Integration of heterogeneous data sources for identification of disease genes using computational techniques

Li, Yongjin

2011

Li, Y. J. (2011). Integration of heterogeneous data sources for identification of disease genes using computational techniques. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/43993>

<https://doi.org/10.32657/10356/43993>

**INTEGRATION OF HETEROGENEOUS DATA SOURCES
FOR IDENTIFICATION OF DISEASE GENES USING
COMPUTATIONAL TECHNIQUES**

**LI YONGJIN
SCHOOL OF COMPUTER ENGINEERING
2011**

**INTEGRATION OF HETEROGENEOUS DATA SOURCES
FOR IDENTIFICATION OF DISEASE GENES USING
COMPUTATIONAL TECHNIQUES**

LI YONGJIN

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2011

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my supervisor, Assistant Professor Jagdish Chandra Patra, for his invaluable advice and guidance throughout my research. I have learned a lot from him, ranging from academic knowledge to research experiences. Besides, he provided so much freedom for me to explore and discover. Without his constant support and help, I could not have finished this thesis.

I would like to thank all my friends and colleagues at BioInformatics Research Center of NTU for their friendship and creating a very pleasant research environment. Special thanks go to Assistant Professor Li Jinming, Mr. Piyushkumar Arjunlal Mundra, Mr. Zheng Bo, Mr. Liu Song and Mr. Liu Qian, for their great and honest discussions, comments and help all along the way. I would also like to thank technicians, Mr. Tan Sing Yau and Mr. Kesavan Asaithambi, for their kind assistance.

Moreover, I would enthusiastically thank my family and my beloved Ting, for their love and support.

Last but not least, many thanks to School of Computer Engineering, Nanyang Technological University for awarding the research scholarship and providing first-rate research facilities.

Contents

Acknowledgments	i
Abstract	viii
Important Abbreviations and Symbols Used	xi
List of Figures	xiv
List of Tables	xviii
1 Introduction	1
1.1 Background	1
1.1.1 Basics in Human Molecular Genetics	1
1.1.2 Human Inheritance Disease	4
1.1.3 Cancer and Cancer Genes	4
1.1.4 Wet-lab Experiments for Disease Gene Identification	5
1.2 Motivations	7
1.3 Major Contributions	10
1.4 Organization of the Thesis	11
2 Literature Survey	13
2.1 Prioritization of Candidate Genes Based on Single Data Source	13

2.1.1	Sequence Based Methods	14
2.1.2	Gene Expression Based Methods	14
2.1.3	Ontology Based Methods	15
2.1.4	PPI Network Based Methods	16
2.2	Integration Methodologies on Candidate Genes Prioritization	18
2.2.1	SUSPECTS	18
2.2.2	CAESAR	19
2.2.3	Prioritizer	19
2.2.4	TOM	20
2.2.5	Candidate Protein Complex Method	20
2.2.6	CIPHER	21
2.2.7	ENDEAVOUR	23
2.2.8	Multiple Kernels Learning	24
2.3	Summary	24
3	Data Collection and Preprocessing	26
3.1	Protein-Protein Interaction Network	26
3.1.1	Protein-Protein Interaction	26
3.1.2	The PPI Database — HPRD	28
3.1.3	Topology of the PPI Network	30
3.1.4	Diffusion Kernel	31
3.2	Gene Ontology and Gene Functional Similarity	32
3.2.1	Gene Ontology	32

3.2.2	Gene Annotation in Gene Ontology	34
3.2.3	Gene Functional Similarity	35
3.2.4	Gene Networks Obtained from GO	36
3.3	Disease Related Data	37
3.3.1	Cancer Genes	37
3.3.2	The OMIM Database	38
3.3.3	Disease and Disease Phenotype	39
3.3.4	The Phenotype Network	40
3.3.5	Disease Genes and Phenotype–Gene Network	42
3.4	List of Websites	43
3.4.1	Websites for the Databases	43
3.4.2	Websites for the Softwares	45
4	Prediction of Cancer Genes from PPI Network	46
4.1	Introduction	47
4.2	Methods	48
4.2.1	Datasets and Topological Features	48
4.2.2	Classification Algorithms	52
4.2.3	Data Normalization	53
4.2.4	Cross Validation	53
4.3	Experiments and Results	55
4.3.1	Significance of the Five Topological Features	55
4.3.2	Correlation Between the Five Topological Features	56

4.3.3	Results of Classification with Different Combination of Features	57
4.3.4	The Best Combination of Features	58
4.4	Summary	60
5	Rank Aggregation-based Data Integration Models for Candidate Gene	
	Prioritization	62
5.1	Introduction	63
5.2	Methods	66
5.2.1	Ranking Based on 1CSVM	66
5.2.2	Random Walk with Restart (RWR)	69
5.2.3	Multiple Kernels Learning (MKL)	73
5.2.4	N-dimensional Order Statistics (NDOS)	75
5.2.5	Discounted Rating System (DRS)	77
5.2.6	Leave-One-Out Cross Validation (LOO-CV)	82
5.3	Experiments and Results	86
5.3.1	Datasets	86
5.3.2	Comparison Between MKL and the RABI Models	86
5.3.3	Comparison Between DRS-based Integration and NDOS-based In- tegration	92
5.3.4	Stable Performance of the RWR+DRS Model	96
5.3.5	Computational Efficiency of NDOS and DRS Algorithms	97
5.4	Summary	99

6	Network-based Data Integration for Candidate Gene Prioritization	101
6.1	Introduction	102
6.1.1	Random Walk with Restart on Multiple networks (RWRM) . . .	103
6.1.2	Random Walk with Restart on the Heterogeneous network (RWRH)	103
6.1.3	Complex Heterogeneous Network (CHN)	106
6.2	Methods	106
6.2.1	The RWRM Model	106
6.2.2	The RWRH Model	110
6.2.3	The CHN Model	115
6.3	Experiments and Results	116
6.3.1	Performance of the RWRM Model on 36 Diseases	117
6.3.2	Comparison Between RWRH and CIPHER	118
6.3.3	Effect of Parameter Variations in RWRH	121
6.3.4	Comparison between RWRH and RWR on Gene Network Only . .	123
6.3.5	Discovering Disease Genes for Alzheimer’s Disease Using the RWRH Model	124
6.3.6	Discovering Hidden Disease-Disease Associations Using the RWRH Model	127
6.3.7	Performance of the CHN Model	130
6.3.8	Discovering Disease Genes of Insulin-Dependent Diabetes Mellitus Using the CHN Model	132
6.4	Summary	134

7	Conclusions and Future Directions	136
7.1	Conclusions and Discussion	136
7.2	Future Directions	140
7.2.1	Cancer Gene Identification	140
7.2.2	Integration of More Data Sources	140
7.2.3	Assignment of Weights to Different Data Sources	141
7.2.4	Prioritization of SNPs from GWAS Data	141
	References	142
	Author's Publications	158
A	Detailed Results on 36 Diseases	159
B	Softwares Used in This Work	162
B.1	MimMiner	162
B.2	BioMart	164

Abstract

Genes related to causing some disease are called disease-causing genes or disease genes. In wet-lab experiments, disease genes are identified by mutation analysis, which is expensive and labor extensive. In this thesis, we propose novel computational techniques to predict disease genes.

In the first part of this thesis, we proposed five novel topological features obtained from the Protein-Protein Interaction (PPI) network. We applied Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) networks to predict new cancer genes using these features. We found that SVM performed slightly better than MLP. We also found that the feature, named 2N-index, is the most discriminative feature between cancer genes and other genes.

With the availability of various data sources related to genes and disease phenotype, accurate prediction of disease genes is possible by integrating the information available from multiple data sources. We propose several novel computational models to integrate multiple data sources for the identification of disease genes. These models are proposed to prioritize set of candidate disease genes, based on their functional similarity to known disease genes. The top ranked candidate gene is most likely to be the real disease gene.

In the second part of this thesis, we focus on rank aggregation based integration

(RABI) models. We integrated four genomic data sources, i.e., protein-protein interaction data, and three ontologies, Biological Process (BP), Cellular Component (CC) and Molecular Function (MF). We first rank candidate genes based on each of the four data sources, using 1CSVM (one-class SVM) and RWR (Random Walk with Restart) algorithm. Then we use the NDOS (N-Dimensional Order Statistics) algorithm to combine the four rank lists into one list, which is used as the final prioritization of the set of candidate genes. Computational models were evaluated in terms of AUC value, using 36 diseases as the benchmark data. The proposed RWR+NDOS and 1CSVM models achieved AUC values of 85.1% and 84.4%, better than the Multiple Kernels Learning (MKL) method which gave an AUC of 79.3%. Next, we proposed another efficient rank aggregation algorithm named Discounted Rating System (DRS). The proposed RWR+DRS and 1CSVM+DRS models achieved AUC values of 87.7% and 85.4%, higher than the NDOS-based models.

In the third part of this thesis, we focus on network based integration (NBI) models. We proposed an improved RWR algorithm named RWR on Multiple networks (RWRM), to integrate the four genomic data sources (PPI, BP, CC and MF). We obtained the AUC value of 89.4% on the same benchmark data as in the second part. In case of disease phenotype without any known disease genes, the candidate genes are prioritized based on their similarity to disease genes associated with similar phenotypes. We proposed a novel heterogeneous network model, named RWR for Heterogeneous network (RWRH), to integrate the PPI network and phenotype information obtained from OMIM. We have shown using the leave-one-out cross validation (LOO-CV) that, the RWRH algorithm correctly identified 814 of 1,428 phenotype-gene relationships (PGRs), the previous work identified

only 709 of them [1]. Finally, we proposed a Complex Heterogeneous Network (CHN) model, in which four genomic data sources and the phenotype data were integrated. Both RWRM and RWRH algorithms are used in the CHN model to prioritize candidate genes. We used the latest OMIM data as benchmark data, containing 3,871 PGRs. We successfully identified 2,105 of them, whereas the DRS-based and NDOS-based integration methods, respectively, could identify 2,008 and 2,048 number of relationships. Thus, the proposed CHN model identified more PGRs than both DRS-based and NDOS-based integration methods, which showed better performance of the proposed CHN model.

Important Abbreviations and Symbols Used

Abbreviations

1CSVM	One-Class Support Vector Machine
AUC	Area Under the Curve
ALI	Artificial Linkage Interval
BP	Biological Process
CC	Cellular Component
CHN	Complex Heterogeneous Network
cRL	combined Rank List
DNA	Deoxyribonucleic Acid
DRS	Discounted Rating System
GO	Gene Ontology
HPRD	Human Protein Reference Database
IC	Information Content
IDDM	Insulin-Dependent Diabetes Mellitus
LOO-CV	Leave-One-Out Cross Validation
MeSH	Medical Subject Headings vocabulary
MF	Molecular Function

MKL	Multiple Kernel Learning
MLP	Multi-Layer Perceptrons
NBI	Network-based Integration
NDOS	N-Dimensional Order Statistics
OMIM	Online Mendelian Inheritance in Man
PGR	Phenotype–Gene Relationship
PPI	Protein-Protein Interaction
RABI	Rank Aggregation Based Integration
RL	Rank List
ROC	Receiver Operating Characteristic
RWR	Random Walk with Restart
RWRH	Random Walk with Restart on Heterogeneous network
RWRM	Random Walk with Restart on Multiple networks
SDP	Semi-Definite Programming
SVM	Support Vector Machine
WSR	Wilcoxon signed-rank

Symbols

g_i	Gene i
X	Feature matrix with the i -th row corresponding to feature vector of g_i
$\mathbf{x}_i$	Feature vector of g_i
N	The number of data sources
n	The number of candidate genes
μ_i	Weight for data source i , used in the MKL algorithm
lb	The lower bound of μ_i
$\mathcal{K}$	The kernel matrix
$\mathcal{K}_i$	The normalized kernel presentation of data source i
ρ	The margin between the hyper-plane and the origin
A	The adjacency matrix of the graph
M	The transition matrix in RWR algorithm
$\mathcal{N}_g$	The number of genes in the gene network
$\mathcal{N}_{dg}$	The number of disease genes
$\mathcal{N}_{Ph}$	The number of phenotypes
$\mathbf{p}_0$	The vector of initial probability in the RWR algorithm
$\mathbf{p}_s$	The output probability vector of RWR at step s
$\mathbf{p}_\infty$	The stable state probability vector for RWR
λ	The probability of coming back to source nodes at each step in RWR
RL_j	Rank list of candidate genes, based on data source j
$RL_{i,j}$	Rank position for candidate gene i based on data source j
$RR_{i,j}$	Rank Ratio for candidate gene i based on data source j

$rating_{i,j}$	Rating for candidate gene i based on data source j
$dr_{i,j}$	Discounted rating score for candidate gene i based on data source j
$(S_{dr})_i$	The combined score of gene i calculated using DRS algorithm
η	The parameter used in RWRH
γ	The jumping probability used in RWRH
$G(V, E)$	A graph with a set of nodes, V and a set of edges, E

List of Figures

1.1	Illustration of relationships among cell, chromosome, gene and DNA. This figure is taken from the website of NIGMS [2].	2
1.2	The procedure of identifying disease genes.	6
2.1	Candidate genes prioritization using random walk with restart (RWR) algorithm, taken from [24].	17
2.2	The procedures of ENDEAVOUR, a rank aggregation based integration method. This figure is taken from [33].	22
3.1	An illustration of Protein-Protein Interaction.	27
3.2	Section of Protein Interaction Data obtained from HPRD [55].	28
3.3	The PPI Network taken from [68].	31
3.4	Part of BP Ontology. This figure is taken from [73].	33
3.5	Illustration of gene annotation matrix.	34
3.6	The procedure of generating KNN graph with $K = 2$	36
3.7	Snapshot of an MIM record. The MIM ID of this record is 605526. It is a phenotype of Alzheimer's Diseases (AD6).	39
3.8	Data processing in MimMiner.	41

4.1	A schematic diagram of cancer gene prediction using the PPI network. .	47
4.2	An illustration of cancer genes in protein-protein interaction network, where dark shaded nodes denote cancer genes.	49
4.3	An example of PPI network showing 2-neighborhood of node 7.	50
4.4	The performance of classifiers with varying number of features.	58
5.1	Two stages of the RABI model '1CSVM+NDOS'. In the first stage, we use 1CSVM ranking algorithms to prioritize candidate genes. In the second stage, NDOS is used to combine multiple rank lists into one list.	64
5.2	The linear function learned by 1CSVM to separate disease genes from other genes. Filled circles denote known disease genes and other circles denote candidate genes.	67
5.3	Illustration of RWR algorithm. The random walker is now at node g_5 , which has three neighbors g_3 , g_6 or g_9 . Therefore, he may choose to move to any of the neighbors of g_5 with a probability of $\frac{1}{3}$	70
5.4	Candidate gene prioritization using random walk with restart (RWR) al- gorithm, taken from [24].	71
5.5	The schematic diagram of the NDOS algorithm.	75
5.6	The schematic diagram of the DRS algorithm.	78
5.7	Illustration of relationships between rank, rating and discounted rating. Here the rank ranges from 1 to 100 and the range of rating is 5.	79
5.8	An illustration of four steps in Discounted Rating System.	81
5.9	Procedures of leave one out cross validation.	83

5.10	Calculation of the AUC value.	84
5.11	ROC curve corresponding to MKL and NDOS-based Integration Models.	88
5.12	Comparison between 1CSVM+NDOS model and the MKL algorithm in terms of AUC values.	89
5.13	Comparison between RWR+NDOS model and the MKL algorithm in terms of AUC values.	90
5.14	Comparison between 1CSVM+NDOS model and the RWR+NDOS model in terms of AUC values.	91
5.15	ROC curves of DRS-based integration and NDOS-based integration. . . .	93
5.16	Comparison between RWR+DRS model and the RWR+NDOS model in terms of AUC values.	94
5.17	Comparison between 1CSVM+DRS model and the 1CSVM+NDOS model in terms of AUC values.	95
6.1	The illustration of graph merging. Three kinds of links denote three kinds of relationships obtained from different data sources.	107
6.2	Illustration of the heterogeneous network. Nodes connecting gene network and phenotype network are called bridging nodes. Other nodes are called internal nodes.	111
6.3	The Complex Heterogenous Network (CHN) is constructed by connecting the phenotype network and the merged gene network. Nodes connecting two kinds of networks are named bridging nodes. Other nodes are named internal nodes.	115

6.4	ROC curves of RWRM and four RABl models, on the benchmark data set of 36 diseases.	117
6.5	ROC curves of RWR and RWRH.	124

List of Tables

3.1	Datasets used in this thesis ^(a)	29
3.2	Prefix and category of MIM entries.	38
4.1	The statistical significance of five topological features obtained from the PPI network.	56
4.2	Correlation between the five topological features.	57
4.3	Performance of classifiers with different combination of five features. . . .	59
5.1	Performance of the MKL algorithm on 36 diseases in terms of overall AUC values.	87
5.2	Performance of five different integration models on 36 diseases in terms of overall AUC values.	94
5.3	Stable performance of RWR with variation of γ	97
5.4	The range of ratings and AUC values.	97
5.5	Results on simulated new data sources.	98
6.1	Performance of RWRH and CIPHER.	119
6.2	Effect of λ in RWRH.	121
6.3	Effect of η in RWRH.	122

6.4	Top five ranked candidate genes for each phenotype of Alzheimer’s disease.	125
6.5	Predicted disease associations.	129
6.6	Number of successful predictions in LOO-CV.	131
6.7	Top five ranked candidate genes for each phenotype of IDDM.	133
A.1	The performance of MKL with different lb values, evaluated on each of the 36 diseases, measured by AUC values (%).	160
A.2	The comparison of several integration methods used in this thesis, on each of 36 diseases, measured by AUC values (%).	161

Chapter 1

Introduction

This chapter begins with a brief introduction of the problem of disease gene identification, followed by the motivation to carry out this research, with an outline of achievements.

1.1 Background

In this section, we start with some basics in human molecular genetics, followed by the description of human inheritance diseases and cancer. Then we introduce the procedure of disease gene identification, which results in a set of candidate genes related to the disease. Finally, we introduce the problem of candidate gene prioritization.

1.1.1 Basics in Human Molecular Genetics

Human beings, *homo sapiens*, are made up of cells. The somatic cell of human contains 23 pairs of different chromosomes. One member of each pair is derived from the individual's father, and the other member is derived from the mother. As shown in Figure 1.1, chromosomes reside in the nucleus of the cell with four-arms structure. The shorter arms are called p arms and the longer arms are called q arms. A chromosome contains a long DNA (Deoxyribonucleic acid) sequence and the proteins, named histone, that help to

Figure 1.1: Illustration of relationships among cell, chromosome, gene and DNA. This figure is taken from the website of NIGMS [2].

pack the DNA.

The most important constituents of DNA are four nucleotide bases: adenine (A), thymine (T), cytosine (C) and guanine (G). According to Watson-Crick Rules, the nucleotide base A specifically binds to T, and C specifically binds to G. The base pairs are connected to form the double helix shape. Each human cell contains approximately three billion nucleotide base pairs.

Gene is the basic unit of inheritance. It is a DNA sequence contained in the chro-

mosome. Usually a gene contains thousands of base pairs. A locus refers to the specific location of a gene or DNA sequence on a chromosome. The chromosomal locus of a gene might be written "6p21.3", which means the gene is located at the short arm of chromosome 6, band 2, section 1, sub-band 3. The bands are visible under a microscope when the chromosome is suitably stained. Each of the bands is numbered, beginning with 1 for the band nearest the centromere. Sub-bands and sub-sub-bands are visible at higher resolution. A range of locations is specified in a similar way. For example, the locus of the gene OCA1 may be written "11q1.4-q2.1", meaning it is on the long arm of chromosome 11, stretches from band 1 sub-band 4 to band 2 sub-band 1. The figure is taken from [2].

