

IDENTIFICATION OF ESSENTIAL PROTEINS BASED ON PROTEIN-PROTEIN INTERACTION (PPI) NETWORK

~Under the guidance of~

Mr. Sovan Saha



**Bachelor of Technology
(Computer Science & Engineering)
Department Of Computer Science & Engineering
Dr.Sudhir Chandra Sur Degree Engineering College
Maulana Abul Kalam Azad University of Technology
Kolkata, West Bengal, India**

**IDENTIFICATION OF ESSENTIAL PROTEINS BASED ON PROTEIN-PROTEIN
INTERACTION (PPI) NETWORK**

By

Ananya Chakraborty

Roll No : 25500115006

Registration No :152550110007

Anannya Bhattacharjee

Roll No :25500115005

Registration No :152550110006

Dyuti Giri

Roll No :25500115017

Registration No :152550110018

Alokananda Ghosh

Roll No : 25500115004

Registration No :152550110005

**Under the Guidance of
Mr. Sovan Saha**

**Bachelor of Technology(Computer Science & Engineering)
Dr.Sudhir Chandra Sur Degree Engineering College
Maulana Abul Kalam Azad University of Technology
Kolkata, West Bengal, India**

ACKNOWLEDGEMENT

We are very happy to acknowledge the numerous personalities involved in lending their help to make our project “IDENTIFICATION OF ESSENTIAL PROTEINS BASED ON PROTEIN-PROTEIN INTERACTION(PPI) NETWORK” a successful one.

Salutations to our beloved and esteemed institute “ DR. SUDHIR CHANDRA SUR DEGREE ENGINEERING COLLEGE ” for having well-qualified staff and labs with necessary equipment.

We express our sincere gratitude to Mrs. Mallika De for providing us the best facilities and without consecutive suggestions, we would not have been able to do this curriculum.

We express our gratitude to our internal guide Mr. Sovan Saha ,for giving the sort of encouragement in preparing the report , presenting the project , spirit and useful guidance.

We also thank our parents whose moral support and constant guidance made our efforts successful.

CERTIFICATE

It is certified that our project work entitled “IDENTIFICATION OF ESSENTIAL PROTEINS BASED ON PROTEIN-PROTEIN INTERACTION (PPI) NETWORK” is a bona fide work carried out by ANANYA CHAKRABORTY, ANANNYA BHATTACHARJEE, DYUTI GIRI, ALOKANANDA GHOSH in partial fulfillment of the award of Bachelor of Technology in Computer Science and Engineering of DR.SUDHIR CHANDRA SUR DEGREE ENGINEERING COLLEGE during the year 2018-2019.

It is certified that this report on this topic has not been submitted for any other examination and does not take part in any other course undergone by the candidates. I have no doubt that they have a good research potential.

Date :

Mrs.Mallika De
(HOD, CSE, DSCSDEC)

CERTIFICATE

It is certified that our project work entitled “IDENTIFICATION OF ESSENTIAL PROTEINS BASED ON PROTEIN-PROTEIN INTERACTION (PPI) NETWORK” is a bona fide work carried out by ANANYA CHAKRABORTY, ANANNYA BHATTACHARJEE, DYUTI GIRI, ALOKANANDA GHOSH in partial fulfillment of the award of Bachelor of Technology in Computer Science and Engineering of DR.SUDHIR CHANDRA SUR DEGREE ENGINEERING COLLEGE during the year 2018-2019.

It is certified that this report on this topic has not been submitted for any other examination and does not take part in any other course undergone by the candidates. I have no doubt that they have a good research potential.

Date :

Mr. Sovan Saha
(Asst.prof, CSE,DSCSDEC)

CERTIFICATE

It is certified that our project work entitled “IDENTIFICATION OF ESSENTIAL PROTEINS BASED ON PROTEIN-PROTEIN INTERACTION (PPI) NETWORK” is a bona fide work carried out by ANANYA CHAKRABORTY, ANANNYA BHATTACHARJEE, DYUTI GIRI, ALOKANANDA GHOSH in partial fulfillment of the award of Bachelor of Technology in Computer Science and Engineering of DR.SUDHIR CHANDRA SUR DEGREE ENGINEERING COLLEGE during the year 2018-2019.

It is certified that this report on this topic has not been submitted for any other examination and does not take part in any other course undergone by the candidates. I have no doubt that they have a good research potential.

Date :

Examiner Signature

ABSTRACT

Protein is one of the most essential components of a living cell. Any living organism requires a modest amount of protein to function well. With the advancement in science and technology, researchers have identified numerous protein sequences, while the functions of most of them remains unannotated. So, a considerable amount of research work is being carried out to study and observe the behaviour and the functions of the unannotated proteins. One of the most common approaches that is being followed since old times is to predict the functions of these target proteins from their corresponding neighbors. However, these predictions involve the presence of false positives in a certain amount. So, in the proposed computational model, a node weight based search for essential protein has been executed initially with the aim of identifying the denser subnetwork, within the entire protein interaction network. Now each protein in this dense subgraph is considered as target protein, the function of which has been evaluated by 6 different centrality combined and also by a recently developed centrality: LBCC, which formed the basis for the prediction of essential proteins. In this project, a statistical analysis has been undertaken to study the effectiveness of a feature ,i.e node weight, on 1) Combination of different centrality measures and 2) Recently developed method LBCC. The proposed methodology achieves an overall higher precision, recall and F-Score than the existing ones which highlights the fact that this work is far more efficient than most of the existing state-of-arts.

CONTENTS

	Page No.
CHAPTER 1: INTRODUCTION	01
1.1. Protein	02
1.2. Protein Classification	04
1.3. Protein Function Classification	09
1.4. Amino Acid Sequence	11
1.5. Protein Structure	13
1.6. Protein-Protein Interaction Network	15
1.7. Types of Protein-Protein Interaction	17
 CHAPTER 2: RELATED WORKS ON PROTEIN	
2.1. Overlay Structure	20
2.2. Previous Works	21

	Page No.
CHAPTER 3: PRESENT WORKS	
3.1.Motivation	23
3.2.Related Terminologies	24
3.3.Methodology	25
3.4.Present Works	27
3.5.Dataset Used	32
3.6.Tools Used	33
 CHAPTER 4: RESULTS	
4.1.Results And Discussions	41
4.2.Conclusion	47
 REFERENCES	48
 APPENDIX	51

FIGURE INDEXING

Figure No.	Figure Description	Page No.
1	Protein Structure	2
2	Fibrous and Globular Proteins	5
3	Conjugated Protein (Hemoglobin)	6
4	FABP2	7
5	Ferritin, The Iron Storage Protein	8
6	A Sample PPI Network	9
7	Amino Acid Sequence	11
8	PPI Network	15
9	Homo-oligomers vs. hetero-oligomers	17
10	Water Molecule Structure	19
11	Cytoscape 3 starting activity	34
12	Tour of Cytoscape Core Functionality	35
13	YMIPS data set	35
14	YDIP data set	36
15	Calculating and exporting the CC,DC,EC,LAC,NC,IC values of YMIPS	37
16	Full view of Sublime Text IDE	38
17	Windowed view of Notepad++	39
18	Windowed view of Google Sheets	40
19	Precision Recall F-score values of different methods on YMIPS dataset	45
20	Precision Recall F-score values of different methods on YDIP dataset	45

TABLE INDEXING

Table No.	Table Description	Page No.
1	Information on the two PPI datasets: YIMP and YDIP	32
2	Statistical Measures of Combined Centrality Method	42
3	Statistical Measures of Modified LBCC Method	43
4	Performance Score of All Methods on YMIPS and YDIP Datasets	43

In a protein-protein interaction network (PPIN), a node represents a protein while an edge between the two provides knowledge about their interaction. It is believed that the target protein performs almost similar functions as that of its neighbourhood. But functional assignment to the target proteins from their corresponding neighbours is a challenging task since all of the neighbourhood proteins do not hold similar importance. Hence, selection of essential proteins is a prime step before function prediction such that a higher accuracy can be achieved. This is where the concept of densely and loosely connected network slides in. Densely connected networks are those networks where interconnectivity between the proteins is maximum while the loosely connected networks are considered to be those networks where interconnectivity is minimum. Introduction of a proper model thus becomes indispensable which can detect essential proteins in the neighbourhood and hence can transmit appropriate functional groups from them to target protein. This dual concept of detection along with the prediction of the target protein functions has been followed in this current work. But before preceding into the detailed implementation of the proposed methodology, few of the relevant existing works have been discussed in the upcoming section to have a clear idea about the working procedures of neighbourhood-based prediction approaches.

Protein name is derived from a Greek word PROTOS which means “the first or the supreme”. Proteins are large biomolecules, or macromolecules, consisting of one or more long chains of amino acid residues. Proteins perform a vast array of functions within organisms, including catalysing metabolic reactions, DNA replication, responding to stimuli, providing structure to cells and organisms, and transporting molecules from one location to another. Proteins differ from one another primarily in their sequence of amino acids, which is dictated by the nucleotide sequence of their genes, and which usually results in protein folding into a specific three-dimensional structure that determines its activity.

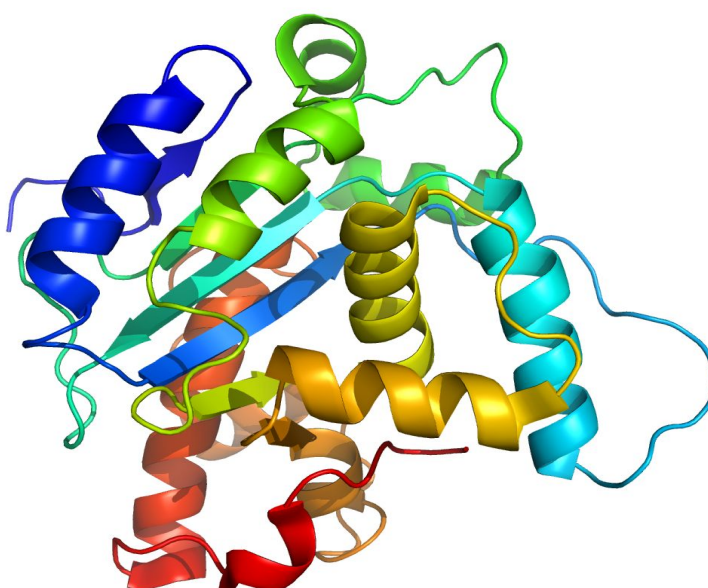


Figure 1 : Protein Structure

A linear chain of amino acid residues is called a polypeptide. A protein contains at least one long polypeptide. Short polypeptides, containing less than 20–30 residues, are rarely considered to be proteins and are commonly called peptides, or sometimes oligopeptides. The individual amino acid residues are bonded together by peptide bonds and adjacent amino acid residues. The sequence of amino acid residues in a protein is defined by the sequence of a gene, which is encoded in the genetic code. In general, the genetic code specifies 20 standard

amino acids; however, in certain organisms the genetic code can include selenocysteine and—in certain archaea—pyrrolysine. Shortly after or even during synthesis, the residues in a protein are often chemically modified by post-translational modification, which alters the physical and chemical properties, folding, stability, activity, and ultimately, the function of the proteins. Sometimes proteins have non-peptide groups attached, which can be called prosthetic groups or cofactors. Proteins can also work together to achieve a particular function, and they often associate to form stable protein complexes.

The chief characteristic of proteins that also allows their diverse set of functions is their ability to bind other molecules specifically and tightly. The region of the protein responsible for binding another molecule is known as the binding site and is often a depression or "pocket" on the molecular surface. This binding ability is mediated by the tertiary structure of the protein, which defines the binding site pocket, and by the chemical properties of the surrounding amino acids' side chains.

Different methods of protein classification have been proposed. Below some examples based on structure, composition and function in different solvents.

Classification based on STRUCTURE of protein

Based on structure, Proteins are classified into 3 groups.

Fibrous Protein

A **Fibrous protein** is a protein with an elongated shape. Fibrous proteins provide structural support for cells and tissues. There are special types of helices present in two fibrous proteins α -keratin and collagen. These proteins form long fibers that serve a structural role in the human body. Fibrous proteins are distinguished from globular proteins by their filamentous, elongated form. Also, fibrous proteins have low solubility in water compared with high solubility in water of globular proteins. Most of them play structural roles in animal cells and tissues, holding things together. Fibrous proteins have amino acid sequences that favour a particular kind of secondary structure which, in turn, confer particular mechanical properties on the proteins.

Globular Protein

Globular proteins or **sphero proteins** are spherical ("globe-like") proteins and are one of the common protein types (the others being fibrous, disordered and membrane proteins). Globular proteins are somewhat water-soluble (forming colloids in water), unlike the fibrous or membrane proteins.

Unlike fibrous proteins which only play a structural function, globular proteins can act as:

- Enzymes, by catalyzing organic reactions taking place in the organism in mild conditions and with a great specificity. Different esterases fulfill this role.
- Messengers, by transmitting messages to regulate biological processes. This function is done by hormones, i.e. insulin etc.

- Transporters of other molecules through membranes
- Stocks of amino acids.
- Regulatory roles are also performed by globular proteins rather than fibrous proteins.

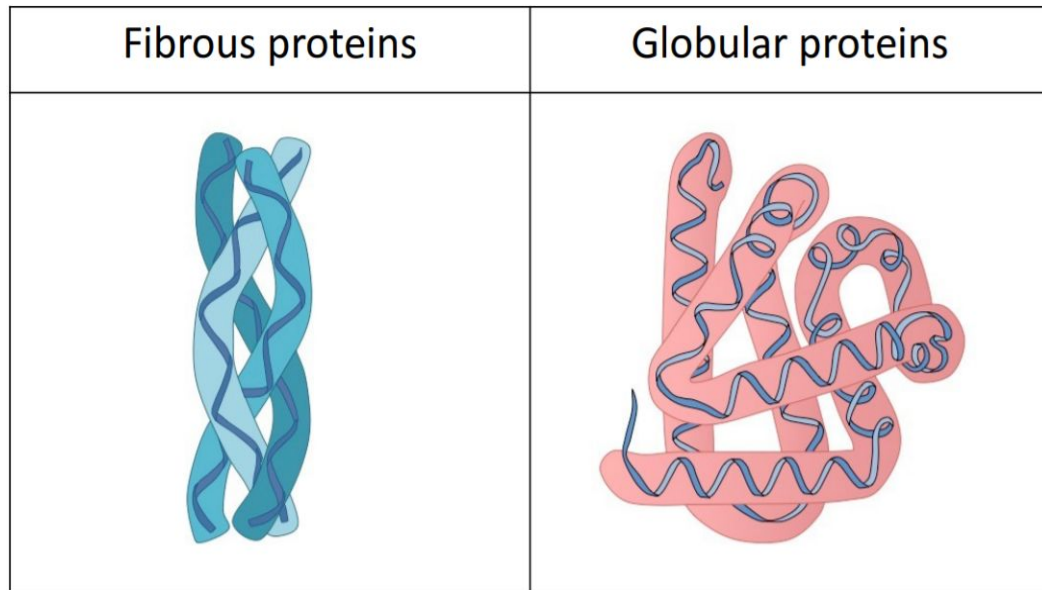


Figure 2 : Fibrous and Globular Proteins

Intermediate Proteins

Their structure is intermediate to linear and globular structures. They are soluble in water. One example of intermediate protein is : *Fibrinogen*.

