a shared partnership between a protein-pair. Farutin *et al.* [40] proposed another SL-based method based on the idea of preferential attachment among the members of a module.

1.2.5 Population-based Stochastic search (PS)

Liu and Liu [41] proposed a genetic algorithm that enumerates maximal cliques. They use it to detect protein complexes in PPI networks. Ravaee *et al.* [42] proposed an immune genetic algorithm to cluster PPI networks for detecting complexes.

1.2.6 Other methods and some issues for detecting complexes

The methods we have already discussed are so-called graph clustering methods and also known as community detection methods. However, there are some techniques that are different from traditional graph clustering methods. In this section, we briefly discuss those methods and some issues [43] that are considered to detect complexes.

Network Alignment (NA)

Conserved topological structures in PPI networks across species may have biological significance, hence are likely to be protein complexes or functional modules [44]. NA algorithms are used to align two or more networks for detecting the conserved modules across the networks. A wide range of NA algorithms has been developed for aligning networks. PathBLAST [44], NetworkBLAST [45], MaWISH [46], the match and split algorithm [47], and UEDAMAlign [48] are some noted local NA techniques. Markov random field-based method [49], IsoRank [50], GRAAL [51], and PINALOG [52] are some NA techniques that align networks globally.

However, NA methods are highly sensitive to the topological structure of networks. As a consequence, graph clustering techniques outperform the NA methods in detecting protein complexes and functional modules, since PPI networks have a significant amount of false positives.

Supervised Graph Clustering (SGC)

The majority of the graph clustering techniques are unsupervised. Qi *et al.* [30] first proposed a supervised graph clustering (SGC) method by considering several topological and biological properties as a feature vector. To train the model, a set of known protein complexes and a set of random subgraphs are used as positive and negative instances, respectively. Recently, Yu *et al.* [53] proposed another SGC method by introducing some new features from the weighted networks. However, SNC methods are highly sensitive to the selected set of features.

Complex detection using protein core attachments

Gavin *et al.* [54] observed that the majority of the complexes have two components - a core and an attachment. The center of a complex is formed by a unique set of core proteins and has relatively more interactions (among the core proteins). Each of the attachment proteins binds with a subgroup of core proteins to form the complex. CORE [55] and COACH [56] are two noted algorithms that have been developed according to the notion of core attachment. Recently, Shen *et al.* proposed an algorithm, DCA (Dynamic Core-Attachment) [57], to incorporate dynamic characteristic of the cell system by utilizing time series information of gene expression profiles.

Complex detection by considering exclusive or cooperative interactions

The majority of the complex detection methods treats all the interactions in a PPI network to be simultaneously occurred, i.e., they occur at the same time. However, two adjacent interactions (interactions with a shared partner) may be mutually exclusive due to their overlapping binding interfaces on the shared partner. Jung *et al.* [58] developed a method, SPIC (Simultaneous Protein Interaction Clusters), that extracts clusters of interactions occurred simultaneously in the PPI networks by addressing the issue. They used traditional graph clustering techniques (MCODE [20] and LCMA [59]) to find initial clusters. Then they filtered out mutually exclusive interactions to get SPICs.

Jin *et al.* [6] introduced another method by considering time series information of gene expression profiles. They determine two adjacent interactions as cooperative (occur simultaneously) if their shared partner has overlapping time-range for both the interactions.

Complex detection by incorporating gene expression data

Interacting proteins (those interact among themselves) are expected to show similar gene expression profiles. Hence proteins involve in a complex are expected to be co-expressed, since they are likely to interact among themselves. Few methods such as GFA [60], DMSP [61], and DyCluster [62] have been developed by considering the above hypothesis.

Complex detection by incorporating functional information

Generally, a protein complex is formed to accomplish a biological function. Consequently, proteins involved in a complex are likely to perform a similar function. The previously mentioned RNSC [31] and SWEMODE [24] integrate protein functional information with

topological properties (PPI networks) by utilizing gene ontology (GO). Cho *et al.* [63] introduced another method that extends STM [37] (a flow simulation method) for detecting functional modules by incorporating functional information using GO.

Complex detection from TAP data with or without constructing the PPI network

The majority of the complex detection methods uses pairwise binary PPI data detected by high-throughput experimental techniques such as yeast two-hybrid (Y2H) system. However, non-binary interaction data taken from TAP (Tandem Affinity Purification) experimental technique is also utilized to detect complexes. Korgan *et al.* [64], Pu *et al.* [65], and Caroline *et al.* [66] have developed different techniques to construct PPI networks from TAP data. All the three techniques finally utilize MCL [34] for detecting complexes. Geva *et al.* [67] proposed a complex detection method, CODEC, without constructing a PPI network by modeling TAP data as a bipartite graph.

1.3 Organization of the thesis

The thesis comprises six chapters. The chapter-wise organization of the thesis is given below:

Chapter 1: Introduction

In this chapter, first, we briefly introduce the thesis work, then we discuss the related work, and finally, an outline of the thesis is provided.

Chapter 2: Identification of protein complexes using overlapping community detection algorithms

This chapter works on the first challenge. In this chapter, we propose an extension (EGCE) of the existing overlapping community detection algorithm, Greedy Clique Expansion (GCE), to detect highly overlapping nodes. We show that the extension outperforms other methods when overlapping nodes (proteins) belong to more than three communities (complexes).

Chapter 3: A new family of similarity measures for scoring confidence of protein interactions using gene ontology

This chapter works on the second challenge. In this chapter, we propose a new family of gene ontology (GO) -based semantic similarity measures (SSMs), namely Relative Depth Specificity (RDS), Relative Node-based Specificity (RNS), and Relative Edge-based Specificity (RES) by redefining the specificity of GO terms. We show that all

the three measures are quite effective, particularly, RNS and RES more effectively distinguish true PPIs from false positives than the existing alternatives.

Chapter 4: Impact of the continuous evolution of gene ontology on the performance of similarity measures

This chapter presents the third contribution made in the thesis. In this chapter, we systematically study the impact of the continuous evolution of GO on the performance of SSMs for the task of scoring confidence of PPIs. We observe that the performance of SSMs gets affected due to the regular updates of GO and they are not robust in all conditions, rather they keep their performance quite similar in certain conditions.

Chapter 5: Impact of low-confidence interactions on computational identification of protein complexes

This chapter presents the fourth contribution made in the thesis. In this chapter, we systematically study the impact of low-confidence PPIs on computational identification of protein complexes using GO-based SSMs. We observe that each complex detection algorithm significantly improves its performance after elimination of low-similarity scored PPIs.

Chapter 6: Conclusion and future work

Finally, this chapter briefly discusses the conclusion of the thesis and some future research directions.

Chapter 2

Identification of protein complexes using overlapping community detection algorithms

Proteins play a central role in performing cellular functions. However, the majority of them perform their functions by interacting with other proteins instead of working alone. Such an assembly formed by physical interactions of more than one proteins to accomplish a biological function is generally referred to as a *protein complex*. Identifying protein complexes are important to unfold the principles of cellular processes. A protein may perform several functions and involve multiple processes. Hence a protein may be the part of several distinct complexes.

Over the years, a large amount of *protein-protein interaction* (PPI) data is becoming increasingly available due to high-throughput technologies and is utilized to detect protein complexes computationally. The problem of detecting protein complexes from PPI data (equivalently, from PPI networks) are usually converted to the problem of detecting *communities* from complex networks, particularly *overlapping communities*. Although there are algorithms for detecting overlapping communities, the majority of these are unable to detect highly overlapping communities accurately. The performance of these algorithms falls sharply when overlapping nodes (proteins) belong to more than three communities (complexes).

In this chapter, we propose an extension of the existing overlapping *community detection algorithm* (CDA), namely the Greedy Clique Expansion (GCE). We refer to the proposed algorithm as EGCE. Due to lack of unavailability of real networks with complete information of ground-truth communities, first, we experiment on state-of-the-art synthetic benchmark datasets. We observe that EGCE exhibits excellent performance when overlapping nodes belong to more than three communities. We also experiment on real datasets and observe good performance. EGCE can be applied to the networks with a substantial amount of

overlapping community structure such as PPI networks. The extension is generic in nature and can be applied to extend any existing overlapping CDAs.

2.1 Introduction

Proteins are the main workers to accomplish cellular functions. However, the majority of the proteins do not perform their functions alone, rather a group of proteins, commonly known as a *protein complex*, physically interact together to accomplish a specific biological function in the cell [68]. Identifying protein complexes are essential to understand the principles of cellular organization.

It is further known that many proteins involve in several cellular functions. Thus they are expected to be the part of multiple complexes, i.e., protein complexes are overlapping in nature. To understand the overlapping nature of protein complexes (percentage of proteins belonging to more than one complex), we calculate the following statistics based on 144 complexes from CYC2008 protein complex dataset [69] that are present in the Combined-AP/MS PPI network of Saccharomyces cerevisiae [70]: approximately 12% of proteins belong to two complexes, 2.16% belong to three complexes, 0.45% belong to four complexes, and 0.8% belong to five complexes, respectively. Even the conservative statistics estimated from the simple model organism indicate $1-2\%$ proteins belong to more than three complexes. Hence we can safely assume that in the case of multicellular eukaryotes, like human, this percentage can easily lead to in the order of thousands of proteins belonging to more than three complexes. Failing to assign these proteins into their all native complexes hinders the progress towards the complete understanding of the cellular organization.

From the last two decades, a large volume of protein-protein interaction (PPI) data has become available due to high-throughput technologies. These PPI data are frequently used to detect protein complexes computationally. The problem of detecting protein complexes from PPI data (equivalently from PPI networks) are usually mapped to the problem of detecting *communities* in complex networks, particularly *overlapping communities*. In fact, many *community detection algorithms* (CDAs) are directly used for the identification of protein complexes. Hereafter, we refer to *community* and *complex* interchangeably.

In the theory of complex networks, detecting communities is one of the fundamental problems. Although there is no precise definition of community [7], the majority of the authors accept relatively dense structures as communities in networks. Community structure is an inherent property found in most of the real networks such as social networks, biological networks. These communities have significant importance in these networks. For example, persons studied from the same school are likely to form a community in social

networks, proteins having similar biological functions are likely to form a complex (i.e., a community) in PPI networks.

From the last two decades, a wide range of community detection algorithms (CDAs) has been developed. The majority of these algorithms considers the network as undirected and unweighted. The algorithms utilized a variety of techniques to identify the community structures such as removal of high-betweenness edges [71], optimization of modularity [72], k-clique percolation [73], statistical inference [74, 75], label propagation technique [76, 77], link partitioning [78], clique expansion by local optimization [8, 79], and many more. For detail review, we refer the readers to the surveys by Fortunato [7] and Xie *et al.* [80].

Problems related to determining the structural properties of graphs are often NP-hard [7]. Hence from the computational viewpoint, detecting communities in complex networks is a computationally challenging task. Initially, CDAs were designed for disjoint communities. However, many real-world communities are overlapping in nature, i.e., a single node may belong to multiple communities. For instance, a typical Facebook user belongs to several distinct communities; similarly, as discussed earlier, many proteins belong to multiple complexes in PPI networks [73, 81].

For the obvious reasons, disjoint CDAs assign a protein into a single complex only even the protein may belong to multiple complexes, and thus are unable to reveal the true picture of complexes. Overlapping CDAs mitigate this issue by assigning one protein into multiple complexes. Algorithms such as [8, 73, 75, 77, 78, 82] have been developed for overlapping communities. Some algorithms, such as [79, 83], even consider the hierarchical structure of communities.

The complexity of detecting communities rises rapidly as the degree of overlapping increases. The existing overlapping CDAs perform satisfactorily when overlapping nodes belong to very few communities [8]. Hence such algorithms fail to assign proteins to all their native complexes if they belong to several (more than 3 to 4) complexes.

In 2010, Lee *et al.* proposed an algorithm Greedy Clique Expansion (GCE) [8] for detecting overlapping communities and claimed that their algorithm performs well compared to several state-of-the-art algorithms. However, we observe that some of the immediate neighboring nodes of the communities detected by GCE are highly overlapping in nature, but not detected by GCE. We identify these nodes from other neighbors and assign them to their respective communities by utilizing the notion of *Interaction Probability* [84] of these neighboring nodes with respect to their communities. Here we propose this extension, EGCE (Extended Greedy Clique Expansion), for detecting highly overlapping nodes (proteins). We evaluate EGCE with five state-of-the-art overlapping CDAs by utilizing both

synthetic and empirical benchmarks.

We briefly describe GCE in section 2.2, and the new extension EGCE is explained in section 2.3. In section 2.4, the experimental setup and evaluation metrics are presented. The results are reported, analyzed, and discussed in section 2.5. Finally, we conclude the chapter in section 2.6.

2.2 Greedy Clique Expansion (GCE)

Let $G(V, E)$ be a graph with vertex and edge sets are denoted by V and E respectively. A community C may be thought of as an induced subgraph S of G such that the number of intra-community edges is more than the number of inter-community edges. By intra-community edge, we mean both end vertices of the edge belong to C whereas inter-community edges are those whose only one end vertex belongs to C.

GCE finds all maximal cliques in G with a minimum clique size k as seeds. The largest seed is selected as a candidate community C'. The candidate community C' is expanded by optimizing a community fitness function F_C locally as long as the addition of any neighboring node u improves the fitness of the subgraph $C' \cup \{u\}$. The fitness function F_C of a community C is defined as

$$F_C = \frac{d_{in}^C}{(d_{in}^C + d_{out}^C)^\alpha} ,$$

where d_{in}^C and d_{out}^C are twice the number of intra-community edges and the number of inter-community edges of C, respectively, and the α is a parameter. The expanded C' is then accepted if it is not within a minimum community distance ϵ of an already accepted community C, i.e., the expanded C' is discarded if it is near-duplicate of another accepted community C. The algorithm continues the procedure by selecting the next largest seed from the remaining seeds and stops when all seeds are expanded. All the parameters k, ϵ, and α are set to default values throughout our experiment except for the experiment done on PPI networks, where k is set to 3.

2.3 Extended Greedy Clique Expansion (EGCE)

The problem of GCE is that it cannot identify nodes belonging to many communities. The problem is associated with the community fitness function used in GCE. Let us try to understand the problem through an illustrative example. Let v be a node in a community C_1 detected by GCE, i.e., $F_{C_1 \setminus v} < F_{C_1}$. If v belongs to one more community C_2 without altering the structure of C_1 except for the addition of enough number of edges between

v and C_2 so that v can belong to both C_1 and C_2, then the fitness of C_1 is decreased as only $d_{out}^{C_1}$ is increased. Similarly, if v belongs to another community C_3 without altering the structure of C_1 and C_2 except for the addition of enough number of edges between v and C_3 so that v can belong to C_1, C_2, and C_3, then the fitness of C_1 is reduced more as $d_{out}^{C_1}$ is increased more. The fitness of C_1 decreases as community membership of v increases. So, $F_{C_1 \setminus v} < F_{C_1}$ will not be satisfied when v belongs to many communities, and v will not be assigned into C_1. In our extension, we identify this kind of highly overlapping nodes not detected by GCE and assign them to their respective communities. We exploit the notion of *Interaction Probability* [84] to detect these nodes. Let $IP_{u,C}$ be the *Interaction Probability* of a node $u \notin C$ with respect to C and defined as

$$IP_{u,C} = \frac{|(u,v) : (u,v) \in E \text{ and } v \in C\}|}{|C|}$$

Intuitively, $IP_{u,S}$ measures the strength of connectivity of a node $u \in G$ to a subgraph S of G. Let $N(C)$ be immediate neighboring nodes of a community C detected by GCE. The IP values of all nodes in $N(C)$ with respect to C show a flat distribution except for few outliers. We observe that the majority of these outliers are those highly overlapping nodes not detected by GCE. To separate these outliers from the general population of IP values of nodes in $N(C)$, we apply the k-means clustering algorithm with $k = 2$. We add one extra individual $min(IP_{u',C'})$ in the population before applying the k-means clustering algorithm, where $min(IP_{u',C'}) = min\{IP_{u',C'} : u' \in C \text{ and } C' = C \setminus \{u'\}\}$. Nodes in $N(C)$ having IP values close to $min(IP_{u',C'})$ are more likely to be true overlapping nodes. The nodes associated with these IP values (outliers) and the node with $min(IP_{u',C'})$ form a cluster and we merge this cluster with C. Figure 2.1 demonstrates the flowchart of EGCE.

Time complexity of EGCE: The authors of GCE mentioned in their paper [8] that the average time complexity of GCE cannot be determined purely in terms of |V| or |E| of the graph; rather, the complexity depends on subtler local characteristics of the graph that are difficult to specify rigorously. Since, EGCE is the extension of GCE, the time complexity of GCE cannot be determined satisfyingly in terms of |V| or |E|.

2.4 Experimental design and evaluation metrics

We have selected five state-of-the-art CDAs to evaluate EGCE. The five algorithms are - GCE [8] (discussed above), CFinder [73] (uses k-clique percolation), COPRA [77] (exploits label propagation technique), Link [78] (uses link partitioning instead of nodes), and OSLOM [75] (exploits statistical significances of clusters). We use the implementations provided by the authors of the above CDAs with the default set of parameters for the experiment with

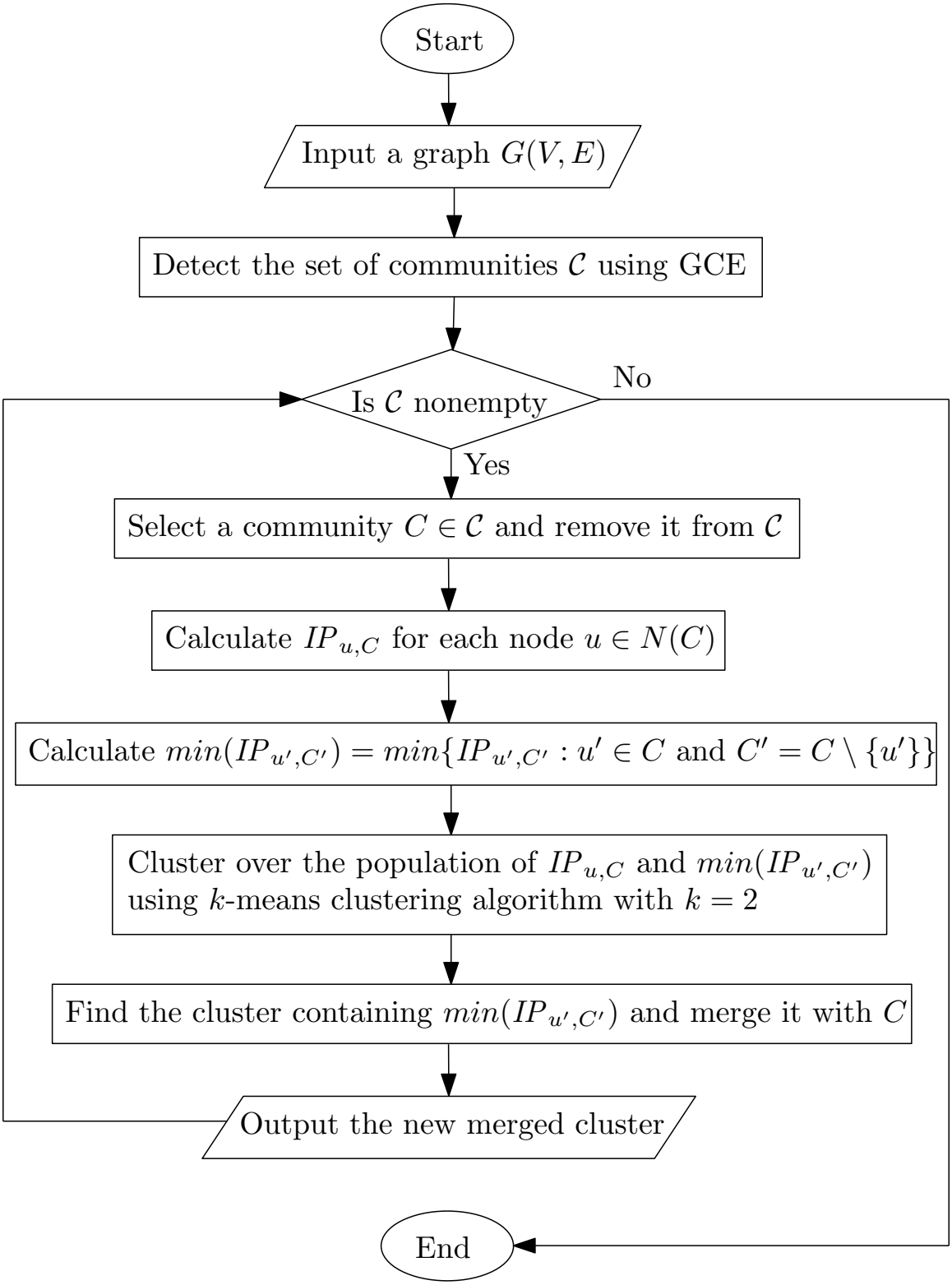


Figure 2.1: Flowchart of EGCE.

synthetic benchmarks. For empirical benchmarks, the parameter k (minimum clique size) of GCE/EGCE, and CFinder is set to 3. To implement EGCE, we modify the implementation provided by the authors of GCE.

We use a variant of Normalized Mutual Information (NMI) introduced by Lancichinetti *et al.* [79] to measure the similarity between detected and true communities, where a node may belong to more than one community, i.e., overlapping communities. NMI is an information-theoretic notion and first used by Danon *et al.* [85] for non-overlapping communities. The NMI between two random variables X and Y is defined as $NMI(X,Y) = {}^{2[H(X)-H(X|Y)]}/_{[H(X)+H(Y)]}$, where $H(X)$ is the entropy of the random variable X. The measure is normalized into [0,1], i.e., an NMI value of 1 indicates that the two sets of communities are exactly the same, whereas 0 indicates both the sets are completely independent. We also consider precision, recall, and F1 score/fmeasure [43] in the evaluation. However, our main discussion is based on NMI, as it is frequently used for community detection [9]. Further, NMI has been extended for overlapping communities by Lancichinetti *et al.* [79], whereas similar extension is not available for most of the other metrics.

2.4.1 Synthetic benchmarks

We use LFR benchmark [9] to create synthetic networks. The LFR benchmark facilitates to create networks with some properties that are claimed to present in real networks, e.g., power-law degree and community size distributions. The parameters of the LFR benchmark are as follows: the number of nodes (N), the average (k) and maximum degree (k_{max}) of a node, the minimum (C_{min}) and maximum (C_{max}) size of a community, number of overlapping nodes (O_n), the number of memberships of the overlapping nodes (O_m), i.e., the number of communities each overlapping node belongs to, exponents of power-law distributions of degrees (τ_1) and community sizes (τ_2), and a mixing parameter (μ), fraction of edges of a node that are ended in some randomly selected communities, e.g., $\mu = 0.3$ indicates approximately 70% of the total edges are planted within communities and remaining 30% are ended in some randomly selected communities. So, the higher value of the mixing parameter indicates a weaker community structure, whereas the lower value represents a stronger community structure.

Lancichinetti *et al.* experimented over a variety of CDAs on a specific set of LFR graphs [86]. This set of LFR graphs is becoming a standard and used in several studies including [77, 8]. We also use the same set of graph specifications. The set contains four LFR graphs: small/large graphs $(N = 1000/5000)$ and small/large communities $(C_{min} = 10/20$ and $C_{max} = 50/100$ i.e., $C_{max} = 5 \times C_{min})$. So, there are four possible combinations. We keep the following parameters fixed throughout our experiments also as in [86]: $k = 20$,

$k_{max} = 50$ i.e., $k_{max} = 2.5 \times k$, $\tau_1 = -2$, and $\tau_2 = -1$.

2.4.2 Empirical benchmarks

We evaluate the performance of EGCE and other algorithms with PPI network as well. It is very difficult to find PPI network data, where the full extent of complexes is known. Hence the performance evaluation and comparison of algorithms on such data have to be analyzed, keeping the above issue in mind. We use Combined-AP/MS PPI network of Saccharomyces cerevisiae [70] to construct the true network and CYC2008 protein complexes [69] as true communities. The same datasets are used by the authors of GCE to evaluate the performance of their algorithm as well. The Combined-AP/MS PPI network dataset contains 1622 proteins and 9070 interactions. We consider those complexes only whose constituent proteins present in the PPI network and then filter out complexes having one or two proteins. There are 880 proteins remain in the final set of true complexes. We keep the minimum clique size 3 to evaluate the performance of both GCE and EGCE.

2.5 Results and discussion

In this section, we report, discuss, and analyze the results of both synthetic and empirical benchmarks.

Synthetic benchmarks

We do our experiment in two folds to capture the extent to which all the six algorithms mentioned above, including EGCE, are able to detect overlapping communities. For each test, we generate 10 realizations of networks with the same parameters and calculate the mean and standard deviation of NMI. The error bars in the plots represent the standard deviation of NMI over 10 runs. Throughout the experiment, we keep fixed the default values of all parameters mentioned by the authors. In the first class of benchmarks, we vary O_m, i.e., the number of memberships of the overlapping nodes, from 3 to 10 by keeping fixed $O_n = 25\%$ of the total nodes and mixing parameter. Figure 2.2 and Figure 2.3 show the performance of all the six algorithms with mixing parameter $\mu = 0.1$ and $\mu = 0.3$.

In the second class of benchmarks, the fraction of the overlapping nodes, i.e., O_n/N, varies from 0.1 to 0.8 by keeping fixed $O_m = 5$ and mixing parameter. In Figure 2.4 and Figure 2.5, we show the performance of all the six algorithms with mixing parameter $\mu = 0.1$ and $\mu = 0.3$.

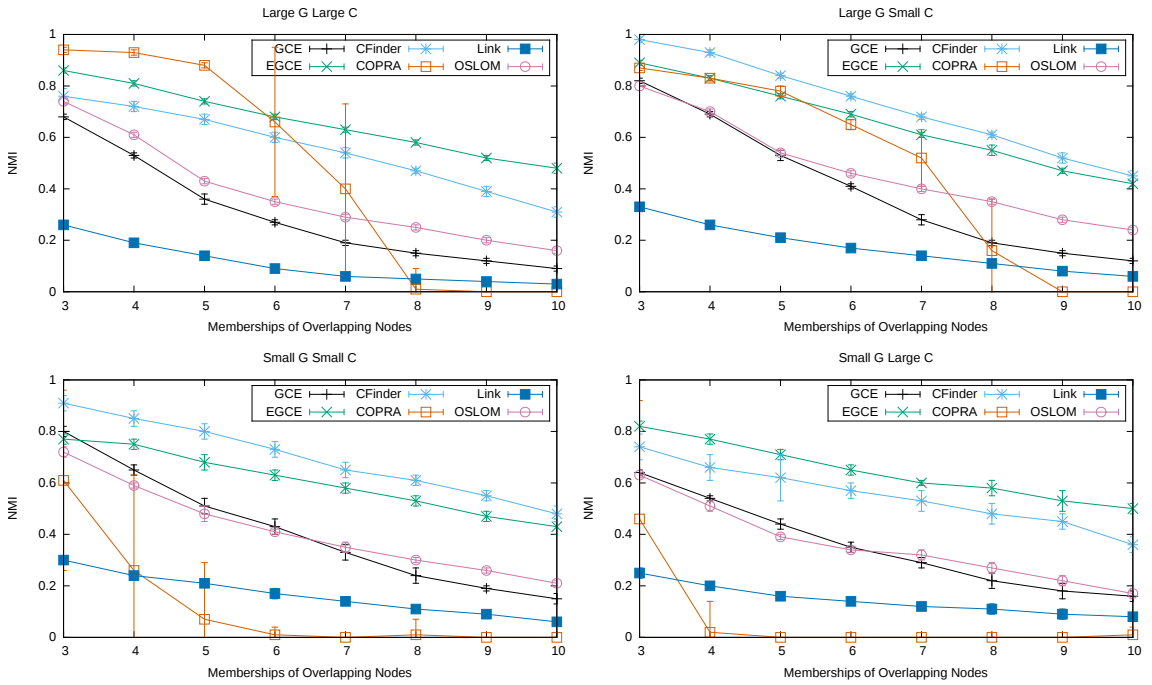


Figure 2.2: The mean NMI of different CDAs by varying the membership of the overlapping nodes (O_m) from 3 to 10 with mixing parameter $\mu = 0.1$. The number of overlapping nodes is kept fixed as 25% of the total nodes. The error bars represent the standard deviation of NMI over 10 runs.