There might be different DNA sequences at the same physical locus, which are called alleles. In any particular diploid organism, with two copies of each chromosome, the genotype for each gene comprises the pair of alleles present at that locus, which are the same in homozygotes and different in heterozygotes. For example, at the gene locus for ABO blood type proteins in humans, there are three alleles, IA, IB, and IO, that determines compatibility of blood transfusions. Any individual has one of six possible genotypes (AA, AO, BB, BO, AB, and OO) that produce one of four possible phenotypes: "A" (produced by AA homozygous and AO heterozygous genotypes), "B" (produced by BB homozygous and BO heterozygous genotypes), "AB" heterozygotes, and "O" homozygotes [3].

1.1.2 Human Inheritance Disease

Human inherited (genetic) disease is a disease or disorder that is inherited genetically. It is caused by abnormalities in genes or chromosomes. Some genetic diseases occur as a direct consequence of single gene being defective. Such disorders are inherited (passed on from parents to children) in a simple pattern according to Mendel's Laws. As such, these disorders are often referred to as Mendelian disorders [4]. The Online Mendelian Inheritance of Man (OMIM) database [5] currently lists over 5000 disorders that have been definitely traced to single-gene defects. Other disorders are caused by a combination of environmental factors and mutations in multiple genes. These kinds of diseases are usually called complex diseases or polygenic diseases. For example, different genes that influence breast cancer susceptibility have been found on chromosomes 6, 11, 13, 14, 15, 17, and 22 [6]. Its more complicated nature makes these diseases much more difficult to analyze than single-gene or chromosomal disorders [7–9]. Many common chronic diseases with adult onset show familial aggregation that usually does not follow Mendelian family patterns but appears to be caused by a usually unknown number of multiple genes, usually interacting with various environmental factors [10]. Such diseases include coronary heart disease, hypertension, diabetes, obesity, various cancers, Alzheimer's disease, Parkinson's disease and others.

1.1.3 Cancer and Cancer Genes

Cancer is a group of complex diseases in which abnormal cells divide without control and can invade other tissues [6]. It is a leading cause of death worldwide. According to the

American Cancer Society, 7.6 million people died from cancer in the world during 2005, which is about 13% of all deaths [11].

Decades of molecular genetic research have shown that cancer is a system failure, which is caused by multiple gene mutations. A central aim of cancer research has been to identify genes that are causally implicated in oncogenesis ('cancer genes'). Traditionally, biologists use mutation analysis to find cancer genes. Up to now, roughly 1% of all predicted human genes have been found to contribute to the development of cancer [12]. However, it has been suggested that 5-10% or more of the 25,000 putative genes encoded in the human genome probably contribute to oncogenesis [13]. Therefore, it is plausible (although unproven), that many more cancer genes remain to be identified. The increasingly identified cancer genes and availability of genomic and proteomics data provide us opportunities to identify cancer genes by computational methods.

1.1.4 Wet-lab Experiments for Disease Gene Identification

The human genome is very large and contains around 25 thousand genes [14]. Therefore, finding the particular gene or genes responsible for any given human disease has always been a challenging task, quite literally like finding a needle in a haystack. Traditionally, the disease genes are found by genetic linkage analysis and positional clone, followed by mutation analysis [9, 15, 16].

Figure 1.2 shows the procedures of identification of disease genes. Firstly, linkage analysis is performed to investigate a pedigree. In this approach, the aim is to find out the susceptible chromosome interval, which is the approximate location of the disease gene [17]. Then a process called 'positional cloning' is used to obtain a set of candidate

Figure 1.2: The procedure of identifying disease genes.

genes in this region [9], one or more of which might be the real disease genes. Finally, the disease gene is identified by performing mutation analysis on each of the candidate genes.

Since 1980, positional cloning techniques have been used to prioritize candidate genes and these techniques have successfully identified disease genes for a number of diseases, such as Duchenne Muscular Dystrophy, Huntington's and Cystic Fibrosis, etc. However the linkage disequilibrium (LD) analysis is not always successful. It heavily depends on two factors. One is the allelic associations between single markers and disease. The other is the degree of genetic heterogeneity underlying a disease sample [9]. Usually, hundreds of candidate genes locate in the resulting susceptible chromosome region. It is a challenging task to identify the real disease gene from hundreds of candidate genes, because both the positional cloning and the mutation analysis experiments are labor-

intensive, tedious and time consuming. Currently, in the OMIM database, thousands of disease loci (chromosome interval) have been associated with disease phenotypes [5]. Each of them contains hundreds of candidate genes that need to be prioritized for identification of real disease-causing genes. To address the challenge, it is preferable to prioritize candidate genes using computational techniques, which will greatly reduce the number of genes for wet-lab experimental analysis. The common strategy is to rank these candidate genes according to their functional similarity to known disease genes. In this thesis, this is called disease gene prioritization.

1.2 Motivations

With the advent of high-throughput technologies, huge amount of genomic data have been generated, therefore, there are many ways to define 'similarity' between genes. A number of methods have been proposed to prioritize candidate genes based on different kinds of genomic data. Several candidate gene prioritization systems rely on Gene Ontology annotations [18–20]. Some others predicted disease genes through sequence-based features, because they found that human genes involved in hereditary disease have some distinct sequence properties in common which render them more susceptible to mutations causing genetic disorders [21, 22]. Tiffin et al. [23] prioritized candidate disease genes according to their expression profiles. Physically interacting proteins tend to be involved in the same cellular process and preferentially interacted with other disease genes significantly, so mutations in these genes may lead to similar disease phenotypes. Therefore, some research groups predict disease genes using PPI (Protein-Protein Interaction) networks [24–26].

Existing methodologies, mining biological and functional information for candidate genes, are based on single data source, e.g., gene sequence or PPI. Most of these data sources are noisy and incomplete. For example, different experiments for PPI detection obtained different results with poor agreement [27]. Different data sources provide different kinds of information. Therefore they are complementary to each other. For example, the PPI data describes which two proteins tend to bind together to perform some function, while the Biological Process (BP) ontology describes what kind of biological processes a gene is involved in [28]. In addition, these data sources are complementary to each other. How to effectively integrate heterogeneous data sources to improve the prediction of disease genes is a major challenge. Several integration algorithms have been proposed to prioritize candidate genes [22, 29–32], however most of them access no more than two data sources. Recently Aerts et al. [33] proposed a tool named ENDEAVOUR to prioritize candidate genes through integrating multiple data sources. De Bie et al. [34] proposed a Multiple Kernels Learning (MKL) algorithm integrating the same data sources as [33] and achieved improved performance.

Aerts et al. [33] use a rank aggregation strategy in ENDEAVOUR, in two stages: the ranking process and the combination process. They integrated nine data sources to prioritize candidate genes. In the ranking process, candidate genes are ranked based on their similarity to known disease genes using the information from each of the data sources. As a result, several rank lists of candidate genes are obtained. In the combination process, these rank lists are combined into one list using N-dimensional order statistics (NDOS) [35]. In this thesis, we follow this rank aggregation framework and attempt to

improve it on both processes, intend to achieve both computational efficiency and better prediction performance.

The above mentioned methods prioritize candidate genes based on their similarity to known disease genes. But for some diseases no confirmed disease genes have yet been found. Therefore candidate genes for this kind of diseases cannot be prioritized using above mentioned methods. To solve this problem, two research groups integrate phenotype information to prioritize candidate disease genes [1, 36]. The assumption is that similar disease phenotypes are caused by the mutation of functionally similar genes [37]. Lage et al. [36] assign candidate gene to protein complexes and then rank the candidate complexes based on the phenotypic similarity using a Bayesian model. Wu et al. [1] employ a regression model to calculate the concordance score between the query phenotype and the candidate gene. Candidate genes are ranked based on this concordance score. For the disease phenotype without confirmed disease genes, these two integration algorithms [1, 36] can prioritize candidate genes using the information of similar disease phenotype. At the same time, integration of phenotype information improved the reliability of the prioritization compared with algorithms using PPI network only. However, both of these algorithms have limitations. Lage et al.'s algorithm [36] rely on the predefined protein complex. However the protein complex data is unreliable. Even if the protein complex data is reliable, it is not proper to prioritize candidate genes using all the subunits in the complex, because protein complexes are not stable *in vivo*, as most of them are assembled just-in-time [38]. Wu et al.'s method has two limitations [1]. One is that they did not utilize the phenotype similarity information obtained from the OMIM. The other is that they use the direct interaction and shortest path from PPI

network to calculate the similarity between genes, which have been found not as effective as the RWR (Random Walk with Restart) algorithm proposed by Köhler et al. [24]. In this thesis, we proposed a RWR-based heterogeneous network model and obtained improved performance.

1.3 Major Contributions

The major achievements of this research are listed below:

- A novel rank aggregation algorithm, named DRS (Discounted Rating System), was proposed to combine rank lists of candidate genes. The algorithm was inspired by DCG (Discounted Cumulated Gain) score [39], which is widely used to evaluate the ranking results of document retrieval. Different data sources might be used for prediction of different kinds of diseases. The real disease gene should be ranked at the top by at least one data source. In the proposed DRS algorithm, genes at the top of any rank list was given higher priority. If one candidate gene is ranked at the top by at least two or more data sources, it is most likely to be the real disease gene and will be selected by the proposed DRS algorithm. On most of the diseases in the benchmark data of 36 genetic diseases, DRS-based integration achieved higher AUC values than NDOS-based integration [35]. Furthermore, the DRS algorithm is computationally more efficient than NDOS algorithm. As a result, the rank aggregation based integration (RABI) model is improved.
- A novel network based integration model RWRM was proposed to integrate multiple genomic data sources. Each genomic data source is represented as a gene network,

we merge multiple gene networks into one network and extend the RWR algorithm on the merged gene network. On the same benchmark data of 36 genetic diseases, the RWRM algorithm works better than the RABI models (DRS-based and NDOS-based integration).

- Another novel integration algorithm, RWRH, was proposed to integrate the PPI network and phenotype data for the candidate gene prioritization. We constructed a heterogeneous network by connecting the PPI network and phenotype network using the gene–phenotype relationships. Based on the heterogeneous network, we proposed an RWRH algorithm, which is a variant of the RWR algorithm. We set source nodes as the target disease phenotype and known disease genes, and let the random walker walk on the heterogeneous network. The most visited gene node and phenotype node in the heterogeneous network are seen as the most relevant genes and phenotypes, respectively. The proposed RWRH algorithm effectively captures the complementarity between gene network and phenotype network. Results of leave one out cross validation showed that it performed better than gene network only RWR algorithm and the previous integration work CIPHER [1]. In addition, the RWRH algorithm can be used to find the association between diseases. We found 18 associations that cannot be found from gene network or phenotype network alone. Most of these disease associations were supported by various type of evidences.

1.4 Organization of the Thesis

The thesis is organized so as to describe the research from basic background to specific topic details. It consists of 7 Chapters.

Chapter 1 begins with a brief introduction of the basics of human molecular genetics and the problem of candidate genes prioritization, followed by the motivation of my research, with an outline of my achieved work.

Chapter 2 gives a review on the framework involved in this research, including prioritization algorithms based on single data source and integration algorithms.

Chapter 3 describes the data used in this work and the corresponding preprocessing methods, including the OMIM database, PPI network, and Gene Ontology, and benchmark disease datasets.

Chapter 4 introduces five topological features derived from the PPI network for cancer gene identification.

Chapter 5 focus on rank aggregation based integration (RABI) models. We propose a number of RABI models, for candidate gene prioritization, including 1CSVM+NDOS, RWR+NDOS, 1CSVM+DRS and RWR+DRS. These models are compared with the kernel-based integration (MKL). Several algorithms are described in this chapter, including 1CSVM, RWR, MKL, NDOS and the proposed DRS. This chapter also includes the detailed description of the leave one out cross validation, which is also used in the subsequent chapters.

Chapter 6 proposes three network based integration (NBI) models: RWRM, RWRH and CHN. The predicted disease genes for Alzheimer's disease and Insulin-Dependent Diabetes Mellitus (IDDM) are shown in this chapter.

Finally, Chapter 7 concludes this thesis and gives possible directions to extend the research done.

Chapter 2

Literature Survey

Linkage analysis localizes the disease genes to a candidate region of the genome. Genes in this region are called candidate genes, one or more of which might be the real disease gene. Identification of the most probable of these candidate disease genes for further wet-lab experimental analysis is a significant challenge. The common strategy is to rank these candidate genes according to their functional similarity to a set of known disease genes, which is called candidate gene prioritization. A number of computational methods have been developed to address this problem. In this chapter, we provide a comprehensive survey on methods to discover disease genes based on individual data source and some integration algorithms on disease gene identification.

2.1 Prioritization of Candidate Genes Based on Single Data Source

In this section, we introduce disease gene discovery methods based on individual data source, including sequence based methods, gene expression based methods, ontology based methods and Protein-Protein Interaction (PPI) network based methods.

2.1.1 Sequence Based Methods

With the completeness of human genome sequence, there is an opportunity to investigate disease susceptibility loci on a large scale, in terms of the likely or known functions of the annotated genes present within them. Some research groups predicted disease genes through sequence-based features, because they found that human genes involved in hereditary disease have some distinct sequence properties in common which render them more susceptible to mutations causing genetic disorders [21, 22].

Lopez-Bigas et al. [21] investigated some features of protein sequence between disease genes and non-disease genes. They found proteins involved in hereditary diseases tend to be long, conserved, phylogenetically extended, and without close paralogues. Adie et al. [22] found more sequence-based features, which are significant differences between genetic disease genes and other genes, e.g. the number of CpG islands at the 5' end, the length of the 3' UTR and the distance to the nearest neighboring gene, etc. Based on these features, they created a software using the alternating decision tree algorithm. It is called PROSPECTR, which ranks genes in the order of likelihood of involvement in disease.

2.1.2 Gene Expression Based Methods

Gene expression measurements on a genome-scale, representing the transcriptome, have been accomplished through the technological advancement of microarrays. Co-expressed genes are likely to be involved in the same cellular process. Co-expression between genes is usually calculated based on the microarray data. However the microarray data are noisy, therefore co-expression does not strongly suggest a functional relation. Conserved

co-expression could be a much stronger criterion than single species co-expression to relate genes involved in similar disease phenotypes, because significant co-expression of two or more orthologous genes is very likely due to selective advantage. Ala et al. [40] showed that reliably disease-relevant relationships can be identified from massive microarray datasets by concentrating only on genes sharing similar expression profiles in both human and mouse. Combining evolutionarily distant species to calculate evolutionary co-expression will further increase the reliability. Oti et al. [41] use co-expression data from yeast (*S. cerevisiae*), nematode worm (*C. elegans*), fruit fly (*D. melanogaster*), mouse and human and found that the use of evolutionary conservation indeed improved the predictive power of co-expression data.

2.1.3 Ontology Based Methods

Ontologies define terms within a specific subject area [42]. There are some ontologies in the field of biomedical research, such as Gene Ontology (GO) [28], the eVOC anatomical ontology [43], the mammalian phenotype ontology (MP) [44] and the human phenotype ontology [45].

Several candidate-gene identification systems that rely on grouping GO terms have been reported [18–20]. Perez-Iratxeta et al. [19] developed a methodology based on biomedical literature that associates pathological conditions with particular Gene Ontology (GO) terms, which then allowed candidate disease genes to be ranked according to the number of these terms they share. Freudenberg and Propping [20] produced clusters of known disease genes based on a measure of phenotypic similarity between diseases.

Candidate genes were then scored according to the GO terms shared with known disease genes in the clusters.

Tiffin et al. [23] used another kind of ontology, the eVOC anatomical system ontology [43], to predict disease genes. The eVOC ontology provide simple sets of controlled terms describing human anatomical systems, cell types, diseases and developmental stages. They used the eVOC anatomical system ontology as a bridging vocabulary that integrates clinical and molecular data through a combination of text- and data-mining, and select candidate disease genes according to their expression profiles within tissues affected by the disease of interest.

2.1.4 PPI Network Based Methods

Proteins usually bind to each other to perform a biological function. This binding is called physically interaction. Interacting proteins tend to be involved in the same cellular process and preferentially interact with other disease genes significantly [46], so disease genes mutations in their genes may lead to similar disease phenotypes. Oti et al. [26] predicted interacting partners of disease genes in the disease loci to be disease genes. Xu and Li [25] found that disease genes share some distinct topological features in the PPI network. Based on these features, they employed the K-nearest neighbor classifier to predict disease genes and the performance of prediction was promising. Köhler et al. use Random Walk with Restart (RWR) algorithm to prioritize candidate genes [24], which is illustrated in Figure 2.1. In the susceptible region, there are n genes called candidate genes. Firstly, candidate genes and known disease genes are mapped to the PPI network. Then, disease genes are used as source nodes to run the RWR algorithm,

Figure 2.1: Candidate genes prioritization using random walk with restart (RWR) algorithm, taken from [24].

and candidate genes are scored by their proximity to known disease genes. Finally, based on the proximity score, candidate genes are prioritized.

In Figure 2.1, one genetic disease is represented as five entries in the database of Online Mendelian Inheritance of Man (OMIM) [5]. The corresponding known disease genes are mapped into the PPI network, represented as yellow nodes without number. And candidate genes are represented as yellow nodes with number. As shown in Figure 2.1, the third candidate gene is well connected to known disease genes, therefore it is very likely to be the real disease gene. After running the RWR algorithm, this gene is given the highest proximity score among all the candidate genes, therefore it is ranked at the top.

2.2 Integration Methodologies on Candidate Genes Prioritization

An ever-increasing wealth of genomic data is now available in public databases [47, 48]. Each data source covers part of the human genes, therefore these data sources are complementary to each other. For example, a gene that has been extensively studied for a long time will have a large amount of associated literature and have a better chance of being annotated in GO [28]. Genes that have well-characterized protein products are more likely to be found in the PPI network. In this section, we introduce some integration algorithms on disease gene identification.

2.2.1 SUSPECTS

Adie et al. [29] proposed a software named SUSPECTS for prioritization of candidate genes. In SUSPECTS, gene annotation, expression and sequence data are used to prioritize disease candidate genes. For each disease, a set of known disease genes is used as training set. Four kinds of gene features have been used to train the SUSPECTS model, i.e., sequence, expression, protein domain and functional annotation. Each candidate gene is scored based on four lines of evidence: first by PROSPECTR [22] on the basis of its sequence features, second by the extent of co-expression with the training set based on expression data [49], third by the number of rare (found in $< 5\%$ of all proteins) protein domains, obtained from the Interpro database [50], shared with the training set, and finally by the functional similarity [51] to genes in the training set. The four scores are then combined. Each score is weighted depending on the amount of information available for each line of evidence.

2.2.2 CAESAR

Gaulton et al. [30] integrated data from several ontologies to discover disease genes associated with user interested disease phenotype. Their integration algorithm is called CAESAR. Four ontologies are used in CAESAR, that are the Biological Process (BP) and Molecular Function (MF) ontologies from GO [28], the mammalian phenotype ontology (MP) [44] and the eVOC anatomical ontology [43].