Classification based on COMPOSITION of protein

Two broad categories of proteins according to its composition, they are :

Simple Proteins

Simple proteins composed of only Amino acids. They maybe fibrous or globular. They possess relatively simple structural organization. Example : *collagen, Insulin, Keratin* etc.

Conjugated Proteins

A **conjugated protein** is a protein that functions in interaction with other (non-polypeptide) chemical groups attached by covalent bonding or weak interactions. The non-amino part of a conjugated protein is usually called its prosthetic group. Most prosthetic groups are formed from vitamins. Conjugated proteins are classified on the basis of the chemical nature of their prosthetic groups. Some examples of conjugated proteins are : *lipoproteins, glycoproteins, phosphoproteins, hemoproteins, flavoproteins, metalloproteins, phytochromes, cytochromes, opsins and chromoproteins.*

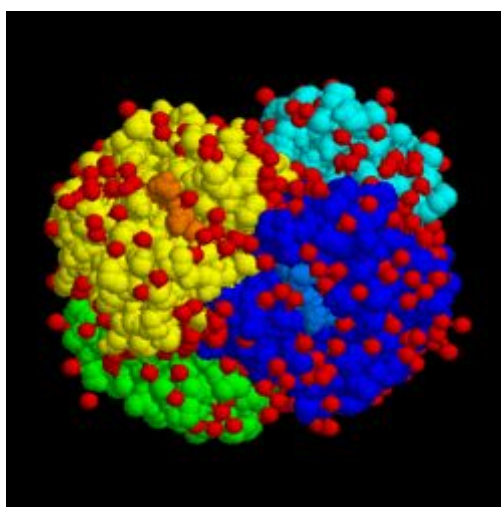


Figure 3 : Conjugated Protein (Hemoglobin)

Classification based on FUNCTION of protein

The multitude of functions that proteins perform is the consequence of both the folding of the polypeptide chain, therefore of their three-dimensional structure, and the presence of many different functional groups in the amino acid side chains, such as thiols, alcohols, thioethers, carboxamides, carboxylic acids and different basic groups.

From the functional point of view, they may be divided into several groups.

- **Enzymes (biochemical catalysts):** In living organisms, almost all reactions are catalyzed by specific proteins called enzymes. They have a high catalytic power, increasing the rate of the reaction in which they are involved at least by factor 10^6 .

Therefore, life as we know could not exist without their “facilitating action.” Almost all known enzymes, and in the human body they are thousand, are proteins (except some catalytic RNA molecules called ribozymes, that is, ribonucleic acid enzymes).

- **Transport proteins:** Many small molecules, organic and inorganic, are transported in the bloodstream and extracellular fluids, across the cell membranes, and inside the cells from one compartment to another, by specific proteins. Examples are: hemoglobin, that carries oxygen from the alveolar blood vessels to tissue capillaries; transferrin, which carries iron in the blood; membrane carriers; fatty acid binding proteins (FABP), that is, the proteins involved in the intracellular transport of fatty acids; proteins of plasma lipoproteins, macromolecular complexes of proteins and lipids responsible for the transport of triglycerides, which are otherwise insoluble in water; albumin, that carries free fatty acids, bilirubin, thyroid hormones, and certain medications such as aspirin and penicillin, in the blood.

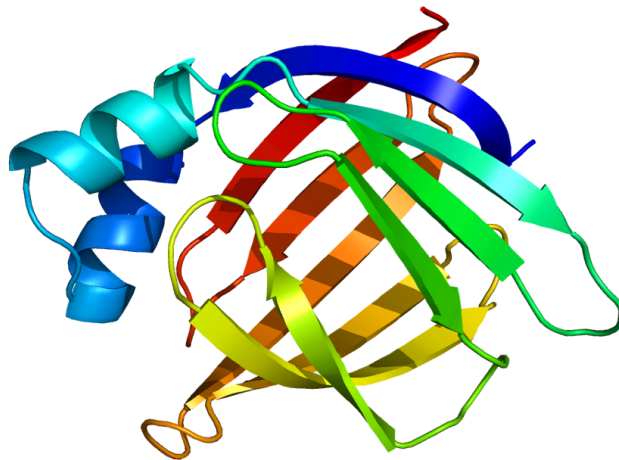


Figure 4 : FABP2

- **Storage proteins :** Examples are: ferritin, that stores iron intracellularly in a non-toxic form; milk caseins, that act as a reserve of amino acids for the milk; egg yolk phosvitin, that contains high amounts of phosphorus; prolamins and glutelins, the storage proteins of cereals.

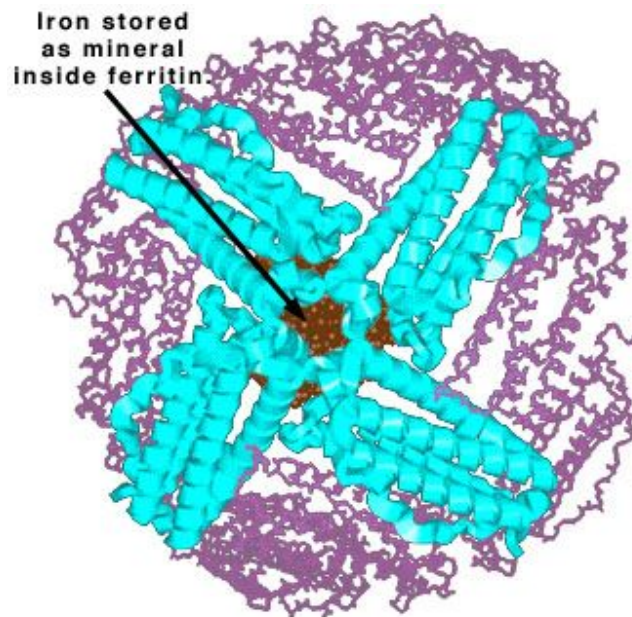


Figure 5 : Ferritin, The Iron Storage Protein

- **Contractile proteins:** They are the force generator of muscles. They can contract with the expense of energy from ATP molecules. Example : Actin, Myosin.
- **Hormones: protein hormones** are hormones whose molecules are peptides or proteins, respectively. The latter have longer amino acid chain lengths than the former. These hormones have an effect on the endocrine system of animals, including humans. Example : TSH, Insulin etc.

The knowledge of protein function plays an essential role in understanding biological cells and has a significant impact on human life in areas such as personalized medicine, better crops and improved therapeutic interventions. Due to expense an inherent difficulty of biology are improving our understanding of biological process and are regularly resulting in new features and characteristics that better describe the role of proteins. It is inevitable to neglect and overlook these anticipated features in designing more effective classification techniques. A key issue in this context, that is not being sufficiently addressed, is how to build effective classification models and approaches for protein function prediction by incorporating and taking advantage from the ever evolving biological information.

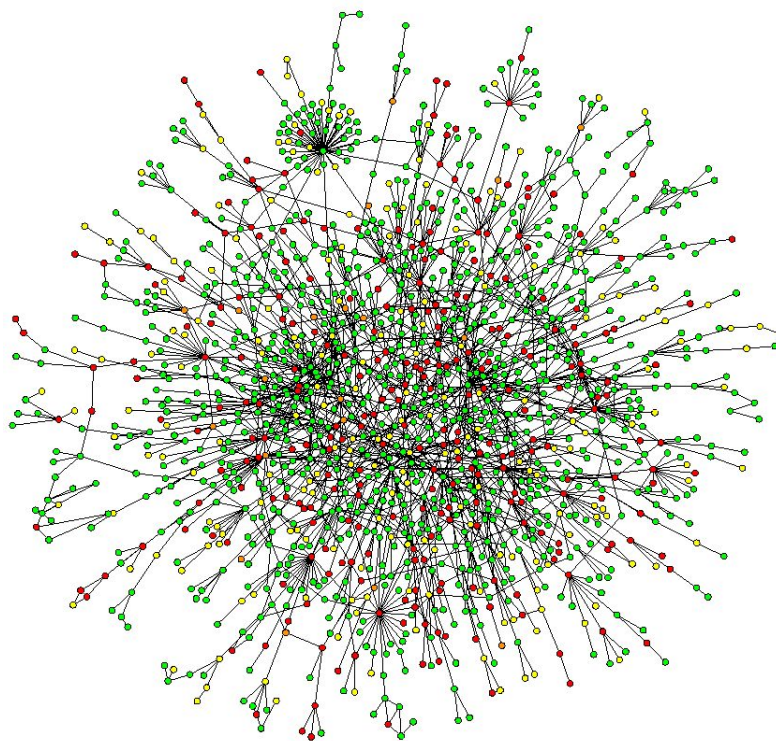


Figure 6 : A sample PPI network

An important factor that impacts the performance of function prediction models is the type of biological information used to infer functional association among proteins. Until recently, many high throughput techniques have been developed to devise mechanisms leading to precise prediction of protein functions. These techniques utilize information derived from sequence similarity, protein 3D structure, phylogenetic profiles, protein complex, PPIs, gene expression profiles.

The multitude of functions that proteins perform in the consequence of both the folding of the polypeptide chain, therefore of their three-dimensional structure, and presence of many different functional groups in the amino acid side chains, such as thiols, alcohols, thioethers, carboxamides, carboxylic acids and different basic groups.

From the functional point of view, they may be divided into several groups.

How is the elaborated three-dimensional structure of proteins attained, and how is the three-dimensional structure related to the one-dimensional amino acid sequence information? The classic work of Christian Anfinsen in the 1950s on the enzyme ribonuclease revealed the relation between the amino acid sequence of protein and its conformation. Ribonuclease is single polypeptide chain consisting of 124 amino acid residue cross-linked by four disulphide bond. Anfinsen's plan was to destroy the three-dimensional structure of the enzyme and then determine what conditions were required to restore the structure.

Agent such as urea or guanidinium chloride effectively disrupt the noncovalent bonds, although the mechanism of action of these agents is not fully understood. The disulfides bonds can be cleaved reversibly by reducing them with a reagent such as β -mercaptoethanol. In presence of a large excess of β -mercaptoethanol, a protein is produced in which the disulfides(Cystines) are fully converted into sulfhydryls (Cysteines).

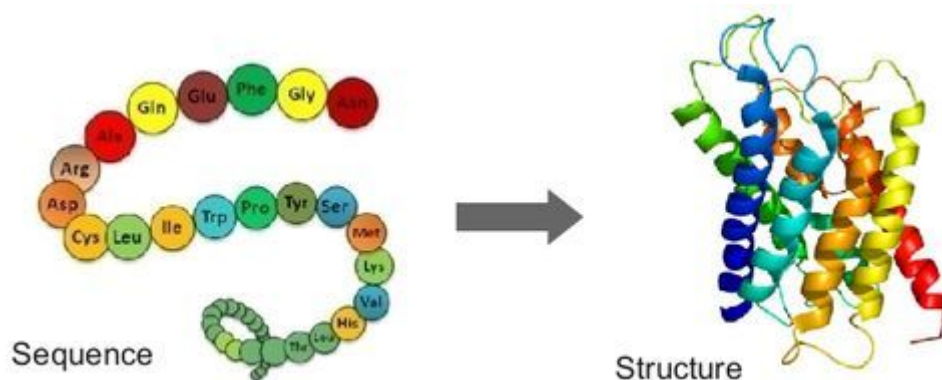


Figure 7 : Amino Acid Sequence

Most polypeptide chains devoid of cross-links assume a random-coil conformation in 8 M urea and 6 M guanidinium chloride, as evidenced by physical properties such as viscosity and optical activity. When ribonuclease was treated with β -mercaptoethanol in 8 M urea, the product was fully reduced, randomly coiled polypeptide chain devoid of enzymatic activity. In other words, ribonuclease was denatured by this treatment.

Anfinsen then made the critical observation that the denatured ribonuclease, freed of urea and β -mercaptoethanol by dialysis, slowly regained enzymatic activity. He immediately perceived the significance of this chance finding: the sulfhydryl groups of the denatured enzyme became oxidized by air, and the enzyme spontaneously refolded into a catalytically active form. Detailed studies then showed that nearly all the original enzymatic activity was regained if the sulfhydryl groups were oxidized under suitable conditions. All the measured physical and chemical properties of the refolded enzyme were virtually identical with those of the native enzymes. These experiments showed that the information needed to specify the catalytically active structure of ribonuclease is contained in its amino acid sequence. Subsequent studies have established the generality of this central principle of biochemistry: sequence specifies conformation. The dependence of conformation on sequence is especially significant because of the intimate connection between conformation and function.

The building blocks of proteins are amino acids, which are small organic molecules that consists of an alpha(central) carbon atom linked into an amino group, a carboxyl group, a hydrogen atom, and a variable component called a side chain. Within a protein, multiple amino acids are linked together by peptide bonds, thereby forming a long chain. Peptide bonds are formed by a biochemical reaction that extracts a water molecule as it joins the amino group of one amino acid to carboxyl group of a neighbouring amino acid. The linear sequence of amino acids within a protein is considered the primary structure of the protein.

As noted earlier, the different types of amino acids are distinguished based on the R group. If a R is a hydrogen atom, for instance, the amino acid is glycine. If R is a methyl group, the amino acid is alanine. If R is the sulfhydryl, the amino acid is cysteine. These are just a few examples, but apart from the R group all amino acids are otherwise the same. At one end, each amino acid has the functional group COOH, called carboxyl. At the other end, each amino acid has an NH₂ group, called amino.