We perform t-test (Aspin-Welch Unequal-variance) to check if the difference between the mean NMI values of 10 realizations (of networks) for two different methods is statistically significant. We observe that the differences are statistically significant for most of the cases. For example, with Large G and Large C at $O_m = 7$ in Figure 2.2, the p-value of the t-test for the mean NMI difference between the best performing CDA (EGCE) and the second-best performing CDA (CFinder) is $4.191212\text{e} - 09$.

We perform our experiment on synthetic benchmarks for different values of mixing parameter (μ). Here we report the results of $\mu = 0.1$ and $\mu = 0.3$ only since the relative performance is quite similar for other values of mixing parameter ($\mu = 0.2$ and $\mu = 0.4$).

We observe that EGCE performs well consistently as long as $\mu < 0.5$. In practice, it is assumed that community structure present in the network as long as $\mu < 0.5$ and for $\mu > 0.5$, communities are not well defined [86]. We also observe that EGCE shows better performance than other algorithms when $O_m \geq 3$, i.e., overlapping nodes belongs to at least 3 communities and particularly for large networks. So, from the experiment on synthetic networks, it is clear that EGCE outperforms other algorithms in detecting highly overlapping communities ($O_m \geq 3$) as long as community structure present in networks. From Figure 2.4 and Figure 2.5, it is also clear that EGCE performs significantly well for $0.1 < O_n/N <$

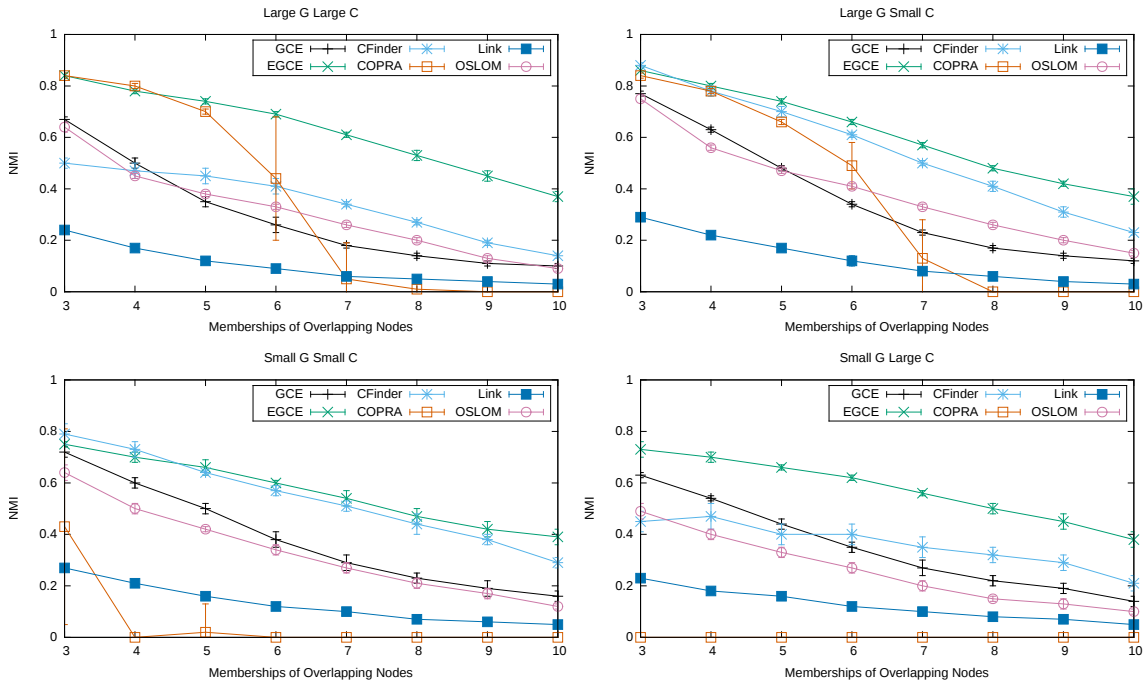


Figure 2.3: The mean NMI of different CDAs by varying the membership of the overlapping nodes (O_m) from 3 to 10 with mixing parameter $\mu = 0.3$. The number of overlapping nodes is kept fixed as 25% of the total nodes. The error bars represent the standard deviation of NMI over 10 runs.

0.6.

Empirical benchmarks

In Table 2.1, we report the performance in terms of NMI, precision, recall, and F1 score for all the algorithms except for CFinder as it fails to terminate on this dataset. We execute OSLOM and COPRA 10 times as they are non-deterministic.

Table 2.1: NMI, F1 score, precision, and recall of different CDAs on the Combined-AP/MS PPI network of S. cerevisiae with the CYC2008 protein complex dataset.

Algorithm	NMI	F1 score	Precision	Recall
GCE	0.556	0.771	0.752	0.792
EGCE	0.550	0.767	0.750	0.785
Link	0.433	0.356	0.217	0.986
OSLOM	0.363	0.366	0.228	0.922
COPRA	0.295	0.329	0.220	0.651

Both GCE and EGCE show almost similar performances (GCE performs slightly better than EGCE) and better than other algorithms. There are several reasons for this similar perfor-

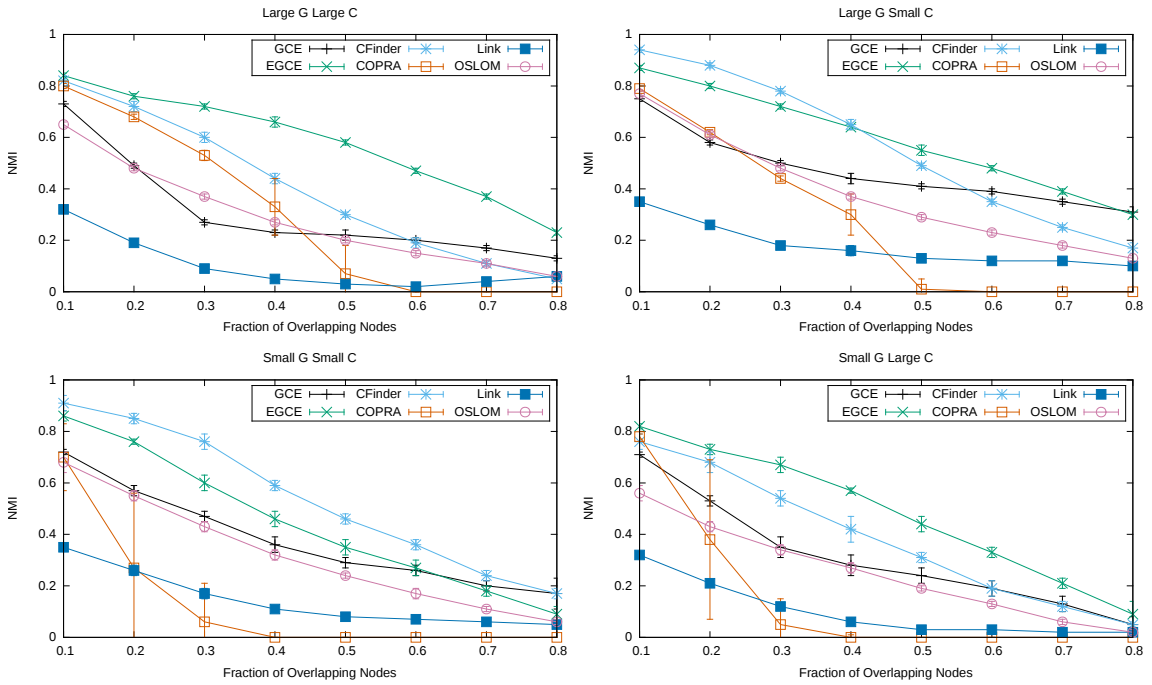


Figure 2.4: The mean NMI of different CDAs by varying the fraction of the overlapping nodes (O_n/N) from 0.1 to 0.8 with mixing parameter $\mu = 0.1$. The membership of the overlapping nodes (O_m) is kept fixed as 5. The error bars represent the standard deviation of NMI over 10 runs.

mance of GCE and EGCE. It is known that PPI networks are incomplete, along with a significant amount of both false positive and false negative interactions [10]. The protein complex dataset we consider as true communities have 880 proteins, and only 136 of these belong to more than one complexes (after necessary elimination). So, the fraction of overlapping nodes in the PPI network is approximately $136/1622 \approx 0.084$ i.e., $O_n/N \approx 0.084 < 0.1$ and it is a small network too. We also find that the average memberships of overlapping nodes (O_m) in the PPI network is approximately 2.35, which is also less than 3. From the aforementioned synthetic benchmarks, we see that EGCE performs well significantly when $(O_m \geq 3)$ and $O_n/N > 0.1$ and particularly for large networks.

However, if we see how overlapping proteins are detected by both the algorithms, we observe that EGCE detects more number of true overlapping proteins than GCE. Out of 136 true overlapping nodes, GCE detects 16 only, whereas EGCE detects 23 as overlapping. While EGCE is able to detect seven true overlapping proteins (with ORF IDs YDR190C, YER025W, YGR090W, YHR099W, YJR007W, YPL235W, and YPR023C) as overlapping too, GCE determines their memberships as one. Further, the protein with ORF ID YFL039C belongs to three complexes in the ground truth, and EGCE correctly determines its membership value as three too, whereas GCE shows its membership value as two. So, EGCE

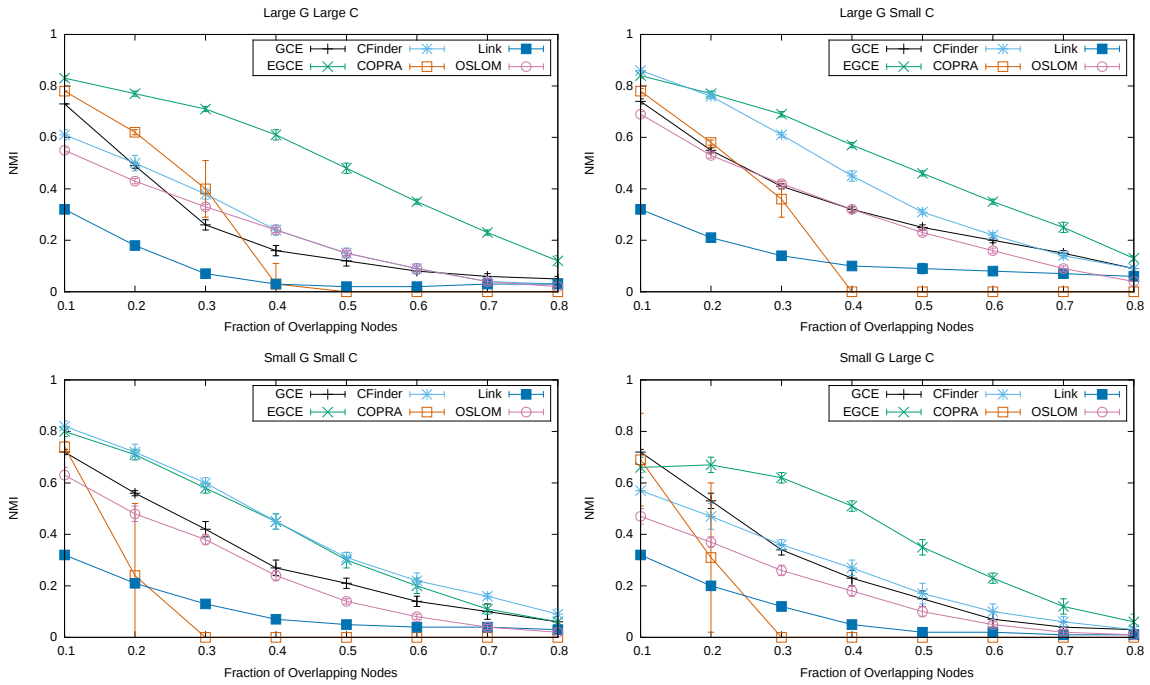


Figure 2.5: The mean NMI of different CDAs by varying the fraction of the overlapping nodes (O_n/N) from 0.1 to 0.8 with mixing parameter $\mu = 0.3$. The membership of the overlapping nodes (O_m) is kept fixed as 5. The error bars represent the standard deviation of NMI over 10 runs.

assigns overlapping proteins into complexes better way than GCE. However, with the increase in high-confidence complexes and protein-protein interactions, we feel EGCE will prove more its worth as shown in the results obtained from synthetic benchmarks. It may be noted that Link and OSLOM produce high recall. However, due to their poor precision, they produce a relatively lower F1 scores.

2.6 Conclusion

In this work, we introduce an extension, EGCE (Extended Greedy Clique Expansion), of existing overlapping CDA, GCE (Greedy Clique Expansion), to identify highly overlapping nodes. Our extensive analysis of synthetic and empirical benchmarks suggests that EGCE outperforms other overlapping CDAs, particularly when overlapping nodes belong to more than three communities. Although GCE performs slightly better than EGCE for the empirical benchmark selected in the evaluation, EGCE assigns overlapping proteins into their native complexes better way than GCE. The new extension is generic in nature and not restricted to GCE only, hence can be applied to extend any existing overlapping CDAs. We notice that some CDAs show better performance (in terms of NMI for detecting overlapping communi-

ties) with LFR benchmark when this extension is applied. Hence the work can be extended by evaluating the new extension with other existing CDAs.

Chapter 3

A new family of similarity measures for scoring confidence of protein interactions using gene ontology

As we have discussed in Chapter 1, protein-protein interaction (PPI) data have a significant amount of false positives. Their presence leads to a negative impact on any downstream analysis of PPI networks. False interactions in PPI data is also likely to affect protein complex detection from PPI networks adversely. In this chapter, we try to identify these false positive PPIs.

The large-scale PPI data have the potential to play a significant role in the endeavor of understanding cellular processes. However, the presence of a considerable fraction of false positives is a bottleneck in realizing this potential. There have been continuous efforts to utilize complementary resources for scoring confidence of PPIs in a manner that false positive interactions get a low confidence score. Gene ontology (GO), a taxonomy of biological terms to represent the properties of gene products and their relations, has been widely used for this purpose. We utilize GO to introduce a new set of specificity measures: Relative Depth Specificity (RDS), Relative Node-based Specificity (RNS), and Relative Edge-based Specificity (RES), leading to a new family of similarity measures. We use these similarity measures to obtain a confidence score for each PPI. We evaluate the new measures using four different benchmarks. We show that all the three measures are quite effective. Notably, RNS and RES more effectively distinguish true PPIs from false positives than the existing alternatives. RES also shows a robust set-discriminating power and can be useful for protein functional clustering as well.

3.1 Introduction

A significant amount of protein-protein interaction (PPI) data has become available due to high-throughput technologies. PPI data play a central role in the systems-level understanding of cellular processes with important applications in disease diagnosis and therapy. A considerable fraction of interactions is false positives due to limitations of experiments used in detecting protein interactions[10]. Hence, a ranking or a scoring mechanism distinguishing between true and false interactions is important for any downstream analysis. There have been continuous efforts to utilize additional knowledge resources, such as Gene ontology (GO) [11], in scoring confidence of PPIs in a manner that false positive interactions get a low confidence score[87]. The primary objective of this work is to introduce a new family of semantic similarity measures (SSMs) between gene products using GO for scoring confidence of PPIs.

GO has been effectively utilized in predicting and validating PPIs [15, 16, 17], and confidence scoring of PPIs [88, 89, 90, 91, 92, 93] among other genomic applications such as predicting protein functions [12, 13, 14], analyzing pathways [18], etc. It is a taxonomy of biological terms to represent the properties of genes and gene products (e.g., proteins) and is organized as a directed acyclic graph (DAG) to describe the relationship among the terms. GO is made up of three independent ontologies: biological process (BP), cellular component (CC), and molecular function (MF). A section of GO DAG (Release March 2015) is shown in Figure 3.1.

Terms closer to the root are more generic in nature, and specificity of terms gradually increases as we move towards the leaves. The more specific a term is, the more informative it is. Ontology-based SSM is a quantitative function that measures the similarity between two terms based upon their relations over a set of terms organized as an ontology. Formally, it is a function of two ontology terms (or two sets of ontology terms) that returns a real number indicating the closeness between the terms in the context of semantic meaning [87]. Gene or gene products in different model organisms are annotated to GO terms based on various evidences and is available through annotation corpora. An annotation corpus of a species (e.g., yeast) is an association between gene products of the species and GO terms.

3.1.1 Motivation and hypothesis

The notion of Information Content (IC) is widely used in defining SSMs. It quantifies the specificity of a term in an ontology, i.e., how specific a term in an ontology is. The IC is explained formally in section 3.2. The IC-based SSMs assume that the given ontology is complete and define the specificity of a term by considering the whole ontology. However,

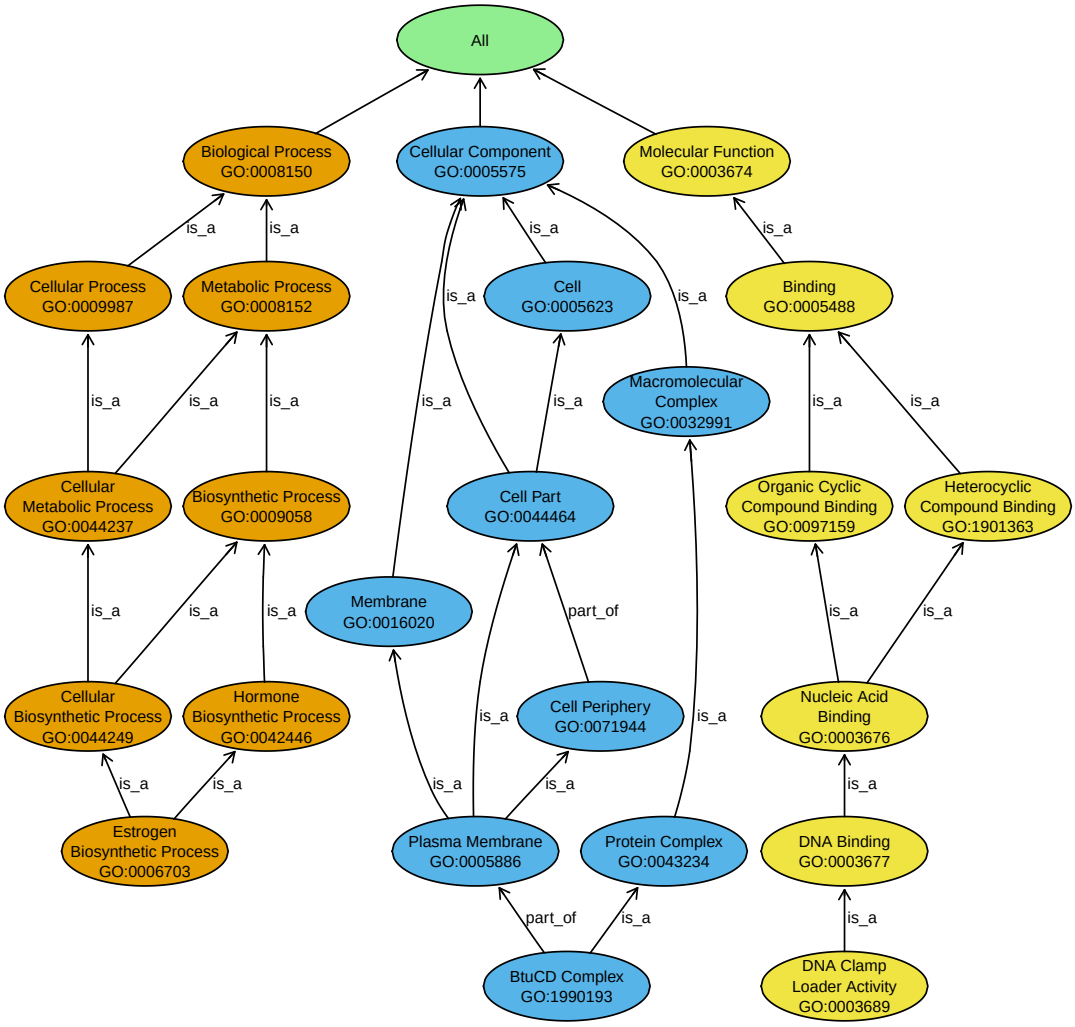


Figure 3.1: A section of GO DAG. Roots of the three ontologies BP, CC, and MF are Biological Process (GO:0008150), Cellular Component (GO:0005575), and Molecular Function (GO:0003674), respectively. The three roots are connected to a dummy node 'All' to make a single GO DAG.

GO is being updated regularly with the addition of new terms and removal of old terms. Furthermore, when new information of gene or gene product is discovered, annotation data corresponding to the appropriate terms are updated as well. Some proteins are annotated with a large number of terms, while many proteins are annotated to one term only, i.e., annotations are not uniformly distributed among the terms (annotation bias). Thus the continuous evolution of the GO DAG, regular updates in annotation and non-uniform distribution of terms (as well as annotations) over the ontology are likely to impact confidence

scores of several PPIs with each update.

A GO term is more closely related to its ancestors and descendants as the ontology is hierarchically organized. The major part of the contribution towards the specificity of a term is accumulated through the properties of its ancestors and descendants. Therefore for quantifying the specificity of a term in an ontology like GO (which is very large, complex, continuously evolving and not uniformly distributed), it is safe to consider the properties of the subgraph consisting of the term itself along with its ancestors and descendants only instead of considering the whole ontology, to minimize the impact of continuous evolution.

Our main hypothesis is that the explicit encoding of the aforementioned unexplored subgraph-based specificity notions into a new family of SSMs could be useful for scoring confidence of PPIs.

3.1.2 Definition of the problem and contribution

The main problem of the current study is to define the specificity of a GO term, based on the properties of the subgraph consisting of the term itself along with its ancestors and descendants only, that could be useful for scoring confidence of PPIs.

With the aforementioned unexplored notion of specificity, we introduce three simple yet effective specificity measures: Relative Depth Specificity (RDS), Relative Node-based Specificity (RNS), and Relative Edge-based Specificity (RES). This new set of specificity measures led to a new family of SSMs.

We compare the performance of the new SSMs with six state-of-the-art SSMs proposed by Resnik [94], Lin [95], Schlicker *et al.* [96], Jiang & Conrath [97], Wang *et al.* [98], and Jain & Bader [99], referred to as *Resnik*, *Lin*, *Rel*, *Jiang*, *Wang*, and *TCSS*, respectively, in the rest of the chapter. Resnik and TCSS have been considered to be the best SSMs for scoring confidence of PPIs by several studies such as Guo *et al.* [100], Xu *et al.* [101], Jain & Bader [99], and Pesquita [102]. We use four different benchmarks to evaluate the new SSMs. The four benchmarks are - 1) correlation with reference dataset from HIPPIE database [103], 2) ROC curve analysis with DIP database [104], 3) set-discriminating power of KEGG pathways [105], and 4) correlation with protein family (Pfam) using CESSM dataset[106]. The first benchmark is for human PPIs only as HIPPIE is an integrated database of human PPIs and the rest of the three benchmarks are applied to both yeast (S. cerevisiae) and human (H. sapiens) PPIs.

The rest of the chapter is organized in the following manner. A brief survey of the literature is presented in section 3.2. The new family of SSMs is explained in section 3.3. Section 3.4 describes the experimental design, evaluation metrics, datasets used, and implementa-

tion. In section 3.5, the results are reported, analyzed, and discussed. Finally, we conclude the chapter in section 3.6.

3.2 Related work

This section introduces a brief review of the literature on PPI confidence scoring methods and GO-based SSMs. For an in-depth review of the family of GO-based SSMs, we refer the reader to the surveys by Pesquita *et al.* [87], Harispe *et al.* [107], Mazandu *et al.* [108], and Pesquita *et al.* [102].

3.2.1 PPI confidence scoring methods

Computational approaches for scoring confidence of PPIs mainly differ in the selection of information used in the prediction model. The common sources of this information are three-dimensional protein structures [109], protein sequences [110], gene expression profiles [111], phylogenetic trees [112, 113], phylogenetic profiles [114], GO [88, 89, 90, 91, 92, 93] etc. Some approaches utilize topology of interaction network from already existing true PPIs [115, 116, 117]. Text mining on peer-reviewed literature is also used for scoring confidence of PPIs [118]. A few approaches utilize multiple sources of information [119, 120]. However, GO is a very comprehensive resource for the properties of gene products and their functional relationships across species. It provides a promising way to infer functional information of gene products. The idea of semantic similarity is a common way to utilize GO for scoring confidence of PPIs. Semantic similarity between two proteins (See section 3.2.3) involved in a PPI may be treated as a confidence score of the interaction. The current study is primarily focused on the SSMs by exploiting GO for scoring confidence of PPIs.

3.2.2 GO-based SSMs

Ontology-based SSMs were originally introduced in the fields of cognitive sciences by Tversky [121] and Natural Language Processing (NLP) and Information Retrieval (IR) by Rada *et al.* [122]. Since then, a plethora of semantic similarity measures based on WordNet (a large lexical database of English) was developed, such as the pioneering works introduced by Resnik [94], Jiang & Conrath [97] and Lin [95]. However, the first pioneering work was introduced by Lord et al. [123, 124] in the field of biology and this work has started the research on the development of GO-based SSMs and their applications in genomics such as [96, 98, 99, 125, 126, 127]. Here, we provide a brief overview of different SSMs.

Existing SSMs are classified broadly into two categories: *edge- and node-based* [87]. Edge-based measures are the natural and direct way of defining SSMs. Rada *et al.* [122] introduced an SSM of this kind in a lexical taxonomy, which was then applied in GO by Nagar and Al-Mubaid [125]. Subsequently, several edge-based SSMs have been developed and used in GO [128, 129, 130, 131]. In the edge-based approach, shared paths between two terms are primarily considered for computing the similarity between them. It assumes that terms at the same level have similar specificity and edges at the same level represent the same semantic distances between two terms [87], which are seldom true in GO. Furthermore, an edge-based approach does not account annotation information of terms and entirely relies on the topological structure of the GO DAG. Hence edge-based methods are more sensitive to the intrinsic structure of the GO DAG.

