B.Comp. Dissertation

Recognizing Small Protein Complexes from Protein Interaction Network

By

Bai Yu

Department of Computer Science

School of Computing

National University of Singapore

15 April 2013

Project ID: H114320

Advisor: Prof Limsoon Wong

Co-Supervisor: Chern Han Yong

ABSTRACT

Protein complexes play important roles in biological systems, thus recognizing protein complexes is important for understanding principles of organization and function in cells. Typical approaches analyze protein-protein interaction (PPI) data produced by high throughput experimental techniques, and predict regions of high density (i.e. clusters) as complexes. However, data from high-throughput experiments are often associated with high false positive and false negative rates, which is difficult to predict complexes accurately. For this project, the goal is to explore the prediction of small protein complexes from PPI networks, where a small complex is defined as being composed of fewer than four distinct proteins. Discovering such small complexes from PPI networks via traditional clustering approaches is challenging, because a huge number of small cliques exist in the network, of which very few are actually complexes. For example, there are many triangles (size-three cliques) exist in the network, while only a few of them correspond to complexes of size three. Another challenge is the noise inherent in PPI networks (such as extraneous edges and missing edges). To solve these challenges, I will explore in two steps: first, investigating the differences in topology between small and large complexes; second, using supervised learning and data integration to predict protein edges belonging to small and large complexes.

Contact: ydwrzzh446@gmail.com

ACKNOWLEDGEMENT

Prof. Limsoon Wong

Prof Wong supervised and advised on the direction of the project.

Chern Han Yong

Chern Han directed and co-supervised the whole project.

Table of Contents

INTRODUCTION

Proteins are biological molecules consisting of one or more chains of amino acids, which perform a vast array of functions within living organisms. While many proteins perform functions independently, the vast majority of proteins interact with each other for proper biological activity. We define a group of two or more associated polypeptide chains as protein complexes.

Protein complexes play important roles in biological systems. Studying and recognizing protein complexes help us better understand the principles of organization and function in cells. Recognizing and predicting unknown protein complexes also contributes to understanding diseases and drug discovery.

BACKGROUND AND LITERATURE REVIEW

In this phase of the study, I have been reading related papers from other researchers, to understand their procedures of clustering PPI networks to predict protein complexes, in either unsupervised or supervised learning ways.

Generally, the papers I have read can be classified into three categories: first, traditional, unsupervised clustering-based approaches; second, approaches that incorporate supervised learning; third, approaches that incorporate prior knowledge of biological complexes as heuristics, for example to pre-process the network or post-process the results.

Traditional Clustering Approaches

In the first category, Gavin, Bosche and Krause (2002)[6] dealt with large-size protein complexes. They used tandem-affinity purification (TAP) and mass spectrometry in a large-scale approach to obtain PPI data in yeast. They clustered this data using the MCL algorithm based on stochastic flow, and characterized multi-protein complexes in yeast. King, Przulj and Jurisica (2004)[8] proposed the RNSC algorithm, which minimizes a cost function based on inter- and intra-cluster edges, then filters the results based on cluster size, density and functional homogeneity. The algorithm appears to be an accurate and scalable method of detecting and predicting protein complexes within a PPI network. Adamcsek, Palla, and Farkas (2006)[1] created an algorithm called CFinder, which became quite popular. They performed clustering using the Clique Percolation Method, and additionally performed visualization. Li, Chen, Wang (2008)[10] created an algorithm IPCA by modifying the DCPlus algorithm. First of all, they generated a seed vertex and extended it by adding new vertices. They looked at the data of the diameter of known complexes and the portion of vertex degree in all edges in graph and decide whether to include the vertex in the complex. Experimental results showed that their algorithm IPCA recalled more known complexes than previously proposed clustering algorithms. Liu, Wong, Chua (2009)[11] used an iterative scoring method, Iterative AdjustCD, to score edges based on topological features, and a clustering method CMC (clustering-based on maximal cliques) to discover complexes from the weighted PPI network. They obtained

more complexes that matched known complexes, and also explored robustness in random addition and deletion of edges.

Supervised Approaches

The second category of papers uses supervised learning methods, by training on known complexes, to predict new complexes. Supervised learning approaches use known protein complexes or other useful datasets as the input training data and infer reasonable functions called classifiers to predict possible protein complexes. In Qi (2006)[13], various classifiers are compared: Support Vector Machines, Naive Bayes, Logistic Regression, Decision Tree-J48 and Random Forest. And for each classifier, they used two styles of feature encoding: "Detailed" and "Summary." They explored learning protein co-complex relationships (MIPS), direct protein - protein interaction (DIP), and protein co-pathway relationship (KEGG) inside the PPI network. The co-complex relationship is shown to be the easiest to predict and RF appears to be the most robust method. In Qi, Balem, Faloutsos and Judith's paper in 2008[12], they argue that in real life complexes are not all formed with the structure of clique in the PPI network. So they proposed a supervised graph local clustering method. The main idea is to learn graph topological patterns from known complexes, using a Bayesian network, search for new complexes that match these learned patterns. In Qiu, Noble's paper in 2008[14], researchers described and applied a framework for combining experimental data to identify pairs of yeast proteins as co-complexed protein pairs (CCPPs). They used kernel methods based on random walks to integrate diverse data such as PPI, gene expression, and annotations to predict CCPPs. They also three different types of networks—yeast two-hybrid, APMS and genetic interaction. In 2009 Wang, Kakaradov, Collins[17], integrated co-expression, co-regulation, interologs, co-localization, and gene ontology annotations, and used boosting, a state-of-the-art machine learning method, to train an affinity function that is aimed at predicting whether two proteins are co-complexed. They used the MIPS complexes as training data, and also proposed a clustering method HACO. They showed that HACO, which progressively merges sets of proteins with strongest affinity, produces the best results for complex reconstruction.

Yong, Liu, Chua, and Wong's[3] integrated PPI, functional associations, and literature co-occurrence with a maximum-likelihood Naïve-Bayes model to weight edges with their probability of being co-complex, and used multiple clustering algorithms to discover protein complexes. Their approach, Supervised Weighting of Composite Network (SWC) achieved high precision and recall for large (size >= 4) yeast and human complexes, but disappointing results for smaller complexes.

Incorporating Biological Knowledge

A third group of researchers incorporated prior biological knowledge in complex prediction, one of which is the core-attachment model of complexes. Gavin et al. (2006)[5] provided a detailed study on the organization of protein complexes and suggested that a complex consists of two parts: a core and an attachment. Many researchers adopted this concept and modified existing algorithms to recognize complexes by finding cores and attachments. In Leung, Xiang and Yiu's paper (2008)[9], they proposed a novel approach to identify complexes from PPI network. Their algorithm identified core proteins by taking two proteins into consideration each time, and calculating the number of interactions between these two proteins and the number of their common neighbors, with threshold p. The attachments were identified as those proteins that are common neighbors of over half of the core proteins. Then they filtered out the noisy cores and ranked the predicted complexes to get final result, The core detection result appears to be much better than Andreopoulos et al. (2007)[2]. In 2010, Srihari, Ning and Leong[15] carried out a different idea that they could couple with the classic MCL clustering method, and created a core-attachment based refinement method to reconstruct yeast complexes from scored (weighted) PPI networks in order to detect meaningful novel complexes. The researchers first partitioned the PPI network into multiple dense clusters using MCL approach, and then they post-processed clusters with the following steps: clustering the PPI network with MCL hierarchically; categorizing cores within clusters; filtering noisy clusters; recruiting proteins as attachments into clusters; extracting out complexes from clusters and finally ranking the predicted complexes. This algorithm, MCL-Caw, improved the predictions of MCL on PPI networks. Wu, Li, Kwoh and Ng (2009)[18] proposed Core-attachment based method (COACH), which first detected protein-

complex cores as the heart and include attachments into these cores to form meaningful complex structures. They defined a core by comparing the degree of the vertices with average degree of the PPI network, then applied a core-removal algorithm and a redundancy-filtering algorithm to get the final complexes. These papers are similar in that they all agree with the concept of the core-attachment model, and searched for cores first, then attachments. Their differences lie in how they identify the core or the attachments.

Another type of biological knowledge is the concept of date hubs, or proteins that inter-connect different biological modules that participate in different biological functions. Liu, Yong, Chua, and Wong[4] proposed that date hubs are a source of noise in PPI networks for complex prediction, as they may connect clusters that correspond to different complexes, leading to the prediction of erroneous clusters of multiple complexes fused together. Thus they removed hubs and decomposed the PPI network into subnetworks of different biological processes before clustering, and obtained improved performance.

Yet another prior knowledge was pointed out by Tatsuke and Maruyama[16], who observed that the frequency of sizes of reference complexes follow a power law distribution. Thus they introduced their Protein Partition Sampler (PPSampler) method, which uses Monte-Carlo Markov Chain to partition the PPI network into complexes whose size frequencies follow this power-law distribution. They obtain good results compare to other methods, and are notable for incorporating the size of complexes into their method.

In summary, many of these papers contain useful ideas for addressing the challenges of finding small complexes in PPI networks. Integrating diverse data, including PPI, sequence similarity, gene co-expression, functional associations, and literature using supervised approaches is a promising direction to pursue. Incorporating biological knowledge such as core-attachment models, date hubs, and complex sizes might be worth exploring as well. Overall, I can see that every approach has its own strengths and weaknesses, especially with regard to the sensitivity and specificity: they perform well in predicting different types of unknown protein complexes.