A peptide bond is formed when the carboxyl carbon atom of one amino acid is joined covalently with the amino nitrogen atom of another amino_acid, expelling a molecule of water. Linking several amino_acids by their carboxyl and amino groups produces small protein, also called a polypeptide, because it contains several peptide bonds. Joining amino acids in this way produces a chain with a COOH at one end and an NH₂ at the other end, called the carboxyl and amino ends, respectively.

By Sanger's Chemist were using acidic chemicals to break the peptide bond, thus separating the individual amino acids. Additionally, they knew that a protein should have more than one polypeptide chain, connected by another by disulfide bonds attaching at areas of a chain that contained cystine. By treating a protein to destroy disulfide bridges, biochemists in early 1940s could find out the number of chains in a protein. Also, by breaking apart the peptide

bonds and running chemical tests, they could determine the identity of the amino acids of a protein and the relative amounts of each amino acid.

However, this did not tell the biochemist the sequence in which those amino acids had been linked together. What set sanger apart from his contemporaries was an insight that the relative amount of each type of amino acid and their sequence could be extremely important. It might be the basis of how each protein functioned. If so, then amino acid sequence would also be the key to how life functioned. Given the prevalence of proteins in organisms, the idea made a lot of sense, but now Sanger's task was to prove it.

CHAPTER : 1.6 PROTEIN-PROTEIN INTERACTION NETWORK

Protein–protein interactions (PPIs) are the physical contacts of high specificity established between two or more protein molecules as a result of biochemical events steered by electrostatic forces including the hydrophobic effect. Many are physical contacts with molecular associations between chains that occur in a cell or in a living organism in a specific biomolecular context.

Proteins rarely act alone as their functions tend to be regulated. Many molecular processes within a cell are carried out by molecular machines that are built from a large number of protein components organized by their PPIs. These interactions make up the so-called interactomics of the organism, while aberrant PPIs are the basis of multiple aggregation-related diseases, such as Creutzfeldt–Jakob, Alzheimer's diseases, and may lead to cancer.

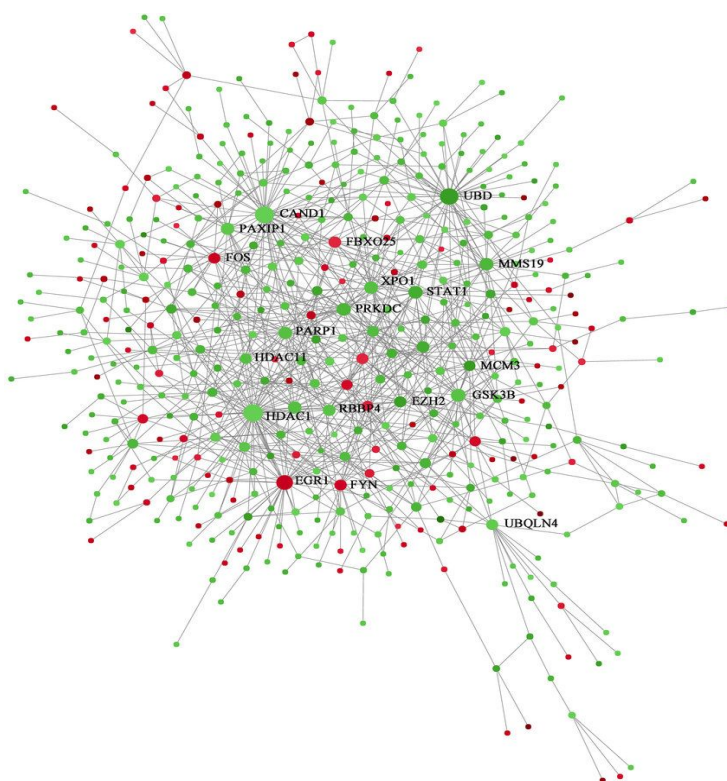


Figure 8 : PPI Network

PPIs have been studied from different perspectives: biochemistry, quantum chemistry, molecular dynamics, signal transduction, among others. All this information enables the creation of large protein interaction networks – similar to metabolic or genetic/epigenetic networks – that empower the current knowledge on biochemical cascades and molecular etiology of disease, as well as the discovery of putative protein targets of therapeutic interest.

CHAPTER : 1.7 TYPES OF PROTEIN-PROTEIN INTERACTION

To describe the types of protein–protein interactions (PPIs) it is important to consider that proteins can interact in a "transient" way (to produce some specific effect in a short time) or to interact with other proteins in a "stable" way to build multiprotein complexes that are molecular machines within the living systems. A protein complex assembly can result in the formation of homo-oligomeric or hetero-oligomeric complexes. In addition to the conventional complexes, as enzyme-inhibitor and antibody-antigen, interactions can also be established between domain-domain and domain-peptide. Another important distinction to identify protein-protein interactions is the way they have been determined, since there are techniques that measure direct physical interactions between protein pairs, named “binary” methods, while there are other techniques that measure physical interactions among groups of proteins, without pairwise determination of protein partners, named “co-complex” methods .

Homo-oligomers vs. hetero-oligomers

Homo-oligomers are macromolecular complexes constituted by only one type of protein subunit. Protein subunits assembly is guided by the establishment of non-covalent interactions in the quaternary structure of the protein. Disruption of homo-oligomers in order to return to the initial individual monomers often requires denaturation of the complex. Several enzymes, carrier proteins, scaffolding proteins, and transcriptional regulatory factors carry out their functions as homo-oligomers.

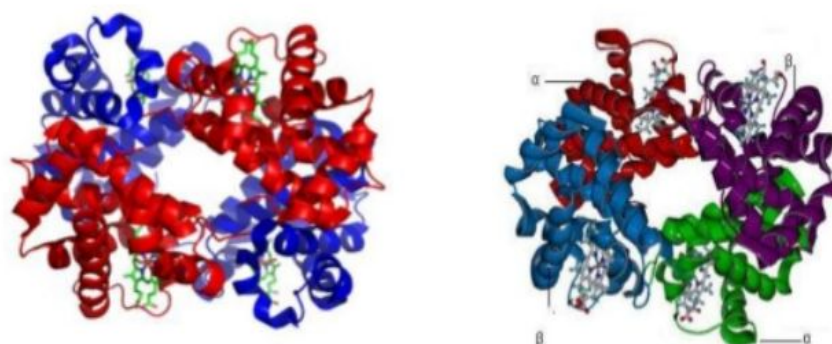


Figure 9 : Homo-oligomers vs. hetero-oligomers

Distinct protein subunits interact in hetero-oligomers, which are essential to control several cellular functions. The importance of the communication between heterologous proteins is even more evident during cell signaling events and such interactions are only possible due to structural domains within the proteins.

Stable interactions vs. transient interactions

Stable interactions involve proteins that interact for a long time, taking part of permanent complexes as subunits, in order to carry out structural or functional roles. These are usually the case of homo-oligomers (e.g. cytochrome c), and some hetero-oligomeric proteins, as the subunits of ATPase. On the other hand, a protein may interact briefly and in a reversible manner with other proteins in only certain cellular contexts – cell type, cell cycle stage, external factors, presence of other binding proteins, etc. – as it happens with most of the proteins involved in biochemical cascades. These are called transient interactions. For example, some G protein-coupled receptors only transiently bind to $G_{i/o}$ proteins when they are activated by extracellular ligands, while some G_q -coupled receptors, such as muscarinic receptor M3, pre-couple with G_q proteins prior to the receptor-ligand binding. Interactions between intrinsically disordered protein regions to globular protein domains (i.e. MoRFs) are transient interactions.

Role of water

Water molecules play a significant role in the interactions between proteins. The crystal structures of complexes, obtained at high resolution from different but homologous proteins, have shown that some interface water molecules are conserved between homologous complexes. The majority of the interface water molecules make hydrogen bonds with both partners of each complex. Some interface amino acid residues or atomic groups of one protein partner engage in both direct and water mediated interactions with the other protein partner. Doubly indirect interactions, mediated by two water molecules, are more numerous in the homologous complexes of low affinity.

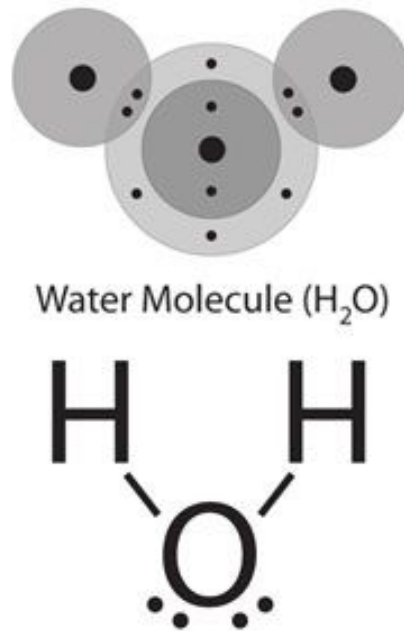


Figure 10 : Water Molecule Structure

Carefully conducted mutagenesis experiments, e.g. changing a tyrosine residue into a phenylalanine, have shown that water mediated interactions can contribute to the energy of interaction. Thus, water molecules may facilitate the interactions and cross-recognitions between proteins.

Essential proteins are indispensable to the viability or reproduction of an organism and play a decisive role in cellular life[1]. Deletion of a single essential protein is sufficient for causing lethality or infertility[2]. Compared to non-essential proteins, essential proteins are more likely to be conserved in biological evolution[3]. Essential proteins provide insights into the molecular mechanisms of an organism at the system level, with significant implications for drug design and disease study[4]. For example, in drug development, essential proteins are excellent targets for potential new drugs and vaccines to treat and prevent diseases and for improved diagnostic tools more reliably to detect infections[5]. The identification of essential proteins is necessary not only for understanding the molecular mechanisms of cellular life but also for disease diagnosis, medical treatments and drug design. Many computational methods have been proposed for discovering essential proteins, but the precision of the prediction of essential proteins remains to be improved. In this paper, we propose a two new methods, where the existing centrality measures are added with a new feature called node-weight. The first method, CCM or Combined Centrality Method, is based on six centrality measures such as closeness centrality (CC), degree centrality (DC), network centrality (NC), eigenvector centrality (EC), information centrality (IC) and the local average connectivity-based method (LAC) along with a new feature called node weight. The next method, MLM or Modified LBCC Method, which is based on the LBCC method[6] with node weight feature. In both the methods the addition of the new feature, i.e. node weight, is used to improve the prediction precision. The experimental results demonstrate that Modified LBCC outperforms traditional topological measures for predicting essential proteins, including degree centrality (DC)[7], betweenness centrality (BC)[8], subgraph centrality (SC)[9], eigenvector centrality (EC)[10], network centrality (NC)[11], local average connectivity-based method (LAC)[12] and the two recently developed methods: LIDC[13] and LBCC. Modified LBCC also shows improved precision and f-score on the YMIPS and YDIP datasets compared to the recently developed method, LBCC.

There are two types of methods for predicting essential proteins. One is experimental procedures, such as RNA interference[14], single gene knockouts[15], and conditional knockouts[16]. However, these experimental procedures require considerable time and resources, even for well-studied organisms, and they are not always practical. The other type of method is bioinformatics computational approaches that take advantage of the abundance of experimental data available for protein interaction networks, such as degree centrality (DC), betweenness centrality (BC), subgraph centrality (SC), eigenvector centrality (EC), network centrality (NC), and the local average connectivity-based method (LAC). Obviously, the latter is faster and less expensive than the former. In 2015, Luo and Qi[13] proposed a method named LIDC for discovering essential proteins based on the local interaction density and protein complexes. The experimental results obtained with the YMIPS dataset demonstrated that the performance of LIDC was superior to that of nine reference methods (i.e., DC, BC, NC, LID[13], PeC[17], CoEWC[18], WDC[19], ION[20], and UC[21]). However, methods based on bioinformatics computational approaches are sensitive to the local or global topological properties of the network, and the prediction precision for identifying essential proteins requires further improvement. In this paper, we first introduce the combination of 6 different centrality measures such as closeness centrality (CC), degree centrality (DC), network centrality (NC), eigenvector centrality (EC), information centrality (IC) and the local average connectivity-based method (LAC) and combined them to get an average centrality. Second a novel method called LBCC is proposed which is based on the works of Qin et al[6]. In both the methods we introduced a new feature, i.e. node weight, which is based on significance of a protein with its neighbouring nodes. If the node weight of a protein is big enough, then that protein has many neighbours surrounding it, which means that the protein complex is dense. This combination of features has not previously been considered for this problem. We performed a statistical analysis based on our proposed methods on different PPI (protein-protein interaction) networks of *Saccharomyces cerevisiae*, YMIPS and YDIP, which will be described in the Experimental data section. The experimental results demonstrate that our Modified LBCC Method (MLM) provides superior

prediction performance compared to centrality measures, including DC, BC, SC, EC, NC, LAC and LIDC. In particular, compared to the most recent method, LBCC, which is a more effective method for predicting essential proteins, MLM improves the prediction precision on the YMIPS and YDIP datasets. Our Combined Centrality Method (CCM) provides superior f-score performance on YMIPS dataset compared to other centrality measures mentioned above.

The observations of the advantages as well as corresponding disadvantages and limitations of the above mentioned works have revealed the fact that there is a scope of improvement in certain fields, some of which are explored while the others are still unexplored. This actually motivates us to analyse this field of study and lift protein function prediction step forward. The proposed work can be disintegrated in two parts: In the part, filtering of the original PPIN of *Saccharomyces cerevisiae* has been performed to identify the essential proteins in the network based on combined centrality score after which three thresholds: High, Medium and Low is estimated based on node weight of each protein by the application of k -sigma[22] finally the filtered proteins are considered as target proteins. In the second part,, LBCC method is used along with the same node weight feature and the three thresholds which resulted in generation of target proteins. The proposed methodology attempts in overcoming some of the shortcomings, to give a more accurate result. It can be divided into two phases. The first phase identifies hub proteins on basis of node weight value and 6 combined centrality score. The second phase identifies the hub proteins on the basis of node weight value and LBCC score. Finally the target proteins in both phases are ranked in a descending order. The top 20% of these ranked proteins has been considered as essential protein by this work. In both phases, the performance score is generated separately and then compared with previous methods.