The most commonly used SSMs are node-based that computes the similarity between two terms by comparing their properties, common ancestors, or their descendants. As mentioned earlier, the majority of the node-based approaches use the notion of information content (IC) to define the specificity of a term. The IC of a term t is defined as

$$IC(t) = -ln\ p(t) \tag{3.1}$$

where $p(t)$ is the probability or frequency of occurrence of t. Usually, the descendants of t are also considered for computing $IC(t)$. The probability of occurrence, $p(t)$ of term t in GO is defined as:

$$p(t) = \frac{|\{t\} \cup Des(t)|}{N} \tag{3.2}$$

where $Des(t)$ is the set of descendants of t and N is the number of terms in the ontology. Since gene products are annotated to terms in GO, $p(t)$ is estimated as the frequency of annotations of t, i.e.,

$$p(t) = \frac{|Ant(\{t\} \cup Des(t))|}{M} \tag{3.3}$$

where $Ant(T)$ is the set of annotations to the set of terms T and M is the total number of annotations in the GO. In words, it is the ratio of the number of annotations to t and its descendants to the total number of annotations. The aforementioned two definitions are commonly known as an *intrinsic* and *extrinsic* way of defining the probability function $p(t)$, respectively.

The most commonly used node-based SSMs are Resnik [94], Lin [95], and Jiang & Conrath [97], which were initially developed for WordNet and subsequently applied to GO by Lord *et al.* [123, 124]. Thereafter, a number of node-based SSMs have been proposed in order to improve the existing SSMs in different perspectives and applications [96, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143]. The major drawbacks of

IC-based SSMs are already pointed out in section 3.1.1. SSMs such as [98, 127, 144, 145] combine both node- and edge-based approaches and are commonly referred to as hybrid approaches. Recently, some complex structural-based SSMs are also developed [99, 146, 147].

3.2.3 SSM between two sets of terms

A gene product may be annotated with more than one term in the same GO. Suppose, p_1 and p_2 are two gene products annotated to the set of terms S and T, respectively. The similarity between p_1 and p_2 are calculated as the similarity between two sets S and T, i.e., $SSM(p_1, p_2) = SSM(S, T)$. Therefore we need to combine GO terms of S and T. Generally, the following three types of strategies used in the literature:

Maximum (MAX) - In MAX strategy [148], similarity between S and T is calculated as the maximum of the set $S \times T$.

$$SSM_{MAX}(S,T) = \max_{s \in S, t \in T} SSM(s,t) \tag{3.4}$$

Average (Avg) - In 'average' strategy [123, 124], similarity between S and T is calculated as the average of the set $S \times T$.

$$SSM_{avg}(S,T) = \frac{\sum_{s \in S, t \in T} SSM(s,t)}{m \times n} \tag{3.5}$$

where $m = |S|$ and $n = |T|$.

Best-match average (BMA) - SSMs between two sets of terms form a matrix. BMA [149, 96, 150] is defined as the average of all maximum SSMs on each row and column of the matrix.

$$SSM_{BMA}(S,T) = \frac{\sum_{i=1}^{m} \max_{1 \le j \le n} SSM(s_i,t_j) + \sum_{j=1}^{n} \max_{1 \le i \le m} SSM(s_i,t_j)}{m+n} \tag{3.6}$$

where $s_i \in S$ and $t_j \in T$.

3.2.4 SSMs used in evaluation

Resnik - Resnik considers IC of the *most informative common ancestor* (MICA) only [94]. The similarity between two terms s and t in Resnik is defined as

$$SSM_{Resnik}(s,t) = \max_{c \in C} IC(c) = IC(MICA(s,t)) \tag{3.7}$$

where C is the set of common ancestors of s and t, and IC is the information content defined earlier. It is the IC of the closest common ancestor or *lowest common ancestor* (LCA) of s and t.

Lin and Jiang - Although Resnik is very effective for computing information shared by two terms, it cannot distinguish between pairs of terms having the same MICA. To overcome the problem, Lin and Jiang are developed by considering ICs of both the terms along with their MICAs in different ways [95, 97]. The similarity between two terms is calculated by these two methods as

$$SSM_{Lin}(s,t) = \frac{2 \times IC(MICA(s,t))}{IC(s) + IC(t)}, \tag{3.8}$$

$$SSM_{Jiang}(s,t) = 1 - [IC(s) + IC(t) - 2 \times IC(MICA(s,t))]. \tag{3.9}$$

Rel - Lin and Jiang overestimate when one term is an ancestor of another. For example, when both the terms are same, the similarity score will be 1, irrespective of its specificity. Rel combines Resnik and Lin in order to capture relevance information by multiplying one minus the *extrinsic* probability of MICA to SSM_{Lin} [96]. As per Rel, the similarity between two terms is calculated as

$$SSM_{Rel}(s,t) = \frac{2 \times IC(MICA(s,t))(1 - p(MICA(s,t)))}{IC(s) + IC(t)}. \tag{3.10}$$

Wang - Wang is a hybrid measure that combines both edge- and node-based approaches [98]. Let $G_t = (V_t, E_t)$ be a DAG for a term t in GO such that V_t is the set of ancestors of t, including t itself, and E_t is the set of edges connecting terms in G_t. Terms closer to term t in G_t contribute more of its semantics to the semantics of term t. The semantic contribution of a term c to the semantics of term t in G_t is denoted as S-value of c or $S_{G_t}(c)$ and defined as:

$$\begin{cases} S_{G_t}(t) = 1 \\ S_{G_t}(c) = \max\{w_e \times S_{G_t}(c') : c' \in \text{ children of } c\} \text{ if } c \neq t \end{cases} \tag{3.11}$$

where w_e $(0 < w_e < 1)$ is semantic contribution factor for edge $e \in E_t$ from term c' to term c. For example, semantic contribution factors (w_e) of *is_a* and *part_of* relationships may be treated as 0.8 and 0.6, respectively. To compare semantics of two terms, a semantic value $SV(t)$ is computed as the aggregate contribution of the semantics of all the terms in G_t to term t and defined as:

$$SV(t) = \sum_{c \in V_t} S_{G_t}(c). \tag{3.12}$$

Now, SSM between two terms s and t with respect to their DAGs $G_s = (V_s, E_s)$ and $G_t =$

(V_t, E_t) is defined as:

$$SSM_{Wang}(s,t) = \frac{\sum_{c \in V_s \cap V_t}(S_{G_s}(c) + S_{G_t}(c))}{SV(s) + SV(t)}. \tag{3.13}$$

The numerator is the summation of S-values of common terms between the two DAGs. S-values of common terms between the two DAGs may not be the same as the locations of s and t may differ in GO.

TCSS - TCSS exploits the unequal depth of biological knowledge representation in different branches of GO DAG [99]. The objective of TCSS is to identify subsets of similar GO terms (e.g., terms related to nucleus and terms related to mitochondrion belong two different subsets) and score PPIs higher if participating proteins belong to the same subset compared to PPIs whose participating proteins belong to different subsets. The authors have introduced a structural-based IC, referred to as topological information content (ICT), to identify subgraph root terms during the preprocessing stage.

$$ICT(t) = -ln\Big(\frac{|Child(t)|}{N}\Big) \tag{3.14}$$

where $Child(t)$ is the set of children of t and N is the number of terms in the ontology.

3.3 New GO-based SSMs

In this section, we introduce the new family of SSMs based on the proposed set of specificity measures. To define the specificity of a GO term, we consider the properties of the subgraph consisting of the term itself along with its ancestors and descendants only and ignore the rest of the ontology. The new specificity models quantify how specific a term in ontology is. The specificity of a parent (term) always will be less than any of its children. RDS considers a specific path of the aforementioned subgraph, while RNS and RES consider the whole subgraph. However, RNS relies on the properties of the nodes only, whereas RES considers the edges of the subgraph as well.

3.3.1 Relative Depth Specificity (RDS)

Let $d_{t,r}$ and $d_{l,t,r}$ are the length of the longest path from term t to the root r and length of the longest path from any leaf l to the root r via the term t, respectively. Then, RDS of a term t in GO is defined as

$$RDS(t) = \frac{d_{t,r}}{d_{l,t,r}} = \frac{d_{t,r}}{d_{l,t} + d_{t,r}}. \tag{3.15}$$

In words, $RDS(t)$ is the ratio between the length of the longest path from the term t to the root and the length of the longest path from any leaf to the root via the term t. This is the simplest SSM that does not consider annotation information. The specificity of the leaves and the root would be highest (1) and lowest (0), respectively. When multiple paths are present between two terms, we consider the longest one as it is likely to be more informative than others.

3.3.2 Relative Node-based Specificity (RNS)

Let $G_1(V_1, E_1)$ be the subgraph consisting of the term t itself along with its ancestors; and $G_2(V_2, E_2)$ be the subgraph consisting of the term t itself along with its ancestors and descendants. The RNS of a term t in GO is defined as

$$RNS(t) = \frac{|Ant(V_1)| + |V_1|}{|Ant(V_2)| + |V_2|} \tag{3.16}$$

where $Ant(T)$ be the set of annotations to the set of terms T as mentioned earlier. In words, it is the ratio of the sum of nodes along with its annotations of the subgraph consisting of the term t and its ancestors to the sum of nodes along with its annotations of the subgraph consisting of t, its ancestors and descendants. Thus, RNS of the leaves and the root would be highest (1), and lowest (close to 0), respectively.

3.3.3 Relative Edge-based Specificity (RES)

We define the weight of an edge $e(t_1, t_2)$ between terms t_1 and t_2 in GO as:

$$w(e) = |Ant(\{t_1\})| + |Ant(\{t_2\})|. \tag{3.17}$$

It is the summation of the number of annotations of terms t_1 and t_2. The weight of a set of edges E is defined as:

$$W(E) = \sum w(e_i) : e_i \in E. \tag{3.18}$$

It is the summation of weights of all edges in the set of edges E. Let $G_1(V_1, E_1)$ be the subgraph consisting of the term t itself along with its ancestors and $G_2(V_2, E_2)$ be the subgraph consisting of the term t itself along with its ancestors and descendants as in RNS. The Relative Edge-based Specificity of a term t in GO is defined as

$$RES(t) = \frac{W(E_1) + |E_1|}{W(E_2) + |E_2|}. \tag{3.19}$$

In words, it is the ratio of the summation of weighted and unweighted edges of the subgraph consisting of the term t itself along with its ancestors to the summation of weighted and unweighted edges of the subgraph consisting of t itself along with its ancestors and descendants. Thus, the specificity of the leaves and the root would be highest (1), and lowest (0), respectively.

The similarities between the two terms s and t are calculated as:

$$SSM_{RDS}(s,t) = \max_{c \in C} RDS(c) = RDS(MICA(s,t)), \tag{3.20}$$

$$SSM_{RNS}(s,t) = \max_{c \in C} RNS(c) = RNS(MICA(s,t)), \tag{3.21}$$

$$SSM_{RES}(s,t) = \max_{c \in C} RES(c) = RES(MICA(s,t)) \tag{3.22}$$

where C is the set of common ancestors of s and t as mentioned earlier. Figure 3.2 demonstrates the flowchart of computing similarity scores between two proteins using the proposed SSMs.

We have chosen the MICA to define the shared specificity between the two terms similar to Resnik. It is noteworthy to mention that the proposed specificity models are different from IC models as they do not rely on probability functions. Therefore we cannot directly apply the new specificity models to other IC-based similarity measures such as Lin, Rel, and Jiang.

3.4 Experimental setup

Our experimental design for evaluation is based on the following two assumptions. First, two proteins involved in the same biological process(es) are more likely to interact than proteins involved in different processes [17, p.953] and [99]. Second, two proteins need to come in close proximity (at least transiently) for interaction. Hence co-localization also provides evidence of interaction [151, p. 689] and [99]. However, if two proteins interact physically, there is no guarantee that they share the same molecular function [152, p. 27]. The 'average' strategy underestimates when two gene products share many similar terms as it considers all possible term pairs of the two gene products [153]. By contrast, the MAX strategy overestimates when two gene products share a few similar terms as it is indifferent to the number of dissimilar terms between the gene products [153]. The BMA strategy, which considers both similar and dissimilar terms [153], does not suffer from the aforementioned limitations. Further, in PPIs, proteins need to be in close proximity (share

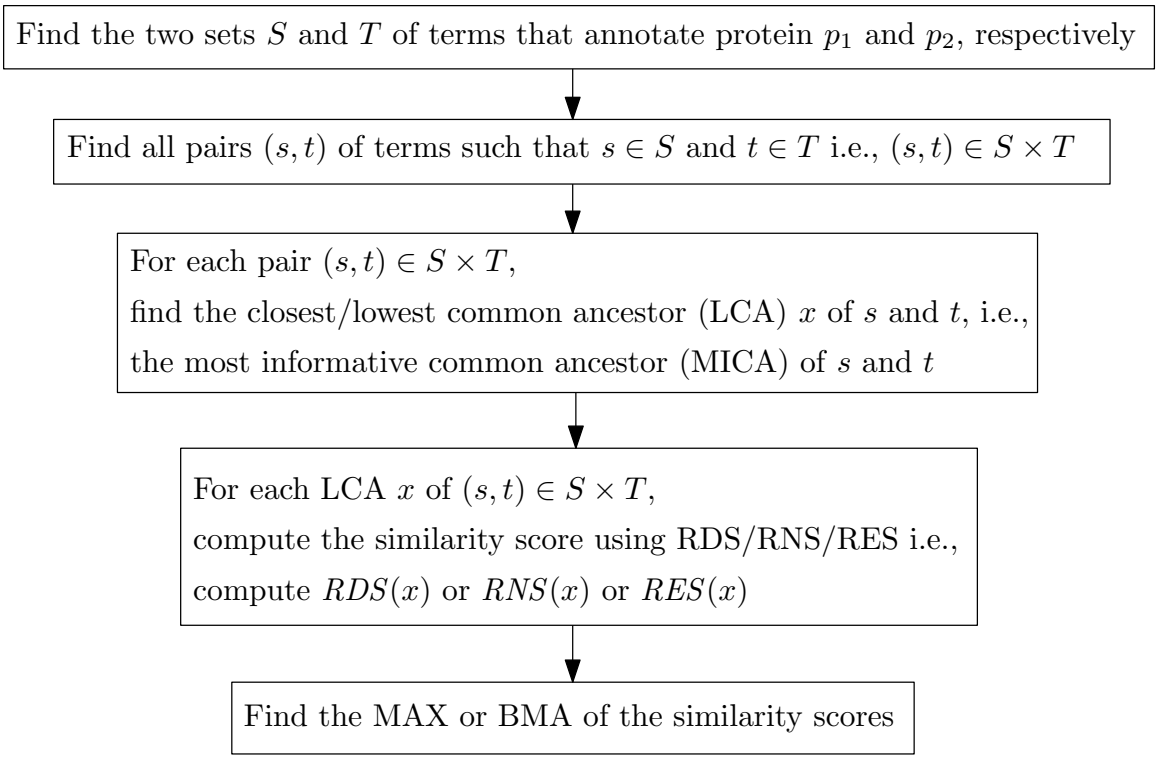


Figure 3.2: Flowchart of computing similarity scores between two proteins using the proposed SSMs.

similar CC terms) and participate in the same biological process (share similar BP terms) once, among all possible combinations, to become biologically relevant [99]. Hence MAX and BMA are considered better strategies than the 'average' for scoring confidence of PPIs. In light of the above discussion, we use BP and CC ontologies of GO along with MAX and BMA strategies for performance evaluation. We exclude electronically inferred annotations (IEA) of GO terms which lack manual curation. We consider only those protein pairs which are having both the proteins annotated with at least one GO term other than the root in their respective ontologies.

As mentioned earlier, the new SSMs are evaluated on the four benchmarks: 1) correlation with reference dataset from HIPPIE database, 2) ROC curve analysis in predicting true PPIs from DIP database, 3) set-discriminating power of KEGG pathways, and 4) correlation with Pfam on CESSM dataset. Evaluation is done using both yeast (S. cerevisiae) and human PPIs except for the first benchmark, as it contains only human PPIs. We use Entrez and ORF gene ids for human and yeast, respectively, except for comparing with TCSS where UniProtKB and SGD gene ids were used for human and yeast, respectively. We have not performed the comparison with TCSS on the second and third benchmarks (for human) as some UniProtKB ids (after mapping from Entrez ids) were not found in the annotation.

3.4.1 Evaluation metrics and baselines

This section introduces how and why each benchmark is used for evaluation. A brief outline and formulation of each metrics used are presented here.

Correlation with reference dataset from HIPPIE database

The HIPPIE database [103] integrates most of the publicly available PPI databases like BioGRID [154], DIP [155], HPRDS [156], IntAct [157], MINT [158], BIND [159], MIPS [160]. It also includes interactions from several manually selected studies. The HIPPIE score of a PPI is defined by considering the following parameters: the number of studies where the PPI was detected, the number and quality of the experimental techniques used to detect the PPI, and the number of non-human organisms where the PPI was reproduced. The authors of HIPPIE showed that their scoring scheme of interactions correlates with the quality of the experimental characterization. We use a reference dataset from HIPPIE database to evaluate different SSMs. Pearson correlation is calculated between the HIPPIE score and PPI confidence score obtained using an SSM.

ROC curve analysis

Similarity measures can be treated as binary classifiers to classify a given PPI as positive or negative with a reasonable cutoff similarity score. PPIs having similarity score greater than the cutoff are treated as positive. Receiver operating characteristic (ROC) curve analysis is used to evaluate the performance of a binary classifier. ROC curve is a graph plotting of true positive rate (TPR or sensitivity) against false positive rate (FPR or 1-specificity) by varying discrimination threshold or cutoff. The area under the ROC curve (AUC) is the measure of discrimination, i.e., the ability of the classifier to classify correctly. An AUC of 1 represents perfect classifier. We utilize the core subsets of yeast and human PPIs from the DIP database [104] to evaluate different SSMs for AUCs.

Set-discriminating power of KEGG pathways

A biological pathway is a sequence of biochemical steps to accomplish a specific biological process within a cell. Therefore proteins involved in a pathway are more likely to interact among themselves than the proteins belonging to different pathways. Proteins within a pathway are likely to be annotated with the same or similar terms in GO too and should show high similarity scores. We consider three sets of selected KEGG pathways [105] to

evaluate different SSMs for their discriminating power as discussed in the following paragraph.

For each KEGG pathway, an *intra-set average similarity* is calculated as the average of all pairwise similarities of proteins within the pathway. An *inter-set average similarity* for every two pathways is also calculated as the average of all pairwise SSMs of proteins between the two pathways. During the calculation of *inter-set average similarity*, we do not consider those pairs whose both the proteins are same. A discriminating power (DP) of a pathway is defined in [161] as the ratio between *intra-set average similarity* and the average of all *inter-set average similarities* between the chosen pathway and rest other pathways. Let $\mathcal{P} = \{P_1, P_2, ..., P_n\}$ be the set of KEGG pathways, each pathway P_k contains m_k number of proteins and p_{ki} denotes i^{th} protein in P_k.

$$Intra\_set\_avg\_sim(P_k) = \frac{\sum_{i=1}^{m_k} \sum_{j=1}^{m_k} SSM(p_{ki}, p_{kj})}{m_k^2}. \tag{3.23}$$

$$Inter\_set\_avg\_sim(P_k, P_l) = \frac{\sum_{i=1}^{m_k} \sum_{j=1}^{m_l} SSM(p_{ki}, p_{lj})}{m_k \times m_l}. \tag{3.24}$$

$$DP(P_k) = \textit{Intra-set average similarity} \text{ of } P_k /$$

$$\text{Avg. of all } \textit{inter-set average similarities} \text{ between } P_k$$

$$\text{and other pathways}$$

$$= \frac{(n-1) \times Intra\_set\_avg\_sim(P_k)}{\sum_{i=1, i \neq k}^{n} Inter\_set\_avg\_sim(P_k, P_i)}. \tag{3.25}$$

Correlation with protein family (Pfam)

A protein family (Pfam) is a group of proteins that are evolutionarily related, i.e., they share a common evolutionary ancestor. Proteins belonging to a family often show functional similarity. The Jaccard index is used to calculate Pfam similarity. The Jaccard index of two proteins is calculated as the ratio of the number of protein families they share to the total number of protein families they belong. We utilize dataset of protein pairs used in CESSM [106]. For each pair, Pfam similarity (Jaccard index) and similarity scores of different SSMs are calculated, and finally, the Pearson correlation between the two scores is obtained.

3.4.2 Datasets

In this section, we describe the sources of different datasets used in the evaluation and the corresponding preprocessing steps. A summary of the datasets used is presented in Table 3.1.

Reference dataset from HIPPIE database

We download Human Integrated Protein-Protein Interaction rEference (HIPPIE) dataset on 09.01.2015 [103]. We extract one reference dataset from HIPPIE consisting of PPIs detected by four top-scored experimental techniques: far-Western blotting, isothermal titration calorimetry, nuclear magnetic resonance, and surface plasmon resonance experiments as in [162]. The interaction detected by any of the chosen four experimental techniques has a high probability of being an actual interaction [162]. The number of PPIs present in the reference datasets is shown in Table 3.1.

Table 3.1: A summary of the datasets used in the evaluation. The fourth column indicates the number of PPIs remains in HIPPIE or DIP datasets, the number of protein pairs remains in the Pfam dataset, and the length of KEGG pathways considered in the evaluation after necessary preprocessing.

Benchmark datasets	**Species**	**Ontology**	**Number of PPIs or protein pairs or length of pathways**
HIPPIE	Human	BP	1748
		CC	1757
DIP	Yeast	BP	4962
		CC	4992
	Human	BP	4279
		CC	4283
Pfam	Yeast	BP	366
		CC	351
	Human	BP	1212
		CC	1211
KEGG	Yeast Set-1	-	11 - 14
	Yeast Set-2		Specified in Table 3.2
	Human	-	11 - 16

Datasets for ROC curve analysis

We download the core subsets of PPIs from the Database of Interacting Proteins (DIP) [104] for S.cerevisiae and H.sapiens on 29.10.2015. DIP is a database of experimentally detected

PPIs from various sources. We assume that these interactions are real and treat them as positive instances of interactions. DIP uses UniProt Ids for proteins. We map UniProt Ids into Entrez and ORF gene Ids for human and yeast, respectively. Table 3.1 shows the number of PPIs of DIP dataset used in this study. As done in [99], an equal number of negative PPI datasets are independently generated by randomly choosing protein pairs annotated in BP and CC, and are not present in the iRefWeb database [163] (version date: 27.11.2015), a combined database of all known PPIs.

KEGG pathways

We extract two sets of KEGG pathways [105] of each of the two organisms, S.cerevisiae and H.sapiens, using org.Sc.sgd.db and org.Hs.eg.db packages with R 3.1.2 version. The first set contains a number of genes between 11 to 14 and the second set 11 to 16. We choose the above ranges so that each set contains the same (11) number of pathways and takes a reasonable time to compute. The two sets have three common pathways: Terpenoid backbone biosynthesis (sec00900 and hsa00900), Riboflavin metabolism (sec00740 and hsa00740), and Pantothenate and CoA biosynthesis (sec00770 and hsa00770). However, each of them is from different organisms and may not show similar results. Another set of 11 yeast KEGG pathways (Table 3.2) with more diverse functionality is also considered to get a broader insight into the inter-set discriminating power.

Table 3.2: The list of 11 yeast KEGG pathways with more diverse functionality used in the study. The number of genes is based on the org.Sc.sgd.db R package with version 3.1.2 (March 2015 release).

Category	Subcategory	Pathway Id	Pathway Name	No. of Genes
Metabolism	Carbohydrate metabolism	sce00040	Pentose and glucuronate interconversions	10
	Energy metabolism	sec00920	Sulfur metabolism	15
	Lipid metabolism	sec00565	Ether lipid metabolism	5
	Amino acid metabolism	sec00360	Phenylalanine metabolism	9
	Glycan biosynthesis and metabolism	sec00514	Other types of O-glycan biosynthesis	13
	Metabolism of cofactors and vitamins	sec00750	Vitamin B6 metabolism	11
	Metabolism of terpenoids and polyketides	sec00900	Terpenoid backbone biosynthesis	13
	Metabolism of other amino acids	sec00410	beta-Alanine metabolism	8
Genetic Information Processing	Folding, sorting and degradation	sec04122	Sulfur relay system	8
	Replication and repair	sec03450	Non-homologous end-joining	10
Environmental Information Processing	Signal transduction	sec04070	Phosphatidylinositol signaling system	15

CESSM dataset for correlation with Pfam

The Collaborative Evaluation of GO-based Semantic Similarity Measures (CESSM) is an online tool for evaluation of GO-based SSMs against sequence, Pfam and EC similarities [106]. Since CESSM has been published around ten years ago, it uses ten years old dataset (August 2008 GO and GOA-UniProt). In the meanwhile, GO DAG, its annotation, as well as Pfam have substantially changed. Moreover, we use GO.db (version:3.1.2) and org.Hs.eg.db (version:3.1.2) R packages that utilize March 2015 GO and annotations, respectively, in the evaluation. Hence we could not use the CESSM automated tool. However, we utilize the dataset of protein pairs used in CESSM to find correlation against Pfam similarity only, since GO captures the functional aspect of gene or gene products primarily. All pairs of proteins are mapped into Entrez and ORF gene Ids for human and yeast, respectively. The dataset involves 13,430 protein pairs of 1,039 proteins from various species. The authors of CESSM reported that both proteins of each pair are manually annotated to at least one term within all the three GOs with a uniform IC of at least 0.5 and have at least one EC class and one Pfam class. The number of protein pairs used for this evaluation is shown in Table 3.1.

3.4.3 Implementation

The new SSMs are implemented in the R programming language [164]. We use GOSemSim R package (version: 1.26.0) [165] for implementations of Resnik, Lin, Rel, Jiang, and Wang SSMs. For GO and corresponding annotations, we use GO.db, org.Sc.sgd.db (for yeast), and org.Hs.eg.db (for human) R packages (version:3.1.2, March, 2015 release) [166, 167, 168]. We maintain versions of all R packages so that they use the same GO and corresponding annotations. For TCSS, we use the implementation provided by the authors with the default set of parameters. The original implementation of TCSS uses MAX strategy only. Therefore we modify it to include BMA strategy as well. The implementation of TCSS needs the ontology and annotation as text files provided by gene ontology consortium. Therefore we use the released version of GO (gene_ontology.obo) dated Mar 13, 2015. The same released version of GO is used in above R packages (version: 3.1.2) and annotation for yeast (gene_association.sgd) and human (gene_association.goa_human) released on Mar 17, 2015. We use ROC and ROCR R packages [169, 170] to plot the ROC curve and to calculate the area under ROC curves (AUC).

3.5 Results and discussion

In this section, we report, analyze, and discuss the results. For each key observation, first, we summarize the conclusion (in bold) with the starting phrase and the rest of the paragraph introduce the discussion endorsing the conclusion.

3.5.1 Correlation with reference dataset from HIPPIE database

Performance, in terms of Pearson correlation, of different SSMs with respect to the reference dataset from HIPPIE, is shown in Table 3.3. The best correlations are shown in bold.

Table 3.3: The Pearson correlation with the reference dataset extracted from the HIPPIE database by considering the four top-scored experimental techniques: far-Western blotting, isothermal titration calorimetry, nuclear magnetic resonance, and surface plasmon resonance experiments. The best correlation for each ontology and strategy is shown in bold.