OBJECTIVES

This project explores prediction of protein complexes, especially small-size protein complexes, defined as protein complexes with size 2 and size 3, accurately. We adopt the SWC approach which integrates diverse data sources (PPI, STRING functional associations, and co-occurrence in PUBMED literature) with a Naïve-Bayes supervised learning approach, as this is effective in removing noisy and transient PPI edges, and filling in missing interactions.

We investigate why SWC performs poorly on small complexes, and hypothesize that this is because small complexes have very different characteristics, especially topological characteristics, from large complexes. Furthermore, the edges from large complexes overwhelm the edges from small complexes during training, leading to SWC learning to accurately score large complex edges, but poorly on smaller complexes. We investigate this hypothesis in the following section.

Next, we modify SWC to learn separate sub-models for scoring co-complex edges from small complexes and large complexes independently (which we call a sub-model approach). We show that this performs better than simply learning one combined model for all co-complex edges (which we call a non-sub-model approach).

ANALYSIS OF TOPOLOGY OF SMALL AND LARGE COMPLEXES

Data Source

This project involves and integrates 3 different data sources: PPI, STRING and PUBMED.

PPI score is obtained from BioGird, IntAct and MIMT database. It indicates the direct interaction score between these two proteins. PPI is scored by the reliabilities. A reliability is estimated for each experimental detection technique (eg. Y2H, TAP-MS, etc), using the proportion of pairs that share GO terms. PPI score is obtained by probabilistically combining the reliabilities of all the experiments that detected it. This dataset contains 55,945 interactions.

STRING (Search Tool for the Retrieval of Interacting Genes/Proteins) data is obtained from STRING database, where the interactions include direct (physical) and indirect (functional) associations. Scores are derived from four different sources: Genomic Context, High-throughput Experiments, Co-expression(Conserved Domain) and Previous knowledge[19]. This dataset contains 175,712 protein pairs. As STRING scores are between 0 and 1000, so we normalize the data to 0 to 1. Also, protein pair with STRING score below 500 are usually not reliable, so we removed data that is smaller than 500 for this project.

PUBMED is a free database accessing primarily the MEDLINE[21] database of references and abstracts on life sciences and biomedical topics, which is maintained by The United States National Library of Medicine (NLM) at the National Institutes of Health[20]. PUBMED indicates the co-occurrences of proteins or genes in papers. For each pair of proteins, the score is the Jaccard Index of the sets of papers that the proteins appear in. This dataset contains 161,213 protein pairs.

Topological Features

For each data source, besides the direct score between two proteins, we calculate 3 topological metrics: DEG, CD, NBC.

DEG [Degree] is identified as the total number of totals neighbors of two proteins. For protein X and Y, define the neighbors of X exclude Y as set N_x, and neighbors of Y exclude X as set N_y. Degree is obtained by calculating the cardinality of the union of all the neighbors of protein X and Y. However, as our edges are weighted, we use the weight values to calculate DEG. If a neighbour is connected to both X and Y, we use the higher weight. Our DEG is defined as

$$\mathrm{DEG} = \sum_{a \in \{Nx \cup Ny\}} \max\{w(X,a), w(Y,a)\}$$

Where w(a, b) is the weight between the two protein a and b.

The larger DEG is, the more neighbors this protein pair has. It is possible that the two protein with high DEG is in a big protein complex.

In Figure 3 below, protein X has neighbors n1, n2 and n4, protein Y has neighbors n2, n3 and n5. Assuming all edge weights are 1, then the DEG score for protein pair [X,Y] is 5.

CD [Czekanowski-Dice] score measures the proportion of shared neighbors of two proteins. It is defined as twice the total number of the common neighbors divided by the sum of the neighbors between these two proteins.

$$\mathrm{CD} = \frac{2*|Nx \cap Ny|}{|Nx| + |Ny|}$$

In our case with weighted edge, we use the weights instead of 1 to calculate the CD score. We use the Iterative Adjust CD program[11] to calculate the CD scores. This program uses multiple iterations to calculate the CD scores, where in each subsequent iteration, the weights from the previous iteration is used. Furthermore it includes a dampening factor to reduce the impact of protein pairs with very few neighbors.

$$CD = w^k(X,Y) = \frac{\Sigma_{x \in N_x \cap N_y}(w^{k-1}(n,Y) + w^{k-1}(n,Y))}{\Sigma_{x \in N_x} w^{k-1}(n,X) + \lambda_x^k + \Sigma_{x \in N_y} w^{k-1}(n,Y) + \lambda_y^k}$$

Where λ is a dampening factor, and $w^{k-1}(X,Y)$ is the previous calculated weight for the edge between X and Y. And dampening factor is defined as

$$\lambda_x^k = \max\{0, \frac{\Sigma_{n\in V}\Sigma_{m\in N_n} w^{k-1}(n,m)}{|V|} - \sum_{n\in N_x} w^{k-1}(n,x)\}$$

$$\lambda_y^k = \max\{0, \frac{\Sigma_{n\in V}\Sigma_{m\in N_n} w^{k-1}(n,m)}{|V|} - \sum_{n\in N_y} w^{k-1}(n,y)\}$$

The larger CD score is, the more shared common neighbors exit, and the two proteins are more likely to exists in the same densely connected protein group.

In Figure 3 below, protein X has neighbors n1, n2 and n4, protein Y has neighbors n2, n3 and n5. The common neighbor is n2. For simplicity, the basic CD score for protein pair [X,Y] is

$$CD = \frac{2*|\{n1,n2,n4\}\cap\{n2,n3,n5\}|}{|\{n1,n2,n4\}|+|\{n2,n3,n5\}|} = \frac{1}{3}.$$

NBC [Neighborhood Connectivity] measures the connectivity between the neighbors of a pair of proteins. NBC is defined as the number of edges between the neighbors of these two proteins divided by the count of all the possible interactions between all the neighbors of the two proteins. Since our edges are weighted, we use the weights of the edges to calculate NBC. Our NBC score is defined as:

$$NBC = \frac{\Sigma(w(n_a,n_b)\,|\,n_a\in(N_x\cup N_y)\wedge n_b\in(N_x\cup N_y))}{|Nx\cup Ny|*(|Nx\cup Ny|-1)+\lambda}$$

Where if two protein a and b are neighbour, function w(a, b) represents the weight between a and b, and λ is a dampening factor.

NBC score indicates how densely connected are the neighbors of the protein pair. The larger NBC score is, the more interactions exist between the neighbors, and the two proteins are more likely to exist in a big densely connected protein group.

In Figure 1.1 below, protein X has neighbors n1, n2 and n4, protein Y has neighbors n2, n3 and n5. Common neighbour is n2. Then the NBC score for protein pair [X,Y], assuming $\lambda = 0$, is

$$NBC = \frac{w(n2, n3) + w(n3, n2)}{5 * 4} = \frac{2}{20}$$

which is very very small. And we know that the two proteins are not in a very dense protein group.

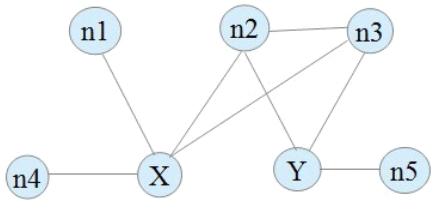

Figure 1.1

Combining the 3 data source and topological features described above, we are able to get 12 feature scores for a protein pair, mainly the PPI score, PPI_CD score, PPI_DEG score, PPI_NBC score, STRING score, STRING_CD score, STRING_DEG score, STRING_NBC score, PUBMED score, PUBMED_CD score, PUBMED_DEG score, and PUBMED_NBC score. These 12 scores will be treated as input training features for supervised learning approach.

In addition to the three data source for protein pairs, we obtain a reference CYC2008 file which contains 408 already known protein complexes with all the proteins inside the complexes. The protein pairs from 3 data sources may not exist in any of the known complexes, or alternatively, they may be inside small protein complexes with size 2 or 3, or large size protein complexes with size 4 and above. When we analysis the protein pair, if an edge is inside multiple protein complexes, they will be classified as in the protein largest complex.

So finally, our dataset consists of 12 features for each protein pair: PPI, PPI_CD, PPI_DEG, PPI_NBC, STRING, STRING_CD, STRING_DEG, STRING_NBC, PUBMED, PUBMED_CD, PUBMED_DEG, PUBMED_NBC. Our class label is called COCOMP_SIZE_CLASS, which has value 0, 2, 3, 4, which corresponds to a pair being not in a complex, in a complex of size 2, or size 3, or size 4 and above, respectively. Later comparison will be made to verify that classification into the separate sub-models of classes 2,3,4 (complexes of different sizes) has better performance than single model (complexes of all sizes).

Now we want to verify the first hypothesis that small and large protein complexes have different topological characteristics in terms of CD, DEG and NBC scores. Below are 2 examples for illustration. Figure 1.2 represents two protein X and Y in a small protein complex of size 2. X has neighbors n1 and n2, and Y has neighbors n1, n3, and n4. They share one common neighbour n1. To make calculation simple and straight forward, each edge is assigned with score 1.

DEG = 4, in total there are 4 neighbors of this protein pair.

CD = $\frac{2*1}{2+3} = 0.4$, only one neighbour acts as the common neighbour.

NBC = $\frac{0}{4*3} = 0$, as none of the neighbour connect to each other.