Protein interaction network: Protein-protein interactions occur when two or more proteins bind together, often to carry out their biological function. Many of the most important molecular processes in the cell such as DNA replication are carried out by large molecular machines that are built from a large number of protein components organized by their protein-protein interactions. These protein interactions form a network like structure which is known as *Protein interaction network*. Here protein interaction network is represented as a graph G_p which consist of a set of vertex (nodes) V connected by edges (links) E . Thus $G_p = (V, E)$. Here each protein is represented as a node and their interconnections are represented by edges.

Subgraph: A graph G'_p is a *subgraph* of a graph G_p if the vertex set of G'_p is a subset of the vertex set of G_p and if the edge set of G'_p is a subset of the edge set of G_p . That is, if $G'_p = (V', E')$ and $G_p = (V, E)$, then G'_p is called as sub graph of G_p if $V' \subseteq V$ and $E' \subseteq E$. G'_p may be defined as a set of $\{K \cup U\}$ where K represents the set of un-annotated proteins while U represents the set of annotated protein.

Level-1 neighbors: In G'_p , the directly connected neighbors of a particular vertex are called *level-1 neighbors*.

Level-2 neighbors: In G'_p , *level-2 neighbors* are those who are directly connected neighbors of level-1 neighbors of that particular vertex.

Node Weight:**Notation**

A PPI network can be modeled as an undirected simple graph $G=(V, E)$, in which V represents the set of nodes (proteins) and E represents the set of edges (protein interactions) in the network. Here, self-interactions (loops) and multiple edges between the same pair of nodes are not considered. Before detail description of our algorithm, some terminologies used in the following algorithm section are presented as follows.

Definition 1

The neighbourhood graph[23] of $v \in V$ consists of v , all its neighbours and the edges among them. It is defined as $G_v=(V',E')$, in which

$$V' = \{v\} \cup \{u \mid u \in V, (u, v) \in E\}, \text{ and}$$

$$E' = \{(u_i, u_j) \mid (u_i, u_j) \in E, u_i, u_j \in V'\}.$$

Definition 2

In G_v , there are some nodes with degree 1 that only have connections with v and the connections among these nodes are often false positive according to topological reliability measures as described in[24], [25],[26]. So all nodes with degree 1 and corresponding edges are removed from G_v . The remaining subgraph of G_v is marked as G_v' . In the algorithm, the node weight w_v of node $v \in V$ in PPI networks is the average degree of all nodes in G_v' . It is represented by Equation (1).[27]

$$w_v = \frac{\sum_{u \in V''} \deg(u)}{|V''|} \quad (1)$$

where, V'' is the set of nodes in G_v' . $|V''|$ is the number of nodes in G_v' . And $\deg(u)$ is the degree of a node $u \in V''$ in G_v' . In our algorithm, the weight w_v of a node $v \in V$ is used in the step of seed chosen. If w_v is big enough, v has many neighbours in G_v' and G_v' is a densely connected region.

Threshold:

Three thresholds (high, medium and low) are set for each of node and edge weight using equation 2.[28]

$$Th_k = \alpha + k.\sigma.(1 - \frac{1}{1+\sigma^2}) \quad \dots(2)$$

where for node weight/edge weight, $k \in \{1,2,3\}$ denotes three different thresholds i.e. low, medium and high respectively. α is the mean of node weight/edge weight values of all proteins. σ is the standard deviation of node weight/edge weight values of all proteins. Proteins and edges having value less value than these node and edge weight thresholds get discarded and are considered as non-essential proteins and unreliable edges in the network respectively.

The PPI network of *Saccharomyces cerevisiae* is selected as the base for essential protein prediction from [6]. Two datasets YMIP and YDIP are considered as the prime focus of the research. This project is divided into two parts: first, a combined centrality approach is formulated. The centralities are combined to form an average centrality based on which node weight feature is applied. Second, LBCC method is considered for application of node weight feature.

Combined centrality method

In this method, 6 centrality measures are selected such as closeness centrality (CC), degree centrality (DC), network centrality (NC), eigenvector centrality (EC), information centrality (IC) and the local average connectivity-based method (LAC). Then, all the centralities are combined into an average value for a particular protein node. Finally, essential proteins are selected using node weight and threshold functionality.

1.1 Average calculation of Combined Centrality.

Algorithm: Average_Centrality

(To generate an average centrality value out of 6 different centralities for each protein)

Input: DataSet containing proteins and its interactions

Output: calculation of average centrality of each protein

Begin:

For each protein node i
 Calculate its centralities such as closeness centrality, degree centrality
 eigenvector centrality, information centrality, network centrality
 local average connectivity-based method using[ref].
End for
Average = 0
For each i
 Average = sum of all the centralities / 6
 Write the average value of each node
End for

End

Modified LBCC method

The LBCC score of proteins are generated using codes provided in [6] . Similar to the previous method, Essential proteins are selected using node weight and threshold.

Node weight calculation

At first a level-0 node is considered and its respective level-1 and level-2 proteins are found out. For eg, In YMIPS, protein Q0275 is considered as the Level-0 protein. The Level-1 proteins are YER154W and YGL187C. The level-2 proteins are YNR003C YOR121C YGL213C YML042W. Then, node weight is calculated as per equation (1). The denominator of the node weight i.e $|V''|$ is calculated as the total of level-1 proteins and level-2 proteins of that particular level-0 protein and the numerator i.e. $\sum_{u \in V''} deg(u)$ is calculated as the sum of the degrees of level-1 proteins or the total number of level-2 proteins of that level-0 protein. Likewise, node weight of each and every protein given in the dataset YMIPS and YDIP is calculated.

1.2 Calculation of Node weight

Algorithm: Node_Weight_Calculator

(to calculate node weight of each protein present in the PPIN)

Input: Dataset containing proteins and its interactions

Output: Node weight of each protein

Begin:

For each protein i

 unique_list = 0 //empty list

 Insert all unique proteins in unique_list

 remove duplicates

 Write unique list

End for

For each protein i in unique_list

 For each protein_pair a,b in dataset

 if protein i == protein a

 write protein a, b

 End if

 End for

End for

to calculate level 1 and level 2 nodes

for each protein i

 level_1 = 0 // empty list

 level_2 = 0 //empty list

 total = 0 //empty list

 Find level_1 protein and its corresponding level_2 proteins

 for each protein of level_0

 total = level_1 + level_2

 Write total

```

        End for
    End for
    For each protein i
        if level_2(i) != 0 then
            node_weight = level_2(i)/total(i)
            Write node_weight
        Else
            Node_weight is NULL
        End if
    End for
End

```

After the calculation of node weight, the mean and standard deviation of node weight of all the proteins present in the dataset is calculated. This mean and standard deviation value is used in the threshold formula as shown in equation (2). The threshold has been set with the implementation of k-sigma where k=1,2,3 represents low, medium, high threshold respectively. The three threshold values for YMIPS dataset are 0.62321(low), 0.69295(medium) and 0.7627(high). For YDIP dataset, the threshold values are 0.90259(low), 0.9564(medium), 1.01022(high). Based on these threshold the proteins having lower node weight value than the threshold value have been discarded.

1.3 Threshold calculation and generation of target protein set

Algorithm: Essential_Protein_Finder

(to find out target proteins based on the threshold value and selecting top 20% among the target proteins as essential proteins)

Input: protein list with node_weight

Output: essential and non essential protein

Begin:

```
calc_p = 0
calculate the threshold value using k-sigma (k=1,2,3) on node_weight
for each protein i
    if node_weight of i threshold
        calc_p = calc_p union i
        Write calc_p
    end if
end for
Sort calc_p in decreasing order
Essen_p = 0      //empty list of essential protein
Non-essen_p = 0  //empty list of non essential protein
For each protein i in calc_p
    If i is present in top 20% then
        essen_P = essen_p union i
    Else
        Non-essen_p = non_essen_p union i
    End if
End for
```

End

Thus, all the non-essential proteins in the PPIN have been eliminated thereby reducing the chance of increasing false positives in the protein complex. For comparison we have considered high threshold value as a benchmark and selected top 20% of the proteins as essential protein.

CHAPTER : 3.5

DATASET USED

To evaluate the performance of the Combined Centrality method and Modified LBCC method, *Saccharomyces cerevisiae* is used as the experimental material because relatively reliable and complete PPI data are available for this organism. The PPI network data are from the MIPS database (Mammalian Protein-Protein Interaction Database) [29], and the DIP database[30]. The first dataset, a MIPS dataset, was marked YMIPS; and the second dataset, a DIP dataset, was marked YDIP. These datasets are acquired from [6].YMIPS included 4546 proteins and 12319 interactions, and its average degree was approximately 5.42. YDIP included 5093 proteins and 24743 interactions, and its average degree was approximately 9.72.

Dataset	Proteins	Interactions	Average degree	Essential proteins
YMIPS	4546	12319	5.42	1016
YDIP	5093	24743	9.72	1167

Table 1 : Information on the two PPI datasets: YIMPS and YDIP

CYTOSCAPE:

Cytoscape is an open source software platform for visualizing molecular interaction networks and biological pathways and integrating these networks with annotations, gene expression profiles and other state data. Although Cytoscape was originally designed for biological research, now it is a general platform for complex network analysis and visualization. Cytoscape core distribution provides a basic set of features for data integration, analysis, and visualization. While Cytoscape is most commonly used for biological research applications, it is agnostic in terms of usage. Cytoscape can be used to visualize and analyze network graphs of any kind involving nodes and edges (e.g., social networks). A key aspect of the software architecture of Cytoscape is the use of plugins for specialized features. Plugins are developed by core developers and the greater user community.

Cytoscape 3 is the mainstream version of Cytoscape with modular architecture. It is designed for long-term maintainability and it replaced 2.x series. It is a Java desktop application designed for large-scale network analysis and visualization. Cytoscape 3 supports a lot of standard network and annotation file formats including: SIF (Simple Interaction Format), GML, XGMML, BioPAX, PSI-MI, GraphML, KGML (KEGG XML), SBML, OBO, and Gene Association. Delimited text files and MS Excel™ Workbook are also supported and one can import data files, such as expression profiles or GO annotations, generated by other applications or spreadsheet programs. Using this feature, one can load and save arbitrary attributes on nodes, edges, and networks. For example, input a set of custom annotation terms for proteins, create a set of confidence values for protein-protein interactions.



Figure 11 : Cytoscape 3 starting activity

Graph Layout and Architecture:

The central organizing metaphor of Cytoscape is a network graph, with molecular species represented as nodes and intermolecular interactions represented as links, that is, edges, between nodes. Cytoscape's Core software component provides basic functionality for integrating arbitrary data on the graph, a visual representation of the graph and integrated data, selection and filtering tools, and an interface to external methods implemented. One of the most fundamental tools for interpreting molecular interaction data is visualization of nodes and edges as a two-dimensional network. Cytoscape supports a variety of automated network layout algorithms, including spring-embedded layout, hierarchical layout, and circular layout. Among these, the *spring embedder* is the most widely used method for arranging general two-dimensional graphs. It models a mechanical system in which edges of the graph correspond to springs, creating an attractive force between nodes that are far apart, and a repulsive force between nodes that are close together.