Ontology	Strategy	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
BP	MAX	**0.358**	0.313	0.346	0.342	0.329	0.277	0.277	0.272	0.286
	BMA	**0.342**	0.332	0.310	0.270	0.238	0.220	0.218	0.211	0.193
CC	MAX	0.204	0.130	0.129	**0.232**	0.231	0.064	0.100	0.064	0.082
	BMA	**0.254**	0.227	0.198	0.232	0.230	0.148	0.164	0.118	0.158

RDS achieves the highest correlation in BP, while TCSS shows the maximum correlation in CC. It may be noted that RDS is the simplest SSM among the proposed measures and does not even consider annotation information. Nevertheless, it shows good correlation. RNS and RES also perform quite well in BP, while Resnik shows good performance in both BP and CC.

All SSMs show greater correlations in BP. The average correlation over all SSMs in BP is 0.311/0.259 (MAX/BMA), whereas, in CC, it is 0.137/0.192 (MAX/BMA). However, all measures show less overall correlation since correlation is computed for positive PPIs only.

3.5.2 ROC curve analysis

AUCs obtained by different SSMs are tabulated in Table 3.4. The best ROC score for each ontology and strategy is shown in bold.

RNS and RES, with both MAX and BMA strategies, effectively classify true PPIs from false in both BP and CC. Resnik-MAX and Rel-MAX too perform well compared to others, while RDS shows competitive performance. Although we could not compare TCSS for human, it performs well with MAX strategy in yeast. All SSMs with MAX strategy have quite similar AUCs in BP for both yeast and human. However, with BMA strategy, AUCs

Table 3.4: The area under the ROC curves of different SSMs for the core subsets of yeast and human PPIs extracted from the DIP database. The result of TCSS with the human is not shown as some UniProtKB ids (after mapping from Entrez ids) are not found in the corresponding annotation corpus. The best ROC score for each ontology and strategy is shown in bold.

Species	Ontology	Strategy	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
Yeast	BP	MAX	0.896	0.908	0.903	0.907	0.908	0.912	**0.914**	0.910	0.895
		BMA	0.868	0.890	**0.893**	0.861	0.879	0.881	0.883	0.874	0.860
	CC	MAX	0.856	0.868	0.850	0.866	**0.870**	0.804	0.868	0.771	0.799
		BMA	0.826	0.848	0.843	0.831	**0.850**	0.805	0.838	0.709	0.783
Human	BP	MAX	0.907	**0.914**	0.904	-	0.908	0.900	0.913	0.887	0.895
		BMA	0.892	**0.903**	0.898	-	0.872	0.865	0.869	0.817	0.867
	CC	MAX	0.848	0.847	0.857	-	0.852	0.794	**0.858**	0.795	0.800
		BMA	0.824	**0**.849	**0.850**	-	0.814	0.773	0.791	0.708	0.791

achieved by RES (yeast:0.893, human:0.898) and RNS (yeast:0.890, human:0.903) are significantly higher than others. Further, RES and RNS exhibit greater consistency, since they show less difference between MAX and BMA strategies in both BP and CC (for both yeast and human).

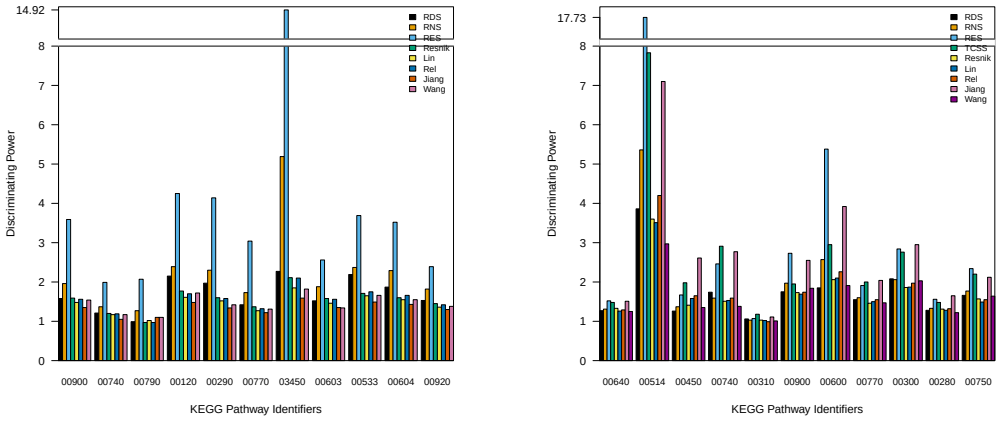
All SSMs show higher AUCs in BP. The average AUCs in BP are 0.906/0.877 (yeast:MAX/BMA) and 0.904/0.873 (human:MAX/BMA), whereas in CC these are 0.839/0.815 (yeast:MAX/BMA) and 0.831/0.800 (human:MAX/BMA).

We also perform statistical analysis of ROC curves using the StAR online tool [171] to see if the difference in AUCs for two different SSMs is statistically significant. We observe that the differences are statistically significant for most of the cases. For example, in BP ontology with BMA strategy for the yeast model organism (second row in Table 3.4), the p-value of the test between the best performing SSM (RES) and the second-best performing SSM (RNS) is 0.00458231.

3.5.3 Set-discriminating power of KEGG pathways

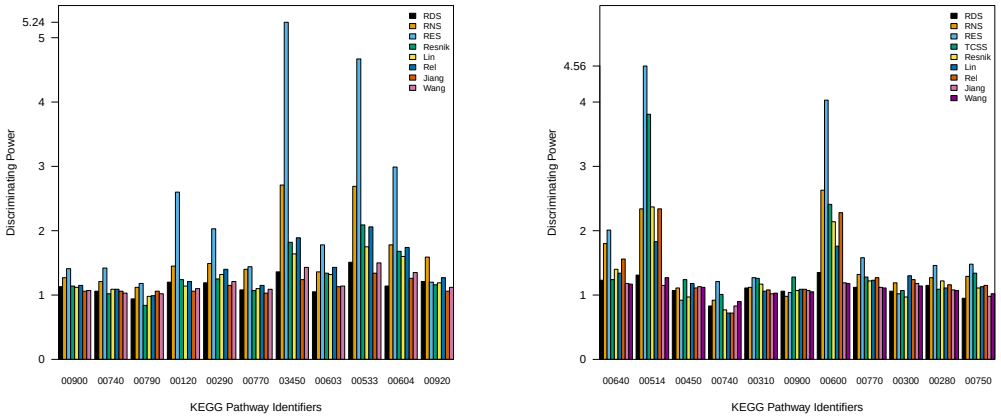
As discussed earlier, the discriminating power quantifies the ability of an SSM to distinguish among various functionally different sets of proteins (e.g., KEGG pathways). Figure 3.3 and 3.4 demonstrate the discriminating power of different SSMs with BMA strategy against KEGG pathways in BP and CC ontology, respectively. Instead of pathway names, KEGG pathway identifiers are shown along the x-axis. The discriminating power for the selected yeast KEGG pathways (listed in Table 3.2) with more diverse functionality is shown in Figure 3.5. The results with MAX strategy are quite similar; hence, they are not reported.

The discriminating power of RES is significantly higher than other SSMs for all the 11 human KEGG pathways. RES produces DP value greater than or equal to 1.81/1.99



(a) Human (b) Yeast

Figure 3.3: The *inter-set discriminating power* of different SSMs with BMA strategy in BP ontology. The y-axis is splitted to accommodate high DP value.

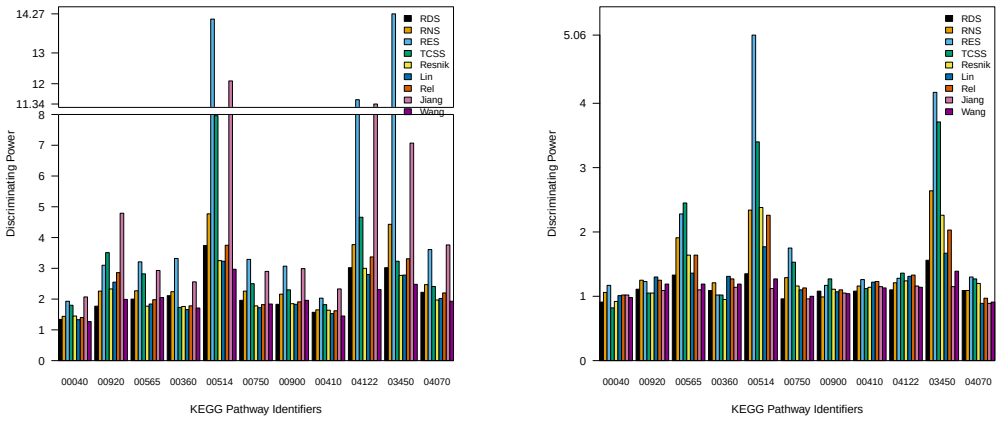


(a) Human (b) Yeast

Figure 3.4: The *inter-set discriminating power* of different SSMs with BMA strategy in CC ontology.

(MAX/BMA) in BP, while the next minimum DP value is 1.17 (produced by RDS - MAX).

RES shows maximum functional discrimination among the pathways. RES produces very high DP value with 11.10/14.92 (MAX/BMA) for *Non-homologous end-joining* (hsa03450) pathway. This is the only pathway that belongs to the *Genetic Information Processing* category, while rest fall in the same *Metabolism* category. So, the functional characteristic of *Non-homologous end-joining* pathway is completely different from the rest. RES nicely captures this functional discrimination by producing very high DP value.



(a) BP ontology

(b) CC ontology

Figure 3.5: The *inter-set discriminating power* of different SSMs with BMA strategy for the selected 11 yeast KEGG pathways with more diverse functionality.

All SSMs produce greater DP values in BP. Although RES almost consistently produces higher DP values in both BP and CC (with both MAX and BMA), it shows comparatively lower DP values in CC.

The overall discriminating power of all the SSMs is quite similar and not so good for the first set of yeast KEGG pathways. If we examine the functional categories of all the 11 pathways, we find that all belong to the same *Metabolism* category with six pathways from two subcategories only. Further, the selected first set of yeast pathways contain merely 134 genes with 16 are shared. In contrast, the selected human pathways include 150 genes with 11 are common only. Hence the selected first set of yeast pathways are functionally closer to each other, and this fact is reflected by low DP values.

To study further, we consider another set of 11 yeast pathways with more diverse functionality, where three pathways (sec00514, sec00750, and sec00900) were taken from the previous set. The pathways are listed in Table 3.2, and corresponding discriminating power for BMA strategy is shown in Figure 3.5.

The discriminating power of all the SSMs is improved significantly for the pathways with more diverse functionality. In particular, the DP values of RES and Jiang are higher than other measures for almost all the pathways. RES and Jiang produce DP value greater than or equal to 2/1.93 (MAX/BMA) and 1.84/2.07 (MAX/BMA), respectively, in BP, while the next minimum DP value is 1.73 (produced by TCSS - BMA). The maximum DP value (MAX/BMA:13.75/14.27 in BP) is again produced by RES for the pathway sec03450 (Non-homologous end-joining).

RES can be used for functional clustering. It may be noted that although Jiang produces competitive DP values with RES for yeast pathways, it is unable to show good DP values for the human pathways. Therefore RES might be used for functional clustering (e.g., to characterize protein functional modules) as it shows consistently high discriminating power.

No SSM produces consistently good DP values in CC, particularly for the yeast pathways. Guo et al. [100] observed that all pairs of proteins involved in the same KEGG pathway have significantly higher similarity scores than randomly selected in BP, whereas similarity decreases exponentially as the distance between two proteins increases within the same pathway in CC and MF. These findings conform with current results as well.

3.5.4 Correlation with Pfam

Finally, Table 3.5 demonstrates the performance of different SSMs on Pfam. The best scores are shown in bold.

Table 3.5: The Pearson correlation of different SSMs with protein family (Pfam) on CESSM dataset. The Jaccard index is used to calculate Pfam similarity as in CESSM. The best score for each ontology and strategy is shown in bold.

Species	Ontology	Strategy	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
Yeast	BP	MAX	0.280	**0.324**	0.283	0.290	0.304	0.308	0.314	0.268	0.302
		BMA	0.306	**0.347**	0.310	0.279	0.307	0.296	0.299	0.272	0.264
	CC	MAX	0.240	0.202	0.252	**0.259**	0.243	0.156	0.183	0.123	0.139
		BMA	0.218	0.204	**0.233**	0.204	0.225	0.226	0.225	0.205	0.201
Human	BP	MAX	0.158	0.157	0.160	0.258	**0.300**	0.152	0.156	0.143	0.156
		BMA	0.231	0.290	0.308	**0.347**	0.302	0.263	0.262	0.258	0.293
	CC	MAX	0.308	0.233	**0.390**	0.314	0.307	0.193	0.223	0.159	0.198
		BMA	0.356	0.383	**0.471**	0.437	0.347	0.349	0.365	0.269	0.349

Overall performances of TCSS, RES, and Resnik are well. Particularly, TCSS - MAX, RES - BMA, and Resnik - MAX perform well. Although RES does not show good correlation with MAX strategy in human, it produces a good correlation with BMA strategy. MAX strategy could overestimate while computing the general measure of functional similarity [99] and protein family captures a general aspect of protein function. Thus, BMA might be a better choice than MAX for Pfam similarity.

Further, it may be noted that the correlation in CC is higher than BP in human for all measures, which are quite unexpected. Therefore it might be challenging to draw comparative inference for the benchmark like Pfam that adopt a very general aspect of protein function with Jaccard index.

3.6 Conclusion

The chapter presents a new family of SSMs for scoring confidence of PPIs utilizing GO. This new family of SSMs is based on a new set of specificity measures namely, RDS, RNS and RES. The specificity of a term is redefined by considering the properties of its ancestors and descendants only along with its own properties so that maximum unwanted noises could be avoided. The evaluation shows that instead of simplicity, they are quite effective. Particularly, RNS and RES more effectively distinguish true interactions from false. RES can be useful for protein functional clustering as well since it shows a robust set-discriminating power over KEGG pathways. It also exhibits greater consistency and shows the best performance in BP with BMA strategy. Similar to the earlier studies, our evaluation also shows Resnik is one of the best SSMs for scoring confidence of PPIs. TCSS with MAX strategy and Rel also show competitive performance. Although RDS is the simplest SSM that does not even consider annotation information, it shows competitive performance as well. For almost all the four benchmarks, each SSM shows comparatively greater and consistent performances in BP. Therefore we believe that BP is more suitable than CC for scoring confidence of PPIs.

Availabilty of data and script

An R script for the new SSMs along with the complete datasets used in the evaluation is freely available at https://github.com/msp-cse/NaiveSSMs.

Chapter 4

Impact of the continuous evolution of gene ontology on the performance of similarity measures

In the previous chapter, we have discussed how GO can be used to define a semantic similarity measure (SSM) between two gene products and assign a confidence score to protein-protein interactions (PPIs). However, GO is being updated regularly with the addition, deletion, and merging of terms along with their annotations. Consequently, the similarity score of interaction may differ from one instance of GO to another. To the best of our knowledge, there is no such study that evaluates the impact of the continuous evolution of GO on the performance of SSMs.

In this chapter, we systematically study the robustness of SSMs over the evolution of GO for the task of scoring confidence of PPIs. By robustness of an SSM, we mean it should either improve or keep its performance similar over the evolution of GO. We consider nine state-of-the-art SSMs, including our proposed ones in the previous chapter and nine different Bioconductor versions of GO. We observe that the performance of similarity measures gets affected due to the regular updates of GO. We further observe that the degree of robustness of a similarity measure is highly influenced by the particular setting we consider.

Some fundamental concepts of this chapter have a natural overlapping with the previous chapter. However, for the sake of completeness, a brief discussion of those concepts are given at appropriate places.

4.1 Introduction

Gene Ontology (GO) [11] is a comprehensive resource for the properties of gene products and their relationships. It is a taxonomy of biological terms to represent the properties of genes and/or gene products (e.g., proteins). It is organized as a directed acyclic graph to describe the relationship among the terms. Gene products are annotated to pertinent GO terms through annotation corpora. There are three GOs: biological process (BP), cellular component (CC), and molecular function (MF). Since gene products are not directly represented in GO, annotation corpora are used to link between a gene product and a GO term. An annotation corpus of a species (e.g., yeast) is an association between gene products of the species and GO terms. Ontology-based *semantic similarity measure* (SSM) is a quantitative function, $SSM(t_1, t_2)$, that measures the closeness between two terms t_1 and t_2 based on their semantic representations in a given ontology. Mathematically, it is a function of two ontology terms (or two sets of ontology terms) that returns a numeric value reflecting the closeness between them in the context of semantic meaning [87]. SSMs were originally defined in the study of linguistics. Lord *et al.* [123] did the first pioneering work by utilizing the ontology-based SSM in the field of genomics. Subsequently, a wide range of GO-based SSMs has been developed and successfully applied to different genomics applications [99, 102].

The high similarity score between two proteins indicates that either they are annotated with similar CC and/or BP terms. This shows an indirect evidence that the two proteins are likely to be interacting compare to other pairs, which has a low similarity. Hence several studies have utilized GO-based SSMs to score the confidence of *protein-protein interactions* (PPIs). However, GO is being updated regularly with the addition of new terms and deletion/merging of obsolete terms along with their annotations. As a consequence, the similarity score of a protein-pair may get changed over different instances of the ontology.

In this work, we systematically study whether changes in GO affect the performance of GO-based SSMs. We further compare the state-of-the-art GO-based SSMs under different settings for scoring confidence of PPIs. We consider nine *Bioconductor* versions of GO and its annotations and nine state-of-the-art GO-based SSMs in the evaluation. We utilize two evaluation metrics: ROC curve analysis with two commonly used experimental PPI datasets and set-discriminating power of KEGG pathways, for S. cerevisiae (yeast) model organism.

Section 4.2 briefly discusses the related work. In section 4.3, we discuss experimental design, datasets, and evaluation metrics. Results are reported, discussed, and analyzed in section 4.4. Finally, we conclude the chapter in section 4.5.

4.2 Related work

Originally, SSMs have been studied in the fields of Natural Language Processing, Artificial Intelligence, Cognitive Science, and Psychology. After the first pioneering work by Lord *et al.* [124, 123], a wide range of GO-based SSMs has been developed for different genomics applications. In this section, we present a brief outline of GO-based SSMs.

As discussed in the previous chapter (Section 3.2.2), SSMs can be classified mainly into two categories: *edge- and node-based* [87]. The first edge-based SSM was introduced in a lexical taxonomy by Rada *et al.* [122]. Subsequently, it was applied to GO by Nagar and Al-Mubaid [125]. Thereafter a variety of edge-based SSMs have been developed and used in GO [129, 131, 130, 128]. This approach mainly considers the shared paths between two ontology terms. It does not account for annotation information of terms and entirely relies on the intrinsic structure of the GO DAG.

The most widely used SSMs are node-based that computes the similarity between two terms by comparing their properties, common ancestors, and their descendants. As discussed in The notion of *Information Content* (IC) (See section 3.2.2) is commonly used to define node-based SSMs. The IC quantifies specificity of a term in an ontology. The specificity decreases as we move towards the root. Therefore the highest and lowest IC would be at leaves and the root, respectively. A node-based approach assumes that if two proteins are annotated with common specific terms, they are likely to be similar in their functions.

The most widely used node-based SSMs are Resnik [94], Lin [95], and Jiang & Conrath [97], which were originally developed for WordNet (a large lexical database of English) and subsequently applied to GO [124, 123]. Resnik considers IC of the *lowest common ancestor* (LCA) between the two terms s and t. It is quite effective to compute information shared by two terms. However, it cannot distinguish among pairs of terms having the same LCA.

To overcome the problem, Lin and Jiang are developed by considering the information contents of both terms, along with their LCA in different ways. Both Lin and Jiang overestimate when one term is an ancestor of another. For instance, the similarity score will be 1 when both the terms are the same, irrespective of its specificity. In order to capture relevance information, Resnik and Lin are combined into Rel by multiplying one minus probability of annotation of LCA to Lin.

Over the years, a number of node-based methods have been developed in order to improve the existing SSMs in different perspectives and applications such as [142, 133, 143, 136, 138, 135, 139, 137, 132, 140, 134, 141]. The IC-based SSMs are less sensitive to the aforementioned issues related to edge-based SSMs. Similarity measures such as [127, 145, 98, 144] try to combine both node- and edge-based approaches and are referred

to as hybrid approaches. Some methods have been developed by considering the complex structure of GO DAG such as TCSS [99], and [147, 146]. Refer to section 3.2.4 for detailed formulations of all the SSMs used in the evaluation.

A gene product may be annotated with more than one term in the same GO. Suppose, p_1, and p_2 are two gene products annotated to the set of terms S and T, respectively. The similarity between p_1 and p_2 is calculated as the similarity between two sets S and T, i.e., $SSM(p_1, p_2) = SSM(S, T)$. Therefore we need to combine GO terms of S and T. Three types of strategies are commonly used in the literature: maximum (MAX) [148], average (Avg) [124, 123] and best-match average (BMA) [150, 149, 96]. In the MAX strategy , the similarity between S and T is calculated as the maximum of the set $S \times T$. In the 'average' strategy, the similarity between S and T is calculated as the average of the set $S \times T$. SSMs between two sets of terms form a matrix, and BMA is defined as the average of all maximum similarity scores on each row and column of the matrix. Refer to section 3.2.3 for detailed formulations of all the three combining strategies.

4.3 Experimental setup

We consider BP and CC ontologies along with MAX and BMA in the evaluation. As discussed in the previous chapter (See section 3.4), these ontologies and strategies are the most relevant for scoring confidence of PPIs. We exclude electronically inferred annotations (IEA) as they are not verified by human experts. Further, we consider only those PPIs where both the interacting proteins are annotated to at least one GO term other than the root.

We select the following nine Bioconductor versions of GO and corresponding annotation corpora: 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6 3.7, and 3.8 released in Sep 2014, Mar 2015, Oct 2015, Apr 2016, Oct 2016, Apr 2017, Sep 2017, Apr 2018, and Oct 2018, respectively, We consider six state-of-the-art SSMs proposed by Resnik [94], Lin [95], Schlicker *et al.* [96], Jiang & Conrath [97], Wang *et al.* [98], and Jain & Bader [99], referred to as *Resnik*, *Lin*, *Rel*, *Jiang*, *Wang*, and *TCSS*, respectively, in the rest of the chapter. The detail formulation of these SSMs is given in the previous chapter (Section 3.2.4). Resnik and TCSS with MAX strategy have been considered to be the best SSMs for scoring confidence of PPIs by several studies [99, 102]. We also consider the new SSMs - RDS, RNS, and RES, proposed in the previous chapter. The selected nine SSMs encompass all types of SSMs, as discussed in section 4.2.

4.3.1 Datasets

Experimental PPIs are usually considered to validate an SSM and Database of Interacting Proteins (DIP) [104] is one of the frequently used experimental PPI databases for the purpose. We utilize the core subset of the yeast PPIs from the DIP database downloaded on 29.10.2015 as positive instances. To make the evaluation scheme more robust, we select another experimental, combined-AP/MS PPI dataset (of S. cerevisiae) [70], which is another experimental dataset commonly used for the same purpose. As done in [99], an equal number of negative PPI instances are generated independently by randomly choosing protein pairs annotated in BP and CC and are not present in the iRefWeb database [163], a combined database of all known PPIs.

Proteins involved in a pathway are more likely to interact among themselves and likely to be annotated to the same or similar GO terms and thus should show high similarity scores. We consider the same set of 11 yeast (S. cerevisiae) KEGG pathways utilized in the previous work (See section 3.4.2). During the selection of pathways, we try to maintain a trade-off between functional diversity and computational time required for the experiment.

4.3.2 Evaluation metrics

A similarity measure can classify a set of PPIs into two groups: positives and negatives, for a given cutoff similarity score. Hence an SSM can be treated as a binary classifier. We utilize the area under the ROC curve (AUC) as an evaluation metric for binary classifiers, as discussed in the previous chapter (3.4.1).

For each KEGG pathway, an *intra-set average similarity* is calculated as the average of all pairwise similarities of proteins within the pathway. An *inter-set average similarity* for every two pathways is calculated as the average of all pairwise cross-similarities of proteins between the two pathways. As discussed in the previous chapter (3.4.1), a discriminating power (DP) of a pathway is defined as the ratio between *intra-set average similarity* and the average of all *inter-set average similarities* between that pathway and other pathways as in [161]. Thus the DP quantifies the ability of an SSM to distinguish among various functionally different sets of proteins (e.g., KEGG pathways).

4.4 Results and discussion

In this section, the results are reported, discussed, and analyzed. The section is divided into two subsections for the two evaluation metrics mentioned earlier. The key observations are

highlighted.

4.4.1 ROC curve analysis

Table 4.1 summarizes the AUC of SSMs for the different versions of BP ontology with the DIP dataset. At first glance, it seems that the evolution of GO has no impact on the classification performance of SSMs as the AUC values achieved by them are quite similar among the different GO versions. The only significant change is observed between GO-BP Ver3.4 and Ver3.5 by RES and TCSS with BMA strategy. It seems that there may be a substantial change between these two GO-BP versions. An AUC of 1 represents a perfect classifier, whereas an area of 0.5 represents a random classifier. So, the practical range of AUC for a reasonably good classifier is very limited (Generally, [0.7,1]). Unless the majority of the PPIs get affected (due to the changes in GO), it is unexpected to observe high variability in AUCs over the different versions of GO. By affected we mean for a given PPI, an SSM produces different similarity scores for different GO versions. In fact, only a few PPIs (in the PPI dataset) gets affected significantly due to the changes in GO.

Table 4.1: The area under the curves (AUCs) of SSMs for the different GO-BP versions (Ver3.0 to Ver3.8) using the DIP-core dataset. The best AUC for each strategy is shown in bold.