The input of CAESAR is user-defined body of text. This input text is ideally an authoritative and comprehensive source of biological knowledge about the disease of interest. It can be the clinical symptom of the disease or an OMIM identifier [5]. If an OMIM identifier is supplied, CAESAR will use the OMIM record as input. For each ontology, CAESAR uses text mining techniques to extract related ontology terms and assign a correlation score $(c_1, c_2, \dots, c_m)$ to each of the m extracted terms. The score ξ_{ij} of gene i for source j is then calculated as either the maximum, sum or mean of the correlation scores of the m matching ontology terms or extracted genes. Then, the normalized score for each gene is defined as

$$z_{ij} = \frac{\xi_{ij} - v_j}{s_j}, \quad (\text{Eq. 2.1})$$

where v_j is the mean and s_j is the standard deviation of the scores from data source j . Finally, the combined score for one candidate gene is obtained by taking the maximum, sum or mean of the modified scores from different data sources.

2.2.3 Prioritizer

Franke et al. [31] proposed a method named Prioritizer to discover disease genes from the susceptibility loci. They compiled a human PPI network containing interactions

from different databases, i.e., the Kyoto Encyclopedia of Genes and Genomes (KEGG) [52], the Biomolecular Interaction Network Database (BIND) [53], Reactome [54], and the Human Protein Reference Database (HPRD) [55]. Then they predicted some other functional relationships from GO and microarray data using the Bayesian framework, and constructed a human gene functional network. Finally, candidate genes were prioritized based on the shortest path distance to known disease genes in the functional network.

2.2.4 TOM

Rossi et al. [32] proposed a web-based system named Transcriptomics of OMIM (TOM) to predict disease genes by integrating gene expression data and GO data [28]. They used disease genes from OMIM [5] as seeds, and defined a co-expression neighborhood, i.e. a set of candidate genes with similar expression to the seeds. Then, they further filtered candidate genes based on their functional annotation in GO to produce a shortlist of set of candidate genes.

2.2.5 Candidate Protein Complex Method

Genes sharing mutant phenotype are highly correlated in their biological functions [56]. The first work to integrate phenotype information was proposed by Lage et al. in 2007 [36]. For a disease phenotype, they found the most probable candidate gene by finding the most probable corresponding protein complex, which comprised of the candidate gene and the disease genes associated with similar disease phenotypes. The procedures are described subsequently, taking Leber Congenital Amaurois (LCA) as an example. The first step is to find protein complexes including the candidate genes. The protein

complex is named as the candidate complex. In the second step, proteins known to be involved in the similar disorders are identified in the candidate complex. In this case, proteins that are involved in different disorders comparable to LCA are scored according to the phenotype similarity. The next step involves scoring and ranking the candidate complexes. Finally the ranking of candidate genes are obtained according to the ranking of corresponding candidate complexes.

2.2.6 CIPHER

Wu et al. [1], proposed an integration method named CIPHER, which integrates PPI network and the phenotype information obtained from OMIM. Given a query phenotype and a set of candidate genes, CIPHER calculates a concordance score between the query phenotype and the candidate gene. In the first step, it calculates a similarity profile of the query phenotype. The similarity profile is a numerical vector, consisting of the similarity scores between the query phenotype and all phenotypes. The similarity score between two phenotypes is calculated based on the topological distance between two sets of associated disease genes in the PPI network. In the second step, they calculate the closeness profile from a candidate gene to all the phenotypes. The closeness profile is a numerical vector, an element of which denotes the proximity from the candidate gene to a phenotype. It is calculated based on the topological distance between the candidate gene and the set of disease genes associated with the phenotype in the PPI network. Finally, the concordance score is calculated as the correlation between similarity profile and closeness profile. The candidate genes are ranked based on corresponding concordance scores.

Figure 2.2: The procedures of ENDEAVOUR, a rank aggregation based integration method. This figure is taken from [33].

2.2.7 ENDEAVOUR

Aerts et al. [33] developed the tool named ENDEAVOUR, the procedure of which is shown in Figure 2.2. In the first step, a ranking model is trained based on each data source using training genes (known disease genes). In the second step, candidate (test) genes are ranked using trained ranking algorithm based on each data source. In the set of candidate genes, the red square is supposed to be the real disease gene (Figure 2.2, Step 2). Finally, these rank lists were integrated into a single list by making use of N-dimensional order statistics (NDOS) [35]. Detailed description of the NDOS algorithm can be found in Section 5.2.4.

Aerts et al. provided a different ranking method for different kinds of data, i.e., numerical vector-based data, attribute-based data and other data types. To rank a set of test genes, vector-based data are scored by Pearson correlation between the test vector and the average of training vectors. Attribute-based data are scored by Fishers omnibus meta-analysis, generating an enrichment p -value for each attribute. The protein sequence data and PPI data are neither vector-based nor attribute-based. The protein sequence is represented as a string, in which each character denotes an amino acid. The similarity score between a test gene (protein) and the training set is the lowest E-value obtained from the BLAST [57] against the protein sequences of the training genes. For PPI network, the similarity score is calculated as the overlap between all protein-protein interaction partners of the training set and those of the test gene. For each data source, candidate genes are ranked based on these similarity scores. As described above, the ranking methods used in ENDEAVOUR are very simple and have much space to be

improved. More effective ranking algorithms need to be introduced, such as one-class SVM (1CSVM) [58] and the RWR algorithm [24].

2.2.8 Multiple Kernels Learning

Kernel-based methods exhibit good classification performance, and they can conveniently be adapted to the problem at hand by designing appropriate kernel functions to represent prior knowledge. Therefore, kernel-based methods, Multiple Kernels Learning (MKL) have been successfully used to integrate heterogeneous data sources [59].

De Bie et al. [34] extended the MKL method to one-class problem. Similar to Lanckriet et al. [59], they generate a gene functional similarity matrix based on each data source, which is called kernel matrix. Each kernel matrix is a representation of the similarity between genes. Then they combine multiple kernels into one. The 1CSVM ranking model is trained based on the combined kernel, which is used to prioritize set of candidate genes. Detailed description of this algorithm is provided in Section 5.2.3.

2.3 Summary

This chapter provided a review on computational methods for candidate gene prioritization, including prioritization algorithms based on single data source and integration algorithms. A future challenge will be to devise a uniform framework for integrating all the data sources, that is, to develop a disease gene predicting system. One aspect of the challenge is that most of the data sources are incomplete and noisy. Another aspect is that genomic data are very different in their characteristics, such as Boolean interactions, real number vectors for expression profiles, gene sequence data, categorical gene

annotation data and text record from OMIM, etc.

Chapter 3

Data Collection and Preprocessing

In this chapter, first of all we introduce data sources used for disease gene identification, protein-protein interaction (PPI) network, gene annotation from the Gene Ontology (GO) database, and the corresponding preprocessing procedures. Then we introduce disease related data investigated in this work: cancer genes, human inheritance disease, disease phenotype and disease gene. Datasets used in this thesis are listed in Table 3.1, and available online at http://www3.ntu.edu.sg/home/aspatra/research/Yongjin_Thesis_Data.zip.

3.1 Protein-Protein Interaction Network

In this section, we introduce protein interaction, the PPI database, topology of PPI network and kernel representation of the PPI network.

3.1.1 Protein-Protein Interaction

A protein can not perform its function in isolation. Instead, its functional expression, involved in regulating a cellular activity, is realized through interacting with other proteins [60]. The proteins involved are physically in contact through the interaction or

Figure 3.1: An illustration of Protein-Protein Interaction.

the formation of a protein complex, as shown in Figure 3.1. Protein-protein interactions are of prime importance for virtually every process in a living cell and form the basis of all biological mechanisms such as DNA replication and transcription, enzyme-mediated metabolism, signal transduction, and cell cycle control [61]. An individual protein plays its cellular role through interacting with other proteins. Therefore, the interactions of an uncharacterized protein can provide a clue about its biological function.

The advent of genome sequencing projects makes it possible to analyze protein-protein interactions at the genome scale. The human genome sequence is almost completed in the year of 2001 [62, 63]. The whole genome sequence provides the information about all the proteins in genome, i.e., the so-called "proteome". On the other hand, high-throughput experiments have populated the public databases with thousands of protein-protein interaction data [64].

Protein A HPRD identifier	Protein A Entrez Gene	Protein B HPRD identifier	Protein B Entrez Gene	Experiment Type	PubMed
02995	4089	04484	4093	yeast 2-hybrid	16189514,15231748
01898	7429	00017	71	in vitro	3510866,6893424
16000	55607	00017	71	in vitro;in vivo	9362513,12052877
16375	29124	00017	71	in vitro	15009185
03014	1173	03690	1856	yeast 2-hybrid	16189514
02319	10134	00017	71	in vivo	10958671
02458	9087	00017	71	in vitro	8416954
02549	4130	00017	71	in vitro	7820861
03014	1173	04318	10320	yeast 2-hybrid	16189514
02991	4306	00017	71	in vitro	8612804
03180	5339	00017	71	in vitro	9808630
03334	8655	00017	71	in vitro	14760703
03415	7408	00017	71	in vitro;in vivo	12372613
03014	1173	05629	9513	yeast 2-hybrid	16189514
03014	1173	10262	9779	yeast 2-hybrid	16189514
03855	5337	00017	71	in vitro;in vivo	12388543
03014	1173	13248	151871	yeast 2-hybrid	16189514
03014	1173	16106	10900	yeast 2-hybrid	16189514
06049	29109	00017	71	in vivo	12665555
08371	4640	00017	71	in vitro	11474115
10808	10486	00017	71	in vivo	8761950
11539	85477	00017	71	in vitro	1331119
12282	93661	00017	71	in vitro	12029070

Figure 3.2: Section of Protein Interaction Data obtained from HPRD [55].

3.1.2 The PPI Database — HPRD

The PPI data were derived from Human Protein Reference Database (HPRD) [55]. All the interactions in HPRD have been manually extracted from literature by expert biologists who read, interpret and analyze the published data. Therefore, PPI data in this database is most reliable. A section of HPRD data is shown in Figure 3.2, where each protein is represented by HPRD identifier and Entrez Gene ID. This contains not only the information of protein interactions, but also the experimental information and literature the interaction comes from. For example, the first row in Figure 3.2 shows that the interaction between proteins encoded by gene 4089 and 4093 is detected by the yeast 2-hybrid experiment technique. This interaction is reported in two articles with PubMed ID 16189514 and 15231748.

Table 3.1: Datasets used in this thesis ^(a).

Dataset	Description	Size	Source
#1	PPI Network (HPRD Release6)	$34,405 \times 2$	HPRD [55]
#2	PPI Network (HPRD Release8)	$36,619 \times 2$	HPRD [55]
#3	Wu PPI Network	$34,364 \times 2$	Wu et al. [1]
#4	Gene ID Mapping	$19,391 \times 3$	HPRD [55]
#5	BP Network	$74,184 \times 3$	Gene Ontology [28]
#6	CC Network	$64,709 \times 3$	Gene Ontology [28]
#7	MF Network	$132,331 \times 3$	Gene Ontology [28]
#8	Cancer Genes	311×1	Cancer Census [12]
#9	House-Keeping Genes	$1,007 \times 1$	Tu et al. [65]
#10	Control Set	$7,738 \times 1$	Excluding cancer genes and house-keeping genes from the PPI Network
#11	36 Diseases	36 text files ^(b)	Köhler et al. [24]
#12	1,284 Diseases	$2,929 \times 6$	Goh et al. [66]
#13	Phenotype Network 2009	$27,100 \times 3$	OMIM [5]
#14	Phenotype Network 2005	$19,729 \times 3$	Wu et al. [56]
#15	PGRs_2009	$3,871 \times 2$	OMIM [5]
#16	Wu PGRs	$1,428 \times 2$	Wu et al. [1]
#17	AD Candidate Genes	12 text files ^(c)	OMIM [5]
#18	IDDM Candidate Genes	14 text files ^(d)	OMIM [5]
#19	Gene Location	$24,476 \times 5$	ENSEMBL [67]

^(a) All the datasets are available online at http://www3.ntu.edu.sg/home/aspatra/research/Yongjin_Thesis_Data.zip.

^(b) Each file is named after the disease, containing a list of respective disease genes.

^(c) File names are given as per the MIM ID of respective phenotypes. Each text file contains a set of candidate genes for respective phenotype of Alzheimer's Disease.

^(d) File names are given as per the MIM ID of respective phenotypes. Each text file contains a set of candidate genes for respective phenotype of Insulin-Dependent Diabetes Mellitus.

In this thesis, we have used three PPI datasets obtained from different versions of HPRD. The first one was obtained from HPRD release 6, containing 34,405 interactions between 9,019 genes (Dataset #1 in Table 3.1). The second one is shown as Dataset #2 in Table 3.1. It was obtained from HPRD release 8, containing 36,619 interactions between 9,474 genes. In both Dataset #1 and Dataset #2, genes are represented as Entrez Gene IDs. The last one contains 34,364 interactions between 8,919 genes, Dataset #3 in Table 3.1. It is the data used in Wu et al.’s work [1], available online at http://bioinfo.au.tsinghua.edu.cn/cipher/interactome_hprd.txt. In Dataset #3, HPRD ID is used to represent a gene. The cross reference between HPRD ID, Entrz ID and gene symbol is provided in Dataset #4.

3.1.3 Topology of the PPI Network

The PPI data can be viewed as a graph (network), where proteins are represented by nodes and interactions between proteins are represented by edges connecting the nodes. Analysis of the global architecture of this large-scale interaction network can give us insights about the cellular mechanisms. Jeong et al. [68] established a positive correlation between the connectivity and lethality; in particular, highly connected proteins are three times more likely to be essential (i.e., lethal upon deletion). This result is not surprising, since a mutation of a highly connected protein tends to affect more significantly the PPI network, and hence, it is more likely to be lethal. Figure 3.3 taken from [68] shows the yeast interaction network. The color of the nodes shows the essentiality of the yeast protein. Red means highly essential and green nodes denotes low essential.

Figure 3.3: The PPI Network taken from [68].

The protein interaction network provides a unique approach to address challenges in this post genome era, especially for understanding the functions of many newly discovered genes whose functions have not been characterized. PPI networks have been used to identify protein complexes [69], pathways [70] and to predict gene functions [71]. The large-scale PPI data allows us to study the general relationship between protein-protein interaction and diseases, especially the relationship between interaction network topology and disease genes.

3.1.4 Diffusion Kernel

The diffusion kernel is a common kernel function for graph-based data. It can be seen as a random walk on the graph [72]. Here the diffusion kernel method is used to represent the similarity between genes in the PPI network.

The PPI network with $\mathcal{N}_g$ number of nodes can be represented by a graph $G(V, E)$ be a graph, in which V is a set of nodes given by $V = \{g_1, g_2, \dots, g_{\mathcal{N}_g}\}$, and E is a set of edges given by $E = \{e(i, j) | \text{if node } g_i \text{ is connected with node } g_j\}$. Let $A_{\mathcal{N}_g \times \mathcal{N}_g}$ be the adjacency matrix, whose (i, j) -th element is given by

$$A_{i,j} = \begin{cases} 1, & \text{if } e(i, j) \in E \\ 0, & \text{otherwise.} \end{cases} \quad (\text{Eq. 3.1})$$

The diffusion kernel is calculated based on the Laplacian matrix of the graph, $L_{\mathcal{N}_g \times \mathcal{N}_g}$, whose (i, j) -th element is given by

$$L_{i,j} = \begin{cases} -A_{i,j}, & \text{if } i \neq j \\ d(i) - A_{i,j}, & \text{otherwise,} \end{cases} \quad (\text{Eq. 3.2})$$

where $d(i)$ is the sum of the i -th column in A . The kernel function is defined as

$$k(g_i, g_j) = e^{\beta L_{i,j}}, \quad (\text{Eq. 3.3})$$

which measures the proximity between two nodes, g_i and g_j , in the network. In this work, the value of β is set as -0.25 . The larger $k(g_i, g_j)$ value indicates the higher similarity between g_i and g_j .

3.2 Gene Ontology and Gene Functional Similarity

In this section, we introduce the Gene Ontology (GO) data and gene functional similarity based on Gene Ontology annotation.

3.2.1 Gene Ontology

The Gene Ontology provides a controlled vocabulary to describe gene and gene product attributes [28]. It is comprised of three independently constructed sub-ontologies, 'biological process' (BP), 'cellular component' (CC) and 'molecular function' (MF).

Figure 3.4: Part of BP Ontology. This figure is taken from [73].

Biological process (BP) refers to a biological objective to which the gene or gene product contributes. It often involves a chemical or physical transformation. A section of BP ontology taken from [73] is shown in Figure 3.4, which consists of different levels of BP terms. An example of general (high level) biological process term is 'signal transduction' and its child node 'intracellular receptor-mediated signalling' is a more specific term.

Cellular component (CC) refers to the place in the cell where a gene product is active. It includes terms such as 'nuclear membrane' or 'Golgi apparatus'. The structure of CC ontology is also the directed acyclic graph similar to BP ontology shown in Figure 3.4.

Molecular function (MF) is defined as the biochemical activity (including specific binding to ligands or structures) of a gene product. This definition also applies to the capability that a gene product (or gene product complex) carries as a potential. It

$$X = \begin{matrix} & \begin{matrix} t_1 & t_2 & t_3 & \dots & t_{\mathcal{N}_t} \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & \dots & 1 \\ 0 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & \dots & 0 \end{bmatrix} & \begin{matrix} g_1 \\ g_2 \\ \vdots \\ g_{\mathcal{N}_g} \end{matrix} \end{matrix}$$

Figure 3.5: Illustration of gene annotation matrix.

describes only what is done without specifying where or when the event actually occurs. Examples of broad functional terms are 'enzyme', 'transporter' or 'ligand'. Examples of narrower functional terms are 'adenylate cyclase' or 'Toll receptor ligand'.

3.2.2 Gene Annotation in Gene Ontology

If a gene is found to be involved in a biological process, such as phosphorylation, then in BP ontology this gene is associated with the term 'Phosphorylation'. This association is named as 'gene annotation'. In an ontology (i.e. BP), one gene is annotated with one or more terms. Suppose $\mathcal{N}_t$ is the number of terms in a ontology, and $\mathcal{N}_g$ is the number of genes annotated in this ontology. In this thesis, the annotation information in a ontology is represented as a gene-term matrix, as shown in Figure 3.5. The annotation vector of gene g_i is represented as a row in the annotation matrix,

$$\mathbf{x}_i = [x_{i,1}, x_{i,2}, \dots, x_{i,\mathcal{N}_t}]. \quad (\text{Eq. 3.4})$$

The annotation vector is binary valued, with '1' representing the presence of the GO term in the gene's annotation and '0' representing its absence. Formally, x_{ij} is defined as

$$x_{ij} = \begin{cases} 1, & \text{if gene } g_i \text{ is annotated with term } t_j \\ 0, & \text{otherwise} \end{cases} \quad (\text{Eq. 3.5})$$

3.2.3 Gene Functional Similarity

The functional similarity between two genes can be measured by the semantic similarity between their GO annotation terms [74–77]. In our research work, the similarity between two genes was measured by their overlap annotation terms [77], because of its computational efficiency and effective. This measure showed comparable performance with three other semantic similarity measures, Jiang [74] and Lin [75] and Resnik [76].