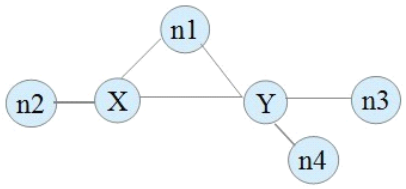

Figure 1.2

In Figure 1.3 below, protein X and Y exists in large complex of size 6, X has neighbors n1 and n2, and Y has neighbors n1, n3, and n4. They share one common neighbour n1. To make the calculation simple and straight forward, each edge is still assigned with score 1.

DEG = 4, in total there are 4 neighbors of this protein pair.

CD = $\frac{2*4}{4+4}=1$, all of the 4 neighbors act as common neighbour.

NBC = $\frac{4+4}{3*3}=0.6667$, there are 4 interactions (n1-n3, n1-n4, n2-n3, n2-n4) between all the neighbors out of total 6 possible interactions.

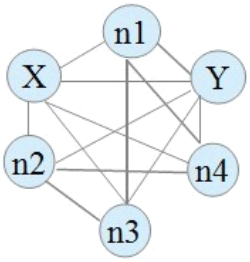

Figure 1.3

Comparing the two protein pairs described above, the DEG scores are the same, however, protein pair in small complex have small CD score and NBC score, protein pair in large complexes will have large CD score and NBC scores because in large protein complexes, proteins normally densely connect to each other. While, there are very few protein pair edges in small complexes, thus the protein pair will have less shared neighbors and not so densely connected neighbour.

Real Examples

Lets study real protein pair examples in CYC file.

Arg2p/Arg5,6p complex[22] is a small protein complex of size 2 composed of protein YER069W and YJL071W. These two proteins are obtained by Affinity Capture-Western method and they share the same PUBMED ID 11553611. All the 12 feature scores we get for the protein pair are shown in Table 1 below:

PPI	PPI_CD	PPI_DEG	PPI_NBC
0	0	7.284109039	0.0157894736842105
STRING	STRING_CD	STRING_DEG	STRING_NBC
0	0	0.8398454514995	0.00155175756015138
PUBMED	PUBMED_CD	PUBMED_DEG	PUBMED_NBC
0.945110528	0.3031	35.0801826734779	0.226732876947333

Table 1

As this proteins pair is obtained with PUBMED database, PPI and STRING features shows very low scores. They don't have direct interaction in PPI database or STRING database. For PUBMED, the edge has PUBMED score 0.945 which is almost near 1, and PUBMED_CD score and PUBMED_NBC score are only 0.3031 and 0.2267 respectively. With DEG score of 35.0802, result shows that this protein pair has many neighbors but these total number of common neighbour is very small and their neighbors are not connected to each other with high probability.

H+-transporting ATPase, Golgi[23] is a large protein complex of size 13 composed of protein YBR127C, YDL185W, YEL027W, YEL051W, YGR020C, YHR026W, YHR039C-A, YKL080W, YLR447C, YMR054W, YOR332W, YPL234C, and YPR036W. These 13 proteins are obtained by Co-localization and Biochemical Activity method and they share the same PUBMED ID 11592965. Let us study the edge scores between protein YBR127C-YDL185W and YBR127C-YLR447C. All the 12 feature scores we get for these two protein pairs are:

For YBR127C-YDL185W pair:

PPI	PPI_CD	PPI_DEG	PPI_NBC
1	0.5371	117.330493221	0.0583901544931359
STRING	STRING_CD	STRING_DEG	STRING_NBC
0.0985915492957746	0.3105	3.06977514368513	0.0279189584720096
PUBMED	PUBMED_CD	PUBMED_DEG	PUBMED_NBC
0.99999990879764	0.3952	114.180010358352	0.0471119268830958

Table 2

Compared to YER069W-YJL071W protein pair inside the small protein complex Arg2p/Arg5,6p, YBR127C-YDL185W has much higher PPI, PPI_CD, PPI_NBC score, and STRING, STRING_CD, STRING_NBC score, where high CD and NBC can explained the topological characteristic difference among complexes of different sizes. They have similar high PUBMED score. PUBMED_CD score equals 0.3952 which is higher than 0.3031 that in YER069W-YJL071W.

Only PUBMED_NBC for YBR127C-YDL185W equals 0.047 and is smaller than 0.2267 in YER069W-YJL071W. In this case, we can still examine the protein to be surrounded in a densely connected protein area. The total number of neighbors of this protein pair YBR127C-YDL185W is 114.1800 which is much bigger than 35.0801 in YER069W-YJL071W. To reach PUBMED_NBC = 0.2267 with PUBMED_DEG = 35.0801{1} and PUBMED_NBC = 0.04711 with PUBMED_DEG = 114.1800{2}:

$$\frac{2*x_1}{35*(35-1)} = 0.2267\ \{1\},\quad x_1 = 127.1787$$

$$\frac{2*x_2}{114*(114-1)} = 0.0471\ \{2\},\quad x_2 = 303.4355$$

X_1 shows that there are 127.1787 interactions between neighbors of protein pair YER069W-YJL071W and X_2 shows that there are 303.4355 interactions between neighbors of protein pair YBR127C-YDL185W, which is much larger.

So as for CD score, 0.3952 out of 114.1800 is much bigger than 0.3031 out of 35.0802.

Same applies to protein pair YBR127C-YLR447C inside complex H+-transporting ATPase, Golgi:

PPI	PPI_CD	PPI_DEG	PPI_NBC
0.999630052	0.4494	158.473566616	0.0258535667713636
STRING	STRING_CD	STRING_DEG	STRING_NBC
0.25	0.7123	5.03592845371551	0.0212086250458227
PUBMED	PUBMED_CD	PUBMED_DEG	PUBMED_NBC
0.939524018194306	0.3752	66.4761311461364	0.0699691181934244

Table 3

PPI score and STRING score are higher and higher PPI, STRING, PPI_CD, PPI_NBC, STRING_CD, STRING_NBC, and PUBMED_CD scores are expected and observed. Data shows a more densely connected neighborhood environment for protein pair YBR127C-YLR447C compared to protein pair YER069W and YJL071W that is in a small protein complex.

We hypothesize that protein in large complexes tend to be densely connected to each other. Thus a co-complex edge inside large size complex has high CD and NBC score, while co-complex edge inside small size complex has low CD and NBC score given similar/same DEG score.

When the protein pair is not inside a complex, DEG, CD, NBC are comparatively small. For the wo proteins YER069W and YOR348C, YER069W is from Arg2p/Arg5,6p complex with size 2 and YOR348C is from not present in CYC file. They are not in the same protein complexes. And the feature scores are all very low as expected:

PPI	PPI_CD	PPI_DEG	PPI_NBC
0	0	13.7756655	0.0898390588368421
STRING	STRING_CD	STRING_DEG	STRING_NBC
0.05	0.01959	2.14489398875756	0.0142303470173485
PUBMED	PUBMED_CD	PUBMED_DEG	PUBMED_NBC
0	0.06674	62.3413267581405	0.135903097307198

Table 4

To illustrate all the different characteristics of the features, chart for edge score frequencies are plotted with regard to different protein complex sizes. Below are the histograms plotted from the arff file we have. For different topological features, frequency of the data ranging from 0 to 1 are displayed according to the sub models: not in the protein complex [class 0], inside small protein complex [class 2-3], and inside large protein complex [class 4]. Frequency in a range is obtained as the total number of scores in this range divided by the total number of the records.

For PPI score shown in Figure 2.1.1, protein pair not inside a protein complex has the most score between 0 and 0.1 and a very small amount of score laying between 0.8 to 1. Protein pairs in small complexes have scores most observed in between 0.9 and 1, and for protein pairs inside large complexes, we observe scores both between 0 - 0.1 and 0.8 - 1. This is reasonable as protein pair in small complexes need to have strong direct interactions to form a small group, while protein pair in large protein complexes may be connected to other proteins inside the complex without connecting to each other.

For PPI_CD score shown in Figure 2.1.2, protein pairs not inside a protein complex has the most score between 0 and 0.1. Protein pair edges in small complexes have most scores between 0 and 0.1 and lesser and lesser when the score gets bigger. And for protein pairs in large complexes, most score are also observed between 0 and 0.1, some of the scores are observed between 0.7 and 0.9. This graph shows the difference between PPI_CD score and PPI score and prove the hypothesis in the topological feature

section, in small protein complexes, PPI score for protein pairs are high and PPI_CD score tend to be small as there should not be too many common neighbors between the two proteins.

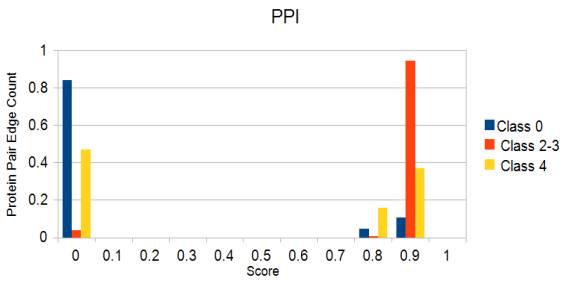

Figure 2.1.1

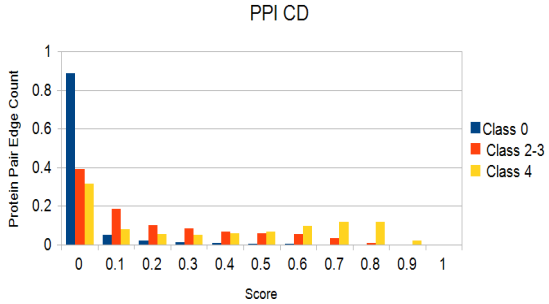

Figure 2.1.2

For PPI_DEG score shown in Feature 2.1.3, DEG scores for PPI scores are more evenly distributed. it shows no significant differences among different classes.