SSM	Str	V3.0	V3.1	V3.2	V3.3	V3.4	V3.5	V3.6	V3.7	V3.8	Mean
RDS	MAX	0.901	0.899	0.896	0.900	0.901	0.906	0.904	0.902	0.903	0.901
	BMA	0.875	0.875	0.870	0.873	0.876	0.880	0.875	0.875	0.875	0.875
RNS	MAX	0.907	0.909	0.908	0.913	0.912	0.915	0.915	0.913	0.912	0.912
	BMA	0.892	0.896	0.893	0.899	0.898	0.903	0.901	0.900	0.900	0.898
RES	MAX	0.902	0.904	0.902	0.907	0.905	0.909	0.909	0.908	0.908	0.906
	BMA	**0.893**	**0.897**	**0.896**	**0.900**	**0.899**	0.935	0.932	0.932	0.932	**0.913**
TCSS	MAX	0.908	0.909	0.909	0.913	0.911	0.914	0.913	0.917	0.914	0.912
	BMA	0.864	0.867	0.863	0.869	0.874	**0.966**	**0.962**	**0.966**	**0.964**	0.911
Resnik	MAX	0.909	0.909	0.908	0.913	0.911	0.915	0.914	0.914	0.912	0.912
	BMA	0.884	0.884	0.883	0.888	0.887	0.894	0.890	0.891	0.888	0.888
Lin	MAX	0.914	0.914	0.913	0.919	0.916	0.919	0.918	0.916	0.914	0.916
	BMA	0.886	0.886	0.885	0.889	0.888	0.894	0.891	0.890	0.888	0.889
Rel	MAX	**0.916**	**0.915**	**0.915**	**0.920**	**0.918**	**0.922**	**0.920**	**0.919**	**0.917**	**0.918**
	BMA	0.888	0.888	0.887	0.891	0.890	0.897	0.894	0.892	0.890	0.891
Jiang	MAX	0.912	0.912	0.912	0.916	0.914	0.911	0.911	0.907	0.907	0.911
	BMA	0.88	0.881	0.880	0.881	0.881	0.875	0.876	0.872	0.873	0.878
Wang	MAX	0.898	0.898	0.898	0.902	0.902	0.906	0.905	0.904	0.903	0.902
	BMA	0.867	0.868	0.867	0.871	0.873	0.876	0.873	0.873	0.873	0.871

To see the closer picture of the impact, we find those PPIs whose similarity scores change over the versions of GO. For each SSM, we select the common PPIs (more than 99% of PPIs are common) among the GO versions. For each of the selected PPIs, the standard deviation of the nine similarity scores corresponding to the nine GO versions is calculated. Then we

sort the PPIs according to their standard deviation (in descending order) and select the top 10% PPIs. Naturally, the selected PPIs will be the most affected 10% PPIs due to the changes in GO. An equal number of negative PPIs are selected from the already generated negative PPIs for the corresponding SSM. Finally, AUC is calculated for the selected positive and negative PPIs for each GO version. The aforementioned process is repeated for each SSM and strategy considered in the evaluation. Figure 4.1 demonstrates the flowchart of the process. The resultant AUCs of SSMs for the different GO versions are demonstrated in Table 4.2.

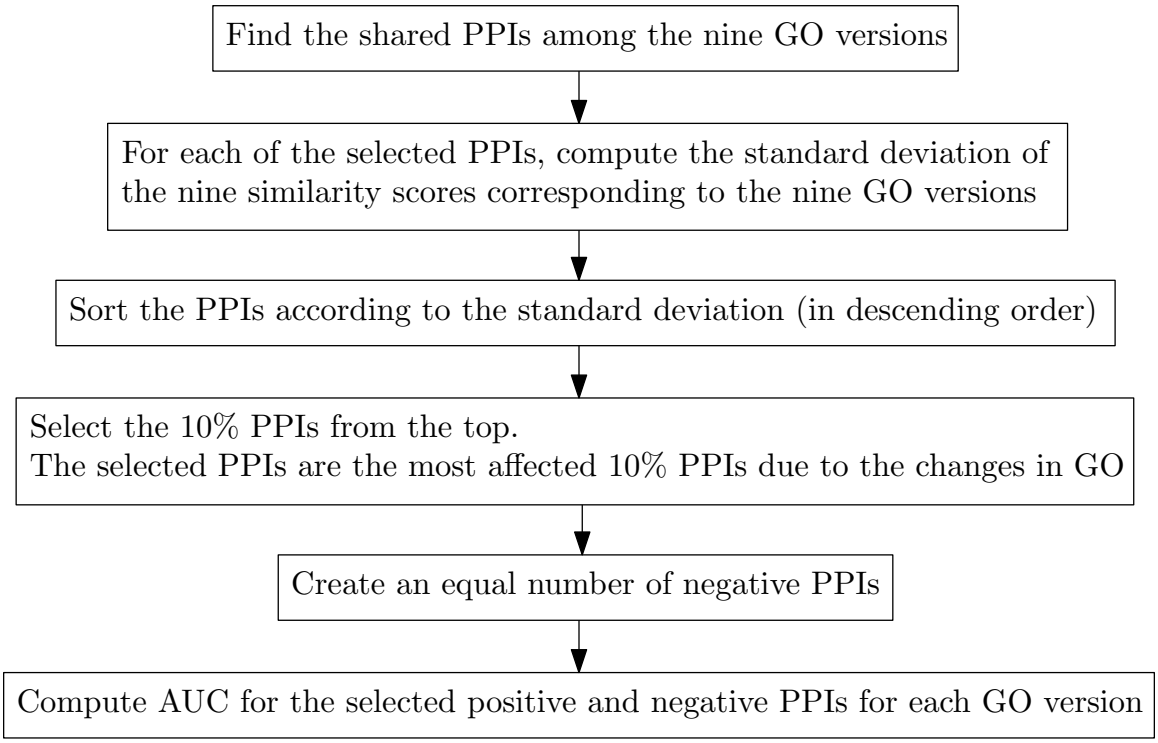


Figure 4.1: Flowchart of the process for computing AUC of an SSM.

Now, noticeable changes are observed among GO versions by most of the SSMs. The relative changes of approximately 7% and 10% are observed between GO-BP versions 3.0 and 3.9 for RES-MAX and RES-BMA, respectively. Similarly, the changes are 5% and 12% for TCSS-MAX and TCSS-BMA, respectively. Similar observations could be made for the other SSMs and/or other ontology. It is also observed that the overall variability is higher in CC than BP across all SSMs.

To find a general pattern of variability among SSMs, we repeat the aforementioned process for different cutoffs (100% to the top 10%) of affected PPIs. Here a cutoff of 100% implies that all PPIs are considered and hence the majority of them have no change in their similarity score. The mean AUCs (of nine GO versions) achieved by SSMs in increasing order of variability of PPIs are shown in Figure 4.2.

Table 4.2: The area under the curves (AUCs) of SSMs for the different GO versions (Ver3.0 to Ver3.8) with the top 10% most affected PPIs using the DIP-core dataset. The best AUC for each strategy (MAX and BMA) and ontology is shown in bold.

SSM	Ont	Str	V3.0	V3.1	V3.2	V3.3	V3.4	V3.5	V3.6	V3.7	V3.8	Mean
RDS	BP	MAX	0.864	0.859	0.848	0.885	0.895	0.903	0.907	0.904	0.911	0.886
		BMA	0.901	0.894	0.887	0.916	0.919	0.918	0.920	0.915	0.921	0.910
	CC	MAX	0.680	0.682	0.682	0.795	0.845	0.879	0.868	0.862	0.844	0.793
		BMA	0.828	0.821	0.832	0.880	0.904	0.930	0.922	0.926	0.911	0.884
RNS	BP	MAX	0.763	0.799	0.798	0.840	0.830	0.822	0.838	0.827	0.827	0.816
		BMA	0.870	0.887	0.887	0.926	0.914	0.915	0.919	0.910	0.906	0.904
	CC	MAX	0.655	0.651	0.643	0.760	0.797	0.826	0.817	0.814	0.798	0.751
		BMA	0.836	0.826	0.818	0.889	0.905	0.937	0.921	0.929	0.916	0.886
RES	BP	MAX	0.846	0.877	0.869	0.907	0.907	0.901	0.908	0.900	0.905	0.891
		BMA	**0.910**	**0.943**	**0.938**	**0.945**	**0.954**	0.994	0.995	0.995	**0.998**	**0.964**
	CC	MAX	0.790	0.789	0.770	0.866	0.901	0.926	0.938	0.936	0.930	0.872
		BMA	**0.911**	**0.906**	**0.888**	**0.944**	**0.950**	0.969	0.971	0.974	0.969	**0.942**
TCSS	BP	MAX	**0.891**	**0.900**	**0.902**	**0.940**	**0.930**	**0.919**	**0.912**	**0.956**	**0.932**	**0.920**
		BMA	0.891	0.910	0.889	0.903	0.919	**0.995**	**0.996**	**1.00**	**0.998**	0.945
	CC	MAX	**0.882**	**0.895**	**0.886**	**0.931**	**0.944**	**0.959**	**0.960**	**0.968**	**0.962**	**0.932**
		BMA	0.876	0.874	0.873	0.903	0.911	**0.997**	**0.997**	**0.998**	**0.994**	0.936
Resnik	BP	MAX	0.807	0.825	0.825	0.881	0.868	0.870	0.872	0.865	0.857	0.852
		BMA	0.858	0.864	0.866	0.889	0.879	0.875	0.873	0.872	0.869	0.872
	CC	MAX	0.749	0.754	0.750	0.843	0.883	0.919	0.909	0.911	0.894	0.846
		BMA	0.891	0.896	0.891	0.926	0.935	0.954	0.948	0.954	0.943	0.926
Lin	BP	MAX	0.712	0.727	0.731	0.780	0.771	0.757	0.758	0.752	0.748	0.748
		BMA	0.848	0.849	0.849	0.874	0.863	0.856	0.856	0.855	0.847	0.855
	CC	MAX	0.553	0.540	0.532	0.634	0.656	0.669	0.652	0.679	0.625	0.616
		BMA	0.768	0.751	0.735	0.783	0.778	0.782	0.775	0.794	0.762	0.770
Rel	BP	MAX	0.741	0.756	0.759	0.813	0.803	0.798	0.799	0.793	0.789	0.783
		BMA	0.850	0.853	0.853	0.881	0.871	0.866	0.866	0.866	0.857	0.863
	CC	MAX	0.657	0.649	0.641	0.748	0.785	0.805	0.795	0.809	0.777	0.741
		BMA	0.823	0.810	0.798	0.836	0.844	0.849	0.843	0.857	0.836	0.833
Jiang	BP	MAX	0.742	0.751	0.738	0.804	0.811	0.777	0.783	0.772	0.773	0.772
		BMA	0.859	0.865	0.852	0.885	0.885	0.858	0.864	0.854	0.854	0.864
	CC	MAX	0.411	0.419	0.402	0.534	0.563	0.552	0.545	0.579	0.534	0.504
		BMA	0.632	0.627	0.594	0.639	0.612	0.562	0.571	0.601	0.554	0.599
Wang	BP	MAX	0.744	0.758	0.750	0.804	0.822	0.801	0.817	0.814	0.822	0.792
		BMA	0.890	0.889	0.875	0.900	0.900	0.887	0.897	0.898	0.896	0.892
	CC	MAX	0.518	0.515	0.507	0.629	0.637	0.645	0.624	0.664	0.617	0.595
		BMA	0.733	0.720	0.704	0.751	0.718	0.713	0.720	0.735	0.699	0.721

SSMs with BMA strategy shows robustness compared to MAX strategy. Almost all SSMs with BMA strategy either improve or keep their performance similar from their initial performance as variability increases in both the ontologies. Particularly in BP, the improvement is more smooth and consistent. However, with MAX strategy, the performance is quite fluctuating, and the irregularity is more in CC. Therefore it seems that MAX strategy overestimates in many cases, especially in CC.

All SSMs exhibit higher robustness in BP than CC. If we examine the same for each SSM separately, we get further insights (See figure 4.3 and 4.4). Here, for an SSM, the four scenarios: BP-MAX, BP-BMA, CC-MAX, and CC-BMA, are plotted together. With the

full dataset considered (100%), some of the SSMs with MAX strategy produce higher AUCs than BMA. However, as variability increases (by filtering out least affected PPIs over GO evolution), SSMs with BMA strategy exhibit higher AUCs.

From the aforementioned observations, it is clear that the majority of the similarity measures exhibit good robustness with BMA strategy and show poor robustness with MAX strategy. Further, the robustness is more in BP than CC.

RES and TCSS with BMA strategy consistently produce high AUCs as variability increases. Overall, RES, TCSS, RNS, and Resnik show comparatively high robustness. With the top 10% variable PPIs, the highest mean AUC is $0.964/0.942$ (BP/CC) produced by RES-BMA while the second-highest mean AUC is $0.945/\ 0.936$ (BP/CC) produced by TCSS-BMA. The highest mean AUC values in different cutoffs (100% to the top 10%) of affected PPIs are shown in Table 4.3. From the table, we can see that RES-BMA consistently produces the highest mean AUC values, particularly in BP.

Further, we consider another commonly used experimental, combined-AP/MS PPI dataset (of S. cerevisiae) in the evaluation. Here, we consider Ver3.4 to Ver3.8 only. In Figure 4.5, the mean AUCs (of five GO versions) achieved by SSMs in increasing order of variability of PPIs are demonstrated. It seems that the results are quite similar to the experiment with the DIP-core PPI dataset.

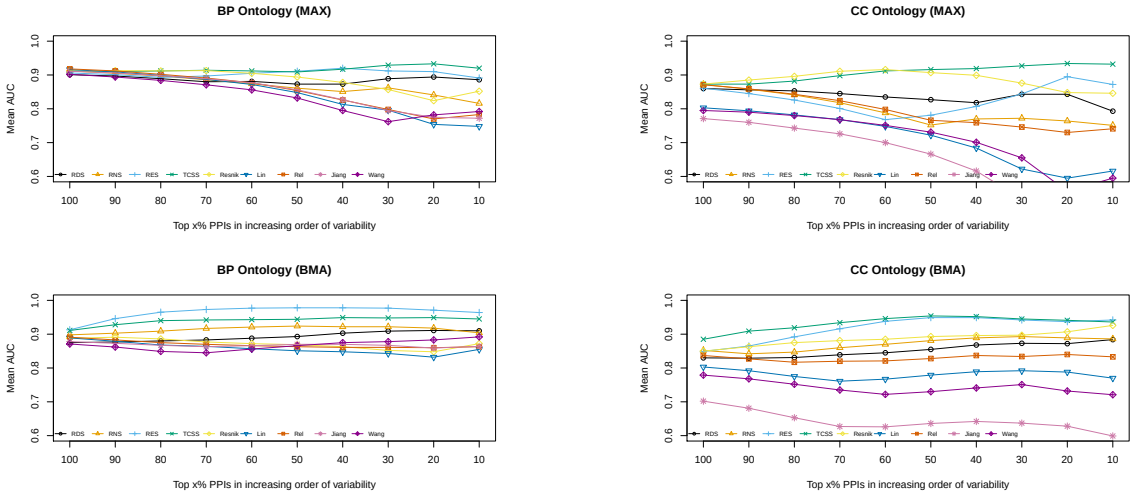


Figure 4.2: The mean AUCs of nine GO versions (Ver3.0 to Ver3.8) achieved by SSMs at different cutoffs of affected PPIs for the DIP-core dataset.

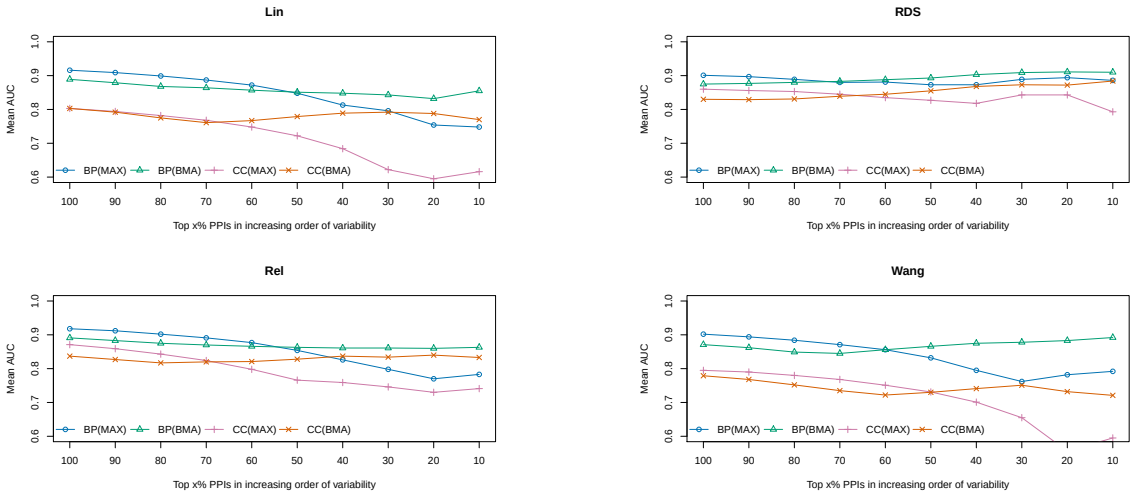


Figure 4.3: The mean AUCs of nine GO versions (Ver3.0 to Ver3.8) achieved by Lin, Rel, RDS, and Wang at different cutoffs of affected PPIs with the plotting of individual SSM for the DIP-core dataset.

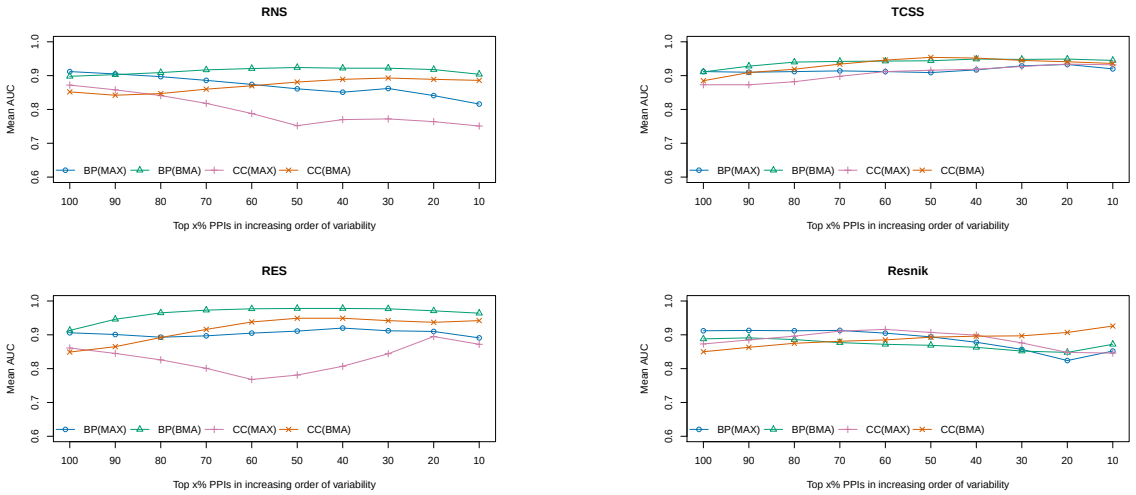


Figure 4.4: The mean AUCs of nine GO versions (Ver3.0 to Ver3.8) achieved by RNS, RES, TCSS, and Resnik at different cutoffs of affected PPIs with the plotting of individual SSM for the DIP-core dataset.

4.4.2 Set-discriminating power of KEGG pathways

For each GO version and SSM, we calculate DP values of each pathway with respect to other 10 pathways. Then we take version-wise (GO) mean DP values. Table 4.4 shows the mean DP values of all the 11 pathways for each GO-BP version and SSM.

The majority of SSMs produces quite similar DP values over the evolution of GO. To compute DP values, we need to consider all the intra-set and inter-set protein-pairs among

PPIs in order of variability	BP AUC	BP SSM	BP Str.	CC AUC	CC SSM	CC Str.
All PPIs(100%)	0.918	Rel	MAX	0.885	TCSS	BMA
Top 90%	0.946	RES	BMA	0.909	TCSS	BMA
Top 80%	0.965	RES	BMA	0.919	TCSS	BMA
Top 70%	0.973	RES	BMA	0.934	TCSS	BMA
Top 60%	0.977	RES	BMA	0.946	TCSS	BMA
Top 50%	0.978	RES	BMA	0.954	TCSS	BMA
Top 40%	0.978	RES	BMA	0.952	TCSS	BMA
Top 30%	0.977	RES	BMA	0.945	TCSS	BMA
Top 20%	0.971	RES	BMA	0.941	TCSS	BMA
Top 10%	0.964	RES	BMA	0.942	RES	BMA

Table 4.3: The highest performance in different levels of variable PPIs for the DIP-core dataset. The first column indicates the top x% variable PPIs. Columns 2, 3, and 4 show the highest mean AUC, corresponding SSM, and strategy, respectively, in BP ontology. Similarly, columns 5, 6, and 7 demonstrate the same in CC ontology.

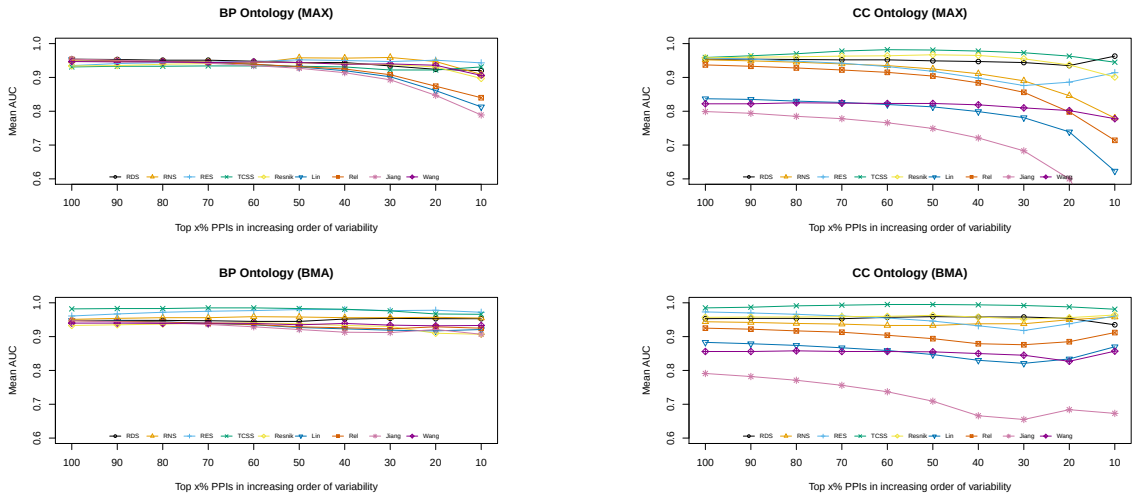


Figure 4.5: The mean AUCs of five GO versions (Ver3.4 to Ver3.8) achieved by SSMs at different cutoffs of affected PPIs for the AP/MS dataset.

the pathways. However, as discussed earlier, the majority of the PPIs does not get affected significantly due to the changes in GO. As a consequence, the mean DP values for most of the SSMs are similar over the GO versions.

DP values achieved in BP are significantly higher than CC, irrespective of SSM and strategy. RES almost continuously produces higher DP values in both the ontologies, particularly, with BMA strategy. TCSS shows competitive performances in both the ontologies while Jiang achieves good DP values in BP only. All the three SSMs produce significantly higher DP values with BMA strategy than MAX in their respective best-case scenarios.

Table 4.4: The mean DP values of all the 11 pathways for each GO-BP version and SSM. The best DP values are shown in bold.

Ver.	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
3.0	MAX	2.45	2.64	**5.22**	2.46	2.24	2.16	2.38	4.17	2.08
	BMA	2.23	2.70	**5.59**	3.16	2.18	2.16	2.41	5.06	2.03
3.1	MAX	2.44	2.65	**5.39**	2.46	2.20	2.13	2.34	4.09	2.06
	BMA	2.23	2.70	**5.76**	3.16	2.14	2.12	2.36	4.98	2.00
3.2	MAX	2.47	2.78	**5.95**	2.48	2.32	2.23	2.48	4.33	2.10
	BMA	2.26	2.84	**6.38**	3.16	2.25	2.22	2.51	5.43	2.05
3.3	MAX	2.52	2.77	**5.65**	2.66	2.28	2.15	2.36	3.70	2.06
	BMA	2.24	2.94	**6.56**	2.76	2.22	2.19	2.48	4.94	2.01
3.4	MAX	2.49	2.74	**5.56**	2.67	2.28	2.14	2.35	3.73	2.06
	BMA	2.18	2.87	**6.50**	2.78	2.23	2.19	2.50	5.04	2.02
3.5	MAX	2.46	2.72	**5.62**	2.51	2.25	2.11	2.31	3.39	2.07
	BMA	2.24	2.91	**6.40**	2.82	2.24	2.19	2.46	4.58	2.02
3.6	MAX	2.38	2.71	**5.76**	2.72	2.40	2.28	2.52	3.76	2.07
	BMA	2.19	2.93	**6.70**	2.82	2.50	2.45	2.81	5.21	2.03
3.7	MAX	2.34	2.73	**5.90**	2.76	2.4	2.28	2.52	3.75	2.07
	BMA	2.15	2.93	**6.77**	3.31	2.49	2.45	2.82	5.16	2.03
3.8	MAX	2.32	2.70	**5.77**	2.71	2.39	2.30	2.53	3.79	2.09
	BMA	2.12	2.89	**6.60**	3.22	2.49	2.46	2.83	5.12	2.02

RES-BMA shows continuous and significant improvement over the evolution of GO. The newer GO version is expected to be more accurate and complete than the older, and the robust SSMs should reflect that in their performance accordingly. RES almost consistently and significantly improves the DP value over the evolution of GO, particularly in BP ontology, while other SSMs keep their performance almost similar. Therefore it seems that the changes in GO are reflected better way with RES than the others.

Table 4.5 summarizes the mean DP values of all the nine GO versions and eleven pathways. The overall mean DP value of RES-BMA is higher than RES-MAX.

Ontology	Strategy	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
BP	MAX	2.43	2.71	5.65	2.60	2.31	2.20	2.42	3.86	2.08
	BMA	2.21	2.86	6.36	3.02	2.30	2.27	2.58	5.06	2.02
CC	MAX	1.17	1.40	2.13	1.67	1.46	1.23	1.36	1.07	1.11
	BMA	1.15	1.50	2.25	1.95	1.40	1.27	1.40	1.08	1.13

Table 4.5: The mean DP values of all the nine GO versions and eleven pathways. The best DP values are shown in bold.

4.5 Conclusion

In this work, we systematically study how similarity measures get affected due to the evolution of GO for the task of scoring confidence of PPIs. In the evaluation, we consider nine state-of-the-art SSMs and nine Bioconductor versions of GO along with their corresponding annotation corpora. We observe that the performance of each measure gets affected due to the regular updates of GO. SSMs show better robustness while considering the BMA strategy and BP ontology. While RES, TCSS, RNS, and Resnik exhibit higher robustness, RES reflects the changes in GO better way than the others. In this work, we consider the yeast model organism only. Therefore the work may be extended in the future by considering PPIs of other species such as human PPIs.

Chapter 5

Impact of low-confidence interactions on computational identification of protein complexes

As we have mentioned in Chapter 1 and 3, protein-protein interaction (PPI) data have a significant amount of false positives that might have a negative impact on any downstream analysis of PPI networks, including detecting protein complexes. Although some studies indicate that filtering PPI networks improve the performance of complex detection techniques, we could not find a systematic study on the effect of PPI filtering using GO. In this chapter, we systematically study the impact of low-confidence PPIs on the performance of complex detection algorithms (CDAs) using GO-based semantic similarity measures.

We consider five representative state-of-the-art CDAs (discussed in Chapter 1 and 2), including our proposed one and nine GO-based similarity measures (discussed in Chapter 3 and 4), including our proposed ones in the evaluation. We find that each CDA significantly improves its performance after filtration of low-similarity scored PPIs. It is also observed that the percentage improvement and the filtration percentage (of low-confidence PPIs) are highly correlated.

Some fundamental concepts of this chapter have already been discussed in previous chapters. However, a brief discussion of those concepts is included in suitable places for the sake of completeness.