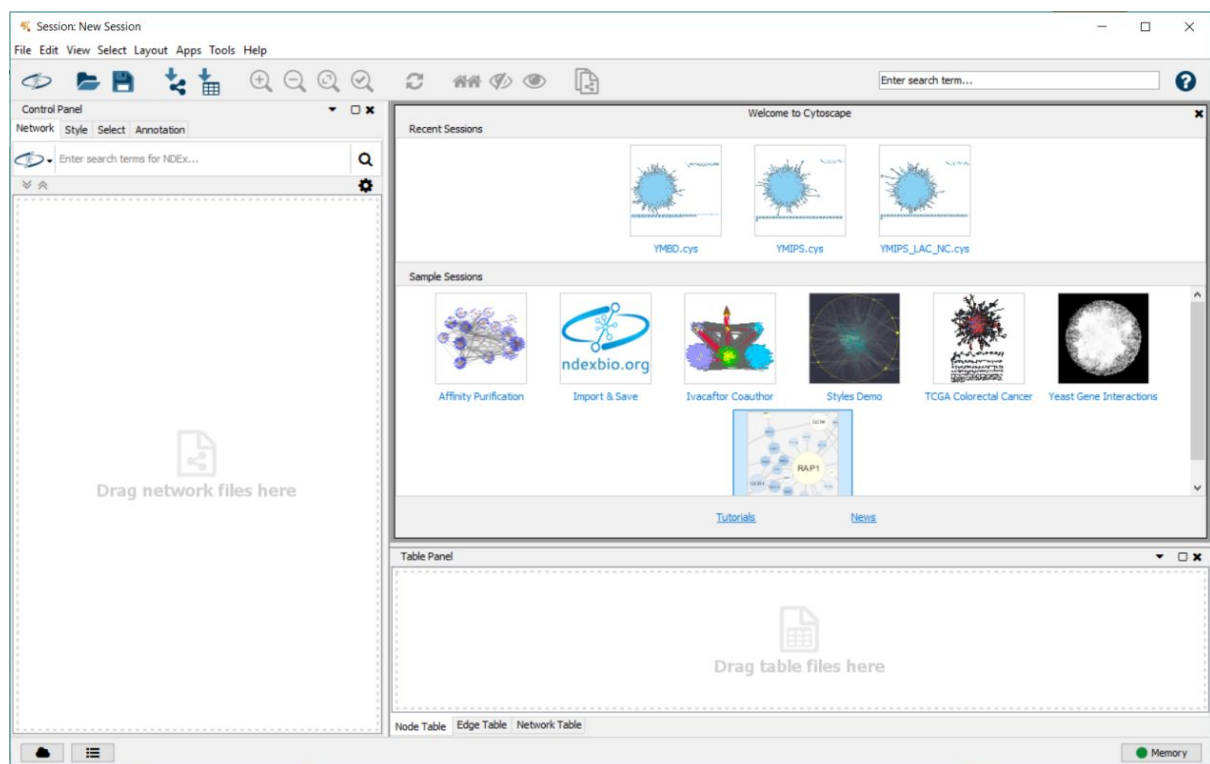


Figure 12 : Tour of Cytoscape core functionality

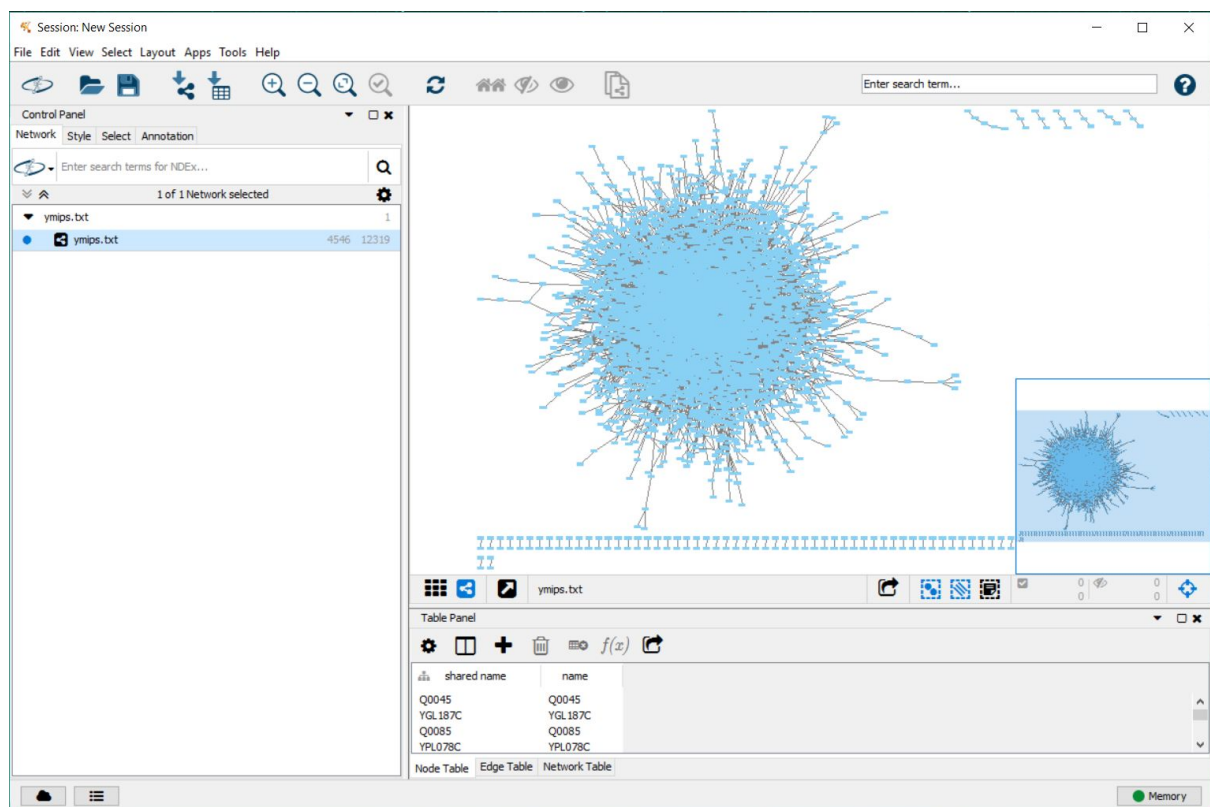


Figure 13 : YMIPS data set

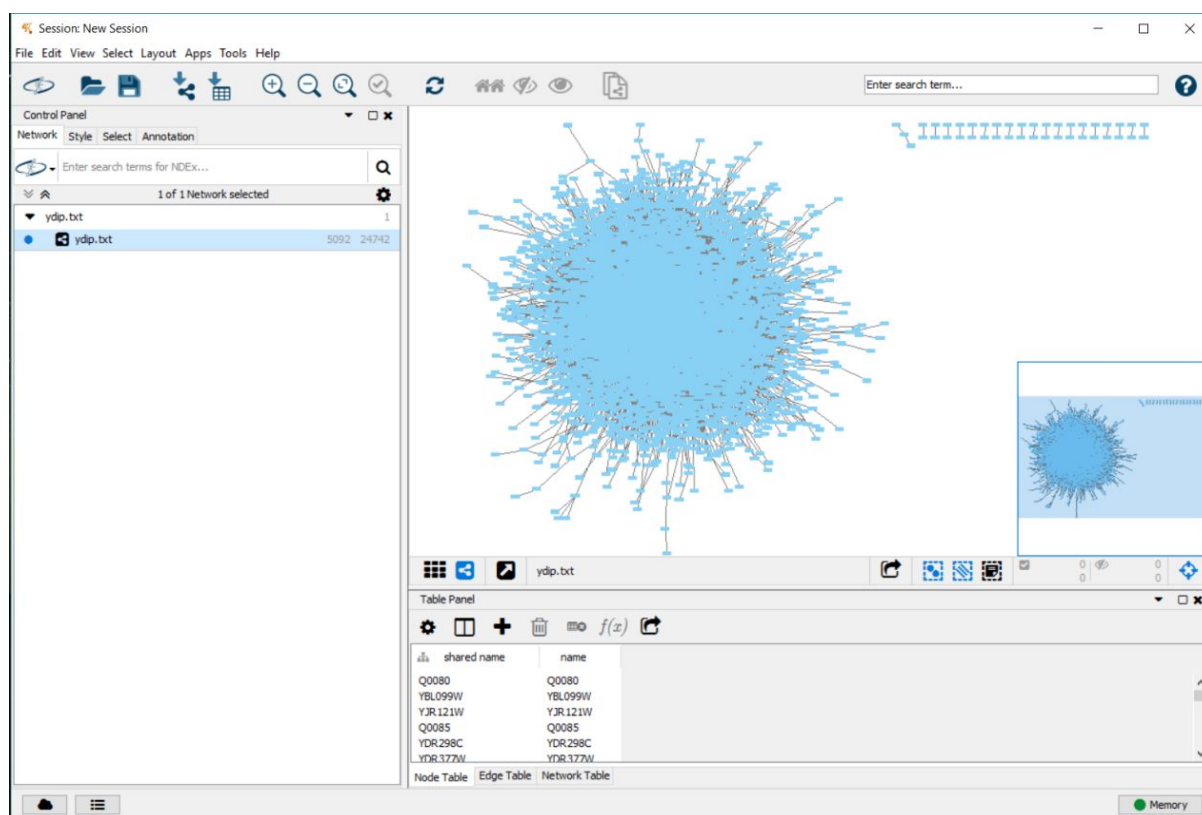


Figure 14 : YDIP data set

Data Integration:

Data are integrated with the graph model using Attributes. These are (name, value) pairs that map node or edge names to specific data values. Attribute values may assume any type (e.g., text strings, discrete or continuous numbers, URLs, or lists) and are either loaded from a data repository or generated dynamically within a session. Graphical browsers allow the user to examine all attributes on the currently selected nodes and edges. Here, an open source plugin called “cytoNCA” is used to determine the Closeness Centrality(CC), Degree Centrality(DC), Eigenvector centrality(EC), Local Average Connectivity based method (LAC), Network Centrality (NC) and Information Centrality(IC).

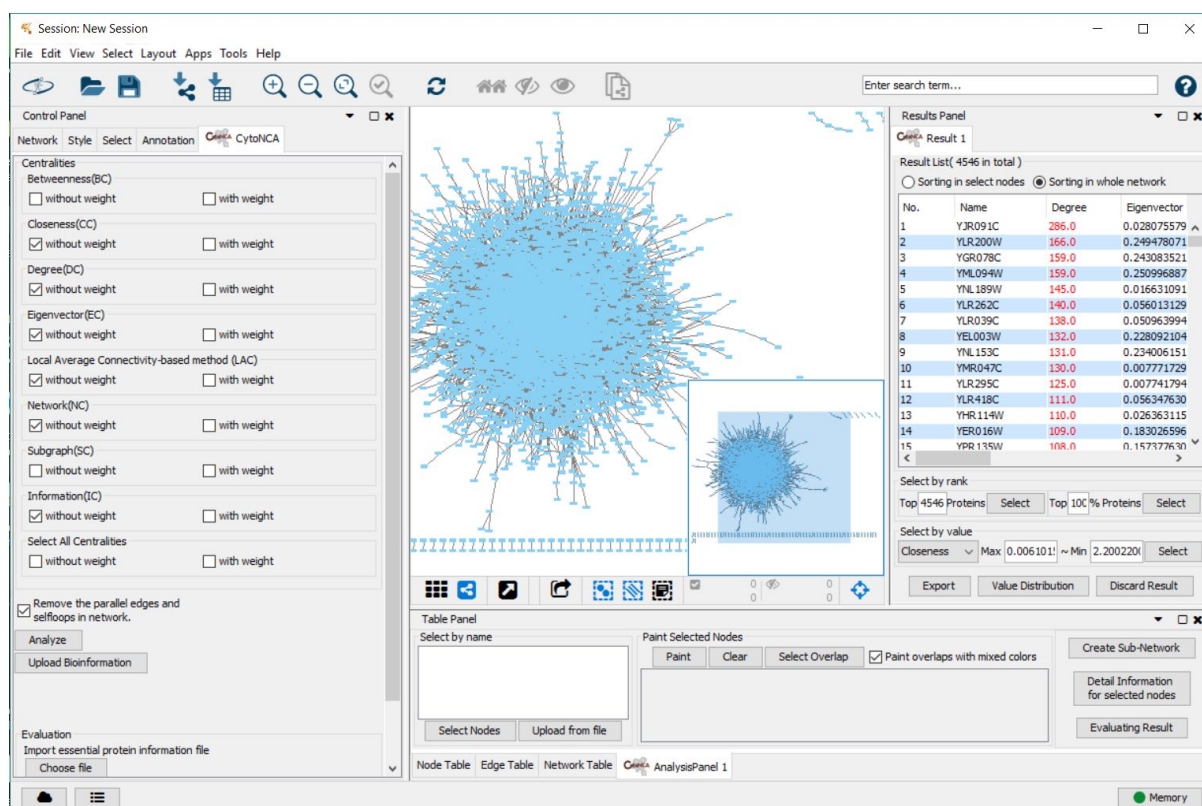


Figure 15 : Calculating and exporting the CC,DC,EC,LAC,NC,IC values of YMIPS

SUBLIME TEXT:

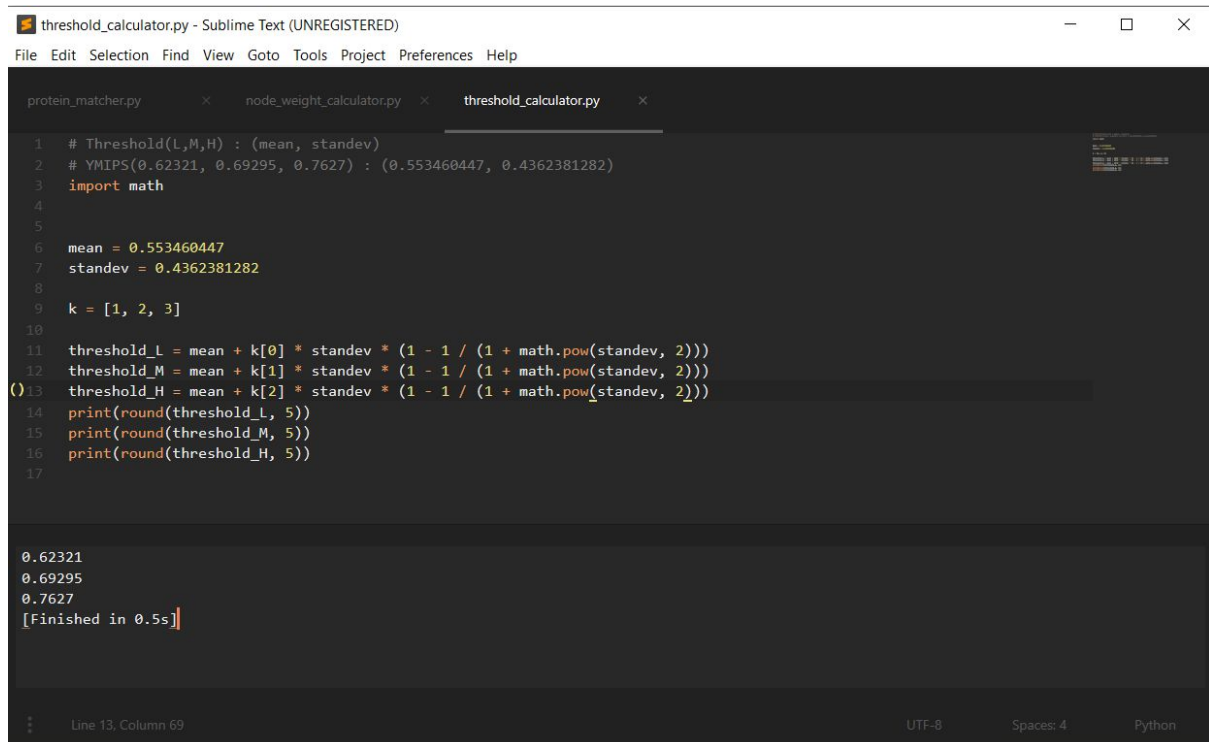
Sublime Text is a proprietary cross-platform source code editor with a Python application programming interface (API). It natively supports many programming languages and markup languages, and functions can be added by users with plugins, typically community-built and maintained under free-software licenses.

Sublime Text Version 3:

Version 3 of Sublime Text entered beta on 29 January 2013. At first available only for registered users who had purchased Sublime Text 2, on 28 June 2013 it became available to the general public. However, the latest development builds still required a registration code. Sublime Text 3 was officially released on 13 September 2017.

Two of the main features that Sublime Text 3 adds include *symbol indexing and pane management*. Symbol Indexing allows Sublime Text to scan files and build an index to

facilitate the features *Goto Definition* and *Goto Symbol in Project*. Pane Management allows users to move between panes via hotkeys.



```
threshold_calculator.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help

protein_matcher.py × node_weight_calculator.py × threshold_calculator.py ×

1 # Threshold(L,M,H) : (mean, standev)
2 # YMIPS(0.62321, 0.69295, 0.7627) : (0.553460447, 0.4362381282)
3 import math
4
5
6 mean = 0.553460447
7 standev = 0.4362381282
8
9 k = [1, 2, 3]
10
11 threshold_L = mean + k[0] * standev * (1 - 1 / (1 + math.pow(standev, 2)))
12 threshold_M = mean + k[1] * standev * (1 - 1 / (1 + math.pow(standev, 2)))
13 threshold_H = mean + k[2] * standev * (1 - 1 / (1 + math.pow(standev, 2)))
14 print(round(threshold_L, 5))
15 print(round(threshold_M, 5))
16 print(round(threshold_H, 5))
17

0.62321
0.69295
0.7627
[Finished in 0.5s]

Line 13, Column 69 UTF-8 Spaces: 4 Python
```

Figure 16 : Full view of Sublime Text IDE

NOTEPAD:

Notepad++ is a text editor and source code editor for use with Microsoft Windows. It supports tabbed editing, which allows working with multiple open files in a single window. The project's name comes from the C increment operator.