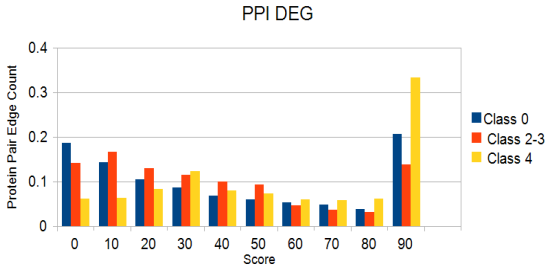

Figure 2.1.3

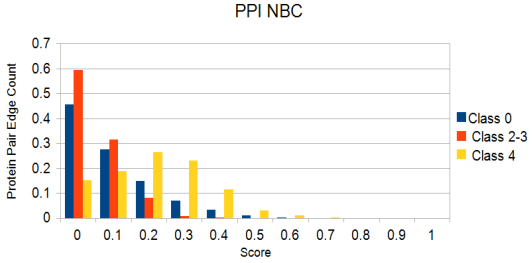

Figure 2.1.4

And for PPI_NBC score shown in Figure 2.1.4, for both protein pairs not inside a protein complex and inside small protein complexes, the scores are mostly in between 0 and 0.1, and become lesser and lesser when the score becomes bigger. For protein pairs in large complexes, most score are also observed between 0.2 and 0.4. The graph also prove the hypothesis that PPI_NBC score tend to be small in small protein complexes and big in large size protein complexes.

When we plot the graphics for STRING, and PUBMED scores, we expect histograms to have similar distribution with PPI histograms, where direct interaction score for small complexes is high and CD NBC scores for large complexes are high.

STRING histograms for different features are more similar to PPI histograms except that the DEG scores that are larger than 90 in STRING have a larger percentage than that in PPI scores.

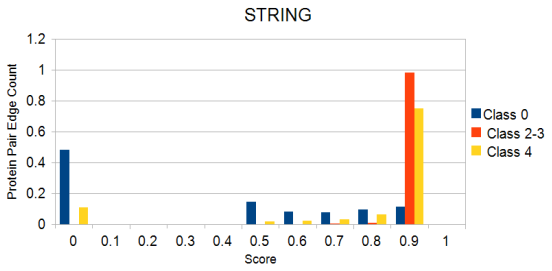

Figure 2.2.1

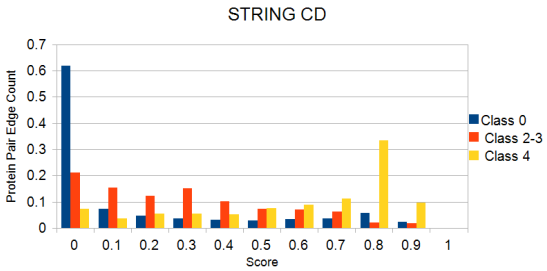

Figure 2.2.2

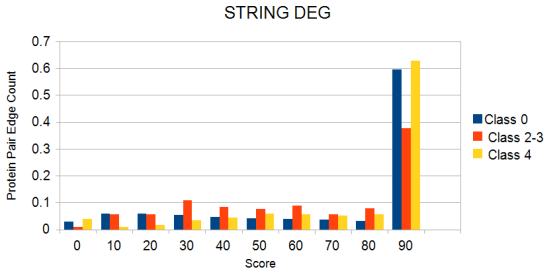

Figure 2.2.3

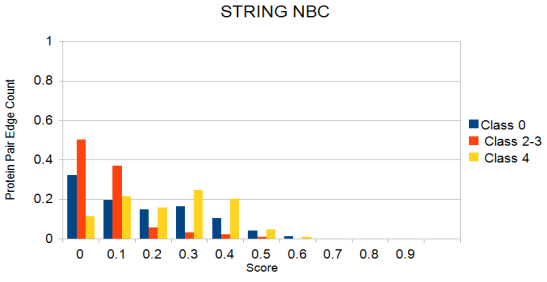

Figure 2.2.4

However, there are also differences in PUBMED graphs compared to PPI histograms. PUBMED histogram does not have high percentage of scores falling between 0.9 and 1 for any of the sub model as shown in Figure 2.3.1. Remember PUBMED represents co-occurrences of proteins or genes in papers and it has different aspect from PPI, which supplements data source from PPI database. Another interesting observation is that PUBMED_NBC scores are all between 0 and 0.1, meaning that neighborhood connectivity of all the PUBMED data are all very low as shown in Figure 2.3.4.

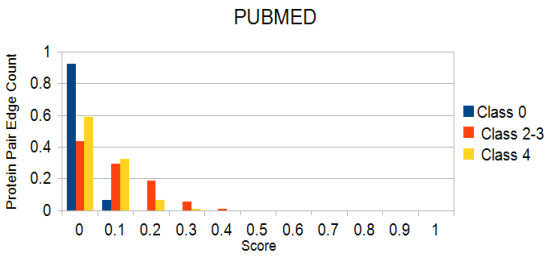

Figure 2.3.1

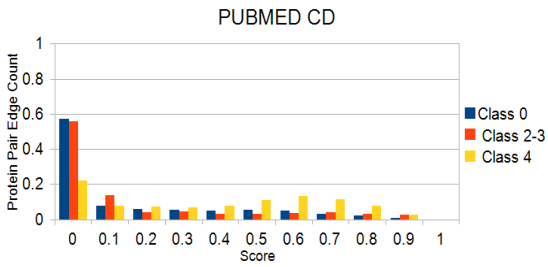

Figure 2.3.2

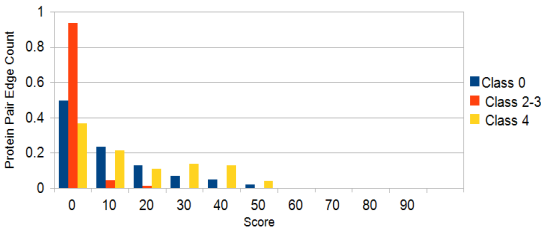

Figure 2.3.3

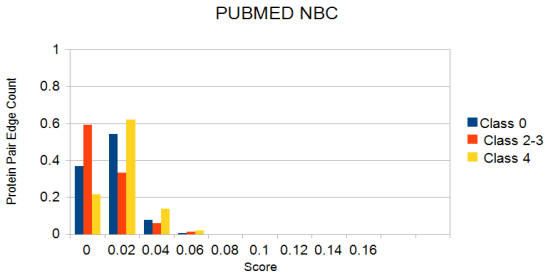

Figure 2.3.4

For each of the features, average and standard deviation are also calculated shown in the table below.

Average and Standard Deviation Table for all the features:

Models / Features	Not in complex	In small Complex	In large Complex
PPI	0.1445 ± 0.3377	0.9534 ± 0.2000	0.4997±0.4747
PPI_CD	0.0405±0.1078	0.2319±0.2241	0.3965±0.3226
PPI_DEG	70.3357±140.6605	48.3165±48.0492	72.3133±57.1353
PPI_NBC	0.1407±0.1238	0.0941±0.0702	0.2593±0.1423

STRING	0.3841±0.3884	0.9912±0.0594	0.8491±0.3117
STRING_CD	0.2004±0.2922	0.2677±0.2405	0.6350±0.2815
STRING_DEG	176.9643±182.1094	98.22711±82.2820	152.3744±95.3350
STRING_NBC	0.2203±0.1637	0.1243±0.0958	0.2875±0.1491
PUBMED	0.0355±0.0422	0.1308±0.1038	0.0861±0.0760
PUBMED_CD	0.2001±0.2641	0.1767±0.2464	0.4274±0.2945
PUBMED_DEG	14.5670±13.0862	3.9091±4.3617	20.2841±16.4095
PUBMED_NBC	0.0243±0.0117	0.0188±0.0151	0.0285±0.0135

Table 5

Significant features and scores to observe are PPI score for small complex, which has a very high average 0.9534 and standard deviation 0.2, PUBMED score in small complex, which has a even higher average of 0.9912 and standard deviation 0.0594, DEG score for STRING has a comparatively smaller average and standard deviation compared to PPI and PUBMED. Meanwhile, DEG for STRING has very small score for protein pair in small protein complexes compared to not in complexes and in large size protein complexes. Comparing NBC scores among different models, values for small protein complexes are much smaller than the values in large protein complexes and non protein complexes.

To check if the differences in feature values' means between the three classes are significant, we perform a two-tailed permutation test. The table below shows the P-values, and shows that all the differences are significant, except for large vs non-complex for PPI_DEG, and small vs non-complex for PUBMED_CD.