5.1 Introduction

Identifying protein complexes is crucial to understanding the principles of cellular organization. As discussed in Chapter 1, experimental detection of protein complexes is inadequate due to the limitation of present high-throughput experimental techniques. From the last few decades, a large volume of *protein-protein interaction (PPI)* data has become available due to high-throughput technologies. Hence computational detection of protein complexes from PPI data is considered a useful complement to the experimental techniques.

PPI data can be visualized as an undirected network, commonly referred to as the *PPI network*, where nodes and edges are the proteins and interactions, respectively. The problem of detecting complexes can be mapped to the problem of detecting dense subgraphs (viz., containing many interactions) from PPI networks. These dense structures are often referred to as *communities*. In fact, many *community detection algorithms* are directly used for the identification of protein complexes. Hereafter, we refer to the community detection algorithm as a *complex detection algorithm (CDA)*.

Generally, the CDAs utilize the structural properties of the graph to detect communities. However, PPI data have a significant amount of false positives [10]. These false positive interactions (edges) lead to infer structural properties that are not present in the original PPI network. Therefore the performance of CDAs gets adversely affected by the false positive interactions present in the network. Although the identification of false positive interactions in PPI data is a challenging task, there are computational techniques that assign confidence scores to a given PPI. The confidence score is the likelihood of a pair of proteins to interact. Hence low-confidence interactions are likely to be false positives.

Ontology-based *semantic similarity measures (SSMs)* are a class of numerical indicators that attempt to estimate the semantic closeness of two terms based on their representation in a given ontology. Gene Ontology (GO) is a very comprehensive resource for the properties of gene or gene products and their functional relationships. GO-based SSMs can be used to measure the functional similarity between two gene products (e.g., proteins). The similarity score can be treated as the likelihood of the two proteins to interact. Hence, the interactions with low-similarity scores may be considered as low-confidence PPIs. The low-confidence PPIs are highly likely to be false positives and can be filtered out. Some studies indicate that filtering PPI networks increase the performance of complex detection techniques [115, 117, 172, 173]. However, we could not find a systematic study on the effect of PPI filtering using GO as it is often used as a benchmark for validating PPIs.

In this chapter, we systematically evaluate the performance of CDAs using GO-based SSMs. In the evaluation, we consider two reference PPI network datasets and two ground

truth complex datasets of the yeast model organism. However, the impact of low-confidence PPIs on CDAs strongly depends on the reference species considered, as the reliability of PPIs changes over the species.

The rest of the chapter is organized in the following manner. A brief outline of PPI confidence scoring methods and complex detection algorithms are presented in section 5.2. We detail the experimental design, datasets, evaluation metrics, and implementation in section 5.3. We report, analyze, and discuss the results in section 5.4. Finally, we conclude the chapter in section 5.5.

5.2 Related work

In this section, we provide a brief outline of existing PPI confidence scoring schemes and complex detection algorithms.

5.2.1 PPI confidence scoring methods

The methods for scoring confidence of PPIs are broadly categorized into two classes: methods depending solely on interaction data [174, 162] and methods taking into account domain information such as functional similarity [99, 175], sequence similarity [176], gene expression profile [177], and combination of sequence, structure, and functional annotation information [119]. The first class of methods relies on the structural properties of the interaction network. For instance, FS-weight [115], CD-distance [178], and Iterative score [117] assume that if two proteins share a number of common interaction partners, they are more likely to have a similar function, hence more likely to interact physically with each other.

In this study, we consider GO-based SSMs, belonging to the second class of methods, to score confidence of PPIs. We consider six state-of-the-art SSMs proposed by Resnik [94], Lin [95], Schlicker *et al.* [96], Jiang & Conrath [97], Wang *et al.* [98], and Jain & Bader [99], referred to as *Resnik*, *Lin*, *Rel*, *Jiang*, *Wang*, and *TCSS*, respectively, in the rest of the chapter. We describe these SSMs in Chapter 3 (Section 3.2.4). Resnik, Lin, Rel, and Jiang are node-based approaches and utilize the notion of *information content (IC)* that quantifies the specificity of a term in an ontology. Wang is a hybrid approach that considers the properties of both nodes and edges. TCSS is a structure-based approach that exploits the unequal depth of biological knowledge representation in different branches of GO DAG. Resnik-MAX and TCSS-MAX have been considered to be the best SSMs for scoring confidence of PPIs by several studies [100, 99, 102, 101]. We also consider the new family of SSMs, namely

RDS, RNS, and RES, proposed in Chapter 3. We use an alternative definition of specificity that is not based on the whole ontology, but rather exploits the properties of considering subgraphs that contain only ancestors and descendants of a given term. We show that the proposed measures, particularly RES and RNS, outperform the others in many cases.

A protein may be annotated with more than one GO term. Therefore, the similarity between two proteins (say, p_1 and p_2) is calculated as the similarity between the two sets S and T of terms that annotate each protein. SSMs are defined on a pair of terms. Hence, to calculate the similarity between two sets of terms, combining or aggregate strategies are used. The three commonly used strategies are - maximum (MAX) [148], average [123], and best-match average (BMA) [150, 149, 96]. The formulations of the combining strategies are given in Chapter 3 (Section 3.2.3).

5.2.2 Complex detection algorithms

Over the years, a wide range of computational techniques has been developed to detect protein complexes from PPI networks [179, 180, 181]. The existing CDAs in the literature can be broadly divided into two categories - disjoint and overlapping CDAs. Obviously, a disjoint CDA does not assign a protein into multiple complexes and thus unable to give the true picture of complexes. Overlapping CDAs mitigate this issue by assigning one protein into multiple complexes. Algorithms such as [78, 77, 75, 8, 73, 82] have been developed for overlapping communities, hence can be applied for overlapping complexes. We select five state-of-the-art representative CDAs, namely GCE [8], EGCE (proposed in Chapter 2), ClusterONE [22], Link [78], and CFinder [73]. GCE (Greedy Clique Expansion) finds candidate complexes as maximal cliques and expands by optimizing a community fitness function. EGCE (Extended Greedy Clique Expansion) is an extension of GCE which detects the highly overlapping nodes (not detected by GCE) and assigns their respective communities (complexes) by utilizing the notion of *Interaction Probability*. Link uses link partitioning instead of nodes, and CFinder exploits k-clique percolation. ClusterONE finds groups of nodes as candidate complexes with high cohesiveness and merges highly overlapping groups.

5.3 Experimental setup

In this section, we discuss the experimental design, datasets, evaluation metrics, and implementation. The following list of CDAs and SSMs are used in the evaluation:
CDAs - GCE [8], EGCE (proposed in Chapter 2), ClusterONE [22], Link [78], and CFinder [73]
SSMs - RDS/RNS/RES (proposed in Chapter 3), TCSS [99], Resnik [94], Lin [95], Rel [96],

Jiang [97], and Wang [98].

5.3.1 Preprocessing

We consider the biological process (BP) and cellular component (CC) ontologies of GO along with the MAX and BMA strategies to aggregate SSMs for protein pairs. Our choices of ontologies and strategies are based on our previous study (discussed in Chapter 3, section 3.4), indicating they are the most relevant for scoring confidence of PPIs. We consider those PPIs whose both the interacting proteins are annotated to at least one GO term other than the root terms in BP or CC ontologies. Further, we do not consider electronically inferred annotations (IEA) as they are not verified by human experts.

5.3.2 Datasets

In this section, we detail different datasets used in the evaluation, including their sources/constructions, preprocessing, and specifications.

Determining thresholds for positive and negative interactions

For each similarity measure, we need to choose a reasonable threshold (a cut-point or discrimination value) of confidence that differentiates between positive and negative interactions. Determining a reasonable threshold is challenging. A number of criteria are found in literature, for example, F1 score, Youden's Index [182], Kappa Index [183], etc. Here, we use Youden's Index as it is the widely used criterion to find optimal cut-point. It maximizes the sum of sensitivity and specificity in a ROC curve.

We consider a high-quality binary gold standard yeast dataset ("Binary-GS") created by Yu et al. [184] as the positive PPI dataset. The number of PPIs remains in the dataset are 1139 and 1196 for BP and CC ontology, respectively, after the necessary preprocessing mentioned in section 5.3.1. The negative PPI datasets with an equal number of PPIs are generated independently by randomly choosing protein pairs annotated in BP and CC ontologies and which are also not present in the iRefWeb database [163] dated 22.01.2018, a combined database of all known PPIs. The thresholds for different similarity measures in both BP and CC ontologies are shown in Table 5.1.

Table 5.1: Discrimination values of different SSMs using Youden's Index to differentiate between positive and negative interactions.

Ontology	Strategy	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
BP	MAX	0.41	0.64	0.34	0.39	0.44	0.75	0.74	0.66	0.56
	BMA	0.31	0.32	0.15	0.16	0.26	0.46	0.41	0.39	0.41
CC	MAX	0.56	0.93	0.69	0.39	0.32	0.73	0.97	0.94	0.82
	BMA	0.40	0.31	0.23	0.18	0.23	0.56	0.47	0.80	0.64

Reference PPI networks

Experimental PPIs are commonly used for the validation of computational protein complex detection methods. Database of Interacting Proteins (DIP) [104] is one of the frequently used databases for the purpose. Hence we consider the DIP-full dataset of S. cerevisiae dated 05.02.2017. To make the evaluation scheme more robust, we select another commonly used experimental combined-AP/MS PPI dataset (of S. cerevisiae) [70]. The sizes of PPIs in AP/MS and DIP-full datasets are 8748 and 21091, respectively, after the necessary preprocessing mentioned in section 5.3.1.

For each PPI dataset, we construct the following four types of unfiltered PPI networks: i) network type-1: PPIs with the interacting proteins annotated to BP terms; ii) network type-2: PPIs with the interacting proteins annotated to CC terms; iii) network type-3: PPIs with the interacting proteins annotated to BP or CC terms and iv) network type-4: PPIs with the interacting proteins annotated to BP as well as CC terms. In other words, network type-3 and 4 are the union and the intersection of network type-1 and type-2, respectively. From each unfiltered network, 18 filtered networks are generated using the nine SSMs along with the two combining strategies (MAX and BMA). To construct the filtered network, we consider only those PPIs whose confidence scores are greater than or equal to the optimal thresholds of the corresponding SSM and strategy mentioned in Table 5.1. The number of interactions remaining for each unfiltered and corresponding filtered networks are presented in Table 5.2 and 5.3.

Ground truth complexes

We utilize two yeast protein complex datasets: CYC2008 [69] and the complex portal of EMBL-EBI [185] (dated 16.07.2018) as ground truths. We remove those proteins (from the ground truth complexes) which are not present in the corresponding unfiltered networks and consider the complexes having at least three proteins. Finally, the number of protein complexes with three or more proteins remains in the CYC2008, and EMBL-EBI datasets are 144 and 346, respectively.

Table 5.2: No. of interactions before and after filtration of the AP/MS network. *Raw* indicates before filtration and headers with the name like *RDS*, *RNS*, etc. indicate after filtration with the corresponding method.

Ont.	Raw	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
BP	8486	MAX	7790	7211	7224	7070	6970	7276	7252	7487	7524
		BMA	7610	7740	7424	7159	7514	7622	7673	7690	7518
CC	8690	MAX	7454	7228	7317	7147	7542	7792	6170	6662	7708
		BMA	7583	7880	7663	7156	7752	7544	7759	5676	7721
BP/CC	8748	MAX	8203	7898	7962	7872	8039	8167	7775	8112	8272
		BMA	8182	8289	8106	7904	8294	8190	8298	7983	8226
BP&CC	8428	MAX	7041	6541	6579	6345	6473	6901	5647	6037	6960
		BMA	7011	7331	6981	6411	6972	6976	7134	5383	7013

Table 5.3: No. of interactions before and after filtration of the DIP-full network. *Raw* indicates before filtration and headers with the name like *RDS*, *RNS*, etc. indicate after filtration with the corresponding method.

Ont.	Raw	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
BP	18750	MAX	6902	6341	6558	6267	5877	5887	5858	6523	6646
		BMA	6025	6816	6496	6424	6990	6360	6667	6424	6239
CC	20721	MAX	5974	4682	5958	5065	6504	11023	4482	10404	11073
		BMA	6121	8946	6739	5262	6446	8464	8164	5725	10425
BP/CC	21091	MAX	8606	7299	8409	7443	8192	11979	6892	11992	12572
		BMA	7962	10645	8774	7879	8962	10180	10035	9133	11778
BP&CC	18380	MAX	4270	3724	4107	3889	4189	4930	3447	4934	5147
		BMA	4183	5117	4461	3807	4474	4644	4796	3016	4886

5.3.3 Evaluation metrics

Many criteria exist in the literature [186] to measure the similarity between the detected and true complexes. We consider Normalized Mutual Information (NMI), precision, recall, F1 score/fmeasure [43], and the prediction of possible novel complexes (described in Sec 5.4) in the evaluation. However, our main discussion is based on NMI, as it is commonly used for community detection [9]. Further, NMI has been extended for overlapping communities by Lancichinetti *et al.* [79], whereas similar extension is not available for most of the other criteria. NMI is an information-theoretic notion and was first used by Danon *et al.* [85] for non-overlapping communities. The NMI between two random variables X and Y is defined as

$$NMI(X,Y) = {}^{2[H(X)-H(X|Y)]}\!/_{[H(X)+H(Y)]},$$

where $H(X)$ is the entropy of the random variable X. We use the overlapping extension of NMI since a protein may belong to more than one complex. The measure is normalized into [0,1], i.e., an NMI value of 1 indicates that the two sets of communities (complexes) are precisely the same, whereas 0 indicates both the sets are entirely different.

5.3.4 Methods and their implementations

We consider five representative state-of-the-art CDAs - GCE, EGCE, ClusterONE, Link, and CFinder, in the evaluation. We use the implementation provided by the authors of GCE, EGCE, ClusterONE, Link, and CFinder with the default values of the parameters except for the k (minimum clique size) of GCE, EGCE, and CFinder. We set the value of k to 3.

We select nine state-of-the-art SSMs - RDS, RNS, RES, TCSS, Resnik, Lin, Rel, Jiang, and Wang, in the evaluation. The GOSemSim R package (version: 2.10.0) [165] is used for the implementation of Resnik, Lin, Rel, Jiang, and Wang. We use the code provided by the authors of TCSS with the default set of parameters. The implementation provided by the authors of TCSS utilizes the MAX strategy only, and we modify the code slightly to include the BMA strategy as well.

For GO and corresponding annotation corpus, GO.db [166], and org.Sc.sgd.db [168], R packages (version:3.8.2, April 2019 release) are used. The versions of all R packages are maintained in such a way that they use the same GO and its annotations. Since the implementation of TCSS expects the ontology and annotation corpus as the text files provided by the Gene Ontology Consortium, we use the corresponding released versions of GO and annotation corpus dated Apr 26, 2019. For the implementation of NMI, we use the code provided by Lancichinetti *et al.* [79].

5.4 Results and discussion

In this section, we report, analyze, and discuss the results. The key observations are highlighted.

Filtering of low confidence PPI enhanced the performance of all CDAs. Each CDA is executed on each type of aforementioned unfiltered and filtered networks for each of the two PPI datasets (AP/MS and DIP-full). The outcome is compared with two ground truth complex datasets (CYC2008 and EMBL) using NMI, as mentioned earlier. The main experimental design is demonstrated in Figure 5.1. The performances of different CDAs on four types of unfiltered and filtered networks for AP/MS PPI dataset with CYC2008 yeast complex dataset are reported in Table 5.4, 5.5, 5.6, and 5.7. As CFinder fails to terminate on AP/MS PPI dataset, we cannot report the corresponding NMI values. The result of network type-4 (BP & CC) from another combination of PPI and complex datasets (DIP - EMBL) is shown in Table 5.8. The NMI values are significantly higher in filtered networks in comparison with unfiltered ones for almost all cases. For example, the range of improvements (in terms of NMI) for network type-1 (BP) after filtration is from 4.68%

(Rel-BMA) to 11.79% (RNS-BMA) while using GCE (Table 5.4). Similar observations can be made in all other cases. However, the improvement is much higher in the AP/MS dataset than DIP.

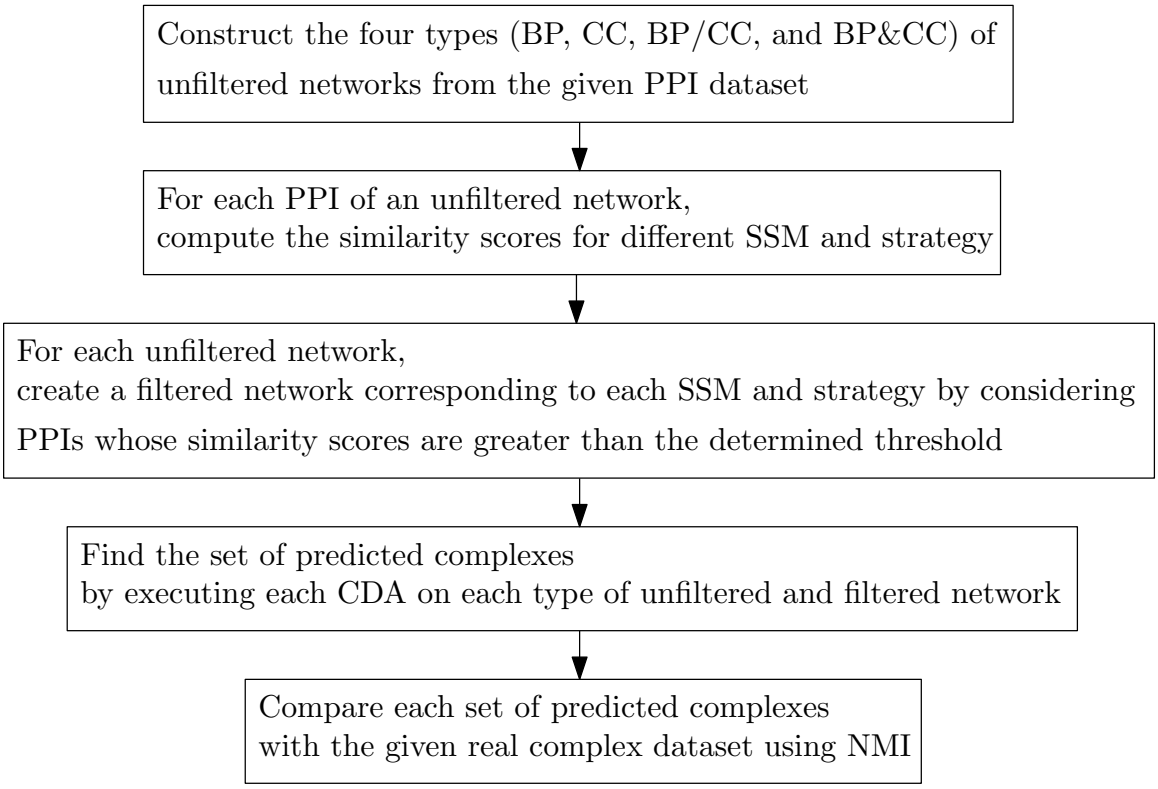


Figure 5.1: Flowchart of the main experimental design for a pair of PPI and complex dataset.

Table 5.4: NMI values of different CDAs on filtered and unfiltered networks with BP ontology for the AP/MS network and CYC2008 complex datasets. The highest improvement for each row is shown in bold.

Met.	Raw	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
GCE	0.577	MAX	0.632	**0.636**	0.623	0.620	0.621	0.631	0.626	**0.636**	0.635
		BMA	0.630	**0.645**	0.631	0.638	0.614	0.616	0.604	0.629	0.616
EGCE	0.569	MAX	0.623	**0.629**	0.616	0.613	0.613	0.619	0.612	0.626	0.612
		BMA	0.608	**0.628**	0.608	0.625	0.596	0.604	0.590	0.607	0.601
Cluster ONE	0.559	MAX	0.598	0.605	0.590	0.585	0.578	**0.612**	0.612	0.599	0.608
		BMA	0.596	0.598	0.599	**0.610**	0.586	0.592	0.597	0.600	0.579
Link	0.448	MAX	0.462	0.458	0.449	0.450	0.449	**0.471**	**0.471**	0.470	0.462
		BMA	0.445	0.456	0.459	**0.460**	0.449	0.459	0.455	0.456	0.463

PPI datasets have a significant amount of false positives. The amount of filtration of low-confidence PPIs is significantly high in both the PPI datasets, with the average percentage of filtration as 40%. It is noteworthy to mention that only approximately 4% of PPIs and 13% of complexes are shared between the two PPI and two complex datasets, respectively. Hence, significant high-filtration in highly dissimilar datasets indicates that both the PPI

Table 5.5: NMI values of different CDAs on filtered and unfiltered networks with CC ontology for the AP/MS network and CYC2008 complex datasets. The highest improvement for each row is shown in bold.

Met.	Raw	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
GCE	0.569	MAX	0.672	**0.687**	0.674	0.671	0.671	0.630	0.704	0.639	0.627
		BMA	0.653	0.648	0.669	**0.675**	0.666	0.644	0.659	0.624	0.610
EGCE	0.551	MAX	0.660	0.677	0.663	0.656	0.657	0.618	**0.690**	0.625	0.619
		BMA	0.644	0.639	0.655	**0.663**	0.644	0.645	0.644	0.621	0.589
Cluster	0.553	MAX	0.639	0.670	0.636	0.649	0.632	0.577	**0.674**	0.595	0.567
ONE		BMA	0.626	0.596	0.623	**0.658**	0.621	0.606	0.609	0.589	0.586
Link	0.437	MAX	0.468	**0.480**	0.472	0.465	0.467	0.455	0.501	0.475	0.458
		BMA	**0.469**	0.457	0.465	0.468	0.461	0.460	0.459	0.428	0.451

Table 5.6: NMI values of different CDAs on filtered and unfiltered networks by considering all the interactions with either BP or CC for the AP/MS network and CYC2008 complex datasets. The highest improvement for each row is shown in bold.

Met.	Raw	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
GCE	0.566	MAX	0.621	**0.627**	0.612	0.610	0.614	0.595	0.628	0.592	0.587
		BMA	0.624	0.621	0.616	**0.631**	0.628	0.600	0.607	0.590	0.589
EGCE	0.548	MAX	0.600	0.597	0.588	0.594	0.593	0.572	**0.611**	0.570	0.570
		BMA	**0.613**	0.609	0.600	0.612	0.609	0.589	0.581	0.582	0.568
Cluster-	0.551	MAX	0.579	**0.595**	0.590	0.578	0.568	0.558	0.588	0.553	0.561
ONE		BMA	0.586	0.568	0.581	**0.594**	0.578	0.561	0.566	0.570	0.559
Link	0.436	MAX	0.451	0.458	0.453	0.458	0.449	0.445	**0.465**	0.447	0.445
		BMA	0.449	0.445	0.451	**0.456**	0.446	0.452	0.447	0.443	0.446

Table 5.7: NMI values of different CDAs on filtered and unfiltered PPIs by considering the common interactions with BP and CC for the AP/MS network and CYC2008 complex dataset. The highest improvement for each row is shown in bold.

Met.	Raw	Str.	RDS	RNS	RES	TCSS	Resnik	Lin	Rel	Jiang	Wang
GCE	0.589	MAX	0.685	**0.703**	0.685	0.686	0.675	0.668	0.702	0.669	0.674
		BMA	0.668	0.662	0.670	**0.687**	0.668	0.681	0.659	0.673	0.647
EGCE	0.579	MAX	0.675	**0.700**	0.673	0.687	0.679	0.672	0.693	0.660	0.668
		BMA	0.656	0.647	0.662	**0.687**	0.651	0.673	0.651	0.674	0.640
Cluster-	0.540	MAX	0.662	0.673	0.655	0.659	0.647	0.636	**0.701**	0.654	0.631
ONE		BMA	0.640	0.621	0.641	**0.676**	0.648	0.656	0.627	0.624	0.618
Link	0.450	MAX	0.485	0.487	0.479	0.471	0.482	0.493	**0.515**	0.495	0.480
		BMA	0.477	0.472	0.474	0.479	0.478	0.476	**0.481**	0.438	0.472

datasets are having a significant amount of false positives. The low-confidence PPIs are likely to be false positives. Therefore, one should be careful in any analysis with a full PPI dataset.

The percentage improvement of CDAs and the filtration percentage (of low-confidence PPIs) are highly correlated. For each of 288 cases (4 combinations of PPI and complex datasets $\times$ 4 network types $\times$ 9 SSMs $\times$ 2 combining strategies), we compute the average percentage of improvement (NMI) of CDAs and the corresponding percentage of filtration

Table 5.8: NMI values of different CDAs on filtered and unfiltered PPIs by considering the common interactions with BP and CC for the DIP-full network and EMBL-EBI complex dataset. The highest improvement for each row is shown in bold.

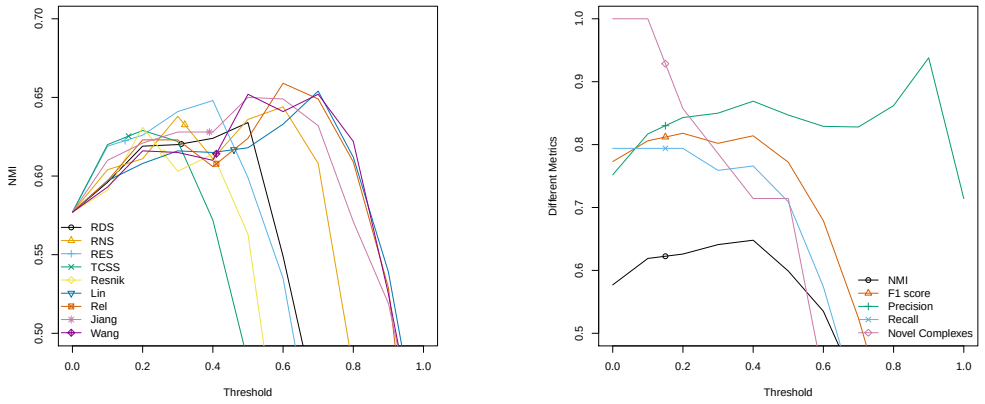
Met.	**Raw**	**Str.**	**RDS**	**RNS**	**RES**	**TCSS**	**Resnik**	**Lin**	**Rel**	**Jiang**	**Wang**
GCE	0.079	MAX	0.411	0.446	0.405	0.417	0.387	0.361	**0.485**	0.393	0.365
		BMA	0.388	0.359	0.386	**0.453**	0.376	0.367	0.377	0.446	0.347
EGCE	0.081	MAX	0.420	0.450	0.405	0.414	0.388	0.376	**0.478**	0.396	0.369
		BMA	0.375	0.346	0.378	**0.455**	0.376	0.370	0.370	0.454	0.341
Cluster-ONE	0.141	MAX	0.375	**0.413**	0.358	0.391	0.368	0.359	0.436	0.368	0.334
		BMA	0.370	0.335	0.354	**0.407**	0.354	0.350	0.339	0.377	0.339
Link	0.262	MAX	0.303	0.328	0.302	0.312	0.307	0.303	**0.348**	0.308	0.315
		BMA	0.313	0.303	0.303	**0.321**	0.309	0.297	0.293	0.288	0.304
CFinder	0.168	MAX	0.407	0.441	0.407	0.412	0.403	0.381	**0.480**	0.399	0.371
		BMA	0.404	0.364	0.382	**0.451**	0.371	0.381	0.378	0.421	0.366

of low-confidence PPIs. The Pearson correlation between the average percentage of improvement and the percentage of filtration of the 288 cases is very high (0.95). Hence, the percentage of improvement of CDAs and the percentage of filtration (of low-confidence PPIs) are highly proportional. This percentage improvement of CDAs is only on the discovery of known complexes, while the prediction of novel complexes may suffer as discussed later.