The number of overlap annotation terms can be calculated as the dot product of the annotation vectors, as mentioned in Section 3.2.2. However, the importance of GO terms is different. Intuitively, genes sharing low level specific term, e.g., 'intracellular receptor-mediated signalling', are expected to be more functionally similar than genes sharing a higher level general term, e.g., 'signal transduction'. Therefore, the '1' values in the annotation vector were replaced by the information content (IC) of the corresponding GO term. The IC of a term t_j is related to how often t_j is associated to genes in the database, thus rarely used terms are assigned with higher IC. The calculation of IC for the term t_j is described as follows:

$$IC(t_j) = -\log_2(\mathcal{N}(t_j)/\mathcal{N}_g) \quad (\text{Eq. 3.6})$$

where $\mathcal{N}(t_j)$ is the number of genes annotated with term t_j .

Let $\tilde{\mathbf{x}}_i$ denote the weighted annotation vector for g_i . The j -th element of $\tilde{\mathbf{x}}_i$ is calculated by

$$\tilde{x}_{ij} = IC(t_j) \cdot x_{ij}. \quad (\text{Eq. 3.7})$$

Suppose $S_{\mathcal{N}_g \times \mathcal{N}_g}$ is the similarity matrix between genes in the ontology. The (i, j) -th element of S measures the similarity between two genes g_i and g_j . It is calculated by dot

Figure 3.6: The procedure of generating KNN graph with $K = 2$.

product between corresponding weighted vectors

$$S_{i,j} = \text{sim}(g_i, g_j) = \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_j^T. \quad (\text{Eq. 3.8})$$

3.2.4 Gene Networks Obtained from GO

Because three sub-ontologies BP, CC and MF are independently created, we calculated gene functional similarity based on each sub-ontology separately and constructed three gene networks.

From each sub-ontology, we first obtained the gene annotation matrix, as shown in Figure 3.5. Secondly we calculated the gene functional similarity matrix using Eq. 3.8. From the similarity matrix, we constructed a K -nearest neighbor (KNN) graph, with K equals to five.

The procedure of constructing a KNN graph with $K = 2$ is shown in Figure 3.6. We start from the similarity matrix for five nodes (a, b, c, d and e). Then for each node, we find its K most similar nodes. As shown in Figure 3.6, the similarity matrix is transformed into the adjacency matrix. We found b and e are two most similar nodes to node a. Therefore, in the first column of the adjacency matrix, the second and the fifth

elements remain the same as the similarity matrix, whereas other elements are set as 0. In Figure 3.6, all the non-zero elements in the adjacency matrix are in blue rectangles. Finally we construct the KNN graph by linking the nodes with its two most similar nodes and use the similarity value as the weight of the edge.

From the annotation matrix mentioned in Section 3.3, we obtained three gene networks from BP, CC and MF respectively. The sizes of the three networks are shown in Table 3.1 (Dataset #5—#7).

3.3 Disease Related Data

In this research, we studied cancer and human inheritance disease. We first introduce the dataset of cancer genes and then introduce datasets related to genetic diseases.

3.3.1 Cancer Genes

The cancer genes are obtained from Cancer Gene Census database (http://www.sanger.ac.uk/genetics/CGP/Census/Table_1_full_2010-03-30.xls), under the Cancer Gene Project. There are 311 cancer genes in the PPI network. We use the 311 cancer genes as positive training set (Dataset #8). However it is difficult or even impossible to get negative sample (non-cancer genes), because we do not know which gene may not contribute to cancer development or progress. The negative genes are randomly selected as follows. Firstly, we exclude 311 cancer genes from all the genes in the PPI network. Secondly, we exclude house-keeping genes because they have distinct features [65]. The set of house-keeping genes is shown as Dataset #9 in Table 3.1, which is obtained from <http://www.biomedcentral.com/content/supplementary/1471-2164-7-31-s2.txt>. After

Table 3.2: Prefix and category of MIM entries.

Prefix	Category of MIM entries
%	Mendelian phenotype or locus, molecular basis unknow
*	gene with known sequence and phenotype
#	phenotype description, molecular basis known
+	gene with known sequence and phenotype
^	the obsoleted record
<i>none</i>	other, mainly phenotypes with suspected Mendelian basis

excluding cancer genes and house-keeping genes, the remaining 7,738 genes are defined as control gene set (Dataset #10). Finally, we randomly select 311 genes, the same number as cancer genes, from the control gene set, which are used as negative samples.

3.3.2 The OMIM Database

Online Mendelian Inheritance in Man (OMIM) [5] is a comprehensive, authoritative and periodically updated knowledgebase of human genes and genetic disorders compiled to support human genetics research. Derived from the biomedical literature, OMIM is written and edited at Johns Hopkins University with input from scientists and physicians around the world. Each OMIM entry has a text summary of a genetically determined phenotype and/or gene.

In the OMIM database, each entry is assigned a unique six-digit MIM ID. MIM entries are categorized by their information on genes, phenotypes or both. This is denoted by the prefix symbol of an MIM ID as given in Table 3.2. Especially, a percentage sign (%) prefix indicates that the entry describes a confirmed Mendelian phenotype or phenotypic locus for which the underlying molecular basis is not known. Genes in this region are

NCBI OMIM Online Mendelian Inheritance in Man

Search for

Limits Preview/Index History Clipboard Details

Display Show Send to

MIM ID %605526

ALZHEIMER DISEASE 6

Alternative titles; symbols

AD6
ALZHEIMER DISEASE 6, LATE-ONSET

Other entities represented by this entry

PLASMA BETA-AMYLOID-42 LEVEL QUANTITATIVE TRAIT LOCUS, INCLUDED

Gene map locus: [10q24](#)

Text [Back to Top](#)

For a phenotypic description and a discussion of genetic heterogeneity of Alzheimer disease (AD), see [104300](#).

Clinical Features [Back to Top](#)

[Bassett et al. \(2005\)](#) performed functional magnetic resonance imaging (fMRI) of 9 asymptomatic female offspring from AD families linked to the chromosome 10q region and 6 females from AD families unlinked to this region. These individuals were on average 11.3 years younger than the onset age of their parents. All were APOE4-[\(107741\)](#) negative. During memory encoding tasks, individuals from the 10q-linked families showed more extensive activation in the bilateral temporal and frontal lobes as well as in the thalamus and right parahippocampal gyrus, compared to offspring from the unlinked families who

Table of Contents

MIM %605526
Text
Clinical Features
Mapping
Molecular Genetics
References
Contributors
Creation Date
Edit History

Links

Selected Gene Related Links
[Entrez Gene](#)
Other NCBI Links
Free in PMC
Gene
Map Viewer
PubChem Compound
PubChem Substance
PubMed (calculated)
PubMed (cited)
UniGene
Related Entries
Nucleotide
Protein
SNP
UniSTS

Figure 3.7: Snapshot of an MIM record. The MIM ID of this record is 605526. It is a phenotype of Alzheimer’s Diseases (AD6).

named candidate genes associated with this phenotype. The object of this research is to prioritize candidate genes for this kind of phenotype. Figure 3.7 shows an example of MIM record with prefix ‘%’. This MIM record corresponds to a phenotype of Alzheimer’s disease (AD6). This phenotype is associated to the locus at 10q24. Using BioMart, we obtained 122 genes from this region, which were used as the set of candidate genes for this phenotype.

3.3.3 Disease and Disease Phenotype

In OMIM database, one disease might be described in one or more MIM records. For example, Alzheimer’s disease is described in 16 MIM records. Similar to previous works [1, 56], the phenotype was defined as an OMIM record. We excluded the records with

the prefix '*' and '^', because the prefix '*' refers to the record of disease gene, and '^' refers to an obsolete record.

Köhler et al. [24] manually categorized thousands of OMIM phenotypes into 110 diseases. The largest one contains 47 genes and the smallest contains only three genes. These diseases include genetically heterogeneous disorders in which mutations in distinct genes are associated with similar or indistinguishable phenotypes; cancer syndromes comprising genes associated with hereditary cancer, increased risk, or somatic mutation in a given cancer type; and complex (polygenic) disorders known to be influenced by variation in multiple genes. In this thesis, we used Köhler et al.'s disease dataset, but we only consider the diseases with at least 6 disease genes, because the training set for 1CSVM (Section 5.2.1) and MKL (Section 5.2.3) should not be too small. Therefore, in our benchmark data, there are 36 diseases, containing 497 disease genes in total. This benchmark data is shown as Dataset #11 in Table 3.1. Names of the 36 diseases are shown in Appendix Table A.1.

Goh et al. grouped MIM phenotype entries into 1,284 diseases based on the description of the entries. In total, 1,778 phenotypes is involved. The data is shown as Dataset #12 in Table 3.1. It was obtained from http://www.barabasilab.com/pubs/CCNR-ALB_Publications/200705-14_PNAS-HumanDisease/Suppl/supplementary_tableS1.txt.

3.3.4 The Phenotype Network

A text mining tool, named MimMiner, is reported by van Driel et al. [56], which can calculate the similarity between phenotypes obtained from OMIM [5] database. Data processing of MimMiner is shown in Figure 3.8, and the usage of MimMiner is described in

Figure 3.8: Data processing in MimMiner.

Appendix B. A disease phenotype is described in two fields of an OMIM record, the 'Text' and 'Clinical Features', as shown in Figure 3.7. The similarity between phenotypes is measured by the co-occurrence of key words in the Medical Subject Headings vocabulary (MeSH) [78]. MeSH is a comprehensive controlled vocabulary for the purpose of indexing journal articles and books in the life sciences; it can also serve as a thesaurus that facilitates searching. As shown in Figure 3.8, the disease phenotype is converted into an MeSH term vector. An element of the vector denotes frequency of the occurrence of a MeSH term in the text description of the disease phenotype. The similarity between two phenotypes is calculated as the correlation coefficient between two vectors.

After obtaining the phenotype similarity matrix, we constructed the KNN graph (phenotype network) using the procedure shown in Figure 3.6. In our study, we have used $K = 5$. In the phenotype network, each node denotes a phenotype and the link between the nodes denotes the similarity between phenotypes. The phenotype network was constructed based on the OMIM data in the year of 2009. It contains 27,100 links between 6,708 phenotypes. The data is shown as Dataset #13 in Table 3.1. The old version phenotype network was constructed based on OMIM data obtained in the year of 2005. The corresponding phenotype similarity matrix was obtained from <http://www.cmbi.ru>.

nl/MimMiner/MimMiner_Exp_AC_T_TXCS_basedonACMESH_filt_RW.mat.gz. The constructed phenotype network contains 19,729 links between 5,080 phenotypes. The data is shown as Dataset #14 in Table 3.1.

3.3.5 Disease Genes and Phenotype–Gene Network

The disease genes for 36 diseases (Dataset #11) were obtained from <http://compbio.charite.de/genewanderer>. Given a disease phenotype, the corresponding disease gene is obtained from OMIM, using BioMart [79]. BioMart is a query-oriented data management system, providing an interface for querying different biological data sources in an integrated manner. One can cross query most of the major biological databases using this system. The brief description of BioMart is provided in Appendix B.

From the OMIM database of the year 2009, we obtained 3,871 phenotype-gene relationships (PGRs) between 2,464 phenotypes and 2,840 genes, shown as Dataset #15 in Table 3.1. The PGR data was represented as a bipartite network, in which genes and phenotypes were represented as nodes and the PGRs as edges.

The old version of PGR data were obtained from Wu et al.’s work [1]. The data is available at http://bioinfo.au.tsinghua.edu.cn/cipher/landscape_gene_phenotype_network_hprd.txt. It contains 1,428 PGRs between 937 genes and 1,126 phenotypes, shown as Dataset #16 in Table 3.1. In this dataset, 168 disease phenotypes are associated with more than one disease gene.

Candidate genes are a set of genes located in the susceptible locus. Given a susceptible locus, a set of candidate genes can be obtained using BioMart. For example, the susceptible locus for MIM 605526 is at 10q24. Using BioMart, we obtained 122 genes

in this region, which were used as candidate genes. Using BioMart, we obtained a list of candidate genes for each of 12 phenotypes of Alzheimer’s disease. The data is shown as Dataset #17 in Table 3.1. The number of candidate genes ranges from 9 to 241. We also obtained candidate genes for 14 phenotypes of Insulin-dependent diabetes mellitus (IDDM), shown as Dataset #18. The number of candidate genes for IDDM ranges from 17 to 263.

In the procedure of LOO-CV (Section 5.2.6), no locus is available. In this case, the held-out disease gene and the 99 nearest genes in the chromosome are used as candidate genes. The chromosome region containing the 100 candidate genes are named as the Artificial Linkage Interval (ALI) for the held-out disease gene. The 99 genes was obtained using BioMart [79]. We firstly obtained the gene location data for 24,477 genes using BioMart. The data is shown as Dataset #19 in Table 3.1. Given a gene, we find its 99 nearest genes using Dataset #19.

3.4 List of Websites

Here we provide the list of websites from which the datasets and softwares are obtained.

3.4.1 Websites for the Databases

1. HPRD

Human Protein Reference Database (HPRD) — <http://www.hprd.org/download>

The data is free for academic users with prior registration. In this page, the PPI data is provided in various file formats, i.e, tab delimited text file, XML file and PSI-XML file. We used the text file. Dataset #1 and #2 were obtained from this website.

2. MeSH

<http://www.nlm.nih.gov/mesh/>

When calculating similarity between phenotypes to obtain KNN graph (Section 3.3.4), key words are obtained from MeSH.

3. OMIM

<http://www.ncbi.nlm.nih.gov/omim>

As shown in Table 3.1, various kinds of data were obtained from this database, including the phenotype network, the phenotype–gene network and the susceptible regions for a phenotype. We obtained Dataset #13, #15, #17 and #18 from this website.

4. Gene Ontology

<http://www.geneontology.org/ontology-archive/>

Ontology file downloads: GO terms and relationships in a variety of formats

[http://www.geneontology.org/gene-associations/gene_association.goa_human.
gz](http://www.geneontology.org/gene-associations/gene_association.goa_human.gz)

Annotation file downloads: associations between gene products and GO terms submitted by members and associates of the GO consortium. We obtained Dataset #5–7 from this website.

5. 36 Diseases Benchmark Data [24]

<http://compbio.charite.de/genewanderer>

The 36 diseases and corresponding disease genes were obtained from this website.

6. Wu Data

<http://bioinfo.au.tsinghua.edu.cn/cipher/>

This website provides all the data used in Wu et al.'s work [1]. Datasets #3, #14 and #16 were obtained from this website.

3.4.2 Websites for the Softwares

1. MimMiner — obtained via email with the author (h.brunner@antrg.umcn.nl).

Used to calculate the similarity between phenotypes.

2. BioMart — <http://www.biomart.org/biomart/martview/>

Used to retrieve genes from given locus and to obtain the PGRs.

3. LibSVM — <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

The SVM algorithm was implemented using this package.

4. NetLab — <http://www.ncrg.aston.ac.uk/netlab>

The MLP algorithm is implemented using this package.

5. GOSim — <http://cran.r-project.org/web/packages/GOSim/index.html>

It is a package used in R environment (<http://www.r-project.org/>). It was used to calculate the functional similarity between genes based on each of the three ontologies, BP, CC and MF.

Chapter 4

Prediction of Cancer Genes from PPI Network

Cancer is a group of complex diseases, in which a relatively large number of genes are involved. One of the main goals of cancer research is to identify genes which are causally relevant to the development and progress of cancer. The increasing identification of cancer genes and availability of genomic and proteomics data provide us opportunities to identify cancer genes by computational methods. In this chapter, we investigated five predictive topological features, derived from the PPI network. Two kinds of neural networks, Support Vector Machine (SVM) and Multi-Layer Perceptrons (MLP), were employed to assess the predictive ability of features. Results of 10-fold cross validation showed that the best feature for one gene measures the proportion of cancer genes in its 2-neighbourhood in the PPI network. At the same time, we found SVM performs slightly better than MLP, and the best feature combination for these two classification algorithms is the same. Using only three features, the best performance of our classification model can achieve accuracy as high as 73.9%, comparable to previous method [80], which combined protein sequence data and Gene Ontology annotation information.

Figure 4.1: A schematic diagram of cancer gene prediction using the PPI network.

4.1 Introduction

The increasingly identified cancer genes [12], the finished human genome sequence [13] and availability of genome wide functional annotation data [28] make it possible to predict cancer genes by computational methods. Furney et al. [80] conducted an analysis of a set of genes known to be involved in cancer [12], in order to unveil their unique features that can assist towards the identification of new candidate cancer genes. They detected key patterns in cancer genes in terms of the molecular function or the biological process in which they are involved as well as sequence properties. Based on these features they have developed a Bayesian classification model to predict cancer genes.

The rapid identification of genome-wide human PPIs [55] now offers new opportunities for identifying cancer genes. In this work, we investigated five predictive topological features, derived from the PPI networks. These five features were proposed by Xu and

Li [25], and have been used to investigate the genetic diseases. In this work, two kinds of neural networks, SVM and MLP, were employed to assess the predictive ability of these features. Results of 10-fold cross validation showed that the best feature for one gene measures the proportion of cancer genes in its 2-neighbourhood in the PPI network. At the same time, we found SVM performs slightly better than MLP. Using only three features, the best performance of our classification model can achieve accuracy as high as 73.9%. Our results are comparable to previous method [80], which combined two data sources, gene sequence data and annotation data. If we combine PPI data with other data sources, we expect to achieve better results.

4.2 Methods

4.2.1 Datasets and Topological Features

In this section, we describe details about the five topological features derived from PPI network. The description of these features is based on Figure 4.2. The node represents one protein and the edge represents the interaction between two proteins. Dark shaded nodes represent proteins encoded by cancer genes, i.e., node 4, 5, 11, 12 and 19. In this work, the PPI Network was obtained from HPRD Release6 (Dataset #1).

Let the set of $\mathcal{N}_g$ genes in the PPI network be represented by $V = \{g_1, g_2, \dots, g_{\mathcal{N}_g}\}$. Let $I_N^{(i)}$ be the indices of neighboring genes connected to g_i in the PPI network. The set of indices of disease genes are given by $I_D = \{i | g_i \in V \text{ and } g_i \text{ is a disease gene}\}$. The set of indices of disease genes those are neighbors of gene g_i is given by $I_{DN}^{(i)} = \{j | j \in I_D \text{ and } j \in I_N^{(i)}\}$. For every gene in the PPI network, we investigate five features derived from PPI network to find discriminative features for cancer gene identification.

Figure 4.2: An illustration of cancer genes in protein-protein interaction network, where dark shaded nodes denote cancer genes.

Feature 1 (Degree)

The degree of a node is the number of other nodes it is connected to by edges. We use $|I_N^{(i)}|$ to represent this feature for g_i , where $|\cdot|$ is the number of elements in the set.

$$F_1(g_i) = |I_N^{(i)}| \quad (\text{Eq. 4.1})$$

For example, referring to Figure 4.2, $F_1(g_7) = |I_N^{(7)}| = 2$, $F_1(g_{11}) = |I_N^{(11)}| = 5$.

Feature 2 (1N-index)

The 1N-index is the proportion of disease genes in its immediate neighbors.

$$F_2(g_i) = \frac{|I_{DN}^{(i)}|}{|I_N^{(i)}|} \quad (\text{Eq. 4.2})$$

For example, referring to Figure 4.2,

$$F_2(g_{11}) = \frac{|I_{DN}^{(11)}|}{|I_N^{(11)}|} = \frac{3}{5} = 0.6$$

Figure 4.3: An example of PPI network showing 2-neighborhood of node 7.