Features Model	Small VS None complex	Large VS None complex	Large VS Small complex
PPI	< 0.0002	< 0.0002	< 0.0002
PPI_CD	< 0.0002	< 0.0002	< 0.0002
PPI_DEG	< 0.0002	0.1556	< 0.0002
PPI_NBC	< 0.0002	< 0.0002	< 0.0002
STRING	< 0.0002	< 0.0002	< 0.0002
STRING_CD	< 0.0002	< 0.0002	< 0.0002
STRING_DEG	< 0.0002	< 0.0002	< 0.0002
STRING_NBC	< 0.0002	< 0.0002	< 0.0002
PUBMED	< 0.0002	< 0.0002	< 0.0002
PUBMED_CD	0.794	< 0.0002	< 0.0002
PUBMED_DEG	< 0.0002	< 0.0002	< 0.0002
PUBMED_NBC	< 0.0002	< 0.0002	< 0.0002

Table 6

The above discussion shows that there are significant differences between protein edges corresponding to small vs. large complexes. Furthermore, a vast majority of co-complex edges come from large complexes rather than small complexes. Figure 3.1 the number of protein complexes of different sizes. There are 172 protein complexes are of size 2, 87 of

size 3, 44 protein complexes of size 4 and remaining are the few large size protein complexes. However, Figure 3.2 shows that the total number of protein pair existing in protein complex of size 2 and 3 are much less than the protein pair in large complexes, as one big protein complex will contribute many protein pairs, which overwhelm total number of protein pairs in small protein complexes. So, during the training period of the supervised learning, the classifier learns characteristics of large complexes and result in the poor performance of the prediction of small protein complexes.

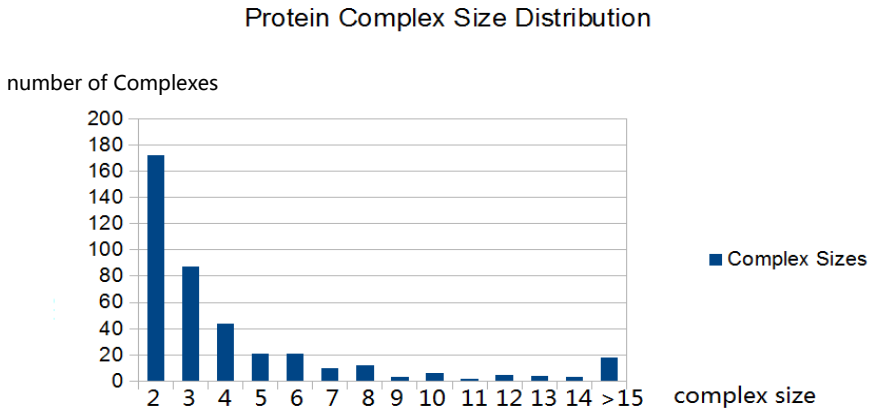

Figure 3.1

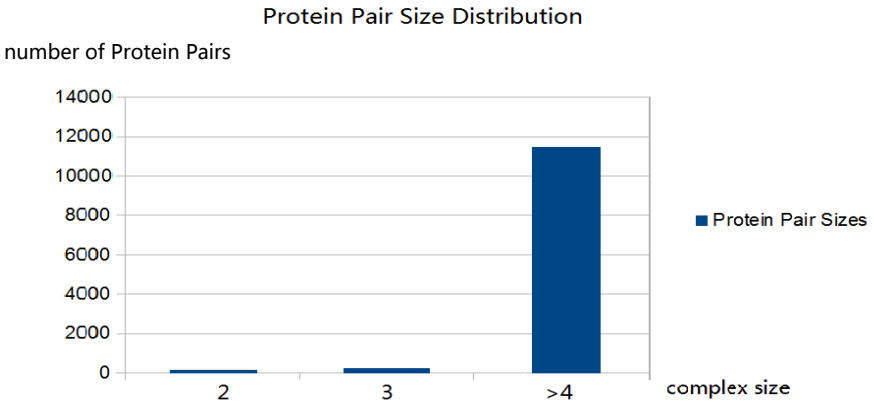

Figure 3.2

SUPERVISED LEARNING FOR CO-COMPLEX EDGE CLASSIFICATION

Naïve-Bayes supervised approach

We perform supervised learning to learn Naïve-Bayes models of co-complex edges belonging to complex of size 2, 3, or 4 and above (which we call the sub-model approach), and compare this against learning a single Naïve-Bayes model of co-complex edges of all complex sizes (which we call the non-sub-model, or single-model approach).

Each data instance is a protein pair, with 12 features: PPI, PPI_CD, PPI_DEG, PPI_NBC, STRING, STRING_CD, STRING_DEG, STRING_NBC, PUBMED, PUBMED_CD, PUBMED_DEG, PUBMED_NBC. For the sub-model approach, the class is called COCOMP_SIZE_CLASS, which has value 0, 2, 3, 4, which corresponds to a pair being not in a complex, in a complex of size 2, or size 3, or size 4 and above, respectively. For the single-model approach, the class is called COCOMP, which has values 0 or 1, corresponding to a pair being not in a complex, or being in a complex, respectively.

The Naïve-Bayes model is a simple probabilistic classifier based on applying Bayes' theorem with the strong independence assumptions[24]. Naive Bayes assumes that give class label, the presence or absence of a feature is unrelated to the presence or absence of any other feature. For all the features F_1, F_2, ..., F_n, the prediction score of a class label C is defined as:

$$p(C \mid F_1, F_2, \ldots, F_n)$$

With the assumption of the independency of all the features, we obtain the probability through Bayes's theorem:

$$\mathrm{P}(C \mid F_1, F_2, \ldots F_n) = \frac{p(C)\, p(F_1, F_2, \ldots, F_n \mid C)}{p(F_1, F_2, \ldots F_n)}$$

Applying chaining rule, and assume that all the features are independent of each other, we derive the probability as following:

$$P(C \mid F_1, F_2, \ldots F_n) = \frac{p(C)\,p(F_1 \mid C)\,p(F_2, \ldots, F_n \mid C, F_1)}{p(F_1, F_2, \ldots F_n)}$$
$$= \frac{p(C)\,p(F_1 \mid C)\,p(F_2, \ldots, F_n \mid C)}{p(F_1, F_2, \ldots F_n)}$$
$$= \frac{p(C)\,p(F_1 \mid C)\,p(F_2 \mid C)\,p(F_3, \ldots, F_n \mid C)}{p(F_1, F_2, \ldots F_n)}$$
...
$$= \frac{p(C)\,p(F_1 \mid C)\,p(F_2 \mid C) \ldots p(F_n \mid C)}{p(F_1, F_2, \ldots F_n)}$$

In our case, there are 12 features, so n = 12, and the probability of

$$P(F_1, F_2, \ldots F_n) = P(F_1)P(F_2)\ldots P(F_{12})$$

based on independency assumption. And then P(C) is the probability of the existence of the class C in all the sample data.

We perform supervised discretization on each feature using Minimum Description Length (MDL) discretization, and model each likelihood function $P(F_i = f|C)$ as:

$$P(F_i = f \mid C) = \frac{n_{Fi\,=\,f,\,C}}{n_C}$$

Where $n_{Fi=f,C}$ is the number of edges whose feature F_i has discretized value f, and is from class C, and n_C is the number of edges from class C.

Although Naïve-Bayes makes strong independence assumptions, an analysis of the Bayesian classification problem in 2004 showed that there are sound theoretical reasons for the apparently implausible efficacy of naive Bayes classifiers[25]. Naïve-Bayes classifiers have also been found to perform well even when the independency assumptions are violated[7].

Now we investigate our second hypothesis: learning separate sub model for small complexes will improve performance, compared to single model for all complexes.

Experiment description

Machine learning software used in this project is WEKA (Waikato Environment for knowledge Analysis). The input training data is the arff file generated from PPI, STRING and PUBMED database combined with CYC file. The data structure is: Protein_Pair_Name, PPI, PPI_CD, PPI_DEG, PPI_NBC, STRING, STRING_CD, STRING_DEG, STRING_NBC, PUBMED, PUBMED_CD, PUBMED_DEG, PUBMED_NBC, COCOMP, COCOMP_SIZE_CLASS. The first attribute is the protein pair that we want to predict. The last 2 attributes are class labels. Remaining are scores for 12 features related to this protein pair.

Cross validation

To reduce variability and guard against Type I and Typy II errors, 10 rounds of cross validation are performed during training phase. The input data set is partitioned into 10 subsets and one round of cross-validation performs the analysis on 9 subsets, and validates the analysis on the remaining subset. Performing cross validation help increase the performance of suggested hypotheses.

Evaluation: precision-recall graph

To evaluate the Naïve-Bayes approach on the input data, we plot precision recall graph.

Precision is the fraction of retrieved instances that are relevant

Recall is the fraction of relevant instances that are retrieved. Definition of the two terms are as following:

$$\Pr ecision = \frac{true\_positive}{true\_positive + false\_positive}$$

$$\mathrm{Re}\, call = \frac{true\_positive}{true\_positive + false\_negative}$$

Good performance of a prediction is that no matter what the recall value is, precision remain high (1 if possible). We select different features each time to perform the training

and validation with 10-cross-validation on Naïve-Bayes at each round. Then the output precision-recall graphs give different performance at each round. By comparing the precision-recall graphs, we are able to evaluate the performance of the selection of features.

Comparison for single model vs sub model

As predicted, training data on sub models have better performance than training data on single model because protein pair in small and large protein complexes have different topological characteristics. Too many protein pairs in large protein complexes will overwhelm the protein pairs in small complexes during training. So we perform two sets of training and learning. One is the non-sub-model-approach using class label COCOMP only with value 1 and 0 indicating whether the protein pair is in the complex or not. And the other one is sub-model-approach using label COCOMP_SIZE_CLASS with value 0, 2, 3, 4 to classify the protein pair as not in protein complex, in protein complex size 2, in protein complex of size 3, or in large protein complex with size 4 and above.

Sub-model approach is simple, after selecting the targeted features, we use the Naïve-Bayes classifier with 10 cross validation to train and validate the input data. Then draw precision-recall graph with regard to 4 sub models {0, 2, 3, 4}.