Notepad++ is distributed as free software. At first the project was hosted on SourceForge.net, from where it has been downloaded over 28 million times, and twice won the SourceForge Community Choice Award for Best Developer Tool.

Since the datasets were too large, notepad++ text editor is used as it comes with a powerful editing interactive user interface and other useful components.

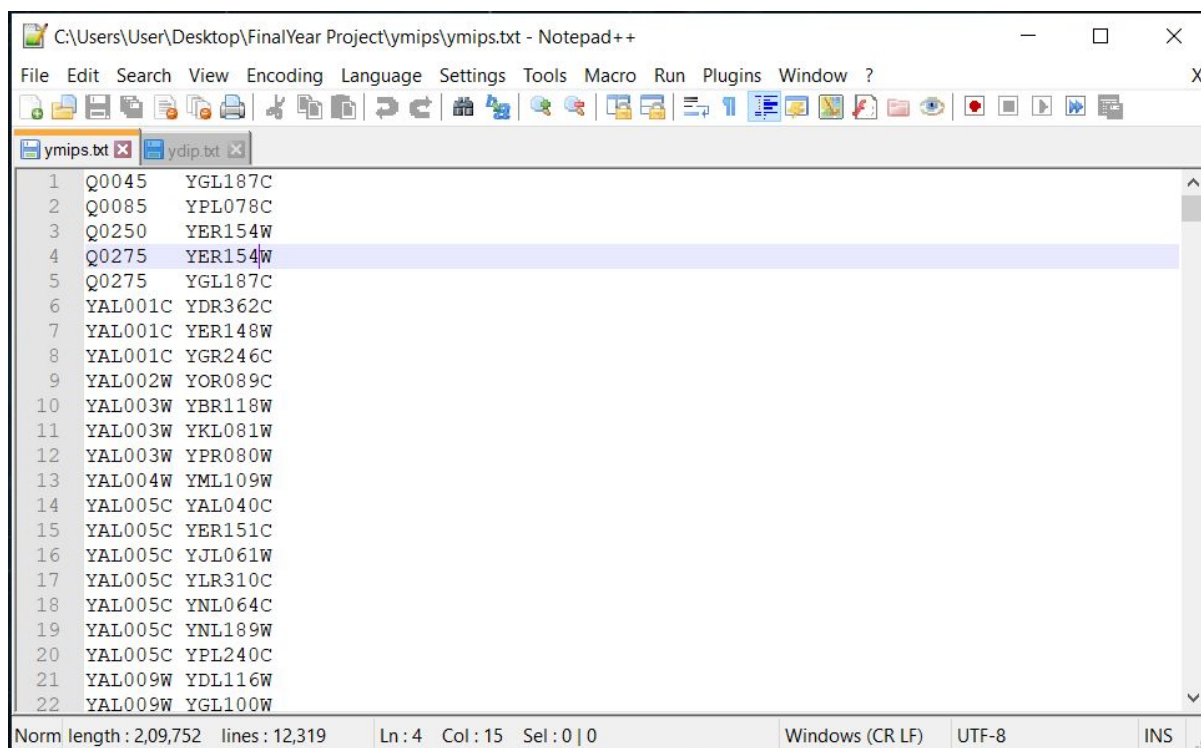


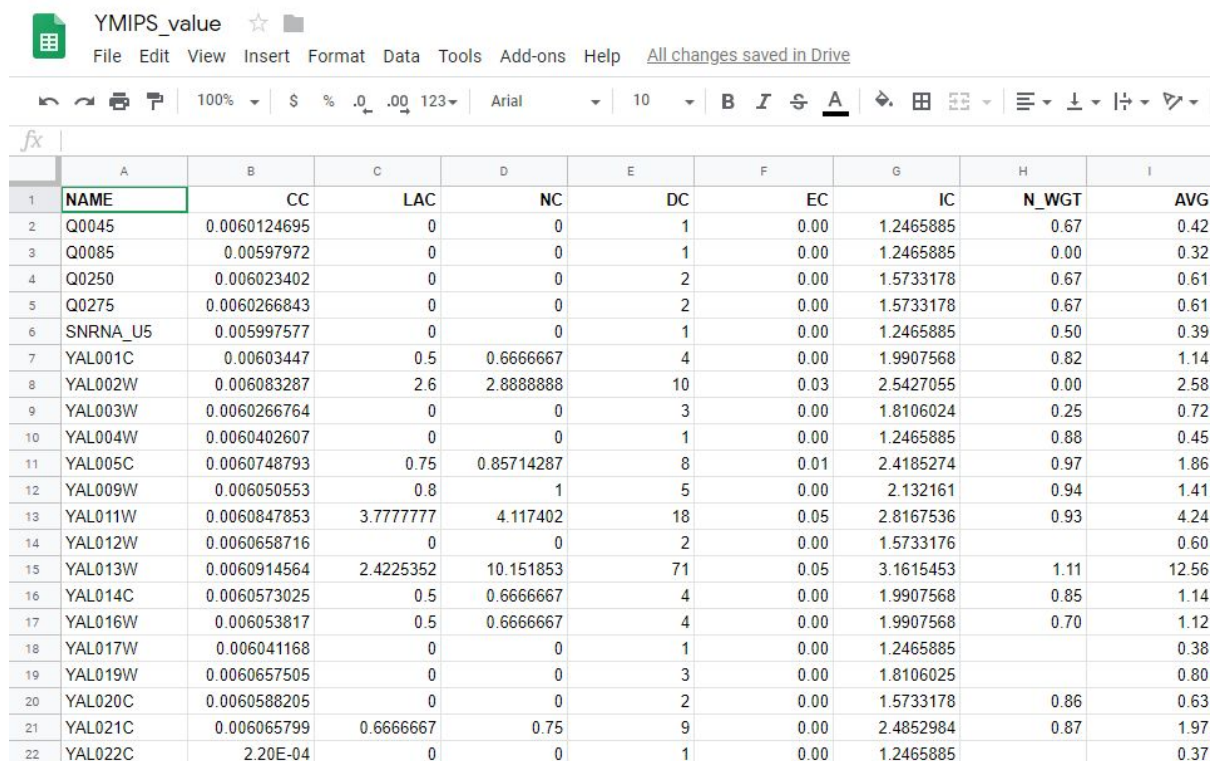
Figure 17 : Windowed view of Notepad++

GOOGLE SHEETS:

Google Sheets is a spreadsheet program included as part of a free, web-based software office suite offered by Google within its Google Drive service. The service also includes Google Docs and Google Slides , a word processor and presentation program respectively. Google Sheets is available as a web application, mobile app for Android, iOS, Windows, BlackBerry, and as a desktop application on Google's Chrome OS. The app is compatible with Microsoft

Excel file formats. The app allows users to create and edit files online while collaborating with other users in real-time. Edits are tracked by user with a revision history presenting changes. An editor's position is highlighted with an editor-specific color and cursor and a permissions system regulates what users can do. Updates have introduced features using machine learning, including "Explore", offering answers based on natural language questions in a spreadsheet.

Google Sheets is very robust, user friendly and is helpful when performing statistical analysis on a very big chunk of data. With the help of Google Sheets, data interpretation, graph construction and calculation of various equations including mean and standard deviation has been fast and hassle-free.



YMIPS_value ☆

File Edit View Insert Format Data Tools Add-ons Help All changes saved in Drive

100% \$ % .0 .00 123 Arial 10 B I S A

	A	B	C	D	E	F	G	H	I
1	NAME	CC	LAC	NC	DC	EC	IC	N_WGT	AVG
2	Q0045	0.0060124695	0	0	1	0.00	1.2465885	0.67	0.42
3	Q0085	0.00597972	0	0	1	0.00	1.2465885	0.00	0.32
4	Q0250	0.006023402	0	0	2	0.00	1.5733178	0.67	0.61
5	Q0275	0.0060266843	0	0	2	0.00	1.5733178	0.67	0.61
6	SNRNA_U5	0.005997577	0	0	1	0.00	1.2465885	0.50	0.39
7	YAL001C	0.00603447	0.5	0.6666667	4	0.00	1.9907568	0.82	1.14
8	YAL002W	0.006083287	2.6	2.8888888	10	0.03	2.5427055	0.00	2.58
9	YAL003W	0.0060266764	0	0	3	0.00	1.8106024	0.25	0.72
10	YAL004W	0.0060402607	0	0	1	0.00	1.2465885	0.88	0.45
11	YAL005C	0.0060748793	0.75	0.85714287	8	0.01	2.4185274	0.97	1.86
12	YAL009W	0.006050553	0.8	1	5	0.00	2.132161	0.94	1.41
13	YAL011W	0.0060847853	3.7777777	4.117402	18	0.05	2.8167536	0.93	4.24
14	YAL012W	0.0060658716	0	0	2	0.00	1.5733176		0.60
15	YAL013W	0.0060914564	2.4225352	10.151853	71	0.05	3.1615453	1.11	12.56
16	YAL014C	0.0060573025	0.5	0.6666667	4	0.00	1.9907568	0.85	1.14
17	YAL016W	0.006053817	0.5	0.6666667	4	0.00	1.9907568	0.70	1.12
18	YAL017W	0.006041168	0	0	1	0.00	1.2465885		0.38
19	YAL019W	0.0060657505	0	0	3	0.00	1.8106025		0.80
20	YAL020C	0.0060588205	0	0	2	0.00	1.5733178	0.86	0.63
21	YAL021C	0.006065799	0.6666667	0.75	9	0.00	2.4852984	0.87	1.97
22	YAL022C	2.20E-04	0	0	1	0.00	1.2465885		0.37

Figure 18 : Windowed view of Google Sheets

Evaluation Methods

In general, several statistical measures, such as sensitivity (SN), specificity (SP), positive predictive value (PPV), negative predictive value (NPV), F-measure (F), and accuracy (ACC), are used to determine how effectively the essential proteins are identified by different methods including proposed CCM and MLM methods. First, we provide four statistical terms:

- *True positives*(TP). The essential proteins that are correctly selected as essential.
- *False positives*(FP). The nonessential proteins that are incorrectly selected as essential.
- *True negatives*(TN). The nonessential proteins that are correctly selected as nonessential.
- *False negatives*(FN). The essential proteins that are incorrectly selected as nonessential.

Next, we provide the definitions of six statistical measures:

1. *Sensitivity*. Sensitivity is the ratio of the proteins that are correctly selected as essential to the total number of essential proteins,

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

2. *Specificity*. Specificity is the ratio of the nonessential proteins that are correctly selected as nonessential to the total number of nonessential proteins,

$$SP = \frac{TN}{TN+FP}$$

3. *Positive predictive value*. Positive predictive value refers to the ratio of the proteins that are correctly selected as essential,

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

4. *Negative predictive value*. Negative predictive value refers to the ratio of the proteins that are correctly selected as nonessential,

$$NPV = \frac{TN}{TN+FN}$$

5. *F-measure*. F-measure refers to the harmonic mean of SN and PPV,

$$SN = \frac{2*SN*PPV}{SN+PPV}$$

6. *Accuracy*. Accuracy refers to the ratio of the proteins that are correctly selected as essential and nonessential in all the results,

$$ACC = \frac{TP+TN}{P+N}$$

in which P represents the number of essential proteins and N represents the number of nonessential proteins.

Statistical Analysis

The statistical measures of both CCM and MLM has been shown below in table 2 and 3:

DATASET	THRESHOLD	SN	SP	PPV	NPV	F-MEASURE	ACC
YMIPS	LOW	0.205	0.790	0.290	0.703	0.240	0.589
	MEDIUM	0.195	0.787	0.277	0.700	0.229	0.586
	HIGH	0.187	0.785	0.265	0.699	0.587	0.263
YDIP	LOW	0.327	0.839	0.451	0.755	0.379	0.654
	MEDIUM	0.281	0.826	0.430	0.711	0.340	0.619
	HIGH	0.219	0.811	0.414	0.631	0.286	0.573

Table 2 : Statistical Measures of Combined Centrality Method

DATASET	THRESHOLD	SN	SP	PPV	NPV	F-MEASURE	ACC
YMIPS	LOW	0.409	0.877	0.582	0.780	0.480	0.706
	MEDIUM	0.401	0.874	0.572	0.777	0.471	0.704
	HIGH	0.411	0.876	0.579	0.782	0.481	0.712
YDIP	LOW	0.417	0.873	0.571	0.788	0.482	0.702
	MEDIUM	0.410	0.883	0.621	0.762	0.494	0.694
	HIGH	0.383	0.907	0.716	0.707	0.499	0.691

Table 3 : Statistical Measures of Modified LBCC Method

The performance score for function prediction of three different sets of target proteins has been evaluated in terms of prediction (P), recall (R) and f-score (F) as has been shown below:

$$P = \frac{TP}{TP+FP} \quad (1)$$

$$R = \frac{TP}{TP+FN} \quad (2)$$

$$F = \frac{2*P*R}{P+R} \quad (3)$$

Here TP represents True Positives. FP represents False Positives and FN represents False Negatives. Precision, recall and f-score obtained by the proposed work for two different sets of target proteins are shown below with other existing methods in Table 4 :

DATASET	METHODS	PRECISION	RECALL	F-SCORE
YMIPS	DC	0.282	0.252	0.266

	LAC	0.300	0.269	0.284
	SC	0.155	0.139	0.146
	EC	0.155	0.139	0.146
	BC	0.278	0.249	0.263
	NC	0.315	0.281	0.297
	LIDC	0.473	0.423	0.447
	LBCC	0.481	0.430	0.454
	CCM	0.265	0.187	0.587
	MLM	0.579	0.411	0.481
YDIP	DC	0.406	0.354	0.378
	LAC	0.465	0.405	0.433
	SC	0.370	0.323	0.345
	EC	0.370	0.323	0.345
	BC	0.354	0.308	0.330
	NC	0.456	0.398	0.425
	LIDC	0.511	0.446	0.476
	LBCC	0.512	0.446	0.477
	CCM	0.414	0.219	0.286
	MLM	0.716	0.383	0.499