Feature 3 (2N-index)

The 2N-index measures the connectivity with cancer genes in the 2-neighborhood of g_i , where 2-neighborhood of one node means a sub-network within two steps of the investigated node. Below is the computation of this feature.

$$F_3(g_i) = \begin{cases} \frac{\sum_{j \in I_N^{(i)}} (|I_{DN}^{(j)}| - 1)}{\sum_{j \in I_N^{(i)}} (|I_N^{(j)}| - 1)}, & \text{if } i \in I_D \\ \frac{\sum_{j \in I_N^{(i)}} |I_{DN}^{(j)}|}{\sum_{j \in I_N^{(i)}} (|I_N^{(j)}| - 1)}, & \text{otherwise} \end{cases} \quad (\text{Eq. 4.3})$$

To make it more clear, we take node 7 in Figure 4.2 as an example. Figure 4.3 shows the 2-neighborhood of node 7.

$$F_3(g_7) = \frac{|I_{DN}^{(4)}| + |I_{DN}^{(11)}|}{(|I_N^{(4)}| - 1) + (|I_N^{(11)}| - 1)} = \frac{1 + 3}{5 + 4} = \frac{4}{9}.$$

Note that nodes 5 and 8 are counted twice. This is reasonable, because these two nodes are well connected in the 2-neighborhood of node 7 and should be given more importance.

Feature 4 (Sum of Topological Coefficients)

This feature is the sum of topological coefficients [81] between g_i and cancer genes. It measures the extent to which g_i shares interaction partners with other cancer genes. The equation is shown below:

$$F_4(g_i) = \sum_{j \in I_D} \frac{C_{i,j}}{\min(|I_N^{(i)}|, |I_N^{(j)}|)} \quad (\text{Eq. 4.4})$$

where $C_{i,j}$ denotes the number of nodes to which both node i and node j are linked. We take node 7 as an example, as shown in Figure 4.2 or Figure 4.3. There are three disease genes sharing neighbors with it, i.e., node 5, 12 and 19. Therefore,

$$\begin{aligned} F_4(g_7) &= \frac{C_{7,5}}{\min(|I_N^{(7)}|, |I_N^{(5)}|)} + \frac{C_{7,12}}{\min(|I_N^{(7)}|, |I_N^{(12)}|)} \\ &\quad + \frac{C_{7,19}}{\min(|I_N^{(7)}|, |I_N^{(19)}|)} \\ &= \frac{2}{\min(2,5)} + \frac{1}{\min(2,6)} + \frac{1}{\min(2,6)} = 2. \end{aligned}$$

Feature 5 (Average Distance to Disease Genes)

This feature calculates the average distance between g_i and all cancer-genes in the network. Below is the computation of *Feature 5*.

$$F_5(g_i) = \begin{cases} \frac{1}{|I_D|-1} \sum_{j \in I_D, j \neq i} d_{i,j}, & \text{if } i \in I_D \\ \frac{1}{|I_D|} \sum_{j \in I_D} d_{i,j}, & \text{otherwise} \end{cases} \quad (\text{Eq. 4.5})$$

where $|I_D|$ is the number of elements in I_D and $d_{i,j}$ is the length of the shortest path between nodes i and node j in the PPI network.

In Figure 4.2, $I_D = \{4, 5, 11, 12, 19\}$. An example for non-cancer gene is g_7 ,

$$F_5(g_7) = \frac{1}{|I_D|} \sum_{j \in I_D} d_{7,j} = \frac{1}{5}(1 + 2 + 1 + 2 + 2) = 1.6.$$

An example for cancer gene is g_{11} ,

$$F_5(g_i) = \frac{1}{|I_D| - 1} \sum_{j \in S_1, j \neq 11} d_{11,j} = \frac{1}{4}(2 + 1 + 1 + 1) = 1.25.$$

4.2.2 Classification Algorithms

In this work, we employ two kinds of classification algorithm, SVM [82] and MLP to assess the predictive ability of topological features. Support vector machines aim at finding a maximal separating hyper-planes. The assumption of SVM is that the larger the margin or distance between these hyper-planes, the better the generalization error of the classifier. Firstly, training vectors are mapped to a higher dimensional space by the function. Then SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. We used the SVM toolbox implementation in LIBSVM [83].

MLP is an important class of neural networks. In this work, we used MLP as implemented in Netlab [84]. Typically, the network consists of a set of inputs that constitute the input layer, one or more hidden layers of computational nodes, and an output layer of computational nodes. During the training process, the input signals propagate through the network in a forward direction, on a layer-by-layer basis, and the weights of every edge in the neural networks are modified to get the minimum error between the predicted label and target label.

4.2.3 Data Normalization

For each gene in the training set, we calculate five features as defined in Section 4.2.1 to train a classifier. However, ranges of the features are not comparable. For example, *Feature1* ranges from 1 to several hundreds, whereas the 1N-index and the 2N-index range between 0 and 1. With such diverse ranges, the features corresponding to degree may dominate in the classifier and nullify the effects of the other features. To avoid this problem, each feature is normalized to $[-1, 1]$. After calculating the features, we generated a matrix X . Suppose $X = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_K)$, where $\mathbf{x}_k = [x_{1k}, x_{2k}, \dots, x_{mk}]^T$ ($1 \leq k \leq K$), K is the number of features and m is the number of genes in training set. Let $\max(\mathbf{x}_k)$ be the maximum value in $\mathbf{x}_k$ and $\min(\mathbf{x}_k)$ is the minimum value in $\mathbf{x}_k$. The normalization method is carried out as shown below:

$$x_{(i,k)} = -1 + 2 \frac{x_{(i,k)} - \min(\mathbf{x}_k)}{\max(\mathbf{x}_k) - \min(\mathbf{x}_k)} \quad (\text{Eq. 4.6})$$

4.2.4 Cross Validation

We perform 10-fold cross validation experiments with all combinations of feature sets to evaluate the performance of two kinds of classification algorithms. We obtained a set of genes consisting of 311 disease genes (Dataset #8) and 311 non-disease genes randomly selected from Dataset #10, as described in Section 3.3.1. The data is divided into ten equal partitions with the same number of positive genes and negative genes. One

partition is singled out in turn as the test set, and genes of the remaining nine partitions are used as the training set. Classifier is trained with training set, and applied to test set to get the performance of the classifier. The performance is assessed by accuracy, which is the proportion of true results (both true positives and true negatives) in the prediction. True positive (TP) means the test sample is cancer gene, and the classifier predicted it as cancer gene. True negative (TN) means the test sample is not cancer gene and the classifier predicted it as non-cancer gene. False positive (FP) means the test sample is not cancer gene but the classifier predicted it as cancer gene. False negative (FN) means the test sample is cancer gene but the classifier predicted it as non-cancer gene. Accuracy is defined below.

$$Accuracy = \frac{\mathcal{N}_{TP} + \mathcal{N}_{TN}}{\mathcal{N}_{TP} + \mathcal{N}_{FP} + \mathcal{N}_{TN} + \mathcal{N}_{FN}} \times 100\% \quad (\text{Eq. 4.7})$$

where $\mathcal{N}_{TP}$, $\mathcal{N}_{FP}$, $\mathcal{N}_{TN}$ and $\mathcal{N}_{FN}$ are the number of samples in the corresponding categories. After each partition is used as a test set, we can get 10 accuracies, and average of these 10 accuracies is the over all performance of 10-fold cross validation.

As our negative samples are randomly selected in the control gene set, in order to avoid selection bias, we randomly sample 20 negative gene sets from the control gene set. Each time, we do cross validation on the positive gene set and one negative gene set, and obtain a prediction accuracy value. We use the average of 20 accuracy values as the over all performance measure.

4.3 Experiments and Results

4.3.1 Significance of the Five Topological Features

Usually, the discriminative power of a feature is assessed by t-test. For a feature, if the values are significantly different between two groups, this feature may contribute to discrimination. To evaluate whether or not these topological features are significantly different between cancer gene set and the control gene, for each feature, first of all we calculate values of this feature for every gene in the cancer gene set and the control gene set. Then, for each feature, we perform two-sided t-test by the equation below.

$$t = \frac{\text{mean}(\mathbf{x}_{\text{pos}}) - \text{mean}(\mathbf{x}_{\text{ctrl}})}{\sqrt{s_p^2(\frac{1}{n_1} + \frac{1}{n_2})}} \quad (\text{Eq. 4.8})$$

$$\text{where } s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}$$

For each investigated feature, $\mathbf{x}_{\text{pos}}$ is a vector of feature values for genes in cancer gene set, and $\mathbf{x}_{\text{ctrl}}$ is the feature values for genes in control gene set. The number of genes in cancer gene set and control gene set are $n_1 = 311$ and $n_2 = 7,738$, respectively. The standard deviation of $\mathbf{x}_{\text{pos}}$ and $\mathbf{x}_{\text{ctrl}}$ are s_1 and s_2 , respectively.

Calculated t-statistics are listed in Table 4.1. For two-sided t-test, the t-statistics corresponding to significance level of 0.001 is 3.29. Therefore, the t-statistics above 3.29 means the difference between cancer genes set and the control gene set are highly significant ($p < 0.001$). As shown in Table 4.1, t-statistics for all the features are much larger than 3.29, indicating that all five features are statistically significant. The results confirmed that cancer genes have more interactions than other genes [85]. In the PPI

Table 4.1: The statistical significance of five topological features obtained from the PPI network.

Feature No.	Feature Name	t-statistics
1	Degree	14.8
2	1N-index	8.31
3	2N-index	11.3
4	Sum of Topological Coefficient	16.1
5	Average Distance to Disease Genes	-14.3

network, cancer genes tend to interact with each other and communicate efficiently, organized as functional modules to work together [86]. Similar observation was found for hereditary genes [25].

4.3.2 Correlation Between the Five Topological Features

As described in Section 4.2.1, *Feature1* measures the centrality of cancer genes, *Features2*, 3 and 4 measure the extent of inter-connection between cancer genes and *Feature5* measures the communication efficiency between cancer genes. All the five features, assess the modularity of cancer genes from different but related aspects. Therefore, correlations may exist among these features. We calculated the Spearman's rank correlation coefficient between these features. Results are shown in Table 4.2. *Feature5* is highly correlated with other features, especially *Feature4*. It is not surprising, if a gene has more links, or has more cancers genes in its neighborhood, it will be more efficient for this gene to communicate with other cancer genes. Correlation between *Feature3* and *Feature4* is also very high, because both features measure the connectivity of a gene in its 2-neighborhood. Since these features are highly correlated, some features may

Table 4.2: Correlation between the five topological features.

Feature No.	1	2	3	4	5
1	1.00	0.39	0.28	0.36	-0.74
2		1.00	0.41	0.46	-0.51
3			1.00	0.76	-0.54
4				1.00	-0.81
5					1.00

be redundant. Therefore, we investigated all the combinations of features to find fewer features and get higher accuracy.

4.3.3 Results of Classification with Different Combination of Features

For every feature combination, we performed 10-fold cross validation to assess the performance of both classifiers, SVM and MLP. For SVM, we used grid search to find the best parameters. The C parameter used in SVM was selected from a range of values of 2^α , with α from -2 to 10. The γ parameter used in SVM was selected from a range values of 2^β , with β from -3 to 5. For MLP, we used two-layer networks with the number of input nodes and output nodes as 5 and 1, respectively. For each training, the number of hidden nodes varies from 2 to 20 to get the best neural network structure. Learning rate and momentum were set at 0.01 and 0.9, respectively. In each training process, the iteration number was set to 50.

Table 4.3 shows results of classification, including mean and standard deviation of 20 accuracy values, according to 20 randomly selected negative sets. The corresponding parameters are also provided in this table. Figure 4.4 shows the performance of SVM

Figure 4.4: The performance of classifiers with varying number of features.

and MLP using varying number of features. As shown in Figure 4.4, for all possible combination of features, the performance of SVM was slightly better than the performance of MLP. For both SVM and MLP, 3-feature combination achieved the best accuracy. And the best performance for SVM was 73.9%, which was better than MLP with 73.3%.

4.3.4 The Best Combination of Features

Each feature can contribute to classification and prediction of cancer genes as can be seen from Table 4.3. When using each feature alone, the accuracy of classifiers were above 67.4%. However, due to the correlation between features, the performance of classifiers did not increase with the number of features increasing. Figure 4.3 shows that when feature number was three, both MLP and SVM got the highest accuracy.

Top three feature combinations for these two classification algorithms are the same, which are highlighted in Table 4.3. The best combination for both classification algo-

Table 4.3: Performance of classifiers with different combination of five features.

SN	Features	MLP		SVM	
		Accuracy	$\mathcal{N}_{\text{hidden}}$	Accuracy	(C, γ)
1	{1}	67.4 ± 1.5	3	67.4 ± 1.8	$(2^{10}, 2^3)$
2	{2}	70.4 ± 1.4	5	70.5 ± 0.7	$(2^{10}, 2^0)$
3	{3}	70.3 ± 1.5	5	70.5 ± 1.2	$(2^{-1}, 2^5)$
4	{4}	68.0 ± 1.6	14	68.7 ± 1.5	$(2^3, 2^3)$
5	{5}	69.9 ± 1.2	10	70.5 ± 1.1	$(2^1, 2^{-3})$
6	{1, 2}	70.0 ± 1.5	18	70.7 ± 1.4	$(2^4, 2^1)$
7	{1 3}	73.1 ± 1.2	7	73.4 ± 0.9	$(2^1, 2^5)$
8	{1, 4}	70.6 ± 1.5	13	69.3 ± 1.6	$(2^2, 2^3)$
9	{1, 5}	69.6 ± 1.4	13	70.4 ± 1.3	$(2^1, 2^{-1})$
10	{2, 3}	71.8 ± 1.1	14	71.8 ± 1.3	$(2^1, 2^5)$
11	{2, 4}	69.8 ± 1.4	6	70.7 ± 1.1	$(2^{-1}, 2^2)$
12	{2, 5}	70.3 ± 1.3	6	70.6 ± 1.4	$(2^{-1}, 2^2)$
13	{3, 4}	70.6 ± 1.6	10	70.6 ± 1.6	$(2^9, 2)$
14	{3, 5}	73.3 ± 1.0	6	73.5 ± 1.2	$(2^{10}, 2^{-2})$
15	{4, 5}	69.3 ± 1.1	5	70.0 ± 1.3	$(2^6, 2^4)$
16	{1, 2, 3}	72.9 ± 1.4	12	72.7 ± 1.3	$(2^4, 2^4)$
17	{1, 2, 4}	70.3 ± 1.7	5	70.9 ± 2.0	$(2^3, 2^4)$
18	{1, 2, 5}	69.9 ± 1.2	9	70.5 ± 1.1	$(2^6, 2^0)$
19	{1, 3, 4}	73.0 ± 1.3	20	73.2 ± 1.4	$(2^9, 2^{-1})$
20	{1, 3, 5}	73.3 ± 1.3	13	73.9 ± 1.2	$(2^{10}, 2^{-1})$
21	{1, 4, 5}	70.0 ± 1.7	14	69.9 ± 1.1	$(2^{10}, 2^1)$
22	{2, 3, 4}	71.3 ± 1.4	10	71.5 ± 1.3	$(2^4, 2^3)$
23	{2, 3, 5}	72.7 ± 1.4	12	73.1 ± 1.6	$(2^3, 2^5)$
24	{2 4 5}	69.8 ± 1.3	9	70.7 ± 1.3	$(2^0, 2^{-2})$
25	{3, 4, 5}	73.1 ± 1.3	15	73.4 ± 1.1	$(2^9, 2^{-2})$
26	{1, 2, 3, 4}	72.8 ± 1.5	6	72.9 ± 1.5	$(2^9, 2^0)$
27	{1, 2, 3, 5}	72.2 ± 1.4	16	72.8 ± 1.3	$(2^2, 2^3)$
28	{1, 2, 4, 5}	70.1 ± 1.6	7	71.1 ± 1.5	$(2^{-2}, 2^4)$
29	{1, 3, 4, 5}	72.6 ± 1.3	9	73.4 ± 1.7	$(2^8, 2^{-1})$
30	{2, 3, 4, 5}	72.4 ± 1.7	2	72.9 ± 1.4	$(2^0, 2^5)$
31	{1, 2, 3, 4, 5}	71.8 ± 1.3	3	72.5 ± 1.3	$(2^9, 2^0)$

gorithms is $\{1, 3, 5\}$. The second and third best combinations are $\{3, 5\}$ and $\{1, 3\}$. *Feature3* occurs in every combination in the top-three list, which indicates that this feature, 2N-index, is the most predictive feature. Using this feature alone, we can get the accuracy higher than 70%.

4.4 Summary

We investigated the topological features derived from PPI network for use in the characterization and classification of cancer genes. Statistical significance of these features indicates that cancer genes tend to be central in the PPI network and tend to interact with each other. Classifiers based on single features are higher than 67.4%, denotes that all the five features can contribute to the discrimination between cancer genes and other genes. But problem is that the performance of classifier does not significantly increase with the increasing of feature numbers.

We investigated all the combinations of features and found that the best feature combination is $\{1, 3, 5\}$. Among three features, *Feature1* measures centrality of a node in the graph, *Feature3* measures proportion of cancer genes in its 2-neighborhood, and *Features5* measures the communication efficiency of cancer genes. Based on this combination, the accuracy for SVM classifier was 73.9%. As we can see from Table 4.2, *Feature1* and *Feature5* are highly correlated, the performance of two-feature combinations $\{1, 3\}$ and $\{3, 5\}$ are similar to $\{1, 3, 5\}$. We also found that *Feature3*, 2N-index is the most predictive feature, that is, the proportion of cancer genes around one gene's 2-neighborhood is an important indicator for prediction. Using this feature alone, we can get the accuracy higher than 70%. In addition, we compared two kinds of classification

algorithms, and found that SVM is slightly better than MLP on all possible combination of features.

In comparison with sequence-based features, PPI network have some advantages. Firstly, PPI has better interpretation of gene function than DNA sequence, because genes can not perform functions, they have to be translated to proteins. Secondly, Proteins always perform functions by interacting with each other. Thirdly, using PPI network as a whole, we can investigate cancer genes in a higher level, e.g. to investigate the modularity nature of cancer genes. Therefore, using only two or three features derived from PPI network, we can get accuracy as high as 73.9%, which is comparable to previous method [80]. However, there are some drawbacks for using PPI network alone to predict cancer genes. Firstly, the PPI network cover only 9,019 genes, that is far from complete. Secondly, PPI network may bias to best studied genes. Furney et al. [80] combined different biological data sources, gene sequence data and gene functional annotation data, and obtained slightly better results than ours. If we combine other data sources, we expect to achieve better results.

Chapter 5

Rank Aggregation-based Data Integration Models for Candidate Gene Prioritization

Several integration algorithms have been proposed to prioritize candidate genes [22, 29–32], however most of them access no more than two data sources. Recently two integration frameworks have been proposed to integrate multiple data sources for the prioritization of candidate genes. One is a rank aggregation based integration (RABI) tool proposed by Aerts et al. [33], named as ENDEAVOUR. The other is a kernel based integration algorithm, named MKL (Multiple Kernels Learning) [34]. De Bie et al. [34] reported that MKL performed better than ENDEAVOUR.