For non-sub-model approach, we firstly use the Naïve-Bayes classifier with 10 cross validation to get the output data with prediction of class {0, 1}. Then combining the CYC file, we classify the protein pairs classified as in class 1 to individual sub classes: size 2, size 3, or size 4 and above. According to the output data, drawing of the precision recall graph can be performed and thus we can compare the precision recall graph between sub-model-approach and non-sub-model-approach to get convincing conclusion.

Results

For the first round of training, we choose all the 12 features as the input features and use COCOMP (0 or 1) as the class label. With Naïve-Bayes approach and 10-cross validation, we are able to get the output prediction score for both classes and the #actual #predicted label. Analyzing the output file, we take out the protein pairs that are defined as in class,

look into the CYC file to find out whether they are in class 2, class 3 or class 4 and above complexes and classify them accordingly. Then we can plot the precision recall graph for each sub classes. In the second round of training, we still use all the 12 features as the input, but with COCOMP_SIZE_CLASS as the class label to proceed with Naïve-Bayes training. And the output file can be proceed to plot precision recall graphs directly.

Below are the precision-recall graphs for class 1 (protein pair is inside protein complex), class 2 (protein pair inside protein complex of size 2), class 3 (protein pair inside protein complex of size 3), class 4 (protein pair inside protein complex of size 4 and above), with respect to non-sub-model and sub-model approaches with all the features. Significant better performance with sub-model approach for class 2 and class 3 are shown in Figure 4.2 and 4.3.

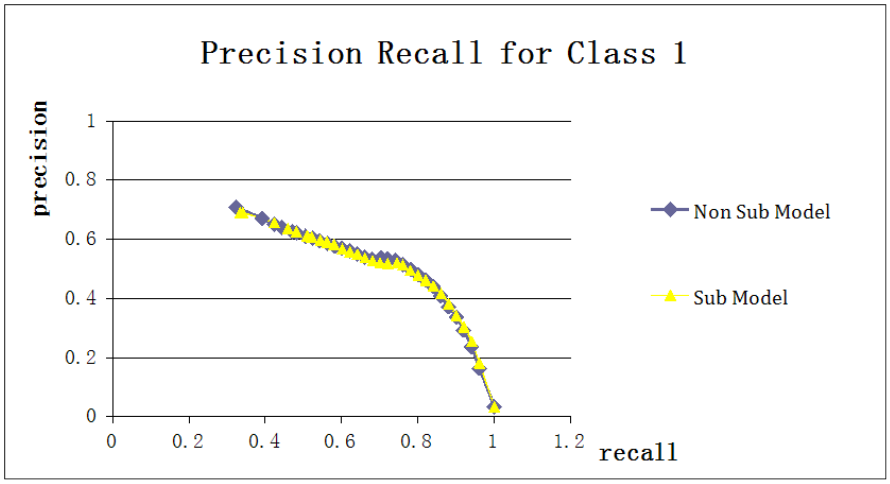

Figure 4.1.1

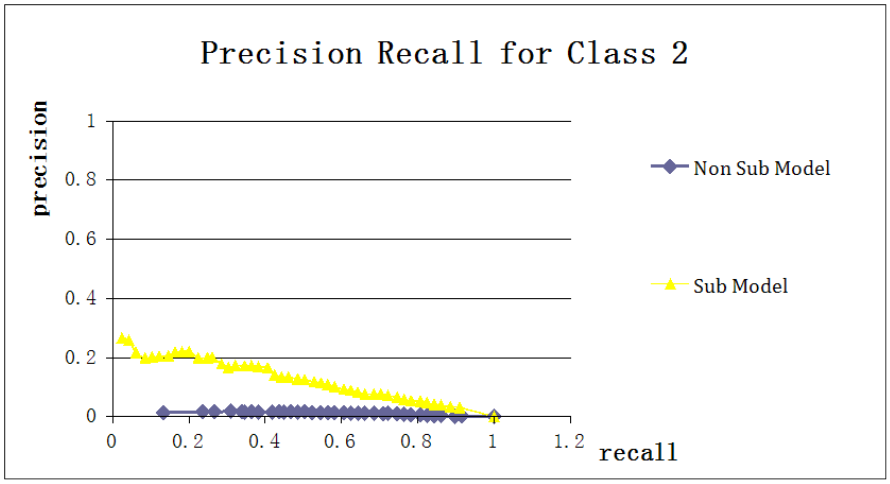

Figure 4.1.2

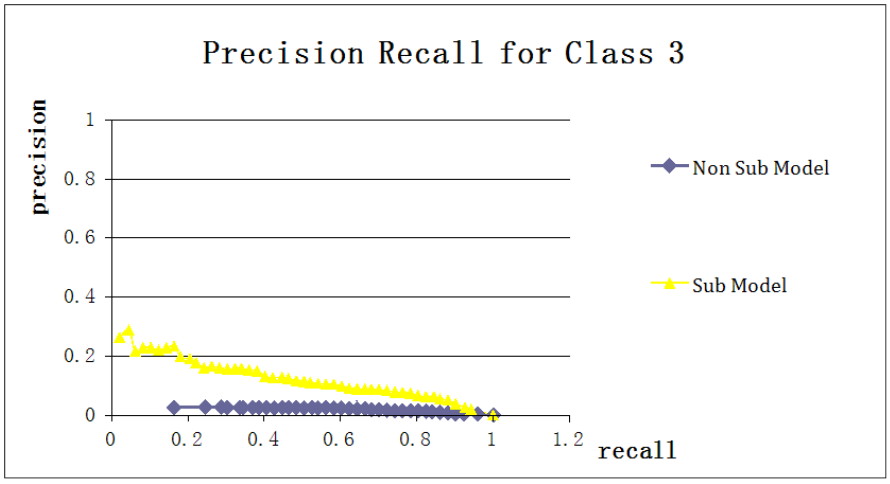

Figure 4.1.3

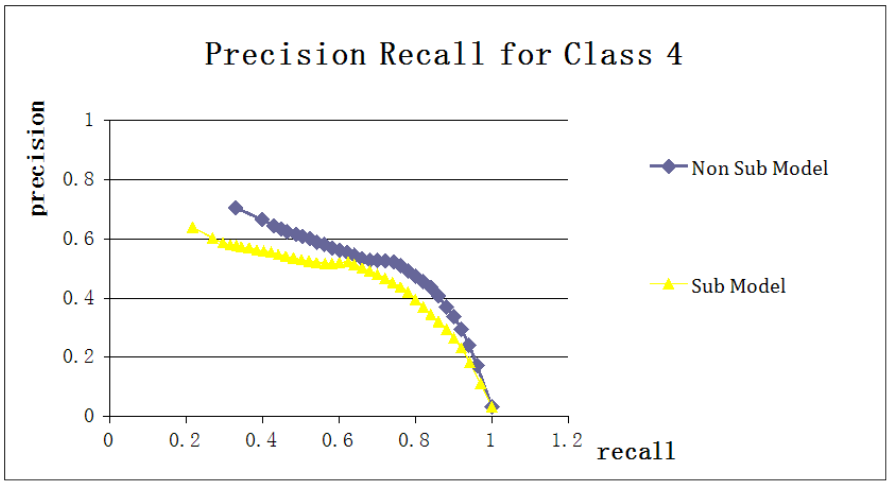

Figure 4.1.4

Training with sub-model have better performance in prediction of protein pairs in small protein complexes.

To verify the different characteristics for the features that we choose, different sets of features are then chosen each time. In total we have 11 sets of features, just now the two runs of learning involves all the 12 features, and the other 10 sets of features are:

ALL_PPI scores (PPI, PPI_CD, PPI_DEG, and PPI_NBC),

ALL_STRING scores (STRING, STRING_CD, STRING_DEG, STRING_NBC),

ALL_PUBMED scores (PUBMED, PUBMED_CD, PUBMED_DEG, PUBMED_NBC),

PPI_STRING_PUBMED,

ALL_CD scores (PPI_CD, STRING_CD, PUBMED_CD),

ALL_DEG scores (PPI_DEG, STRING_DEG, PUBMED_DEG),

ALL_NBC (PPI_NBC, STRING_NBC, PUBMED_NBC),

PPI,

STRING,

PUBMED.

Below are the Precision-Recall graphs for all the predicted sub classes with both non-sub-model and sub-model approaches in terms of different feature sets.