Table 4 : Performance Score of All Methods on YMIPS and YDIP Datasets

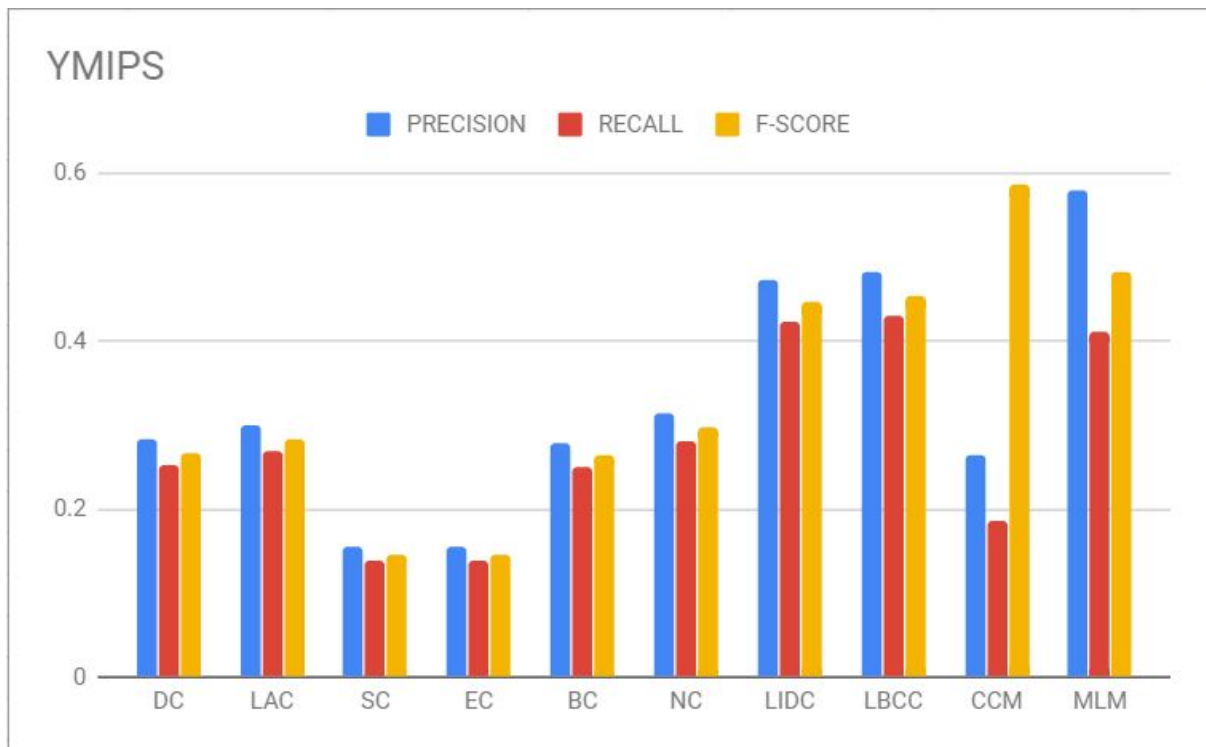


Figure 19 : Precision Recall F-score values of different methods on YMIPS dataset

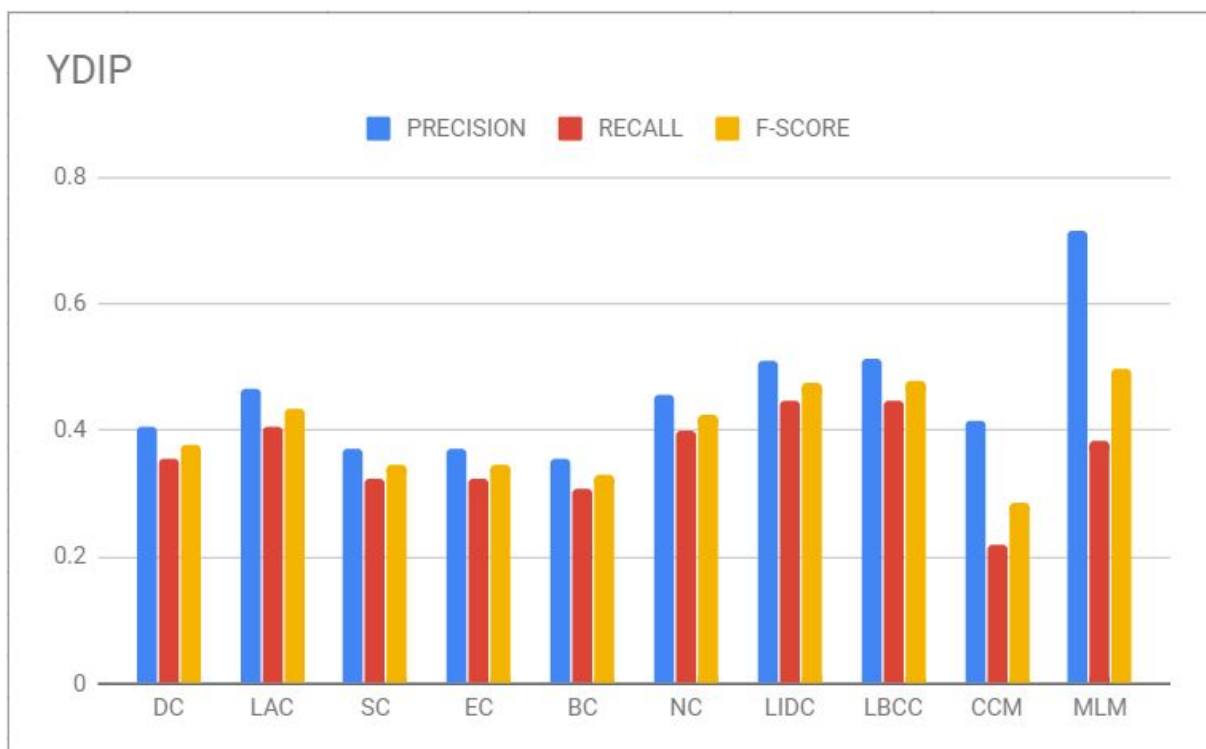


Figure 20 : Precision Recall F-score values of different methods on YDIP dataset

Comparison with other prediction measures

To evaluate the performance of CCM(combined centrality method) and MLM(modified LBCC method), we compared them with other prediction measures using the two datasets described in the [reference]Experimental data section. The compared prediction measures included LBCC, LIDC, DC, BC, SC, EC, NC, and LAC. The algorithm for LBCC was implemented using[link] and the other algorithms were implemented using CytoNCA, a plugin of Cytoscape for centrality analysis of PPI networks[reference]. First, proteins are ranked in descending order based on their node weights; second, we used threshold using [reference] selected those proteins which are higher than the threshold value; third, the selected proteins are ranked in descending order based on their centrality measures; fourth, we selected top 20% of the protein as essential; and finally, the number of true essential proteins was determined.

For the YMIPS dataset, in terms of f-score, CCM achieved the best result with 0.587. In terms of precision, MLM is having the highest score of 0.579. Finally in terms of recall value LBCC is highest with 0.430.

For the YDIP dataset, except for recall value, MLM exhibit superior performance with high f-score of 0.499 and high precision of 0.716. The recall value is highest in both LIDC and LBCC each with 0.446

Thus, our experiments indicate that modified LBCC can identify more essential proteins than the other methods in most cases.

The identification of essential proteins is helpful for comprehending the minimal requirements for cellular life, and many approaches based on topological properties have been proposed for discovering essential proteins in PPI networks. Most of the topology-based methods only concentrate on either local or global characteristics and are also sensitive to the network structure. LBCC outperformed classical topological centrality measures. In this paper, we propose two new methods : Combined Centrality(CCM), based on average of 6 different centralities i.e closeness centrality (CC), degree centrality (DC), network centrality (NC), eigenvector centrality (EC), information centrality (IC) and the local average connectivity-based method (LAC) with addition of a feature called node weight; and another method, Modified LBCC (MLM), based on the LBCC method proposed by [link] with node weight feature . We applied both CCM and MLM to two PPI networks of *Saccharomyces cerevisiae*: YMIPS and YDIP. We then conducted comprehensive comparisons of these two methods MLM and CCM

with the other eight previously proposed methods, including DC, BC, SC, EC, NC, LAC, LIDC and LBCC, in terms of precision, recall and f-score value. For YMIPS dataset, in terms of f-score, CCM outperformed recent prediction methods. Whereas MLM achieved the best precision score out of all prediction methods. For YDIP dataset, MLM outperformed all prediction methods in terms of precision and recall value. Hence, we conclude that CCM is more effective in terms of f-score in YMIPS dataset and MLM is a more effective in terms of precision in both YMIPS and YDIP dataset and in terms of f-score in YDIP dataset.

A combination of other centrality features , domains, structures of proteins along with the feature of node weight in disease specific datasets can be exploited in the future works for better predictions.

REFERENCE

- [1] C. Pál, B. Papp, and L. D. Hurst, “Genomic function: Rate of evolution and gene dispensability,” *Nature*, vol. 421, no. 6922, pp. 496–7; discussion 497–8, Jan. 2003.
- [2] E. A. Winzeler, “Functional Characterization of the *S. cerevisiae* Genome by Gene Deletion and Parallel Analysis,” *Science*, vol. 285, no. 5429, pp. 901–906, Aug. 1999.
- [3] H. B. Fraser, A. E. Hirsh, L. M. Steinmetz, C. Scharfe, and M. W. Feldman, “Evolutionary rate in the protein interaction network,” *Science*, vol. 296, no. 5568, pp. 750–752, Apr. 2002.
- [4] Y. Wang *et al.*, “Identification of essential proteins based on ranking edge-weights in protein-protein interaction networks,” *PLoS One*, vol. 9, no. 9, p. e108716, Sep. 2014.
- [5] S. T. Cole, “Comparative mycobacterial genomics as a tool for drug target and antigen discovery,” *Eur. Respir. J.*, vol. 20, no. Supplement 36, p. 78S–86s, Jul. 2002.
- [6] C. Qin, Y. Sun, and Y. Dong, “A New Method for Identifying Essential Proteins Based on Network Topology Properties and Protein Complexes,” *PLoS One*, vol. 11, no. 8, p. e0161042, Aug. 2016.
- [7] C. P. Loomis and R. M. Powell, “Sociometric Analysis of Class Status in Rural Costa Rica-A Peasant Community Compared with an Hacienda Community,” *Sociometry*, vol. 12, no. 1/3, p. 144, 1949.
- [8] L. C. Freeman, “A Set of Measures of Centrality Based on Betweenness,” *Sociometry*, vol. 40, no. 1, p. 35, Mar. 1977.
- [9] E. Estrada and J. A. Rodríguez-Velázquez, “Subgraph centrality in complex networks,” *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, vol. 71, no. 5 Pt 2, p. 056103, May 2005.
- [10] P. Bonacich, “Power and Centrality: A Family of Measures,” *Am. J. Sociol.*, vol. 92, no. 5, pp. 1170–1182, Mar. 1987.
- [11] J. Wang, M. Li, H. Wang, and Y. Pan, “Identification of essential proteins based on edge clustering coefficient,” *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 9, no. 4, pp. 1070–1080, Jul. 2012.

- [12] M. Li, J. Wang, X. Chen, H. Wang, and Y. Pan, "A local average connectivity-based method for identifying essential proteins from the network level," *Comput. Biol. Chem.*, vol. 35, no. 3, pp. 143–150, Jun. 2011.
- [13] J. Luo and Y. Qi, "Identification of Essential Proteins Based on a New Combination of Local Interaction Density and Protein Complexes," *PLoS One*, vol. 10, no. 6, p. e0131418, Jun. 2015.
- [14] L. M. Cullen and G. M. Arndt, "Genome-wide screening for gene function using RNAi in mammalian cells," *Immunol. Cell Biol.*, vol. 83, no. 3, pp. 217–223, Jun. 2005.
- [15] G. Giaever *et al.*, "Functional profiling of the *Saccharomyces cerevisiae* genome," *Nature*, vol. 418, no. 6896, pp. 387–391, Jul. 2002.
- [16] T. Roemer *et al.*, "Large-scale essential gene identification in *Candida albicans* and applications to antifungal drug discovery: *C. albicans* essential gene identification and antifungal drug discovery," *Mol. Microbiol.*, vol. 50, no. 1, pp. 167–181, Aug. 2003.
- [17] M. Li, H. Zhang, J.-X. Wang, and Y. Pan, "A new essential protein discovery method based on the integration of protein-protein interaction and gene expression data," *BMC Syst. Biol.*, vol. 6, p. 15, Mar. 2012.
- [18] X. Zhang, J. Xu, and W.-X. Xiao, "A new method for the discovery of essential proteins," *PLoS One*, vol. 8, no. 3, p. e58763, Mar. 2013.
- [19] X. Tang, J. Wang, J. Zhong, and Y. Pan, "Predicting Essential Proteins Based on Weighted Degree Centrality," *IEEE/ACM Trans. Comput. Biol. Bioinform.*, vol. 11, no. 2, pp. 407–418, Mar. 2014.
- [20] W. Peng, J. Wang, W. Wang, Q. Liu, F.-X. Wu, and Y. Pan, "Iteration method for predicting essential proteins based on orthology and protein-protein interaction networks," *BMC Syst. Biol.*, vol. 6, p. 87, Jul. 2012.
- [21] M. Li, Y. Lu, Z. Niu, and F.-X. Wu, "United Complex Centrality for Identification of Essential Proteins from PPI Networks," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 14, no. 2, pp. 370–380, 2017.
- [22] L. Zhu, J. Zhang, L. He, J. Wang, Z. Peng, and Z. Jian, "Essential Proteins Discovery Methods based on the Protein-Protein Interaction Networks," *American Journal of Biochemistry and Biotechnology*, vol. 13, no. 4, pp. 242–251, 2017.