In this chapter, we proposed several RABI models for candidate gene prioritization. We first rank candidate genes based on each of the four genomic data sources, PPI (Protein-Protein Interaction) data and three sub-ontologies of Gene Ontology (GO), using One-Class Support Vector Machine (1CSVM) and RWR (Random Walk with Restart) algorithms. Then we use the N-Dimensional Order Statistics (NDOS) algorithm to combine the four rank lists into one list, which is used as the final prioritization of the

set of candidate genes. Computational models were evaluated in terms of AUC value, using 36 diseases as the benchmark data. The proposed RWR+NDOS and 1CSVM models achieved AUC values of 85.1% and 84.4%, better than the Multiple Kernels Learning (MKL) method which gave an AUC of 79.3%. Next, we proposed another efficient rank aggregation algorithm named Discounted Rating System (DRS). The proposed RWR+DRS and 1CSVM+DRS models achieved AUC values of 87.7% and 85.4%, higher than the NDOS-based models.

5.1 Introduction

How to effectively integrate heterogeneous data sources to obtain reliable prioritization of candidate genes is a major challenge. Notably, there are two integration methods ENDEAVOUR [33] and MKL [34]. Aerts et al. [33] developed the tool ENDEAVOUR for candidate gene prioritization, in which nine data sources were integrated, e.g., sequence data, gene annotation data, etc. They obtained a rank list of candidate genes according to their similarity to the known disease genes based on each data source. In the subsequent step, these rank lists were integrated into one list by making use of the NDOS algorithm [35]. De Bie et al. [34] proposed an MKL algorithm to integrate multiple data sources. They used 1CSVM and the combined kernel to prioritize the set of candidate genes. They reported that MKL performed better than ENDEAVOUR proposed by Aerts et al. [33].

In this chapter, we focus on RABI models for the prioritization of set of candidate genes. As shown in Figure 5.1, there are two stages in the rank aggregation model, the ranking procedure and the combination of rank lists. In the first stage, the set of

Figure 5.1: Two stages of the RABI model '1CSVM+NDOS'. In the first stage, we use 1CSVM ranking algorithms to prioritize candidate genes. In the second stage, NDOS is used to combine multiple rank lists into one list.

candidate genes are ranked based on each data source. In the second stage, rank lists are combined into one list. In this work, we attempt to improve both stages.

The improvement in stage 1 is as follows. We used two effective ranking algorithms, 1CSVM and RWR algorithm, to prioritize the set of candidate genes in the ranking procedure. And then rank lists of the set of candidate genes are combined into one list using the NDOS algorithm. On the benchmark data of 36 diseases (Dataset #11), the RWR+NDOS and 1CSVM+NDOS models achieved the AUC value of 85.1% and 84.4%, respectively. Both models performed better than the MKL algorithm, which achieved the AUC value of 79.3%.

To improve the second stage, i.e., the combination procedure, we proposed a new rank aggregation algorithm, named Discounted Rating System (DRS). It is inspired by Discounted Cumulated Gain (DCG) score, which is widely used to evaluate the ranking results of document retrieval [39]. DCG score is used to measure the effectiveness of web search engine algorithms. Lower ranked documents are penalized according to its ranking position. Therefore the DCG value of a searching result is dominated by the relevance of top ranked documents. In the proposed DRS algorithm, one candidate gene is given a set of discounted rating scores, each of which corresponds to one data source. Then, a combined score is calculated for the candidate gene, which is the mean value of discounted rating scores corresponding to each data source. On the benchmark data, the RWR+DRS and 1CSVM+DRS model achieved AUC values of 87.7% and 85.4%, respectively. Therefore DRS-based integration performed better than NDOS-based integration and the MKL algorithm on the benchmark data. In addition, the RWR

algorithm is more preferable than the 1CSVM model in the prioritization of candidate genes.

The rest of this chapter is organized as follows. We begin with the description of the ranking algorithm used in this work, the 1CSVM algorithm [58] and RWR algorithm [24], then introduce the rank aggregation algorithm NDOS [35] and propose the new rank aggregation algorithm named DRS. We evaluate performance of these algorithms on the benchmark data of 36 diseases. The last section concludes this chapter with a brief summary.

5.2 Methods

In this section, we introduce two ranking algorithms, 1CSVM and RWR. Then we introduce the ranking aggregation algorithm NDOS and propose the new rank aggregation algorithm named DRS. At the end of this section, we introduce the evaluation method leave-one-out cross validation (LOO-CV).

5.2.1 Ranking Based on 1CSVM

Schölkopf et al. [58] extends the binary SVM classification scheme into one-class learning by mapping the training data that contains just one class into a high-dimensional Hilbert space via a kernel function. The objective of 1CSVM is to find a hyper-plane that best separates the training data from the rest. For example, in Figure 5.2, data points corresponding to filled circles denote known disease genes, which are used as training genes. Other circles denote candidate genes. The 1CSVM attempt to find a hyper-plane that best separates the disease genes from others. Schölkopf et al. [58] proposed to find

Figure 5.2: The linear function learned by 1CSVM to separate disease genes from other genes. Filled circles denote known disease genes and other circles denote candidate genes.

the optimal hyper-plane by maximizing the perpendicular distance from the origin, and ensuring all the training points are above the hyper-plane.

Let X be the feature matrix of training genes (the number of $\mathcal{N}_{dg}$ disease genes), the i -th row of which $\mathbf{x}_i^T$ represents the feature vector of gene g_i . Let

$$\mathcal{K} = XX^T \quad (\text{Eq. 5.1})$$

represent the kernel matrix of the training genes. The i -th row and j -th column of $\mathcal{K}$ can be viewed as similarity between g_i and g_j given by the kernel function

$$k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \mathbf{x}_j^T. \quad (\text{Eq. 5.2})$$

Suppose the optimal hyper-kernel is found to be

$$\mathbf{w} \mathbf{x}^T = \rho, \quad (\text{Eq. 5.3})$$

where $\mathbf{w}$ is the weight of the hyper-plane. Then the function

$$f(\mathbf{x}) = \mathbf{w} \mathbf{x}^T - \rho. \quad (\text{Eq. 5.4})$$

can be used to prioritize test genes (unfilled circle in Figure 5.2). The value of $f(\mathbf{x})$ measures the distance from the origin in the direction of the hyper-plane, the larger $f(\mathbf{x})$, the higher is the rank position of this test gene in the prioritization.

To find the optimal hyper-plane, a formal definition of the optimization problem is given as

$$\begin{aligned} \max_{\mathbf{w}} & \left(\frac{\|\mathbf{w}\|^2}{2} + \rho \right) \\ \text{s.t.} & \mathbf{w}\mathbf{x}_i^T \geq \rho \quad (\forall i). \end{aligned} \quad (\text{Eq. 5.5})$$

Using the Lagrange multiplier α , we introduce a Lagrangian

$$L(\rho, \mathbf{w}, \alpha) = \frac{\|\mathbf{w}\|^2}{2} + \rho + \sum_{i=1}^{\mathcal{N}_{dg}} \alpha_i (\mathbf{w}\mathbf{x}_i^T - \rho), \quad (\text{Eq. 5.6})$$

and set the derivatives with respect to the primal variables ρ and $\mathbf{w}$ equal to zero, yielding

$$\mathbf{w} = \alpha X \quad (\text{Eq. 5.7})$$

$$\text{s.t.} \quad \sum_{i=1}^{\mathcal{N}_{dg}} \alpha_i = 1, \quad \alpha_i \geq 0 \quad (\forall i). \quad (\text{Eq. 5.8})$$

α_i is the weight corresponding to the training data point $\mathbf{x}_i$. The training data points with $\alpha_i > 0$ are called support vectors. Substituting Eq. 5.7 and Eq. 5.8 into Eq 5.6 and using Eq 5.1, one can obtain the dual problem of Eq 5.5

$$\begin{aligned} \min_{\alpha} & (\alpha^T \mathcal{K} \alpha) \\ \text{s.t.} & \sum_{i=1}^{\mathcal{N}_{dg}} \alpha_i = 1, \quad \alpha_i \geq 0 \quad (\forall i), \end{aligned} \quad (\text{Eq. 5.9})$$

from which the value of α is obtained, and thereby the value of $\mathbf{w}$ and ρ is obtained.

Substituting Eq. 5.7 into Eq. 5.4, the prioritization function $f(\mathbf{x})$ can be given by

$$f(\mathbf{x}) = \sum_{i=1}^{\mathcal{N}_{dg}} \alpha_i k(\mathbf{x}, \mathbf{x}_i) - \rho. \quad (\text{Eq. 5.10})$$

After training the 1CSVM model, we use Eq.5.10 to calculate the $f(\mathbf{x})$ value for each of the candidate genes. Then the set of candidate genes are ranked based on the corresponding $f(\mathbf{x})$ values.

5.2.2 Random Walk with Restart (RWR)

A random walk on a graph is a special case of a Markov chain [87]. It simulates a random walker walking on the graph. Let $G(V, E)$ be a graph, i.e., a gene network, with $\mathcal{N}_g$ number of nodes, $V = \{g_1, g_2, \dots, g_{\mathcal{N}_g}\}$. Let $E = \{e(i, j) | \text{if node } g_i \text{ is connected with node } g_j\}$ be the set of edges and $A_{(\mathcal{N}_g \times \mathcal{N}_g)}$ be the adjacency matrix of the graph.

Suppose the random walker is currently at node g_i , he can walk to any node connecting to node g_i , i.e., its neighbors. The random walker moves from node g_i to node g_j with a transition probability, which is determined by the topology of the graph. Figure 5.3 shows an unweighted graph with nine nodes, and the random walker is on node g_5 . He may move to node g_3 , g_6 or g_9 with a probability of $\frac{1}{3}$.

Based on the topology of the gene network, the transition matrix $M_{(\mathcal{N}_g \times \mathcal{N}_g)}$ is calculated, in which $M_{i,j}$ denotes its (i, j) -th element, representing the probability of transition from node i to node j . The calculation of $M_{i,j}$ is given by

$$M_{i,j} = \begin{cases} A_{i,j}/d(i), & \text{if } e(i, j) \in E \\ 0, & \text{otherwise,} \end{cases} \quad (\text{Eq. 5.11})$$

where $d(i)$ is the sum of the i -th column in A . Let $\mathbf{p}_s = [p_s(1), p_s(2), \dots, p_s(\mathcal{N}_g)]^T$ be a vector, in which $p_s(i)$ holds the probability of finding the random walker at node g_i at step s . The probability at step $s + 1$ is given by

$$\mathbf{p}_{s+1} = M^T \mathbf{p}_s, \quad (\text{Eq. 5.12})$$

Figure 5.3: Illustration of RWR algorithm. The random walker is now at node g_5 , which has three neighbors g_3 , g_6 or g_9 . Therefore, he may choose to move to any of the neighbors of g_5 with a probability of $\frac{1}{3}$.

where M^T is the transpose matrix of M . In the random walk algorithm, $\mathbf{p}_0$ is given such that any element in $\mathbf{p}_0$ is between 0 and 1, and the sum of elements in $\mathbf{p}_0$ is 1. Given any $\mathbf{p}_0$ value, the above equation will converge to $\mathbf{p}_\infty$ as s goes to infinity [88]. The value of $p_\infty(i)$ indicates the connectivity of node g_i in the graph. A large $p_\infty(i)$ value indicates that g_i is highly connected in the graph, which means the importance of node g_i .

Random walk with restart (RWR) is a variant of the random walk algorithm. It can be used as a ranking algorithm, given a set of source nodes [24]. The random walker starts from one of the source nodes. At any node on the graph, he moves to a randomly selected immediate neighbor or goes back to any of the source nodes with a probability of γ . The RWR algorithm is given by

$$\mathbf{p}_{s+1} = (1 - \gamma)M^T \mathbf{p}_s + \gamma \mathbf{p}_0, \quad (\text{Eq. 5.13})$$

where $\mathbf{p}_0$ is the initial probability vector.

Figure 5.4: Candidate gene prioritization using random walk with restart (RWR) algorithm, taken from [24].

In this work, nodes corresponding to known disease genes are used as source nodes for the RWR algorithm. The initial probability vector $\mathbf{p}_0$ is constructed such that equal probabilities are assigned to all the source nodes with the sum of the probabilities equals to 1. This is equivalent to letting the random walker start from any of the source nodes with equal probability. The i -th element of $\mathbf{p}_0$ is given by

$$p_0(i) = \begin{cases} 1/\mathcal{N}_{dg}, & \text{if node } g_i \text{ is source node} \\ 0, & \text{otherwise,} \end{cases} \quad (\text{Eq. 5.14})$$

where $\mathcal{N}_{dg}$ is the number of known disease genes, i.e., the number of source nodes. In Figure 5.3, g_3 and g_6 are source nodes, i.e., $\mathcal{N}_{dg} = 2$, therefore $\mathbf{p}_0 = [0 \ 0 \ 0.5 \ 0 \ 0 \ 0.5 \ 0 \ 0 \ 0]^T$.

Eq. 5.13 will converge to $\mathbf{p}_\infty$ as s goes to infinity [88]. This is obtained when the difference between $\mathbf{p}_s$ and $\mathbf{p}_{s+1}$ falls below 10^{-10} . The difference between $\mathbf{p}_s$ and $\mathbf{p}_{s+1}$ is given by

$$\delta = \sum_{j=1}^{\mathcal{N}_g} (|p_{s+1}(j) - p_s(j)|). \quad (\text{Eq. 5.15})$$

The probability $p_\infty(i)$ is the probability of finding the random walker at node g_i in the steady state. If g_i is well connected to source nodes, it is easy for the random walker to visit this node. Therefore, the value of $p_\infty(i)$ gives a measure of proximity between g_i and source nodes. If $p_\infty(i) > p_\infty(j)$, then node g_i is more proximate to source nodes than node g_j .

Algorithm 5.1 Random Walk with Restart (RWR) algorithm.

Input:

$A_{(\mathcal{N}_g \times \mathcal{N}_g)}$ /* The adjacency matrix of the graph, where $\mathcal{N}_g$ is the number of nodes in the graph. */
 $\mathbf{p}_0$ /* The initial probability vector */
 γ /* The probability of coming back to source nodes at each step in RWR */

Output:

$\mathbf{RL}_{cdg}$ /* The rank list of candidate genes.*/;
1: using Eq. 5.11 to calculate the transition matrix M .
2: $\mathbf{p}_s = \mathbf{p}_0$; $\delta = 1$
3: **while** $\delta > 10^{-10}$ **do**
4: $\mathbf{p}_{s+1} = (1 - \gamma)M^T \mathbf{p}_s + \gamma \mathbf{p}_0$;
5: $\delta = \sum_{j=1}^{\mathcal{N}_g} (|p_{s+1}(j) - p_s(j)|)$;
6: $\mathbf{p}_s = \mathbf{p}_{s+1}$;
7: **end while**
8: $\mathbf{p}_\infty = \mathbf{p}_{s+1}$;
9: obtain $\mathbf{RL}_{cdg}$, the rank list of candidate genes, from $\mathbf{p}_\infty$.
10: **return** $\mathbf{RL}_{cdg}$;

The schematic diagram of RWR algorithm for the prioritization of candidate genes is shown in Figure 5.4. Firstly, a set of known disease genes and candidate genes are mapped to the gene network. Secondly, nodes corresponding to known disease genes are

used as source nodes to run the RWR algorithm. Then, each candidate gene is given a probability value in the vector $\mathbf{p}_\infty$. Finally, a set of candidate genes are prioritized based on the probability value. The higher the value, the higher prioritization position is given. The pseudocode of the RWR algorithm is shown in Algorithm 5.1.

5.2.3 Multiple Kernels Learning (MKL)

Kernel methods exhibit good classification performance, and they can conveniently be adapted to the problem at hand by designing appropriate kernel functions to represent prior knowledge. To integrate multiple data sources, a kernel representation for each data source is generated. Let N be the number of data sources to be integrated and $\mathcal{N}_{dg}$ be the number of disease genes. Using Eq. 5.1, a kernel matrix $(\mathcal{K}_i)_{(\mathcal{N}_{dg} \times \mathcal{N}_{dg})}$ is calculated for data source i , $i = 1, 2, \dots, N$. The kernel matrix is a representation or view on the similarity between training genes (known disease genes). And then, N kernels are combined into one by the equation below:

$$\begin{aligned} \mathcal{K} &= \sum_{i=1}^N \mu_i \mathcal{K}_i \\ s.t. \quad &\sum_{i=1}^N \mu_i = 1, (\mu_i \geq 0) \end{aligned} \tag{Eq. 5.16}$$

where μ_i is the weight for data source i . Weights of different data sources could be tuned by expert knowledge on individual diseases or by solving a convex optimization [59].

De Bie et al. [34] extended the MKL method, proposed by Lanckriet et al. [59], to one-class problem. In the same approach as Lanckriet et al. [59], they proposed to convexly combine the N different kernels, so as to maximize the margin ρ between the

hyper-plane and the origin data. Similar to Eq. 5.9, the optimization problem with the summarizing kernel can be given by

$$\begin{aligned}
 & \min_{\alpha} (\alpha^T \mathcal{K} \alpha) \\
 & s.t. \sum_{i=1}^{\mathcal{N}_{dg}} \alpha_i = 1, \quad \alpha_i \geq 0 \quad (\forall i) \\
 & \text{where } \mathcal{K} = \sum_{j=1}^{\mathcal{N}_{ds}} \mu_j \mathcal{K}_j \quad (\mu_j \geq 0, \sum_{j=1}^{\mathcal{N}_{ds}} \mu_j = 1).
 \end{aligned} \tag{Eq. 5.17}$$

It is equivalent to the following quadratically constrained linear program problem

$$\begin{aligned}
 & \min_{\alpha} \tau \\
 & s.t. \sum_{i=1}^{\mathcal{N}_{dg}} \alpha_i = 1, \quad \alpha_i \geq 0, \\
 & \tau \geq \alpha^T \mathcal{K}_j \alpha \quad (1 \leq j \leq N).
 \end{aligned} \tag{Eq. 5.18}$$

This optimization problem can be solved using MATLAB package SeDuMi [89]. After optimization, the combined kernel and value of α could be determined. Therefore, the prioritization function Eq. 5.10 is obtained. Then, candidate genes can be ranked based on the corresponding $f(\mathbf{x})$ values.

In addition, De Bie et al. [34] found that, in some cases, over-fitting may occur by the optimization problem, due to the large freedom of μ_i allowed (i.e., $0 \leq \mu_i \leq 1$). Therefore, they narrow down the scope of μ_i by introducing a lower bound lb ($0 \leq lb \leq 1$) for the weight of each individual kernel [34]. That is, the weight for each individual kernel is at least lb/N , ($\mu_i \geq lb/N$). Increasing lb means to reduce the scope of searching for the optimal $\mathcal{K}$ and hence reduces the risk of over-fitting. For $lb = 1$, Eq. 5.16 becomes the average of the normalized kernels.

Figure 5.5: The schematic diagram of the NDOS algorithm.

5.2.4 N-dimensional Order Statistics (NDOS)

N-dimensional order statistics (NDOS) used in this chapter was proposed by [35]. It has been used by Aerts et al. [33] to combine multiple data source and prioritize disease genes [33]. Let us consider a set of candidate genes, given by $G_{cg} = \{g_1, g_2, \dots, g_n\}$. Let RL_j with size $(n \times 1)$ be the rank list of n candidate genes obtained by using data source j ($j = 1, 2, \dots, N$). The matrix $RL_{(n \times N)} = [RL_1, RL_2, \dots, RL_N]$ holds N rank lists of candidate genes obtained based on individual data sources. The (i, j) -th element of RL , $RL_{i,j}$ is the rank position of g_i based on data source j .