For Class 1

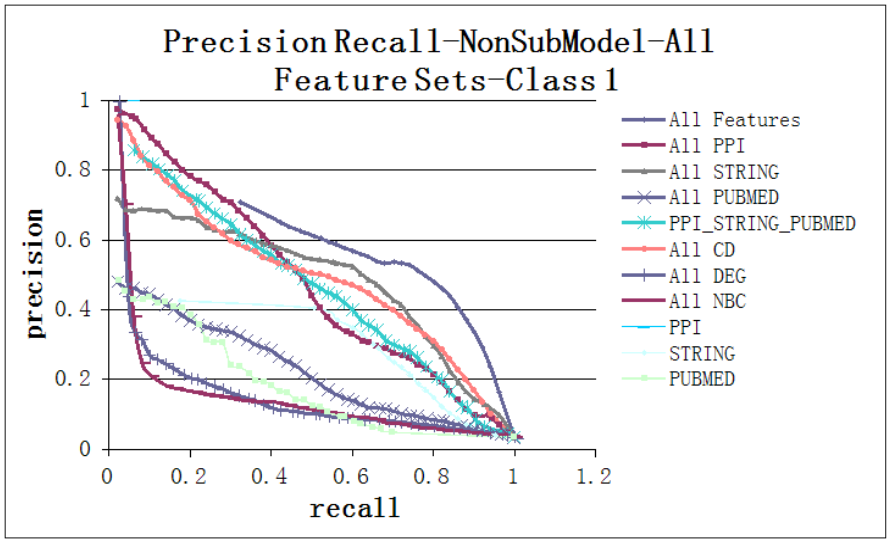

Figure 4.2.1

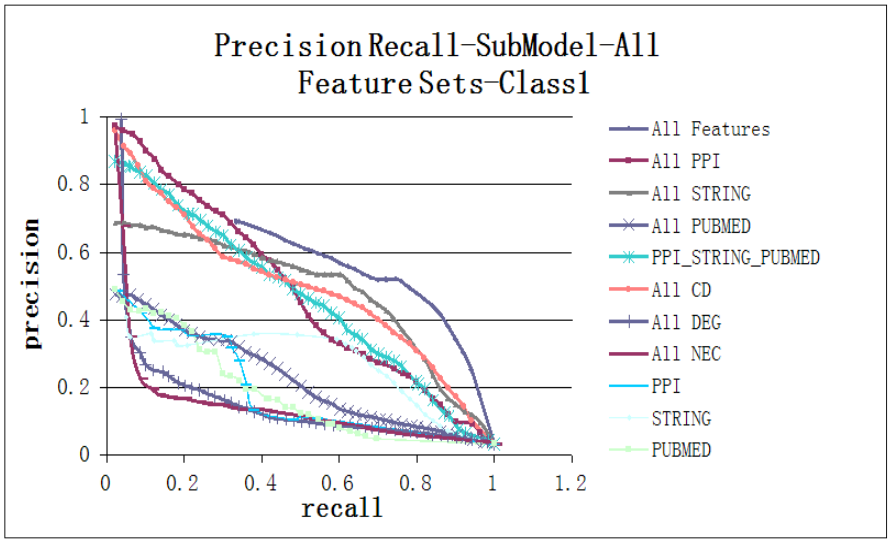

Figure 4.2.2

For Class 2

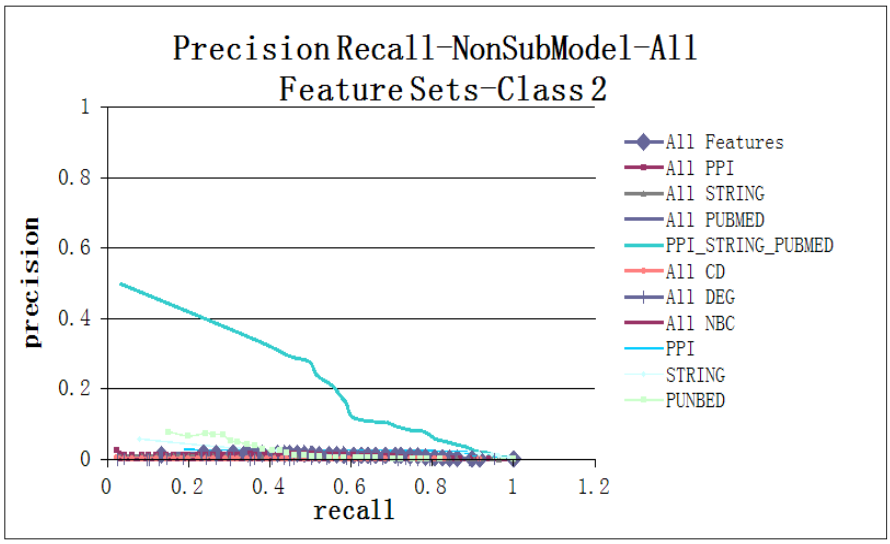

Figure 4.2.3

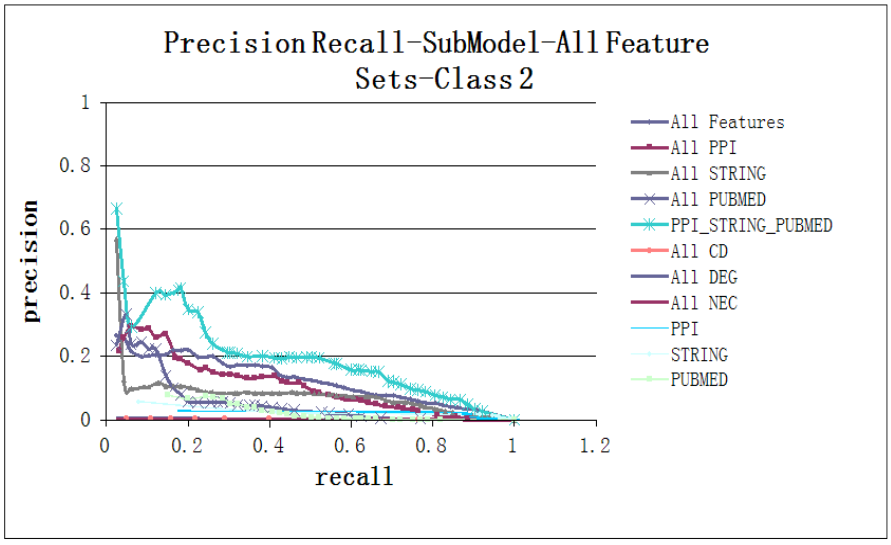

Figure 4.2.4

For Class 3

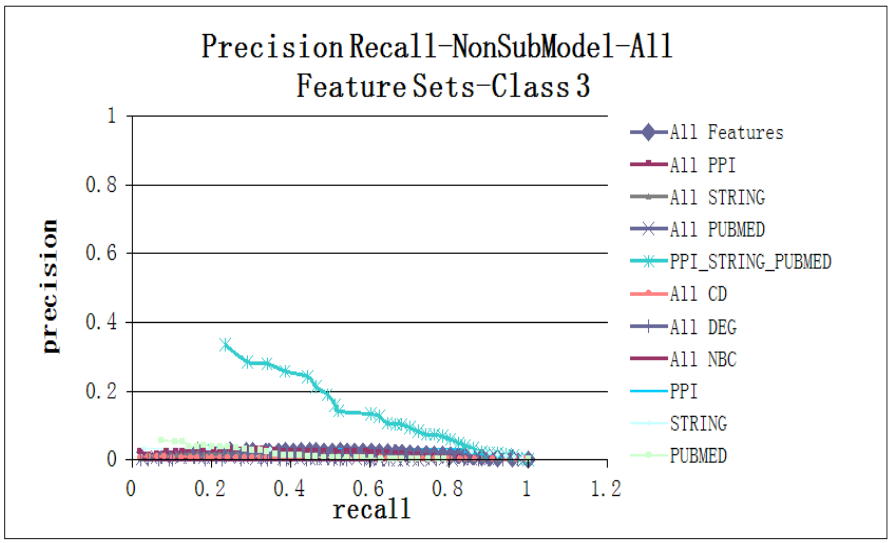

Figure 4.2.5

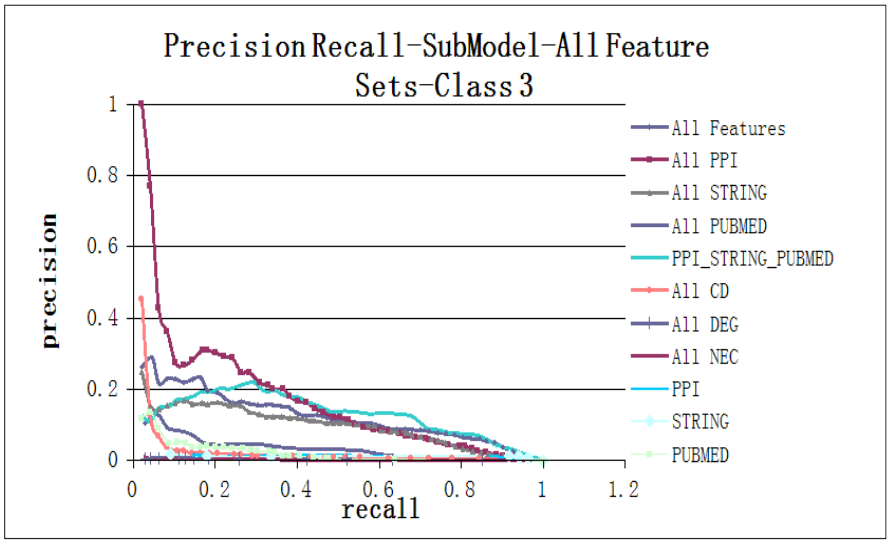

Figure 4.2.6

non-sub model sub-model

For Class 4

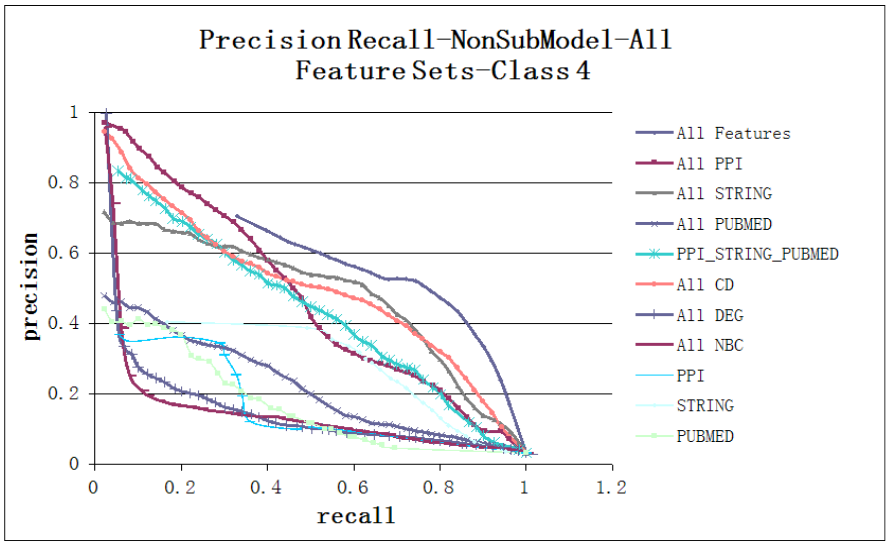

Figure 4.2.7

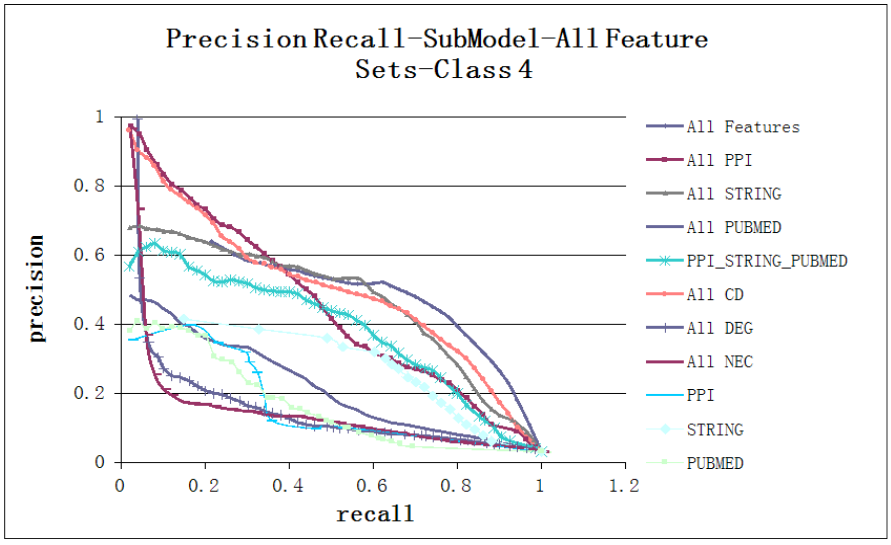

Figure 4.2.8

Comparing figures of non-sub-model and sub-model, class 2 and class 3 model shows much higher precision scores in sub-model approach. Class 1 and class 4 model shows similar performance between two approaches. This validate our second hypothesis that learning separate sub model for small complexes will improve performance, compared to single model for all complexes.

Comparing different class models, in sub-model approach, the performances for ALL_Features are all very significant in different class models, and PPI_STRING_PUBMED score has a great performance in class 2 and class 3 prediction as with the same recall value, precision scores are much higher than other feature sets. Same as ALL_PPI scores, it has a better performance in predicting class 2 and class 3 model than in class 4 model. On the other hand, however, all precision scores for ALL_CD in class 2 and class 3 are very low, so as ALL_DEG scores. But, ALL_CD have a much better performance in class 4 model prediction. We conclude CD as a meaningful characteristic for large protein complex prediction. This also verify the hypothesis that larger complexes have different topological characteristics compared to smaller complexes. All graphs show comparatively good performance with all the features as the input training data.

IMPLEMENTATION DETAILS

Firstly, we write a Perl script, combining the input three data source and CYC file, to calculate and obtain different topological feature scores. PPI, STRING and PUBMED scores are got directly from the database. DEG and NBC scores are calculated using functions discussed above. CD scores are calculated using the Iterative Adjust CD program[11].After generating the output arff file, first we look into the CYC file again and calculate the total number of protein complexes and the protein pairs inside different complexes, and obtain the Figure AAAAA and BBBB above. Then we use WEKA to perform Naïve-Bayes supervised learning on the output file and obtain the data for all the protein pairs with their classified class, prediction scores, and #actual, #predicted score. For non-sub-model output data, reference to CYC file and find out whether they are in class 2, class 3 or class 4 and above complexes and classify them accordingly. Then we can use Perl to calculate the precision and recall score with #actual #predicted score and plot the precision recall graph for each sub classes. For sub-model approach, as COCOMP_SIZE_CLASS is the class label, we use Perl to calculate precision recall for each class without referring to CYC file. And then we can plot precision recall graph for sub-model approach.

CONCLUSION

In this paper, we studied the prediction of protein complexes, especially small protein complexes (of size 2 and 3).

First, we investigated why SWC, a supervised approach with data integration, performs well on large complexes but poorly on smaller complexes. We integrated three data sources, PPI, functional associations (from STRING), and literature co-occurrence, and extracted three additional topological features from each data source: DEG (degree), CD (shared neighbors), and NBC (neighborhood connectivity). We found that there are significant differences in topological characteristics between edges belonging to small complexes, and edges belonging to large complexes. Furthermore, a vast majority of co-complex edges during training come from the large complexes. Thus, SWC learns the characteristics of large complexes and scores such edges accurately, while it performs poorly on edges from smaller complexes.