- [23] M. Wu, X. Li, C.-K. Kwoh, and S.-K. Ng, "A core-attachment based method to detect protein complexes in PPI networks," *BMC Bioinformatics*, vol. 10, no. 1. p. 169, 2009.
- [24] R. Saito, H. Suzuki, and Y. Hayashizaki, "Construction of reliable protein-protein interaction networks with a new interaction generality measure," *Bioinformatics*, vol. 19, no. 6. pp. 756–763, 2003.
- [25] R. Saito, "Interaction generality, a measurement to assess the reliability of a protein-protein interaction," *Nucleic Acids Research*, vol. 30, no. 5. pp. 1163–1168, 2002.
- [26] H. N. Chua, W. -K. Sung, and L. Wong, "Exploiting indirect neighbours and topological weight to predict protein function from protein-protein interactions," *Bioinformatics*, vol. 22, no. 13. pp. 1623–1630, 2006.
- [27] S. Wang and F. Wu, "Detecting overlapping protein complexes in PPI networks based on robustness," *Proteome Sci.*, vol. 11, no. Suppl 1, p. S18, Nov. 2013.
- [28] Y. Zhang, H. Lin, Z. Yang, J. Wang, Y. Liu, and S. Sang, "A method for predicting protein complex in dynamic PPI networks," *BMC Bioinformatics*, vol. 17 Suppl 7, p. 229, Jul. 2016.
- [29] H. W. Mewes *et al.*, "MIPS: analysis and annotation of proteins from whole genomes in 2005," *Nucleic Acids Res.*, vol. 34, no. Database issue, pp. D169–72, Jan. 2006.
- [30] I. Xenarios, D. W. Rice, L. Salwinski, M. K. Baron, E. M. Marcotte, and D. Eisenberg, "DIP: the database of interacting proteins," *Nucleic Acids Res.*, vol. 28, no. 1, pp. 289–291, Jan. 2000.

APPENDIX

CODE

1. node_weight_calculator.py

```
# Program to calculate node weight
# from a given text file of datasets
import csv

# To read from dataset file (.csv)
with open('dataset_YMIPS_copy.csv', 'r') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter='\t')

# New file for writing unique nodes
with open('YMIPS_NodeSet.csv', 'w', newline='') as new_file:
    csv_writer = csv.writer(new_file, delimiter='\t')
    unique_list = []
    for line in csv_reader:
        unique_list.extend(line)
    unique_list = list(dict.fromkeys(unique_list)) # to take unique values of the list
    unique_list.sort()
    for items in unique_list:
        csv_writer.writerow([items])

# To read from new file
with open('YMIPS_NodeSet.csv', 'r') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter='\t')
```

```

# Opening data set file

with open('dataset_YMIPS_copy.csv', 'r') as csv_file2:
    csv_reader2 = csv.reader(csv_file2, delimiter='\t')

# To write nodes, its surrounding nodes and its degree

with open('YMIPS_NodeSet_2.csv', 'w', newline='') as new_file:
    csv_writer = csv.writer(new_file, delimiter='\t')
    for item in csv_reader:
        item = ".join(item)
        chain = [item]
        csv_file2.seek(0)
        for line in csv_reader2:
            if(item == line[0]):
                chain.append(line[1])
        csv_writer.writerow(chain)

# to calculate level 1 and level 2 nodes

with open('YMIPS_NodeSet_2.csv', 'r') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter='\t')

with open('YMIPS_NodeSet_2.csv', 'r') as csv_file2:
    csv_reader2 = csv.reader(csv_file2, delimiter='\t')

with open('YMIPS_NodeSet_3.csv', 'w', newline='') as new_file:
    csv_writer = csv.writer(new_file, delimiter='\t')

    for line in csv_reader:
        level_1 = line[1:]

```

```

level_2 = []
total = []
for item in level_1:
    csv_file2.seek(0)
    for row in csv_reader2:
        if(row[0] == item):
            level_2 = level_2 + row[1:]

total = level_1 + level_2
total = list(dict.fromkeys(total))
total.append(len(level_2))
csv_writer.writerow(total)

# to calculate node-weight
# this file contains parent nodes and their child nodes (level 1)
# with the help of this file node degree is calculated
with open('YMIPS_NodeSet_2.csv', 'r') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter='\t')

# this file contains both level 1 and level 2 nodes
with open('YMIPS_NodeSet_3.csv', 'r') as csv_file2:
    csv_reader2 = csv.reader(csv_file2, delimiter='\t')

# to write the node weight of respective nodes
with open('YMIPS_NodeSet_4.csv', 'w', newline='') as new_file:
    csv_writer = csv.writer(new_file, delimiter='\t')

    for line in csv_reader:
        value = [line[0]]    # for writing parent node

```



```

for item in csv_reader2:
    node_set = len(item) - 1  # no. of nodes in a subgraph
    degree = int(item[-1])    # total number of level 2 node
    value.append(degree)
    value.append(node_set)
    if (node_set != 0):
        node_weight = degree / node_set  # by formula
        value.append(node_weight)
        node_weight_list = [value[0], round(value[-1], 5)]
        csv_writer.writerow(node_weight_list)
    else:
        value.append('NULL')
        csv_writer.writerow([value[0]])
    break

```

2. threshold_calculator.py

```

# Threshold(L,M,H) : (mean, standev)
# YMIPS(0.62321, 0.69295, 0.7627) : (0.553460447, 0.4362381282)
import math

mean = 0.553460447
standev = 0.4362381282

k = [1, 2, 3]
# threshold[]
threshold_L = mean + k[0] * standev * (1 - 1 / (1 + math.pow(standev, 2)))
threshold_M = mean + k[1] * standev * (1 - 1 / (1 + math.pow(standev, 2)))

```

```

threshold_H = mean + k[2] * standev * (1 - 1 / (1 + math.pow(standev, 2)))
print(round(threshold_L, 5))
print(round(threshold_M, 5))
print(round(threshold_H, 5))

```

3. protein_matcher.py

```

# segregation of essential and non-essential proteins
# into separate files
import csv

with open('average.csv', 'r') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter='\t')

    with open('calc_EP.csv', 'w', newline='') as new_file:
        csv_writer = csv.writer(new_file, delimiter='\t')

        with open('calc_NEP.csv', 'w', newline='') as new_file2:
            csv_writer2 = csv.writer(new_file2, delimiter='\t')

            count = 0
            for line in csv_reader:
                count = count + 1

            cutoff = round(count * 0.2)

            csv_file.seek(0)
            count = 1
            for line in csv_reader:
                if(count <= cutoff):

```

```

        csv_writer.writerow(line)
        E_Pro = count
        count = count + 1
    else:
        csv_writer2.writerow(line)
        NE_Pro = count - E_Pro
        count = count + 1

# evaluation
with open('essen_pro.csv', 'r') as csv_file1:
    csv_reader1 = csv.reader(csv_file1, delimiter='\t')

with open('non-essen_pro.csv', 'r') as csv_file2:
    csv_reader2 = csv.reader(csv_file2, delimiter='\t')

with open('calc_EP.csv', 'r') as csv_file3:
    csv_reader3 = csv.reader(csv_file3, delimiter='\t')

with open('calc_NEP.csv', 'r') as csv_file4:
    csv_reader4 = csv.reader(csv_file4, delimiter='\t')

# True Positive calculation
count = 0
csv_file3.seek(0)
for line in csv_reader3:
    line = ".join(line)
    csv_file1.seek(0)
    for item in csv_reader1:
        item = ".join(item)

```

```

        if(line == item):
            count = count + 1
True_Pos = count
print('{} : {}'.format('True_Pos', count))

# False Positive calculation
count = 0
csv_file3.seek(0)
for line in csv_reader3:
    line = ".join(line)
    csv_file2.seek(0)
    for item in csv_reader2:
        item = ".join(item)
        if(line == item):
            count = count + 1
False_Pos = count
print('{} : {}'.format('False_Pos', count))

# True negative calculation
count = 0
csv_file4.seek(0)
for line in csv_reader4:
    line = ".join(line)
    csv_file2.seek(0)
    for item in csv_reader2:
        item = ".join(item)
        if(line == item):
            count = count + 1
True_Neg = count
print('{} : {}'.format('True_Neg', count))

```

```

# False negative calculation
count = 0
csv_file4.seek(0)
for line in csv_reader4:
    line = ".join(line)
    csv_file1.seek(0)
    for item in csv_reader1:
        item = ".join(item)
        if(line == item):
            count = count + 1
False_Neg = count
print('{} : {}'.format('False_Neg', count))

# 6 statistical measures
print('{} : {}'.format('Total_essential_Proteins', E_Pro)) # = 246 # 1285
print('{} : {}'.format('Total_non-essential_Proteins', NE_Pro)) # = 984 # 4394
Sensitivity = round(True_Pos / (True_Pos + False_Neg), 3)
Specificity = round(True_Neg / (True_Neg + False_Pos), 3)
Positive_predictive_value = round(True_Pos / (True_Pos + False_Pos), 3)
Negative_predictive_value = round(True_Neg / (True_Neg + False_Neg), 3)
F_measure = round((2 * Sensitivity * Positive_predictive_value) / (Sensitivity +
Positive_predictive_value), 3)
Accuracy = round((True_Pos + True_Neg) / (E_Pro + NE_Pro), 3)
print('{} : {}'.format('Sensitivity', Sensitivity))
print('{} : {}'.format('Specificity', Specificity))
print('{} : {}'.format('Positive_predictive_value', Positive_predictive_value))
print('{} : {}'.format('Negative_predictive_value', Negative_predictive_value))
print('{} : {}'.format('F_measure', F_measure))
print('{} : {}'.format('Accuracy', Accuracy))

```

RAW DATA

1. YMIPS_datafile.txt

Combined Centrality Method

Threshold:

Low : Total Proteins: 1230

True_Pos : 71

False_Pos : 174

True_Neg : 654

False_Neg : 276

Total_essential_Proteins : 246

Total_non-essential_Proteins : 984

Sensitivity : 0.205

Specificity : 0.79

Positive_predictive_value : 0.29

Negative_predictive_value : 0.703

F_measure : 0.24

Accuracy : 0.589

Medium : Total Proteins: 1013

True_Pos : 56

False_Pos : 146

True_Neg : 538

False_Neg : 231

Total_essential_Proteins : 203

Total_non-essential_Proteins : 810

Sensitivity : 0.195

Specificity : 0.787

Positive_predictive_value : 0.277

Negative_predictive_value : 0.7

F_measure : 0.229

Accuracy : 0.586

High: Total Proteins: 854

True_Pos : 45

False_Pos : 125

True_Neg : 456

False_Neg : 196

Total_essential_Proteins : 171

Total_non-essential_Proteins : 683

** Sensitivity (Recall[top 20%]): 0.187

Specificity : 0.785

** Positive_predictive_value (Precision[top 20%]) : 0.265

Negative_predictive_value : 0.699

** F_measure : 0.219

Accuracy : 0.587

Modified LBCC Method

Low: Total Proteins: 1230

True_Pos : 142

False_Pos : 102

True_Neg : 726

False_Neg : 205

Total_essential_Proteins : 246

Total_non-essential_Proteins : 984

Sensitivity : 0.409

Specificity : 0.877

Positive_predictive_value : 0.582

Negative_predictive_value : 0.78

F_measure : 0.48

Accuracy : 0.706

Medium: Total Proteins: 1013

True_Pos : 115

False_Pos : 86

True_Neg : 598

False_Neg : 172

Total_essential_Proteins : 203

Total_non-essential_Proteins : 810

Sensitivity : 0.401

Specificity : 0.874

Positive_predictive_value : 0.572

Negative_predictive_value : 0.777

F_measure : 0.471

Accuracy : 0.704

High: Total Proteins: 854

True_Pos : 99

False_Pos : 72

True_Neg : 509

False_Neg : 142

Total_essential_Proteins : 171

Total_non-essential_Proteins : 683

** Sensitivity (Re-call[top 20%]) : 0.411

Specificity : 0.876

** Positive_predictive_value (Precision[top 20%]) : 0.579

Negative_predictive_value : 0.782

** F_measure : 0.481

Accuracy : 0.712

2. YDIP_datafile.txt

Combined Centrality Method

Threshold:

Low : Total Proteins: 2023

True_Pos : 180

False_Pos : 219

True_Neg : 1144

False_Neg : 371

Total_essential_Proteins : 405

Total_non-essential_Proteins : 1618

Sensitivity : 0.327

Specificity : 0.839

Positive_predictive_value : 0.451

Negative_predictive_value : 0.755

F_measure : 0.379

Accuracy : 0.654

Medium : Total Proteins: 1594

True_Pos : 135

False_Pos : 179

True_Neg : 851

False_Neg : 346

Total_essential_Proteins : 319

Total_non-essential_Proteins : 1275

Sensitivity : 0.281

Specificity : 0.826

Positive_predictive_value : 0.43

Negative_predictive_value : 0.711

F_measure : 0.34

Accuracy : 0.619

High: Total Proteins: 927

True_Pos : 75

False_Pos : 106

True_Neg : 456

False_Neg : 267

Total_essential_Proteins : 185

Total_non-essential_Proteins : 742

** Sensitivity(Recall[top 20%]) : 0.219

Specificity : 0.811

** Positive_predictive_value (Precision[top 20%]) : 0.414

Negative_predictive_value : 0.631

** F_measure : 0.286

Accuracy : 0.573

Modified LBCC Method

Low: Total Proteins: 2023

True_Pos : 230

False_Pos : 173

True_Neg : 1190

False_Neg : 321

Total_essential_Proteins : 405

Total_non-essential_Proteins : 1618

Sensitivity : 0.417

Specificity : 0.873

Positive_predictive_value : 0.571

Negative_predictive_value : 0.788

F_measure : 0.482

Accuracy : 0.702

Medium: Total Proteins: 1594

True_Pos : 197

False_Pos : 120

True_Neg : 910

False_Neg : 284

Total_essential_Proteins : 319

Total_non-essential_Proteins : 1275

Sensitivity : 0.41

Specificity : 0.883

Positive_predictive_value : 0.621

Negative_predictive_value : 0.762

F_measure : 0.494

Accuracy : 0.694

High: Total Proteins: 927

True_Pos : 131

False_Pos : 52

True_Neg : 510

False_Neg : 211

Total_essential_Proteins : 185

Total_non-essential_Proteins : 742

** Sensitivity (Recall[top 20%]) : 0.383

Specificity : 0.907

** Positive_predictive_value (Precision[top 20%]) : 0.716

Negative_predictive_value : 0.707

** F_measure : 0.499

Accuracy : 0.691