Figure 5.5 shows the schematic diagram of NDOS algorithm. It starts with a set of rank lists of candidate genes. Five steps of NDOS are shown below, and the pseudocode of NDOS is shown in Algorithm 5.2.

i) Step 1: Calculate the rank ratio for g_i in RL_j , $RR_{i,j} = \frac{RL_{i,j}}{\max(RL_j)}$.

ii) Step 2: Sort the rank ratios for g_i in ascending order and store in a variable ζ ,

$$\zeta = \text{sort}([RR_{i1}, RR_{i2}, \dots, RR_{iN}]), \text{ such that } \zeta_1 \leq \zeta_2 \leq \dots \leq \zeta_N.$$

Algorithm 5.2 N-dimensional Order Statistics (NDOS).

Input:

$RL_{n \times N} = [RL_1, RL_2, \dots, RL_N]$ /* A matrix, each column is a rank list of a set of candidate genes based on an individual data source. n is the number of candidate genes. N is the number of data sources. */

Output:

cRL /* The combined ranking of candidate genes, the size of the vector is $n \times 1$. */

- 1: **for** $j = 1$ to N **do**
 - 2: $RR_j = RL_j / \max(RL_j)$; /* Transform the rank list corresponding to each data source into a rank ratio list. */
 - 3: **end for**
 - 4: $RR_{n \times N} = [RR_1, RR_2, \dots, RR_N]$
 - 5: **for** $i = 1$ to n **do**
 - 6: $\zeta = \text{sort}([RR_{i1}, RR_{i2}, \dots, RR_{iN}])$; /* Sort the rank ratios for g_i in ascending order and store in a variable ζ */
 - 7: $V_0 = 1$;
 - 8: **for** $k = 1$ to N **do**
 - 9: $V_k = \sum_{j=1}^k (-1)^{(j-1)} \left[\frac{V_{k-j}}{j!} \right] \cdot [\zeta_{(N-k+1)}]^j$, ($1 \leq k \leq N$)
 - 10: **end for**
 - 11: $CS_i = N!V_N$;
 - 12: **end for**
 - 13: Prioritize candidate genes based on the CS_i , to obtain the combined rank list of candidate genes **cRL**. The smaller CS_i value, the higher is the rank given for g_i .
 - 14: **return cRL**;
-

iii) Step 3: Calculate CS_i , the combined score for g_i , by the following recursive formula:

$$\begin{aligned} CS_i &= Q(\zeta_1, \zeta_2, \dots, \zeta_N) \\ &= N! \sum_{j=1}^N (\zeta_{N-i+1} - \zeta_{N-i}) Q(\zeta_1, \zeta_2, \dots, \zeta_{N-i}, \zeta_{N-i+2}, \dots, \zeta_N), \end{aligned} \quad (\text{Eq. 5.19})$$

with $\zeta_0 = 0$ and $Q(\zeta_i) = \zeta_i (i = 1, 2, \dots, N)$.

iv) Step 4: Rank candidate genes based on the combined score to obtain cRL, the combined rank list of candidate genes. The smaller CS_i value, the higher rank position is given for g_i .

The combined score are calculated using Eq 5.20a-c.

$$\begin{cases} CS_i = N!V_N & (\text{Eq. 5.20a}) \\ V_k = \sum_{j=1}^k (-1)^{(j-1)} \left[\frac{V_{k-j}}{j!} \right] \cdot [\zeta_{(N-k+1)}]^j, \quad (1 \leq k \leq N) & (\text{Eq. 5.20b}) \\ V_0 = 1. & (\text{Eq. 5.20c}) \end{cases}$$

Here we give an example to calculate CS_i using Eq 5.20a-c. Given $\zeta = [0.1, 0.2, 0.3, 0.4]$, $N = 4$. The calculation of CS_i is as follows.

$$V_1 = (-1)^0 \cdot V_0 \cdot \zeta(4) = 0.4.$$

$$V_2 = (-1)^0 \cdot V_1 \cdot \zeta(3) + (-1)^1 \cdot \frac{V_0}{2!} \cdot [\zeta(3)]^2 = 0.075.$$

$$V_3 = (-1)^0 \cdot V_2 \cdot \zeta(2) + (-1)^1 \cdot \frac{V_1}{2!} \cdot [\zeta(2)]^2 + (-1)^2 \cdot \frac{V_0}{3!} \cdot [\zeta(2)]^3 = 0.0083.$$

$$V_4 = (-1)^0 \cdot V_3 \cdot \zeta(1) + (-1)^1 \cdot \frac{V_2}{2!} \cdot [\zeta(1)]^2 + (-1)^2 \cdot \frac{V_1}{3!} \cdot [\zeta(1)]^3 + (-1)^3 \cdot \frac{V_0}{4!} \cdot [\zeta(1)]^4 = 5.2 \times 10^{-4}.$$

$$CS_i = 4! \cdot V_4 = 0.0125.$$

5.2.5 Discounted Rating System (DRS)

Here we propose a new rank aggregation algorithm, named discounted rating system (DRS) to combine N rank lists $(RL_1, RL_2, \dots, RL_N)$ into one. The DRS algorithm

Figure 5.6: The schematic diagram of the DRS algorithm.

attempts to enhance the ranking of the gene that is ranked at the top position by one or more data sources. Our assumption is that the real disease gene is ranked somewhere in the top of the rank lists according to some data sources. Figure 5.6 shows the procedures of the DRS algorithm. Detailed description of the procedures are shown below.

- i) Step1: We transform the rank lists into rating lists as follows: The RL is divided into five equal-sized rate zones. For example, in an RL of 100 candidate genes, rate zone 1 corresponds to rank position 1 to 20, rate zone 2 corresponds to rank position 21 to 40, and similarly rate zone 5 corresponds to rank position 81 to 100. The rating for rate zone 1 is 5, rating for rate zone 2 is 4, ..., similarly the rating for rate zone 5 is 1. Let n represent the number of candidate genes and z represent the range of ratings. The rating of a gene g_i in RL_j is given by

$$rating_{i,j} = z - int(\frac{RL_{i,j}}{bin_size}), \quad (\text{Eq. 5.21})$$

where $bin_size = n/z$ is the size of the rate zone.

- ii) Step2: The discounted rating (dr) of g_i in RL_j is calculated as follows:

$$dr_{i,j} = \frac{rating_{i,j}}{\log_2(RL_{i,j} + 1)}. \quad (\text{Eq. 5.22})$$

Figure 5.7: Illustration of relationships between rank, rating and discounted rating. Here the rank ranges from 1 to 100 and the range of rating is 5.

The discounting function $\log_2(RL_{i,j} + 1)$ reduces the gene's rating as its rank increases, so that genes ranked near the top are more highlighted. Figure 5.7 shows the relationship between rank, rating and discounted rating, where z is set as 5. As can be seen from the figure, the dr score for the top ranked gene is much larger than other genes with the same rating.

- iii) Step 3: The combined score for g_i is calculated as the mean value of discounted ratings, which is given by

$$CS_i = \frac{1}{N} \sum_{j=1}^N dr_{i,j}. \quad (\text{Eq. 5.23})$$

- iv) Step 4: The candidate genes are ranked according to the combined score. The larger CS_i value, the higher is the rank position given to g_i .

The pseudocode of DRS is shown in Algorithm 5.3.

Algorithm 5.3 Discounted Rating System (DRS).

Input:

$RL_{(n \times N)}$ /* A matrix, each column is a rank list of n candidate genes based on an individual data source. N is the number of data sources. */
 z /*the range of ratings. */

Output:

cRL /* The combined ranking of candidate genes, the size of the vector is $n \times 1$. */
1: $\text{bin_size} = n/z$;
2: using Eq. 5.21 to calculate the rating of candidate gene i in data source j , $\text{rating}_{i,j}$.
3: using Eq. 5.22 to calculate the discounted rating of candidate gene i in data source j , $dr_{i,j}$.
4: **for** $i = 1$ to n **do**
5: $CS_i = \frac{1}{N} \sum_{j=1}^N dr_{i,j}$; /* combined score for candidate gene i . */
6: **end for**
7: Prioritize candidate genes based on the combined scores to find the combined rank position of candidate genes **cRL**. The larger CS_i value, the higher is the rank position for g_i .
8: **return cRL**;

Figure 5.8: An illustration of four steps in Discounted Rating System.

Considering 100 candidate genes, an illustrative example of DRS is provided in Figure 5.8. Figure 5.8(a) shows the input of DRS, rank list of 100 candidate genes obtained by using RWR based on each of the four individual data sources. RL_1 , RL_2 , RL_3 and RL_4 were obtained using data from PPI, BP, CC and MF, respectively. Figure 5.8(b) and Figure 5.8(c) shows ratings and discounted ratings, respectively. The combined score of candidate gene is shown in Figure 5.8(d). And the output of DRS is shown in Figure 5.8(e).

For example, the gene g_8 is ranked at 4th, 2nd, 13rd and 3rd in RL_1 , RL_2 , RL_3 and RL_4 respectively. That is, for g_8 , $RL_{8,1} = 4$, $RL_{8,2} = 2$, $RL_{8,3} = 13$ and $RL_{8,4} = 3$. After transformed into ratings, they are all 5. In the next step, their dr values are 2.15, 3.15, 1.31, and 2.5, respectively. The combined score for g_8 is 2.28. Finally, each candidate gene has a combined score, based on which, all the candidate genes are ranked. The most likely real disease gene is g_8 , because it is finally ranked at the top. If we select a threshold at 3, in the combined rank list, since g_3 and g_{100} are ranked 2nd and 3rd, they may also be real disease genes. Thus, depending on the threshold value the shortlisted candidate genes can be given.

5.2.6 Leave-One-Out Cross Validation (LOO-CV)

To assess the performance of the prioritization methods used in our study, we use leave-one-out cross validation (LOO-CV) strategy proposed by Aerts et al. [33], which is illustrated in Figure 5.9. LOO-CV is suitable for this problem because of small-sized training set, i.e., only few disease genes are available for some diseases. In a small dataset, LOO-CV gives almost unbiased estimation of the performance, while k-fold

Figure 5.9: Procedures of leave one out cross validation.

cross-validation has a high bias [90]. For a given disease, suppose there are $\mathcal{N}_{dg}$ disease genes. We hold out one gene from the set of disease genes. We obtain 99 genes closest to the held-out gene in the chromosome. The 99 genes and the held-out gene is used as test set. The rest $\mathcal{N}_{dg} - 1$ genes are used as training set to train the ranking model (1CSVM or RWR etc.). We use the trained model to rank the set of test genes. Ideally, the held-out gene in the test set will be on top of the rank list of 100 test genes. Next, another disease gene of the selected disease is held-out. And above procedures are repeated, until all the disease genes are prioritized. Therefore, for each disease gene of the selected disease, we obtain a rank list of 100 test genes. Now we have $\mathcal{N}_{dg}$ rank lists, one for each disease gene.

Next, we calculate sensitivity and specificity values of the selected disease at varying thresholds and plot the receiver operating characteristic (ROC) curve, and use the AUC

Figure 5.10: Calculation of the AUC value.

value as a qualitative evaluation measure (Figure 5.10). Given a threshold, Sensitivity is calculated as

$$\text{Sensitivity} = \frac{\mathcal{N}_{TP}}{\mathcal{N}_{TP} + \mathcal{N}_{FN}}, \quad (\text{Eq. 5.24})$$

where $\mathcal{N}_{TP}$ is the number of known disease genes ranked above and below the threshold, and $\mathcal{N}_{FN}$ is the number of known disease genes ranked below the threshold. The value of 1 - specificity is calculated as

$$1 - \text{Specificity} = \frac{\mathcal{N}_{FP}}{\mathcal{N}_{FP} + \mathcal{N}_{TN}}, \quad (\text{Eq. 5.25})$$

where $\mathcal{N}_{FP}$ is the number of non-disease genes ranked above the threshold, and $\mathcal{N}_{TN}$ is the number of non-disease genes ranked below the threshold.

Figure 5.10 gives an example of a disease associated with 11 disease genes. Using the procedure shown in Figure 5.9, a rank list of 100 test genes is obtained for each of 11 disease genes. In this case, the number of real disease genes is 11, therefore $\mathcal{N}_{TP} + \mathcal{N}_{FN} =$

11. The number of real non-disease genes is $11 \times 99 = 1089$, therefore $\mathcal{N}_{FP} + \mathcal{N}_{TN} = 1,089$.

Eleven rank lists are shown in Figure 5.10, in which known disease genes are shown as red squares. Out of 11 disease genes, 3 genes occupy $rank = 1$, 2 genes occupy $rank = 2$, 1 genes occupy $rank = 3$, and 3 genes occupy $rank = 4$ in their respective rank lists. Let us set the threshold as 1, then $\mathcal{N}_{TP} = 3$. Therefore, sensitivity is calculated as $3/11 = 0.27$. $\mathcal{N}_{FP} = 11 - 3 = 8$, therefore, '1 - specificity' is $8/1089 = 0.0073$. Next, set the threshold as 2. The sensitivity and '1-specificity' are calculated as 0.45 and 0.016, respectively. Pairs of (1-specificity, sensitivity) values are obtained with the threshold ranging from 0 to 100, as shown in the middle part of Figure 5.10. Then, the ROC curve is plotted as the sensitivity vs. 1 - specificity. Finally, the AUC value is used as a standard measure of the performance. The higher the AUC value, the better is the performance. For instance, an AUC value of 100% indicates that every held-out gene is ranked first.

We evaluate integration methods using 36 diseases as benchmark data, described in Section 3.3.3. To obtain a comprehensive comparison between these methods, we calculate both disease-specific AUC value and the overall AUC value for all the 36 diseases. In the process of calculating AUC, if the ranking lists corresponding to one disease are used, the calculated AUC is called disease specific AUC. If ranking lists corresponding to all the 36 diseases are used, the AUC is called overall AUC.

5.3 Experiments and Results

In this section, we evaluated the performance of integration algorithms using the 36 diseases as benchmark data. We firstly compared the performance of MKL and two RABI models, 1CSVM+NDOS and RWR+NDOS. Then we compared the NDOS and DRS algorithms. Finally, we investigated the parameters in the RWR+DRS model.

5.3.1 Datasets

In this work, we integrate four data sources, PPI network, BP, CC and MF. Benchmark data are 36 diseases obtained from Köhler et al.'s work [24], (Dataset #11). The number of genes in one disease, $\mathcal{N}_{dg}$ ranges from 6 to 47. In total, there are 497 disease genes involved. For each disease gene, we obtained a set of candidate genes (test genes in Figure 5.9), containing this gene and the 99 nearest genes in the chromosome. This was done based on the gene location information (Dataset #19).

5.3.2 Comparison Between MKL and the RABI Models

In the first, we investigate the performance of the MKL algorithm on the benchmark data (Dataset #11). As described in Section 5.2.3, we obtain a kernel matrix for training genes based on each data source. Then we use Eq. 5.18 to find the combined kernel and the value of α . Therefore the prioritization function $f(\mathbf{x})$ (Eq. 5.10) is obtained. Candidate genes are ranked based on the $f(\mathbf{x})$ value. For three data sources from GO (BP, CC, and MF), the kernel matrix is calculated using Eq. 3.8. That is the functional similarity between g_i and g_j is calculated as the dot product of corresponding weighted

Table 5.1: Performance of the MKL algorithm on 36 diseases in terms of overall AUC values.

lb	Overall AUC (%)
0	76.3
0.25	77.4
0.5	78.1
0.75	78.5
1	79.3

annotation vectors. For the PPI data, the kernel matrix is calculated using the diffusion kernel (Eq. 3.3).

5.3.2.1 Performance of the MKL Algorithm

After obtaining the kernel matrix, we run the MKL algorithm with five different lb values, 0, 0.25, 0.5, 0.75 and 1. As described in Section 5.2.3, the value of lb is the lower bound of μ_i , the weight for the i -th data source. Increasing lb value implies reducing the searching space for the combined kernel. In the extreme case of $lb = 1$, all the data sources are weighted equally. Results of the MKL algorithm are shown in Table 5.1 in terms of the overall AUC value. The calculation of overall AUC value is provided in Section 5.2.6. The performance of MKL increases with the increasing of lb value. The lower performance at lower lb value might be caused by the over-fitting problem in the optimization (Eq. 5.18). The disease-specific data AUC values are shown in Table A.1 in Appendix A.

5.3.2.2 Performance of NDOS-based Models

Here we investigate the performance of two RABl models, 1CSVM+NDOS and RWR+NDOS.

As shown in Figure 5.9, for one disease gene, we use each of the four data sources, i.e.,

Figure 5.11: ROC curve corresponding to MKL and NDOS-based Integration Models.

PPI, BP, CC and MF, to train the 1CSVM ranking model. The set of candidate genes are ranked using the trained 1CSVM model. Therefore, we obtain four rank lists of candidate genes, each of which corresponds to one data source. Then the rank lists are combined into one cRL (combined Rank List) using the NDOS algorithm. For each of the 497 disease genes in Dataset #11, we obtained a cRL. From these 497 cRLs, we plotted the overall ROC curve and obtained the overall AUC value as 84.4%.

Following the similar procedure, we rank candidate genes using the RWR algorithm based on the four data sources, and combine separate rank lists into one cRL using NDOS. Then, based on 497 cRLs obtained by the RWR+NDOS model, we plotted the

Figure 5.12: Comparison between 1CSVM+NDOS model and the MKL algorithm in terms of AUC values.

ROC curve and obtained the AUC value as 85.1%. Figure 5.11 shows the ROC curve of three integration methods, RWR+NDOS, 1CSVM+NDOS and MKL. The curves corresponding to RWR+NDOS and 1CSVM+NDOS are above the curve corresponding to the MKL algorithm, which means RABI models perform better than MKL.

We calculate the disease-specific AUC values for 36 diseases, shown in Appendix Table A.2. We use Figure 5.12 to illustrate the performance of MKL and 1CSVM+NDOS model on 36 diseases. The coordinate of each data point shows the AUC values on one disease. The x-coordinate is the AUC value of MKL and the y-coordinate is the AUC value of the 1CSVM+NDOS. Therefore, a data point above the line $y = x$ represents a disease, on which the 1CSVM+NDOS model performs better than MKL. As can be seen from Figure 5.12, the 1CSVM+NDOS model performs better than MKL on most of the 36 diseases.

Figure 5.13: Comparison between RWR+NDOS model and the MKL algorithm in terms of AUC values.

Similarly, the RWR+NDOS model also performs better than MKL, as shown in Figure 5.13. On the one hand, for one disease, the size of training set (known disease genes) is pretty small. On the other hand, there are many parameters in MKL (α_i and μ_j in Eq. 5.17, Page 74) to be tuned. Over-fitting generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations. The lower performance of MKL might be caused by over-fitting problem.

To find the statistical significance of the difference between integration methods, we performed Wilcoxon signed-rank (WSR) test [91] between AUC values of 1CSVM+NDOS and MKL. WSR test is a non-parametric alternative to the pair-wise t-test. It can be used to test the difference between pair-wised samples, when the population cannot be assumed to be normally distributed. The null hypothesis is that there is no significant difference between two methods. The p value of 7.73×10^{-4} , obtained from the WSR

Figure 5.14: Comparison between 1CSVM+NDOS model and the RWR+NDOS model in terms of AUC values.

test, indicates that the 1CSVM+NDOS model performs significantly better than the MKL algorithm. Similarly, the RWR+NDOS model performs significantly better than the MKL algorithm ($p = 4.32 \times 10^{-6}$).