Then, we performed Naïve-Bayes supervised learning of separate sub-models for complexes of size 2, 3, and 4 and above, and showed that this outperformed learning a single model of all complex sizes (the SWC approach). The separate sub-models approach learned to score edges from small complexes much more accurately.

In future work, we can investigate using such a sub-models approach to give separate scores to each edge, corresponding to probability of being in a small complex, and probability of being in a large complex, and using clustering algorithms on these different scores to find clusters. Hopefully this will improve the performance of the detection of small complexes.

REFERENCE

[1] Adamcsek, Palla, Farkas, CFinder, 2009, locating cliques and overlapping modules in biological networks, BIOINFORMATICSAPPLICATIONS NOTE, Vol. 22 no. 8 2006, pages 1021?023

[2] Andreopoulos, B., et al. 2007. Clustering by common friends finds locally significant proteins mediating modules. Bioinformatics 23, pages 1124?131.

[3] Chern Han Yong, Guimei Liu, Limsoon Wong,Hon Nian Chua, 2012. Supervised Maximum-Likelihood Weighting of Composite Protein Networks for Complex Prediction. BMC Systems Biology, 2012, 6(Suppl 2):S13.

[4] Guimei Liu, Chern Han Yong, Hon Nian Chua, Limsoon Wong, 2011. Decomposing PPI Networks for Complex Discovery. Proteome Science, 2011, 9(Suppl 1):S15.

[5] Gavin, A.C., et al. 2006. Proteome survey reveals modularity of the yeast cell machinery. Nature 440, 631?36.

[6] Gavin, Bosche and Krause, 2002, Functional organization of the yeast proteome by systematic analysis of protein complexes, Nature, VOL 415

[7] Hand DJ, Yu K, 2001. Idiot's Bayes not so stupid after all? International Statistical Review, 2001, 69(3):385-398.

[8] King, Przulj, Jurisica, 2004, Protein complex prediction via cost-based clustering, BIOINFORMATICS, Vol. 20 no. 17 2004, pages 3013?020

[9] Leung, Xiang and Yiu, 2009, Predicting protein complexes from PPI data - a core-attachment approach, JOURNAL OF COMPUTATIONAL BIOLOGY, Vol 16

[10] Li, Chen, Wang, 2008, Modifying the DPClus algorithm for identifying protein complexes based on new topological structures, *BMC Bioinformatics* 2008, **9**:398

[11] Liu, Wong, Chua, 2009 Complex discovery from weighted PPI networks, BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 15 2009, pages 1891?897

[12] Yanjun Qi, Fernanda Balem, Christos Faloutsos, Judith Klein-Seetharaman and Ziv Bar-Joseph1 Protein complex identification by supervised graph local clustering, BIOINFORMATICS Vol. 24 ISMB 2008, pages i250–i258

[13] Yanjun Qi, Ziv Bar Joseph, and Judith Klein Seetharaman, 2006, Evaluation of Different Biological Data and Computational Classification Methods for Use in Protein Interaction Prediction, NIH Public Access Author Manuscript

[14] Qiu J, Noble WS, Predicting Co-Complexed Protein Pairs from Heterogeneous Data. PLoS Comput Biol 4(4): e1000054.

[15] Srihari et al.: MCL-CAw: a refinement of MCL for detecting yeast complexes from weighted PPI networks by incorporating core-attachment structure. BMC Bioinformatics 2010 11:504.

[16] Tatsuke, D., Maruyama, O., Sampling strategy for protein complex prediction using cluster size frequency, Gene 2012.

[17] Wang, Kakaradov, Collins, 2009, A complex-based reconstruction of the S cerevisiae interactome, Molecular & Cellular Proteomics 8.6, pages 1362-1381

[18] Wu, Li, Kwoh and Ng, 2009, A core-attachment based method to detect protein complexes in PPI networks, BMC Bioinformatics 2009, 10:169

[19] http://string-db.org

[20] http://en.wikipedia.org/wiki/PubMed

[21] http://en.wikipedia.org/wiki/MEDLINE

[22] http://wodaklab.org/cyc2008/complex/show/21

[23] http://wodaklab.org/cyc2008/complex/show/153

[24] http://en.wikipedia.org/wiki/Naive_Bayes_classifier

[25] Zhang, Harry. "The Optimality of Naive Bayes". FLAIRS2004 conference http://www.cs.unb.ca/profs/hzhang/publications/FLAIRS04ZhangH.pdf.