Further, we investigate the impact of changing the discrimination threshold on the performance for a particular setting. We observe that each SSM improves its performance until a certain threshold value, and after that, performance deteriorates sharply, particularly with BMA strategy. The sharp declination at a high threshold is quite natural. As we increase the threshold, we tend to remove even true PPIs but likely with relatively poor annotation, which in turn also impacts the network structure. Both of these intertwined aspects lead to poor performance. Figure 5.2a demonstrates this for the AP/MS network and CYC2008 complex datasets in BP ontology with BMA strategy while using GCE. However, with MAX strategy, some of the SSMs, in some cases, keep improving their performance even at the highest threshold. The reason may be due to the overestimation nature of the MAX strategy as discussed in Chapter 3 (Section 3.4). The marked points on the plots indicate the NMI values at the cut-off thresholds achieved in Table 5.1. It is clear that the determined cut-off thresholds are quite reasonable as they are not far from the thresholds at the highest NMI.

We further compute the performance in terms of precision, recall, F1 score, and the prediction of possible novel complexes by varying the threshold (Figure 5.2b). We consider two criteria to determine a predicted complex as a possible novel complex: i) it must not match any of the real complexes, i.e., the match or affinity score between the predicted complex and any of the real complexes must be less than 0.2 (the usual affinity score [43]), ii) the *intra-set-avg similarity* of the predicted complexes must be greater than the determined

threshold for the SSM considered (Table 5.1). The *intra-set-avg similarity* is calculated as the average similarity score of the PPIs involved in the predicted complex. Both F1 score and precision improve until a certain threshold and then decline sharply, similar to NMI, as discussed earlier. However, recall and the prediction of possible novel complexes get degraded relatively sooner as threshold increases in many cases, particularly with BMA strategy. Therefore, care should be taken while filtering PPIs at a high threshold. It is noteworthy to mention that the determined thresholds (Table 5.1) are not far from the optimal precision and recall (i.e., the highest F1 score).



(a) (b)

Figure 5.2: (a) Threshold vs. NMI for AP/MS network and CYC2008 complex datasets in BP ontology with BMA strategy while considering GCE. The marked points on the plots indicate the NMI values at the thresholds determined for the evaluation. (b) Threshold vs. different metrics for RES-BMA by considering the same setting. The predicted number of novel complexes is normalized by dividing with the maximum number of predicted novel complexes (14) of the setting being considered.

Although each CDA improves its performance with each SSM and combining strategy, the overall improvement is better with TCSS comparatively for the discrimination thresholds determined in Table 5.1. We do not observe any significant difference (in performance) between the MAX and BMA strategy. However, the objective of this study is not to show the performance comparison among CDAs and SSMs, rather show how low-similarity scored interactions impact on the performance of CDAs. In fact, it would be difficult to draw any such indirect inference, because the structure of the filtered network primarily depends on the chosen threshold of the SSM. If we change the threshold, some PPIs will be added or deleted. Consequently, the performance of a CDA on the filtered network gets affected, as demonstrated in Figure 5.2.

CDAs show the best performance on the network type-4. Since all the PPIs in a pro-

tein complex need to occur simultaneously at one location, the interacting proteins of the complex are likely to share similar CC terms. Further, a biological process is carried out by one or more protein complexes. Hence all the proteins of a complex are likely to share similar BP terms as well. Therefore, proteins sharing both similar BP and CC terms are most likely to be true positives (in the context of protein complexes). Hence CDAs show the best performance on the network type-4 (BP&CC), even though it is the smallest one.

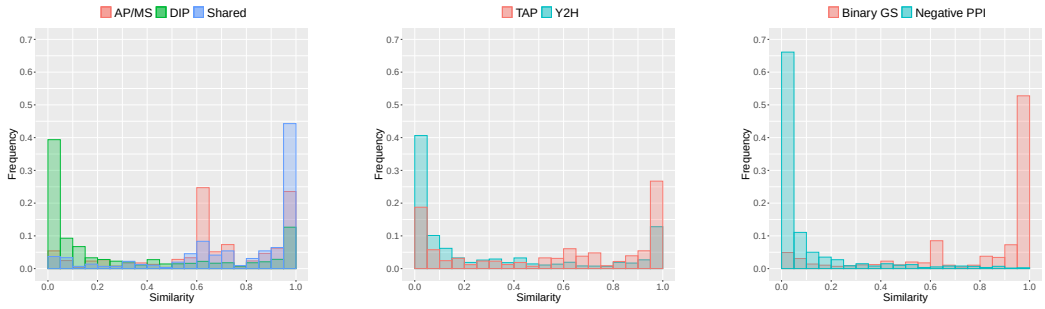
CDAs show the lowest performance on the network type-3. If two proteins share similar BP terms only without sharing CC terms, and vice versa are less likely to interact, equivalently, more likely to be noisy (likely to be false positive). Hence other than the network type-4 (BP & CC), the rest of the three network types may have some amount of these (probable) noisy PPIs. Naturally, the maximum amount of (probable) noisy PPIs present in the network type-3 (BP/CC), which is the union of network type-1 (BP) and type-2 (CC). As a consequence, CDAs exhibit the least performance in the network type-3, even though it is the largest one.

PPIs sharing similar BP terms are more reliable than PPIs sharing similar CC terms. From the last two observations, it is clear that the performance of CDAs is highly affected by the amount of (probable) noisy PPIs present in the PPI network rather than the size of the network. Further, it is observed that NMI values and the percentage of noisy PPIs remain in the filtered networks (#noisy PPIs /#edges of the filtered network considered) are negatively correlated. We also observe that the (probable) percentage of noisy PPIs is significantly higher in the network type-2 (CC) than type-1 (BP). The average percentage of noisy PPIs in filtered network type-2 (CC) is 5.37%, whereas it is 0.69% in type-1 (BP). Therefore although both BP and CC are relevant for protein complexes, PPIs sharing similar BP terms are more likely to be true positives than PPIs sharing similar CC terms.

CDAs show better performance on the AP/MS network. The DIP-full network is more than two times larger than the AP/MS. The average percentage of PPI filtration in the AP/MS network is 14.18%, while it is 65.64% in DIP-full. Further, the percentage of probable noisy PPIs present is significantly higher in the DIP-full network. The average percentage of noisy PPIs (present after filtration) in the DIP-full network is 5.46%, whereas, in the AP/MS network, it is 1.84%. From Figure 5.3a, it is also clear that the overall similarity score is higher in the AP/MS network than DIP-full. Therefore, DIP-full is noisier than the AP/MS network. Consequently, CDAs exhibit better performance on the AP/MS network, even though it is more than two times smaller than DIP-full. However, the shared PPIs between the AP/MS and DIP-full networks produce higher similarity scores as expected (Figure 5.3a).

DIP database provides information about experimental methods used to detect PPIs.

There are hundreds of experimental methods, and the majority of them are associated with a very little number of PPIs. We estimate distributions of similarity scores for *Yeast Two-Hybrid (Y2H)* and *Tandem Affinity Purification (TAP)* methods as they are used to detect a large number of PPIs in the dataset. It is observed that TAP-detected PPIs show higher scores than Y2H (Figure 5.3b). We also estimate distributions of similarity scores for the positive and generated negative datasets used in the evaluation. The positive and negative PPI datasets exhibit very high and very low similarity scores, respectively, as expected (Figure 5.3c). Therefore, the discrimination values achieved (Table 5.1) in the evaluation are quite reasonable.



(a) (b) (c)

Figure 5.3: Histograms of similarity scores of RES-MAX with BP ontology of different PPI datasets: (a) AP/MS, DIP-full, and Shared PPIs between the two datasets (b) TAP-detected and Y2H-detected PPIs in DIP-full dataset (c) Binary GS and Negative PPIs.

5.5 Conclusion

The objective of the study is to observe the impact of low-confidence protein interactions on the performances of CDAs using GO-based SSMs. Five representative CDAs and nine SSMs, along with two combining strategies (MAX and BMA), are used in the evaluation. We consider two different PPI and protein complex datasets. Both PPI and complex datasets are highly dissimilar. However, each CDA significantly improves its performance after filtration of low-similarity scored PPIs (i.e., low-confidence PPIs) for almost all the cases. It seems that there is a significant amount of false positives present in PPI datasets, and the number of false positives (present in PPI networks) greatly impacts the performance of CDAs. We find that the percentage improvement and the filtration percentage (of low-confidence PPIs) are proportionally related. Further, we observe that PPIs sharing similar BP terms are less likely to be false positives than PPIs sharing similar CC terms in the context of protein complexes. In the current study, we discuss the impact of low-confidence PPIs on CDAs by considering the yeast PPIs only. However, the impact strongly depends on the reference

species considered. Therefore, the work may be extended in the future by considering human PPIs, which are less reliable than yeast.

Chapter 6

Conclusion and future work

In the thesis, first in Chapter 2, we explore community detection algorithms (CDAs) in order to identify protein complexes in PPI networks more accurately. Particularly, we investigate overlapping CDAs primarily since protein complexes are overlapping in nature. We observe that existing overlapping CDAs are unable to work appropriately when overlapping nodes (proteins) belong to more than three communities (complexes). Hence we introduce an extension, EGCE (Extended Greedy Clique Expansion), of existing overlapping CDA, namely GCE (Greedy Clique Expansion), to identify highly overlapping nodes.

Our extensive analysis of synthetic and empirical benchmarks suggests that EGCE outperforms other state-of-the-art overlapping CDAs, particularly when overlapping nodes belong to more than three communities. Although the overall performance of GCE is slightly better than EGCE for the particular empirical benchmark considered, we show that EGCE assigns overlapping proteins into their native complexes better way than GCE.

The new extension is generic and not restricted to GCE only. Hence it can be applied to extend any existing overlapping CDAs. In fact, we observe that some CDAs improve their performance (in terms of NMI for detecting overlapping communities) with LFR benchmark when this extension is applied. Therefore it might be worthy of evaluating the new extension with other existing CDAs as future work.

PPI data involve a considerable fraction of false positive interactions. These false positives have a negative impact on any downstream analysis of PPI networks, including complex detection. In Chapter 3, we investigate how gene ontology (GO) can be utilized to isolate false positive PPIs. We introduce a new family of GO-based semantic similarity measures (SSMs) - RDS (Relative Depth Specificity), RNS (Relative Node-based Specificity), and RES (Relative Edge-based Specificity) for scoring confidence of PPIs. We use an alternative definition of specificity that is not based on the whole ontology, but rather exploits the

properties of considering subgraphs that contain only ancestors and descendants of a given term.

We show that the new SSMs effectively distinguish true PPIs from false positives than the existing alternatives. Further, RES can be utilized for protein functional clustering since it shows a robust set-discriminating power over KEGG pathways. It also shows excellent consistency. Further, we observe that BP is more appropriate than CC for scoring confidence of PPIs.

The proposed SSMs are evaluated only on GO for scoring confidence of PPIs. However, they are not restricted to any particular ontology. Therefore it would be worthy of evaluating the new SSMs on other ontologies and applications as future work.

GO is updated regularly by the addition, deletion, and merging of terms. Hence the similarity score of a protein-pair may get changed over different instances of the ontology. However, we could not find any study that evaluates the impact of the continuous evolution of GO on the performance of SSMs. Therefore in Chapter 4, we systematically study how similarity measures get affected due to the evolution of GO, i.e., the robustness of SSMs, under different settings for the task of scoring confidence of PPIs.

We find that SSMs exhibit better robustness while considering the BMA strategy and BP ontology. We notice that the MAX strategy tends to overestimate, particularly in CC. Further, we observe that while RES, TCSS, RNS, and Resnik exhibit higher robustness, RES reflects the changes in GO better way than the others.

However, in this work, the study of robustness is done on SSMs for the task of PPI confidence scoring. It might be worthy of doing similar studies on other genomics applications as future work since GO has been effectively utilized in various genomics applications. Further, in the study, we consider the yeast model organism only. Therefore the work may be extended in the future by considering PPIs of other species such as human PPIs.

As we have already mentioned, the presence of a significant amount of false positive PPIs is a bottleneck in detecting protein complexes from PPI data. Although some studies show that filtering PPI networks increase the performance of complex detection techniques, we could not find a systematic study on the effect of PPI filtering using GO. Hence in Chapter 5, we systematically do the same study using GO-based SSMs.

We find that CDAs significantly improve their performance after the filtration of low-confidence PPIs (likely to be false positives). We find that the percentage improvement and the filtration percentage (of low-confidence PPIs) are highly correlated. We notice that PPI data have a significant amount of false positives. Further, we observe that PPIs sharing similar BP terms are more reliable than sharing CC terms in the context of protein

complexes.

In this work, we study the impact of low-confidence PPIs on CDAs by considering the yeast PPIs only. However, the impact highly depends on the reference species considered. Therefore, the work may be extended in the future by considering human PPIs, which are less reliable than other species. Further, in the study, we use SSMs to score confidence of PPIs. There are various complex detection approaches (discussed in Chapter 1) and PPI scoring schemes (discussed in Chapter 5). Hence similar studies may be done using other complex detection approaches and/or PPI scoring schemes as future work.

The primary focus of the thesis is on detecting protein complexes. There is another closely related construct, *protein functional modules* (discussed in Chapter 1). However, most existing methods are unable to differentiate between the two. Therefore another interesting challenge might be distinguishing between protein complexes and functional modules. One approach may be the exploration of interaction dynamics using time series information of gene expression profiles. Another approach may be the exploration of functional information using GO. In fact, we may develop new methods based on GO solely for detecting protein complexes and functional modules.

Bibliography

[1] Christopher A Ross and Sarah J Tabrizi. Huntington's disease: from molecular pathogenesis to clinical treatment. *The Lancet Neurology*, 10(1):83–98, 2011.

[2] Guillaume Rigaut, Anna Shevchenko, Berthold Rutz, Matthias Wilm, Matthias Mann, and Bertrand Séraphin. A generic protein purification method for protein complex characterization and proteome exploration. *Nature biotechnology*, 17(10):1030, 1999.

[3] Kirill Tarassov, Vincent Messier, Christian R Landry, Stevo Radinovic, Mercedes M Serna Molina, Igor Shames, Yelena Malitskaya, Jackie Vogel, Howard Bussey, and Stephen W Michnick. An in vivo map of the yeast protein interactome. *Science*, 320(5882):1465–1470, 2008.

[4] Stanley Fields and Ok-kyu Song. A novel genetic system to detect protein-protein interactions. *Nature*, 340(6230):245, 1989.

[5] Victor Spirin and Leonid A Mirny. Protein complexes and functional modules in molecular networks. *Proceedings of the National Academy of Sciences*, 100(21):12123–12128, 2003.

[6] Ruoming Jin, Scott McCallen, Chun-Chi Liu, Yang Xiang, Eivind Almaas, and Xianghong Jasmine Zhou. Identifying dynamic network modules with temporal and spatial constraints. In *Biocomputing 2009*, pages 203–214. World Scientific, 2009.

[7] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3-5):75–174, 2010.

[8] Conrad Lee, Aaron McDaid, Fergal Reid, and Neil J Hurley. Detecting highly overlapping community structure by greedy clique expansion. In *Proceedings of the 4th Workshop on Social Network Mining and Analysis held in Conjunction with the International Conference on Knowledge Discovery and Data Mining (SNA/KDD'10)*, pages 33–42, 2010.

[9] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1):016118, 2009.

[10] Robert Gentleman and Wolfgang Huber. Making the most of high-throughput protein-interaction data. *Genome biology*, 8(10):112, 2007.

[11] Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[12] Lars Juhl Jensen, Ramneek Gupta, H-H Staerfeldt, and Søren Brunak. Prediction of human protein function according to gene ontology categories. *Bioinformatics*, 19(5):635–642, 2003.

[13] Yu Chen and Dong Xu. Genome-scale protein function prediction in yeast saccharomyces cerevisiae through integrating multiple sources of high-throughput data. In *Pacific Symposium on Biocomputing*, volume 10, pages 471–482. World Scientific, 2005.

[14] Naoki Nariai, Eric D Kolaczyk, and Simon Kasif. Probabilistic protein function prediction from heterogeneous genome-wide data. *PLoS One*, 2(3):e337, 2007.

[15] Xiaomei Wu, Lei Zhu, Jie Guo, Da-Yong Zhang, and Kui Lin. Prediction of yeast protein-protein interaction network: insights from the gene ontology and annotations. *Nucleic acids research*, 34(7):2137–2150, 2006.

[16] Ulrich Stelzl, Uwe Worm, Maciej Lalowski, Christian Haenig, Felix H Brembeck, Heike Goehler, Martin Stroedicke, Martina Zenkner, Anke Schoenherr, Susanne Koeppen, et al. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–968, 2005.

[17] Daniel R Rhodes, Scott A Tomlins, Sooryanarayana Varambally, Vasudeva Mahavisno, Terrence Barrette, Shanker Kalyana-Sundaram, Debashis Ghosh, Akhilesh Pandey, and Arul M Chinnaiyan. Probabilistic model of the human protein-protein interaction network. *Nature biotechnology*, 23(8):951–959, 2005.

[18] Ronglai Shen, Arul M Chinnaiyan, and Debashis Ghosh. Pathway analysis reveals functional convergence of gene expression profiles in breast cancer. *BMC medical genomics*, 1(1):28, 2008.

[19] Clara Pizzuti, Simona E Rombo, and Elena Marchiori. Complex detection in protein-protein interaction networks: a compact overview for researchers and practitioners. In *European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, pages 211–223. Springer, 2012.

[20] Gary D Bader and Christopher WV Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC bioinformatics*, 4(1):2, 2003.

[21] Balázs Adamcsek, Gergely Palla, Illés J Farkas, Imre Derényi, and Tamás Vicsek. Cfinder: locating cliques and overlapping modules in biological networks. *Bioinformatics*, 22(8):1021–1023, 2006.

[22] Tamás Nepusz, Haiyuan Yu, and Alberto Paccanaro. Detecting overlapping protein complexes in protein-protein interaction networks, 2012.

[23] Md Altaf-Ul-Amin, Yoko Shinbo, Kenji Mihara, Ken Kurokawa, and Shigehiko Kanaya. Development and implementation of an algorithm for detection of protein complexes in large interaction networks. *BMC bioinformatics*, 7(1):207, 2006.

[24] Zelmina Lubovac, Jonas Gamalielsson, and Björn Olsson. Combining functional and topological properties to identify core modules in protein interaction networks. *Proteins: Structure, Function, and Bioinformatics*, 64(4):948–959, 2006.

[25] Xiao-Li Li, Chuan-Sheng Foo, and See-Kiong Ng. Discovering protein complexes in dense reliable neighborhoods of protein interaction networks. In *Computational Systems Bioinformatics: (Volume 6)*, pages 157–168. World Scientific, 2007.

[26] Clara Pizzuti and Simona E Rombo. Pincoc: a co-clustering based approach to analyze protein-protein interaction networks. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 821–830. Springer, 2007.

[27] Hon Nian Chua, Kang Ning, Wing-Kin Sung, Hon Wai Leong, and Limsoon Wong. Using indirect protein-protein interactions for protein complex prediction. *Journal of bioinformatics and computational biology*, 6(03):435–466, 2008.

[28] Christos Dimitrakopoulos, Konstantinos Theofilatos, Andreas Pegkas, Spiros Likothanassis, and Seferina Mavroudi. Predicting overlapping protein complexes from weighted protein interaction graphs by gradually expanding dense neighborhoods. *Artificial intelligence in medicine*, 71:62–69, 2016.

[29] Jie Wang, Wenping Zheng, Yuhua Qian, and Jiye Liang. A seed expansion graph clustering method for protein complexes detection in protein interaction networks. *Molecules*, 22(12):2179, 2017.

[30] Yanjun Qi, Fernanda Balem, Christos Faloutsos, Judith Klein-Seetharaman, and Ziv Bar-Joseph. Protein complex identification by supervised graph local clustering. *Bioinformatics*, 24(13):i250–i268, 2008.

[31] Andrew D King, Natasa Pržulj, and Igor Jurisica. Protein complex prediction via cost-based clustering. *Bioinformatics*, 20(17):3013–3020, 2004.

[32] Jianhua Ruan and Weixiong Zhang. Identifying network communities with a high resolution. *Physical Review E*, 77(1):016104, 2008.

[33] István A Kovács, Robin Palotai, Máté S Szalay, and Peter Csermely. Community landscapes: an integrative approach to determine overlapping network module hierarchy, identify key nodes and predict network dynamics. *PloS one*, 5(9):e12528, 2010.

[34] Anton J Enright, Stijn Van Dongen, and Christos A Ouzounis. An efficient algorithm for large-scale detection of protein families. *Nucleic acids research*, 30(7):1575–1584, 2002.

[35] Kathy Macropol, Tolga Can, and Ambuj K Singh. Rrw: repeated random walks on genome-scale protein networks for local cluster discovery. *BMC bioinformatics*, 10(1):283, 2009.

[36] Young-Rae Cho, Woochang Hwang, and Aidong Zhang. Identification of overlapping functional modules in protein interaction networks: information flow-based approach. In *Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference on*, pages 147–152. IEEE, 2006.

[37] Woochang Hwang, Young-Rae Cho, Aidong Zhang, and Murali Ramanathan. A novel functional module detection algorithm for protein-protein interaction networks. *Algorithms for Molecular Biology*, 1(1):24, 2006.

[38] Zehua Zhang, Jian Song, Jijun Tang, Xinying Xu, and Fei Guo. Detecting complexes from edge-weighted ppi networks via genes expression analysis. *BMC systems biology*, 12(4):40, 2018.

[39] Manoj Pratim Samanta and Shoudan Liang. Predicting protein functions from redundancies in large-scale protein interaction networks. *Proceedings of the National Academy of Sciences*, 100(22):12579–12583, 2003.

[40] Victor Farutin, Keith Robison, Eric Lightcap, Vlado Dancik, Alan Ruttenberg, Stanley Letovsky, and Joel Pradines. Edge-count probabilities for the identification of local protein communities and their organization. *Proteins: Structure, Function, and Bioinformatics*, 62(3):800–818, 2006.

[41] Hongbiao Liu and Juan Liu. Clustering protein interaction data through chaotic genetic algorithm. In *Asia-Pacific Conference on Simulated Evolution and Learning*, pages 858–864. Springer, 2006.

[42] Hamid Ravaee, Ali Masoudi-Nejad, Saeed Omidi, and Ali Moeini. Improved immune genetic algorithm for clustering protein-protein interaction network. In *2010 IEEE International Conference on BioInformatics and BioEngineering*, pages 174–179. IEEE, 2010.

[43] Xiaoli Li, Min Wu, Chee-Keong Kwoh, and See-Kiong Ng. Computational approaches for detecting protein complexes from protein interaction networks: a survey. *BMC genomics*, 11(1):S3, 2010.

[44] Brian P Kelley, Roded Sharan, Richard M Karp, Taylor Sittler, David E Root, Brent R Stockwell, and Trey Ideker. Conserved pathways within bacteria and yeast as revealed by global protein network alignment. *Proceedings of the National Academy of Sciences*, 100(20):11394–11399, 2003.

[45] Roded Sharan, Silpa Suthram, Ryan M Kelley, Tanja Kuhn, Scott McCuine, Peter Uetz, Taylor Sittler, Richard M Karp, and Trey Ideker. Conserved patterns of protein interaction in multiple species. *Proceedings of the National Academy of Sciences*, 102(6):1974–1979, 2005.

[46] Mehmet Koyutürk, Yohan Kim, Umut Topkara, Shankar Subramaniam, Wojciech Szpankowski, and Ananth Grama. Pairwise alignment of protein interaction networks. *Journal of Computational Biology*, 13(2):182–199, 2006.

[47] Manikandan Narayanan and Richard M Karp. Comparing protein interaction networks via a graph match-and-split algorithm. *Journal of Computational Biology*, 14(7):892–907, 2007.

[48] Wei Peng, Jianxin Wang, Fangxiang Wu, and Pan Yi. Detecting conserved protein complexes using a dividing-and-matching algorithm and unequally lenient criteria for network comparison. *Algorithms for Molecular Biology*, 10(1):21, 2015.

[49] Sourav Bandyopadhyay, Roded Sharan, and Trey Ideker. Systematic identification of functional orthologs based on protein network comparison. *Genome research*, 16(3):428–435, 2006.

[50] Rohit Singh, Jinbo Xu, and Bonnie Berger. Global alignment of multiple protein interaction networks with application to functional orthology detection. *PNAS*, 105(35):12763–12768, 2008.

[51] Oleksii Kuchaiev, Tijana Milenković, Vesna Memišević, Wayne Hayes, and Nataša Pržulj. Topological network alignment uncovers biological function and phylogeny. *Journal of The Royal Society Interface*, 7(50):1341–1354, 2010.

[52] Hang TT Phan and Michael JE Sternberg. Pinalog: a novel approach to align protein interaction networks – implications for complex detection and function prediction. *Bioinformatics*, 28(9):1239–1245, 2012.

[53] Feng Ying Yu, Zhi Hao Yang, Nan Tang, Hong Fei Lin, Jian Wang, and Zhi Wei Yang. Predicting protein complex in protein interaction network-a supervised learning based method. *BMC systems biology*, 8(3):S4, 2014.

[54] Anne-Claude Gavin, Patrick Aloy, Paola Grandi, Roland Krause, Markus Boesche, Martina Marzioch, Christina Rau, Lars Juhl Jensen, Sonja Bastuck, Birgit Dümpelfeld, et al. Proteome survey reveals modularity of the yeast cell machinery. *Nature*, 440(7084):631, 2006.

[55] Henry CM Leung, Qian Xiang, Siu-Ming Yiu, and Francis YL Chin. Predicting protein complexes from ppi data: a core-attachment approach. *Journal of Computational Biology*, 16(2):133–144, 2009.

[56] Min Wu, Xiaoli Li, Chee-Keong Kwoh, and See-Kiong Ng. A core-attachment based method to detect protein complexes in ppi networks. *BMC bioinformatics*, 10(1):169, 2009.

[57] Xianjun Shen, Li Yi, Xingpeng Jiang, Tingting He, Jincai Yang, Wei Xie, Po Hu, and Xiaohua Hu. Identifying protein complex by integrating characteristic of core-attachment into dynamic ppi network. *PloS one*, 12(10):e0186134, 2017.

[58] Suk Hoon Jung, Woo-Hyuk Jang, HEE-YUNG HUR, Bora Hyun, and Dong-Soo Han. Protein complex prediction based on mutually exclusive interactions in protein interaction network. In *Genome Informatics 2008: Genome Informatics Series Vol. 21*, pages 77–88. World Scientific, 2008.

[59] Xiao-Li Li, Chuan-Sheng Foo, Soon-Heng Tan, and See-Kiong Ng. Interaction graph mining for protein complexes using local clique merging. *Genome Informatics*, 16(2):260–269, 2005.

[60] Jianxing Feng, Rui Jiang, and Tao Jiang. A max-flow based approach to the identification of protein complexes using protein interaction and microarray data. In *Computational Systems Bioinformatics: (Volume 7)*, pages 51–62. World Scientific, 2008.

[61] Ioannis A Maraziotis, Konstantina Dimitrakopoulou, and Anastasios Bezerianos. Growing functional modules from a seed protein via integration of protein interaction and gene expression data. *Bmc Bioinformatics*, 8(1):408, 2007.

[62] Eileen Marie Hanna, Nazar Zaki, and Amr Amin. Detecting protein complexes in protein interaction networks modeled as gene expression biclusters. *PloS one*, 10(12):e0144163, 2015.

[63] Young-Rae Cho, Woochang Hwang, Murali Ramanathan, and Aidong Zhang. Semantic integration to identify overlapping functional modules in protein interaction networks. *BMC bioinformatics*, 8(1):265, 2007.

[64] Nevan J Krogan, Gerard Cagney, Haiyuan Yu, Gouqing Zhong, Xinghua Guo, Alexandr Ignatchenko, Joyce Li, Shuye Pu, Nira Datta, Aaron P Tikuisis, et al. Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature*, 440(7084):637, 2006.

[65] Shuye Pu, Jim Vlasblom, Andrew Emili, Jack Greenblatt, and Shoshana J Wodak. Identifying functional modules in the physical interactome of saccharomyces cerevisiae. *Proteomics*, 7(6):944–960, 2007.

[66] Caroline C Friedel, Jan Krumsiek, and Ralf Zimmer. Bootstrapping the interactome: unsupervised identification of protein complexes in yeast. In *Annual International Conference on Research in Computational Molecular Biology*, pages 3–16. Springer, 2008.

[67] Guy Geva and Roded Sharan. Identification of protein complexes from co-immunoprecipitation data. *Bioinformatics*, 27(1):111–117, 2010.

[68] Jose B Pereira-Leal, Anton J Enright, and Christos A Ouzounis. Detection of functional modules from protein interaction networks. *PROTEINS: Structure, Function, and Bioinformatics*, 54(1):49–57, 2004.

[69] Shuye Pu, Jessica Wong, Brian Turner, Emerson Cho, and Shoshana J Wodak. Up-to-date catalogues of yeast protein complexes. *Nucleic acids research*, 37(3):825–831, 2009.

[70] Sean R Collins, Patrick Kemmeren, Xue-Chu Zhao, Jack F Greenblatt, Forrest Spencer, Frank CP Holstege, Jonathan S Weissman, and Nevan J Krogan. Toward a comprehensive atlas of the physical interactome of saccharomyces cerevisiae. *Molecular & Cellular Proteomics*, 6(3):439–450, 2007.

[71] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

[72] Mark EJ Newman. Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133, 2004.

[73] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[74] Jake M Hofman and Chris H Wiggins. Bayesian approach to network modularity. *Physical review letters*, 100(25):258701, 2008.

[75] Andrea Lancichinetti, Filippo Radicchi, José J Ramasco, and Santo Fortunato. Finding statistically significant communities in networks. *PloS one*, 6(4):e18961, 2011.

[76] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

[77] Steve Gregory. Finding overlapping communities in networks by label propagation. *New Journal of Physics*, 12(10):103018, 2010.

[78] Yong-Yeol Ahn, James P Bagrow, and Sune Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, 2010.

[79] Andrea Lancichinetti, Santo Fortunato, and Janos Kertesz. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, 2009.

[80] Jierui Xie, Stephen Kelley, and Boleslaw K Szymanski. Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM Computing Surveys (CSUR)*, 45(4):43, 2013.

[81] Erin N Sawardecker, Marta Sales-Pardo, and Luıs A Nunes Amaral. Detection of node group membership in networks with group overlap. *The European Physical Journal B-Condensed Matter and Complex Systems*, 67(3):277–284, 2009.

[82] Shihua Zhang, Rui-Sheng Wang, and Xiang-Sun Zhang. Identification of overlapping community structure in complex networks using fuzzy c-means clustering. *Physica A: Statistical Mechanics and its Applications*, 374(1):483–490, 2007.

[83] Aaron Clauset, Cristopher Moore, and Mark EJ Newman. Hierarchical structure and the prediction of missing links in networks. *Nature*, 453(7191):98, 2008.

[84] Min Li, Jian-er Chen, Jian-xin Wang, Bin Hu, and Gang Chen. Modifying the dpclus algorithm for identifying protein complexes based on new topological structures. *Bmc Bioinformatics*, 9(1):398, 2008.

[85] Leon Danon, Albert Diaz-Guilera, Jordi Duch, and Alex Arenas. Comparing community structure identification. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(09):P09008, 2005.

[86] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: A comparative analysis. *Physical review E*, 80(5):056117, 2009.

[87] Catia Pesquita, Daniel Faria, Andre O Falcao, Phillip Lord, and Francisco M Couto. Semantic similarity in biomedical ontologies. *PLoS computational biology*, 5(7):e1000443, 2009.

[88] Stefan R Maetschke, Martin Simonsen, Melissa J Davis, and Mark A Ragan. Gene ontology-driven inference of protein-protein interactions using inducers. *Bioinformatics*, 28(1):69–75, 2012.

[89] George D Montanez and Young-Rae Cho. Assessing reliability of protein-protein interactions by gene ontology integration. In *Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2012 IEEE Symposium on*, pages 21–27. IEEE, 2012.

[90] George Montañez and Young-Rae Cho. Predicting false positives of protein-protein interaction data by semantic similarity measures §. *Current Bioinformatics*, 8(3):339–346, 2013.

[91] Guangyu Cui and Kyungsook Han. Scoring protein-protein interactions using the width of gene ontology terms and the information content of common ancestors. *Emerging Intelligent Computing Technology and Applications*, 2:31–36, 2013.

[92] Guangyu Cui, Byungmin Kim, Saud Alguwaizani, and Kyungsook Han. Assessing protein-protein interactions based on the semantic similarity of interacting proteins. *International journal of data mining and bioinformatics*, 13(1):75–83, 2015.

[93] Shu-Bo Zhang and Qiang-Rong Tang. Protein-protein interaction inference based on semantic similarity of gene ontology terms. *Journal of theoretical biology*, 401:30–37, 2016.

[94] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th international joint conference on Artificial intelligenc*, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[95] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, volume 98, pages 296–304, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[96] Andreas Schlicker, Francisco S Domingues, Jörg Rahnenführer, and Thomas Lengauer. A new measure for functional similarity of gene products based on gene ontology. *BMC bioinformatics*, 7(1):302, 2006.

[97] Jay J Jiang and David W Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of 10th International Conference on Research In Computational Linguistics (ROCLING'97)*, 1997.

[98] James Z Wang, Zhidian Du, Rapeeporn Payattakool, S Yu Philip, and Chin-Fu Chen. A new method to measure the semantic similarity of go terms. *Bioinformatics*, 23(10):1274–1281, 2007.

[99] Shobhit Jain and Gary Bader. An improved method for scoring protein-protein interactions using semantic similarity within the gene ontology. *BMC bioinformatics*, 11(1):562, 2010.

[100] Xiang Guo, Rongxiang Liu, Craig D Shriver, Hai Hu, and Michael N Liebman. Assessing semantic similarity measures for the characterization of human regulatory pathways. *Bioinformatics*, 22(8):967–973, 2006.

[101] Tao Xu, LinFang Du, and Yan Zhou. Evaluation of go-based functional similarity measures using s. cerevisiae protein interaction and expression profile data. *BMC bioinformatics*, 9(1):472, 2008.

[102] Catia Pesquita. Semantic similarity in the gene ontology. In *The Gene Ontology Handbook*, pages 161–173. Springer, 2017.

[103] Martin H Schaefer, Jean-Fred Fontaine, Arunachalam Vinayagam, Pablo Porras, Erich E Wanker, and Miguel A Andrade-Navarro. Hippie: Integrating protein interaction networks with experiment based quality scores. *PloS one*, 7(2):e31826, 2012.

[104] Ioannis Xenarios, Danny W Rice, Lukasz Salwinski, Marisa K Baron, Edward M Marcotte, and David Eisenberg. Dip: the database of interacting proteins. *Nucleic acids research*, 28(1):289–291, 2000.

[105] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[106] Catia Pesquita, Delphine Pessoa, Daniel Faria, and Francisco Couto. Cessm: Collaborative evaluation of semantic similarity measures. *JB2009: Challenges in Bioinformatics*, 157:190, 2009.

[107] Sébastien Harispe, David Sánchez, Sylvie Ranwez, Stefan Janaqi, and Jacky Montmain. A framework for unifying ontology-based semantic similarity measures: A study in the biomedical domain. *Journal of biomedical informatics*, 48:38–53, 2014.

[108] Gaston K Mazandu, Emile R Chimusa, and Nicola J Mulder. Gene ontology semantic similarity tools: survey on features and challenges for biological knowledge discovery. *Briefings in bioinformatics*, 18(5):886–901, 2016.

[109] Qiangfeng Cliff Zhang, Donald Petrey, Lei Deng, Li Qiang, Yu Shi, Chan Aye Thu, Brygida Bisikirska, Celine Lefebvre, Domenico Accili, Tony Hunter, et al. Structure-based prediction of protein-protein interactions on a genome-wide scale. *Nature*, 490(7421):556, 2012.

[110] Christian Von Mering, Lars J Jensen, Michael Kuhn, Samuel Chaffron, Tobias Doerks, Beate Krüger, Berend Snel, and Peer Bork. String 7 – recent developments in the integration and prediction of protein interactions. *Nucleic acids research*, 35(suppl_1):D358–D362, 2006.

[111] Trey Ideker, Owen Ozier, Benno Schwikowski, and Andrew F Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18(suppl_1):S233–S240, 2002.

[112] Florencio Pazos and Alfonso Valencia. Similarity of phylogenetic trees as indicator of protein-protein interaction. *Protein engineering*, 14(9):609–614, 2001.

[113] Raja Jothi, Maricel G Kann, and Teresa M Przytycka. Predicting protein-protein interaction by searching evolutionary tree automorphism space. *Bioinformatics*, 21(suppl_1):i241–i250, 2005.

[114] Matteo Pellegrini, Edward M Marcotte, Michael J Thompson, David Eisenberg, and Todd O Yeates. Assigning protein functions by comparative genome analysis: protein

phylogenetic profiles. *Proceedings of the National Academy of Sciences*, 96(8):4285–4288, 1999.

[115] Hon Nian Chua, Wing-Kin Sung, and Limsoon Wong. Exploiting indirect neighbours and topological weight to predict protein function from protein-protein interactions. *Bioinformatics*, 22(13):1623–1630, 2006.

[116] Jin Chen, Wynne Hsu, Mong Li Lee, and See-Kiong Ng. Increasing confidence of protein interactomes using network topological metrics. *Bioinformatics*, 22(16):1998–2004, 2006.

[117] Guimei Liu, Limsoon Wong, and Hon Nian Chua. Complex discovery from weighted ppi networks. *Bioinformatics*, 25(15):1891–1897, 2009.

[118] Samira Jaeger, Sylvain Gaudan, Ulf Leser, and Dietrich Rebholz-Schuhmann. Integrating protein-protein interactions and text mining for protein function prediction. In *BMC bioinformatics*, volume 9, page S2. BioMed Central, 2008.

[119] Ashwini Patil and Haruki Nakamura. Filtering high-throughput protein-protein interaction data using a combination of genomic features. *BMC bioinformatics*, 6(1):100, 2005.

[120] Yue Deng, Lin Gao, and Bingbo Wang. ppipre: predicting protein-protein interactions by combining heterogeneous features. *BMC systems biology*, 7(2):S8, 2013.

[121] Amos Tversky. Features of similarity. *Psychological review*, 84(4):327, 1977.

[122] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *IEEE transactions on systems, man, and cybernetics*, 19(1):17–30, 1989.

[123] Phillip W. Lord, Robert D. Stevens, Andy Brass, and Carole A. Goble. Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation. *Bioinformatics*, 19(10):1275–1283, 2003.

[124] PW Lord, RD Stevens, A Brass, and CA Goble. Semantic similarity measures as tools for exploring the gene ontology. In *Pacific Symposium on Biocomputing*, pages 601–612, 2003.

[125] Anurag Nagar and Hisham Al-Mubaid. A new path length measure based on go for gene similarity with evaluation using sgd pathways. In *Computer-Based Medical Systems, 2008. CBMS'08. 21st IEEE International Symposium on*, pages 590–595. IEEE, 2008.

[126] Kamal Taha. Determining the semantic similarities among gene ontology terms. *IEEE journal of biomedical and health informatics*, 17(3):512–525, 2013.

[127] Sanghamitra Bandyopadhyay and Koushik Mallick. A new path based hybrid measure for gene ontology similarity. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 11(1):116–127, 2014.

[128] Hui Yu, Lei Gao, Kang Tu, and Zheng Guo. Broadly predicting specific gene functions with expression similarity and taxonomy similarity. *Gene*, 352:75–81, 2005.

[129] Jill Cheng, Melissa Cline, John Martin, David Finkelstein, Tarif Awad, David Kulp, and Michael A Siani-Rose. A knowledge-based clustering algorithm driven by gene ontology. *Journal of biopharmaceutical statistics*, 14(3):687–700, 2004.

[130] Hongwei Wu, Zhengchang Su, Fenglou Mao, Victor Olman, and Ying Xu. Prediction of functional modules based on comparative genome analysis and gene ontology application. *Nucleic acids research*, 33(9):2822–2837, 2005.

[131] Angela del Pozo, Florencio Pazos, and Alfonso Valencia. Defining functional distances over gene ontology. *BMC bioinformatics*, 9(1):50, 2008.

[132] Nuno Seco, Tony Veale, and Jer Hayes. An intrinsic information content metric for semantic similarity in wordnet. In *ECAI*, volume 16, page 1089, 2004.

[133] Francisco M Couto, Mário J Silva, and Pedro M Coutinho. Semantic similarity over the gene ontology: family correlation and selecting disjunctive ancestors. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 343–344. ACM, 2005.

[134] Zili Zhou, Yanna Wang, and Junzhong Gu. A new model of information content for semantic similarity in wordnet. In *Future Generation Communication and Networking Symposia, 2008. FGCNS'08. Second International Conference on*, volume 3, pages 85–89. IEEE, 2008.

[135] Meeta Mistry and Paul Pavlidis. Gene ontology term overlap as a measure of gene functional similarity. *BMC bioinformatics*, 9(1):327, 2008.

[136] Bo Li, James Z Wang, F Alex Feltus, Jizhong Zhou, and Feng Luo. Effectively integrating information content and structural relationship to improve the go-based similarity measure between proteins. In *Proceedings of BIOCOMP'10*, pages 166–172, 2010.

[137] David Sánchez, Montserrat Batet, and David Isern. Ontology-based information content computation. *Knowledge-Based Systems*, 24(2):297–303, 2011.

[138] Gaston K Mazandu and Nicola J Mulder. A topology-based metric for measuring term similarity in the gene ontology. *Advances in bioinformatics*, 2012:975783–975783, 2012.

[139] David Sánchez and Montserrat Batet. A new model to compute the information content of concepts from taxonomic knowledge. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 8(2):34–50, 2012.

[140] Xuebo Song, Lin Li, Pradip K Srimani, Philip S Yu, and James Z Wang. Measure the semantic similarity of go terms using aggregate information content. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 11(3):468–476, 2014.

[141] Shu-Bo Zhang and Jian-Huang Lai. Semantic similarity measurement between gene ontology terms based on exclusively inherited shared information. *Gene*, 558(1):108–117, 2015.

[142] Abhijit Adhikari, Shivang Singh, Animesh Dutta, and Biswanath Dutta. A novel information theoretic approach for finding semantic similarity in wordnet. In *TENCON 2015-2015 IEEE Region 10 Conference*, pages 1–6. IEEE, 2015.

[143] Juan J Lastra-Díaz and Ana García-Serrano. A new family of information content models with an experimental survey on wordnet. *Knowledge-Based Systems*, 89:509–526, 2015.

[144] Xiaomei Wu, Erli Pang, Kui Lin, and Zhen-Ming Pei. Improving the measurement of semantic similarity between gene ontology terms and gene products: insights from an edge-and ic-based hybrid method. *PloS one*, 8(5):e66745, 2013.

[145] Lizhen Liu, Xuemin Dai, Chao Du, Hanshi Wang, and Jingli Lu. A new hybrid semantic similarity computation method based on gene ontology. In *Software Engineering and Service Science (ICSESS), 2014 5th IEEE International Conference on*, pages 849–853. IEEE, 2014.

[146] Yungang Xu, Maozu Guo, Wenli Shi, Xiaoyan Liu, and Chunyu Wang. A novel insight into gene ontology semantic similarity. *Genomics*, 101(6):368–375, 2013.

[147] Zhixia Teng, Maozu Guo, Xiaoyan Liu, Qiguo Dai, Chunyu Wang, and Ping Xuan. Measuring gene functional similarity based on group-wise comparison of go terms. *Bioinformatics*, 29(11):1424–1432, 2013.

[148] Jose L Sevilla, Victor Segura, Adam Podhorski, Elizabeth Guruceaga, Jose M Mato, Luis A Martinez-Cruz, Fernando J Corrales, and Angel Rubio. Correlation between gene expression and go semantic similarity. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(4):330–338, 2005.

[149] Francisco M Couto, Mário J Silva, and Pedro M Coutinho. Measuring semantic similarity between gene ontology terms. *Data & knowledge engineering*, 61(1):137–152, 2007.

[150] Francisco Azuaje, Haiying Wang, and Olivier Bodenreider. Ontology-driven similarity approaches to supporting gene functional assessment. In *Proceedings of the ISMB'2005 SIG meeting on Bio-ontologies*, pages 9–10, 2005.

[151] Won-Ki Huh, James V Falvo, Luke C Gerke, Adam S Carroll, Russell W Howson, Jonathan S Weissman, and Erin K O'Shea. Global analysis of protein localization in budding yeast. *Nature*, 425(6959):686–691, 2003.

[152] Pingzhao Hu, Gary Bader, Dennis A Wigle, and Andrew Emili. Computational prediction of cancer-gene function. *Nature Reviews Cancer*, 7(1):23–34, 2007.

[153] Catia Pesquita, Daniel Faria, Hugo Bastos, António EN Ferreira, André O Falcão, and Francisco M Couto. Metrics for go based protein semantic similarity: a systematic evaluation. In *BMC bioinformatics*, volume 9, page S4. BioMed Central, 2008.

[154] Chris Stark, Bobby-Joe Breitkreutz, Andrew Chatr-Aryamontri, Lorrie Boucher, Rose Oughtred, Michael S Livstone, Julie Nixon, Kimberly Van Auken, Xiaodong Wang, Xiaoqi Shi, et al. The biogrid interaction database: 2011 update. *Nucleic acids research*, 39(suppl 1):D698–D704, 2011.

[155] Lukasz Salwinski, Christopher S Miller, Adam J Smith, Frank K Pettit, James U Bowie, and David Eisenberg. The database of interacting proteins: 2004 update. *Nucleic acids research*, 32(suppl 1):D449–D451, 2004.

[156] TS Keshava Prasad, Renu Goel, Kumaran Kandasamy, Shivakumar Keerthikumar, Sameer Kumar, Suresh Mathivanan, Deepthi Telikicherla, Rajesh Raju, Beema Shafreen, Abhilash Venugopal, et al. Human protein reference database - 2009 update. *Nucleic acids research*, 37(suppl 1):D767–D772, 2009.

[157] Bruno Aranda, P Achuthan, Yasmin Alam-Faruque, I Armean, Alan Bridge, C Derow, Marc Feuermann, AT Ghanbarian, Samuel Kerrien, Jyoti Khadake, et al. The intact molecular interaction database in 2010. *Nucleic acids research*, 38(suppl 1):D525–D531, 2010.

[158] Arnaud Ceol, Andrew Chatr Aryamontri, Luana Licata, Daniele Peluso, Leonardo Briganti, Livia Perfetto, Luisa Castagnoli, and Gianni Cesareni. Mint, the molecular interaction database: 2009 update. *Nucleic acids research*, page gkp983, 2009.

[159] Gary D Bader, Ian Donaldson, Cheryl Wolting, BF Francis Ouellette, Tony Pawson, and Christopher WV Hogue. Bind - the biomolecular interaction network database. *Nucleic acids research*, 29(1):242–245, 2001.

[160] Philipp Pagel, Stefan Kovac, Matthias Oesterheld, Barbara Brauner, Irmtraud Dunger-Kaltenbach, Goar Frishman, Corinna Montrone, Pekka Mark, Volker Stümpflen, Hans-Werner Mewes, et al. The mips mammalian protein-protein interaction database. *Bioinformatics*, 21(6):832–834, 2005.

[161] Sidahmed Benabderrahmane, Malika Smail-Tabbone, Olivier Poch, Amedeo Napoli, and Marie-Dominique Devignes. Intelligo: a new vector-based semantic similarity measure including annotation origin. *BMC bioinformatics*, 11(1):588, 2010.

[162] Xueping Yu, Anders Wallqvist, and Jaques Reifman. Inferring high-confidence human protein-protein interactions. *BMC bioinformatics*, 13(1):79, 2012.

[163] Sabry Razick, George Magklaras, and Ian M Donaldson. irefindex: a consolidated protein interaction database with provenance. *BMC bioinformatics*, 9(1):1, 2008.

[164] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015.

[165] Guangchuang Yu, Fei Li, Yide Qin, Xiaochen Bo, Yibo Wu, and Shengqi Wang. Gosemsim: an r package for measuring semantic similarity among go terms and gene products. *Bioinformatics*, 26(7):976–978, 2010.

[166] Marc Carlson. Go.db: A set of annotation maps describing the entire. gene ontology. 2013. *R package version*, 3(2), 2013.

[167] M Carlson, S Falcon, H Pages, and N Li. org.hs.eg.db: Genome wide annotation for human, 2013.

[168] M Carlson, S Falcon, H Pages, and N Li. org.sc.sgd.db: Genome wide annotation for yeast. *R package version*, 2(1), 2014.

[169] V Carey and H Redestig. Roc: Utilities for roc, with uarray focus. r package version 1.16. 0, 2008.

[170] Tobias Sing, Oliver Sander, Niko Beerenwinkel, and Thomas Lengauer. Rocr: visualizing classifier performance in r. *Bioinformatics*, 21(20):3940–3941, 2005.

[171] Ismael A Vergara, Tomás Norambuena, Evandro Ferrada, Alex W Slater, and Francisco Melo. Star: a simple tool for the statistical comparison of roc curves. *BMC bioinformatics*, 9(1):1–5, 2008.

[172] Guimei Liu, Chern Han Yong, Hon Nian Chua, and Limsoon Wong. Decomposing ppi networks for complex discovery. In *Proteome science*, volume 9, page S15. BioMed Central, 2011.

[173] Chern Han Yong, Guimei Liu, Hon Nian Chua, and Limsoon Wong. Supervised maximum-likelihood weighting of composite protein networks for complex prediction. In *BMC systems biology*, volume 6, page S13. BioMed Central, 2012.

[174] Michael A Gilchrist, Laura A Salter, and Andreas Wagner. A statistical framework for combining and interpreting proteomic datasets. *Bioinformatics*, 20(5):689–700, 2004.

[175] Xiaotong Lin, Mei Liu, and Xue-wen Chen. Assessing reliability of protein-protein interactions by integrative analysis of data in model organisms. *BMC bioinformatics*, 10(Suppl 4):S5, 2009.

[176] Charlotte M Deane, Łukasz Salwiński, Ioannis Xenarios, and David Eisenberg. Protein interactions two methods for assessment of the reliability of high throughput observations. *Molecular & Cellular Proteomics*, 1(5):349–356, 2002.

[177] Minghua Deng, Fengzhu Sun, and Ting Chen. Assessment of the reliability of protein-protein interactions and protein function prediction. In *Pac. Symp. Biocomputing (PSB 2003)*, pages 140–151, 2002.

[178] Christine Brun, François Chevenet, David Martin, Jérôme Wojcik, Alain Guénoche, Bernard Jacq, et al. Functional classification of proteins for the prediction of cellular function from a protein-protein interaction network. *Genome biology*, 5(1):R6–R6, 2004.

[179] Sylvain Brohee and Jacques Van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC bioinformatics*, 7(1):488, 2006.

[180] Sriganesh Srihari and Hon Wai Leong. A survey of computational methods for protein complex prediction from protein interaction networks. *Journal of bioinformatics and computational biology*, 11(02):1230002, 2013.

[181] Sriganesh Srihari, Chern Han Yong, and Limsoon Wong. *Computational prediction of protein complexes from protein interaction networks*. Morgan & Claypool, 2017.

[182] William J Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.

[183] J Cohen. A coefficient of agreement for nominal scales. educational and psychosocial measurement, 20, 37-46, 1960.

[184] Haiyuan Yu, Pascal Braun, Muhammed A Yıldırım, Irma Lemmens, Kavitha Venkatesan, Julie Sahalie, Tomoko Hirozane-Kishikawa, Fana Gebreab, Na Li, Nicolas Simonis, et al. High-quality binary protein interaction map of the yeast interactome network. *Science*, 322(5898):104–110, 2008.

[185] Birgit HM Meldal, Oscar Forner-Martinez, Maria C Costanzo, Jose Dana, Janos Demeter, Marine Dumousseau, Selina S Dwight, Anna Gaulton, Luana Licata, Anna N Melidoni, et al. The complex portal - an encyclopaedia of macromolecular complexes. *Nucleic acids research*, 43(D1):D479–D484, 2014.

[186] Marina Meilă. Comparing clusteringsâĂŤan information based distance. *Journal of multivariate analysis*, 98(5):873–895, 2007.

Publications

Journals

- **Madhusudan Paul**, Rishav Anand, Ashish Anand, "Detection of highly overlapping communities in complex networks," Journal of Medical Imaging and Health Informatics, vol. 5, no. 5, pp. 1099–1103, 2015.
- **Madhusudan Paul**, Ashish Anand, "A new family of similarity measures for scoring confidence of protein interactions using gene ontology," IEEE/ACM Transactions on Computational Biology and Bioinformatics (**Under Review**).
- **Madhusudan Paul**, Ashish Anand, Saptarshi Pyne, "Impact of the continuous evolution of gene ontology on the performance of similarity measures for scoring confidence of protein interactions," Computational Biology and Biomedical Informatics, SN Computer Science, Springer (**Under Review**).
- **Madhusudan Paul**, Ashish Anand, "Impact of low-confidence interactions on computational identification of protein complexes," Journal of Bioinformatics and Computational Biology (**Accepted**), DOI: 10.1142/S0219720020500250.

Conferences

- **Madhusudan Paul**, Rishav Anand, Ashish Anand, "Detection of highly overlapping communities in complex networks," 5th International Conference on Computational Systems-Biology and Bioinformatics (CSBio 2014), Singapore, 2014.
- **Madhusudan Paul**, Ashish Anand, Saptarshi Pyne, "Impact of the Continuous Evolution of Gene Ontology on Similarity Measures," 8th International Conference on Pattern Recognition and Machine Intelligence (PReMI 2019), Published in Lecture Notes in Computer Science (LNCS), vol. 11942, pp. 122–129, Springer.