5.3.2.3 Performance of Two Ranking Algorithms 1CSVM and RWR

Here we compared the performance of two ranking algorithm, RWR and 1CSVM. Figure 5.14 shows the comparison between RWR+NDOS and 1CSVM+NDOS based on AUC values for the 36 diseases. The lowest AUC value obtained for RWR+NDOS is 62.8%, while 1CSVM+NDOS obtained two AUC values lower than 60%. Therefore, the variance of AUC values obtained by RWR+NDOS is lower than 1CSVM+NDOS.

From Figure 5.11, we obtained the AUC value of RWR+NDOS and 1CSVM at 85.1% and 84.4%, respectively. Therefore, the RWR+NDOS model performed slightly bet-

ter than the 1CSVM model. As shown in Figure 5.11, when the specificity is high (the left part of the figure), the sensitivity of the RWR+NDOS model is higher than 1CSVM+NDOS model. It indicates that among top ranked genes (top 5, top 10, etc.), the RWR+NDOS model selects out more real disease genes. Therefore, the RWR+NDOS is more preferable than the 1CSVM+NDOS model, because in the prioritization of candidate genes we focus on the top ranked genes. There are two possible reasons why the RWR algorithm performed better than 1CSVM. One is the possible over-fitting problem of 1CSVM, the other is the kernel function used in SVM might not be the best one. However, in order to optimize the kernel function, we need to introduce more parameters, which may lead to over-fitting. Whereas, such problems do not occur in the RWR algorithm. It works well even if there is only one known disease gene.

5.3.3 Comparison Between DRS-based Integration and NDOS-based Integration

Here we investigate the performance of the proposed DRS algorithm, described in Section 5.2.5. We integrated the PPI network and gene function annotation information from BP, CC and MF.

In cross validation, for one disease gene, we obtained four rank lists using RWR or 1CSVM algorithm. Then four rank lists are merged into one cRL using DRS as illustrated in Figure 5.8. Finally, cRLs are used to plot the ROC curve and calculate the AUC value as described in Section 5.2.6. Figure 5.15(a) shows the ROC curve of RWR+DRS and RWR+NDOS, and Figure 5.15(b) shows the ROC curve of 1CSVM+DRS and 1CSVM+NDOS. In both cases, RWR ranking and 1CSVM ranking, the DRS-based integration achieved higher AUC values than the NDOS-based integration. Detailed overall

Figure 5.15: ROC curves of DRS-based integration and NDOS-based integration.

Table 5.2: Performance of five different integration models on 36 diseases in terms of overall AUC values.

Integration Models	Overall AUC (%)
MKL	79.3
1CSVM+NDOS	84.4
RWR+NDOS	85.1
1CSVM+DRS	85.4
RWR+DRS	87.7

AUC values for five integration methods is shown in Table 5.2. From Table 5.2, we may get three conclusions, in terms of overall AUC value, (i) RABI models performed much better than the MKL algorithm, (ii) DRS-based integration performed better than NDOS-based integration, and (iii) the ranking of RWR was better than 1CSVM.

In the next, we compared the performance of DRS-based integration and NDOS-based

Figure 5.16: Comparison between RWR+DRS model and the RWR+NDOS model in terms of AUC values.

Figure 5.17: Comparison between 1CSVM+DRS model and the 1CSVM+NDOS model in terms of AUC values.

integration in terms of disease-specific AUC values. All the disease-specific AUC values for 36 diseases are shown in Appendix Table A.2. Figure 5.16 shows the comparison between RWR+DRS and RWR+NDOS models. Among 36 diseases, the RWR+DRS model achieved higher AUC values in 20 of them than RWR+NDOS model. The comparison between 1CSVM+DRS and 1CSVM+NDOS models is shown in Figure 5.17. For most of the diseases, the performance of two methods are comparable, but 1CSVM+DRS achieved higher AUC values than 1CSVM+NDOS in 23 diseases. Comparing with WSR test [91], we found that the RWR+DRS is statistically significant better than RWR+NDOS ($p = 2.28 \times 10^{-2}$), and the 1CSVM+DRS is statistically significant better than 1CSVM+NDOS ($p = 2.72 \times 10^{-2}$).

NDOS selects out candidate genes which are highly ranked in most of the data sources, whereas DRS prefers candidate genes ranked at the top in one or two data sources.

The real gene might not be highly ranked in most of the data sources, because of the incompleteness of current genomic data sources. Even if the genomic data sources are complete, the real disease gene may not ranked well in most of the data sources. In addition, different data sources might be effective for different diseases. For one disease, the disease genes may interact with each other, whereas, for another disease, the disease genes may share the same molecular function.

5.3.4 Stable Performance of the RWR+DRS Model

Results in Table 5.2 show that the RWR+DRS model performs better than MKL and the other four integration models in terms of overall AUC value. In this section, we investigate whether or not the performance of this model is stable against parameter variations. There are two parameters in the RWR+DRS model, the restart probability γ in the RWR algorithm (Eq. 5.13) and the range of ratings z (Eq. 5.21) in the DRS algorithm. We investigate the effect of these two parameters on the performance of the RWR+DRS model.

To investigate the effect of γ in the RWR algorithm, we set γ at 0.6, 0.7, 0.8 and 0.9, and did LOO-CV for both the RWR+NDOS and RWR+DRS models. Results are shown in Table 5.3. For both models, the AUC values did not change much with the change of γ value. In addition, for each γ value, performance of DRS was better than NDOS.

As described in Section 5.2.5, there is only one parameter in the proposed DRS algorithm, the range of ratings z (Eq. 5.21). To investigate whether or not this parameter has any effect on the performance of DRS, we varied the range of ratings from 2 to 10

Table 5.3: Stable performance of RWR with variation of γ .

γ	AUC Value(%)	
	NDOS	DRS
0.6	85.3	87.9
0.7	85.1	87.7
0.8	85.2	87.9
0.9	85.1	87.8

Table 5.4: The range of ratings and AUC values.

The Range of Ratings	AUC value(%)
2	87.4
5	87.7
10	87.8

and calculated the AUC values of LOO-CV. As can be seen from Table 5.4, the variation of this parameter did not effect the results significantly.

5.3.5 Computational Efficiency of NDOS and DRS Algorithms

Here we show the computational efficiency of our proposed DRS algorithm. In this work, four data sources, PPI, BP, CC and MF, have been integrated to prioritize candidate genes. In cross validation, 100 test genes (candidate genes) have been considered. Therefore, for each disease gene, we obtain four rank lists, RL_1 , RL_2 , RL_3 and RL_4 , each of which holds ranks of 100 candidate genes. These rank lists are combined into one cRL using the rank aggregation algorithm, DRS or NDOS. There are 497 disease genes in the benchmark data of 36 diseases. Therefore, in the cross validation, the rank aggregation program is run 497 times. On a desktop with a 2.4GHz Intel processor and 1GB RAM, in

Table 5.5: Results on simulated new data sources.

Number of Data Sources	Computational Time(s)	
	NDOS	DRS
4	9.4	0.18
8	19.0	0.26
12	34.7	0.30
16	51.5	0.36
20	75.4	0.41

cross validation of 36 diseases, the computational time of the rank aggregation procedure using DRS and NDOS were found to be 0.18s and 9.4s, respectively. Both DRS and NDOS were implemented in MATLAB.

In the updated version of ENDEAVOUR [92], there are 20 data sources available for *H. sapiens*. Here we compare the performance of NDOS and DRS, with increasing the number of data sources. As shown in Figure 5.1, a rank list of the set of candidate genes is obtained based on each data source. In this work, we integrated four data source, therefore we obtained four rank lists of the set of candidate genes, RL_1 , RL_2 , RL_3 and RL_4 , as shown in Figure 5.8 (a). Here we simulate some prioritization results based new data sources, which means more rank lists are involved in Figure 5.8 (a). The prioritization result of candidate genes according to new data source is simulated as a rank list, randomly chosen from the set of rank lists from real data source $\{RL_1, RL_2, RL_3$ and $RL_4\}$. Then, simulated rank lists and four real rank lists were put together and fed into DRS and NDOS algorithm respectively, and the corresponding combined rank list was generated.

Computation time on simulated data is shown in Table 5.5. It is the computation time of the rank aggregation procedure on 497 disease genes of the benchmark data. When the number of data sources increasing, the computational time for NDOS dramatically increased, while it for DRS did not change much. When the number of data sources increases to 20, DRS was more than 180 times faster than NDOS.

5.4 Summary

In this work, we integrated four data sources, i.e., PPI, BP, CC and MF, to prioritize candidate genes.

In this chapter, we focused on the RABI model, which is shown in Figure 5.1. There are two stages: ranking of candidate genes based on individual data source to generate an RL of candidate genes, and in the second stage the RLs are combined to form a combined rank list. To improve the first stage, we used the 1CSVM and RWR algorithm to prioritize candidate genes and proposed two RABI models, 1CSVM+NDOS and RWR+NDOS. The performance of rank aggregation based models are compared with the MKL algorithm. On the benchmark data set, the RABI models achieved much higher AUC values than MKL. In addition, we found that the ranking algorithm RWR is slightly more preferable for the problem of candidate gene prioritization, than the 1CSVM algorithm.

We proposed a new rank aggregation algorithm, named discounted rating system (DRS). We compared the performance of the proposed DRS algorithm with NDOS on the benchmark data (Dataset #11). Results showed that the RWR+DRS model works better than other integration methods. In addition, the performance of the RWR+DRS is stable

with variation of parameters. We simulated several new data sources to compare the computational efficiency of DRS and NDOS. Results showed that DRS is computationally more efficient than NDOS with increasing number of data sources.

Chapter 6

Network-based Data Integration for Candidate Gene Prioritization

In this chapter, we focus on network-based integration (NBI) models for the prioritization of a set of candidate genes. We proposed an algorithm named Random Walk with Restart on Multiple networks (RWRM) to integrate four genomic data sources, e.g., Protein-Protein Interaction (PPI) network and three gene networks obtained from Biological Process (BP), Cellular Components (CC) and Molecular Function (MF) ontologies.

Candidate genes are usually prioritized based on their similarity to known disease genes. But, in case of one disease phenotype without any known disease genes, the set of candidate genes can not be prioritized. To solve this problem, Wu et al. [1] proposed a tool named CIPHER (discussed in Section 2.2.6) to integrate the PPI network and phenotype information obtained from the Online Mendelian Inheritance in Man (OMIM) database for the prioritization of candidate genes. Here, we proposed a novel heterogeneous network model, named Random Walk with Restart on Heterogeneous network (RWRH). We have shown using the leave-one-out cross validation (LOO-CV) that, the RWRH algorithm correctly identified 814 of 1,428 phenotype–gene relationships (PGRs)

(Dataset #16), while Wu et al. [1] identified only 709 of them.

At the end of this chapter, we proposed a Complex Heterogeneous Network (CHN) model, in which four genomic data sources (PPI network, BP, CC, and MF), and the phenotype information obtained from OMIM are integrated. We used Dataset #15 as benchmark data containing 3,871 PGRs. We successfully identified 2,105 of them, whereas the DRS-based and NDOs-based integration methods, respectively, could identify 2,008 and 2,048 number of relationships. This shows the better performance of the proposed CHN model.

6.1 Introduction

In molecular biology, there are many kinds of relationships among genes, including protein interaction, co-expression, activation, inhibitory, co-annotation and so on. With the huge amount of data produced by high throughput techniques, tens of thousands of biological relationships among genes have been discovered. Nowadays network based models are gaining popularity to represent these relationships. Several types of biological networks have been proposed: protein interaction network [27, 55], gene regulatory network [93, 94], metabolic network [95, 96], gene co-expression network [97, 98] and gene co-annotation network [99], etc. These networks describe the relationship among genes from different perspectives. Combining different networks will help better understand the relationships among genes. How to integrate multiple networks to prioritize a set of candidate genes remains a challenging problem. In this chapter, we proposed three NBI models to predict disease genes, RWRM, RWRH and the CHN model.

6.1.1 Random Walk with Restart on Multiple networks (RWRM)

In the first part of this chapter, we extended the RWR algorithm (given in Section 5.2.2) to multiple networks and proposed a RWRM algorithm to prioritize a set of candidate genes. The relationships between genes in a specific data source are represented by a gene network. We constructed several gene networks considering different data sources, and then merged these networks into a single network. We ran the proposed RWRM algorithm on the merged network. The transition probability is calculated as the expected value of transition probabilities from multiple networks. On the benchmark data of 36 genetic diseases (Dataset #11), the proposed RWRM algorithm achieved the AUC value of 89.4%, much higher than the kernel base integration [34] and the RABl models proposed in Chapter 5.

6.1.2 Random Walk with Restart on the Heterogeneous network (RWRH)

A set of candidate genes are usually prioritized by their similarity to known disease genes. A number of prioritization algorithms have been proposed, some are based on single data source, e.g., [18–26, 40, 41], others are integration algorithms e.g., [1, 29–34, 36]. We have described these methods in Chapter 2. Though these methods perform well, they still have some limitations. One limitation is the scope of application. For some diseases in OMIM [5], there is no known disease genes. In other words, no disease gene has already been associated with such kind of diseases. Therefore these algorithms are not applicable in this case. Another problem comes from the ambiguous boundary between different diseases. Clinical diseases often encompass a variety of phenotypes and biological

mechanisms, making it difficult to distinguish between similar diseases. Traditionally, diseases have been categorized on the basis of pathophysiology or on etiology, but often these characterizations break down and more *ad hoc* approaches are used, resulting in the celebrated debate between splitters and lumpers [100]. The ambiguous boundary between different diseases prevents the direct inference of gene-disease association. For example, the disease, Leber Congenital Amaurosis (LCA), turns out to be highly heterogeneous on a molecular basis, but these molecular subtypes appear clinically homogeneous [101]. Using all the LCA genes to prioritize a list of genes responsible to a subtype of LCA may not be correct.

Most recently, two methods have been proposed to identify phenotype–gene relationship (PGR) instead of finding the disease–gene relationship directly [1, 36]. The assumption of these two methods is that similar phenotypes are caused by functionally related genes [37]. Lage et al. [36] proposed a candidate complex based algorithm. They assign a candidate gene to protein complexes and then rank these complexes using phenotypic data. Finally, a set of candidate genes are ranked based on the phenotypes associated with the protein complexes. Wu et al. [1] employ a regression model, named CIPHER, to quantify the concordance between the candidate gene and the target phenotype. A set of candidate genes are then ranked by the concordance score. Detailed description of these two methods have been provided in Section 2.2.5 and 2.2.6. It was reported by Wu et al. [1] that CIPHER performed better than the candidate complex based algorithm proposed by Lage et al. [36] on the overlapped benchmark data.

In the second part of this chapter, we propose a RWRH algorithm to infer the PGRs. The RWRH algorithm is inspired by the co-ranking framework [102]. It ranks phenotypes

and genes at the same time. We construct a heterogeneous network by connecting the gene network and phenotype network using PGRs. Then, we extend the RWR algorithm to the heterogeneous network, using the target phenotype and corresponding disease genes as source nodes. In the prioritization of a set of candidate genes, we attempt to make better use of the phenotypic data. On benchmark data set (Dataset #16) the proposed algorithm performed better than the CIPHER algorithm [1]. In addition, the proposed RWRH algorithm achieved higher AUC values than the RWR on gene network only [24].

The RWRH algorithm was also used to discover the relationship between diseases. Considering one disease, the disease phenotypes and disease genes were used as source nodes to run the RWRH algorithm. Other phenotypes were ranked based on their relevance to the disease. Then the top ranked phenotype was used to identify disease associations. Recently, several large-scale studies have systematically evaluated the relationship between human genetic diseases. The method based on the overlap between disease genes [66] is limited by the relative paucity of knowledge of disease causing genes. The phenotypic similarity based methods [56, 103] may be affected by the standardization and quantification of phenotypic description [104]. However, the RWRH algorithm well captured the complementarity between PPI network and phenotype network. We found 18 associations that cannot be found by using gene network or phenotype network alone. Most of these disease associations were supported by various types of evidence from available literature.

6.1.3 Complex Heterogeneous Network (CHN)

Even though the RWRH algorithm performs well, it is necessary to integrate multiple genomic data sources to improve the performance, because the current PPI network covers only a small part of human protein interactions [105]. Finally, we proposed a CHN model to integrate multiple gene networks and the phenotype network for the prioritization of a set of candidate genes. We used Dataset #15 as benchmark data, containing 3,871 PGRs. We successfully identified 2,105 of them, whereas the DRS-based and NDOS-based integration methods, respectively, could identify 2,008 and 2,048 number of relationships. This shows the better performance of our proposed CHN model.

6.2 Methods

In this section, we first explain the proposed RWRM algorithm to integrate multiple gene networks. Then, we introduce a heterogeneous network model and the RWRH algorithm to integrate PPI network and the phenotype network obtained from OMIM. Finally, we propose the CHN model to integrate multiple gene networks and the phenotype network.

6.2.1 The RWRM Model

In real life, different kinds of relationships exist between entities, which can be represented as various networks. In the case of genes, there are many ways to define 'functional similarity' between genes, such as co-expression, co-annotation, and substrate-sharing in the metabolism network, etc. Based on each type of similarity, a corresponding gene network can be constructed. Thus a set of genes can be represented by different types of gene networks. In this section, we extend the RWR algorithm (given in Section 5.2.2) to

Figure 6.1: The illustration of graph merging. Three kinds of links denote three kinds of relationships obtained from different data sources.

multiple gene networks and propose an RWRM (Random Walk with Restart for multiple networks) algorithm to prioritize a set of candidate genes.

As illustrated in Figure 6.1, we merge multiple networks into one network. In Figure 6.1, (a), (b) and (c) are three different networks, with the same set of nodes, $\{g_1, g_2, g_3, g_4, g_5\}$. The merged network is shown in Figure 6.1(d). It contains all the nodes and edges in separate networks

Let $G(V, E^{(1)})$, $G(V, E^{(2)})$, $\dots$, $G(V, E^{(N)})$ be N networks with the same set of nodes $V = \{g_1, g_2, \dots, g_{N_g}\}$, and different edge sets $(E^{(1)}, E^{(2)}, \dots$ and $E^{(N)})$. The merged network is represented as $G(V, E)$, where $E = E^{(1)} \cup E^{(2)} \cup \dots \cup E^{(N)}$. Here $A \cup B$ represents the union of sets A and B .

There might be more than one link between a pair of nodes in the merged network,

each link coming from different networks. For example, there are two links between node g_1 and node g_2 , from Network1 and Network2, respectively. The random walker might move from g_1 to g_2 via any of the two links. The transition probability from node g_1 to node g_2 on the merged network is calculated as the expected value of the transition probabilities corresponding to each link between node g_1 and node g_2 .

Considering node g_i in the merged network, let N_i be the number of networks to which node g_i is associated (for example, in Figure 6.1(d), $N_1 = 2$, $N_2 = 3$ and $N_3 = 3$). For each of the N_i networks, we calculated the transition matrix using Eq. 5.11. Let $M_{(\mathcal{N}_g \times \mathcal{N}_g)}^{(k)}$ be the transition matrix of the network k and $M_{i,j}^{(k)}$ is the (i, j) -th element of the matrix. The transition probability from node g_i to node g_j is calculated as

$$M_{i,j} = \sum_{k=1}^{N_i} q^{(k)} M_{i,j}^{(k)} \quad (1 \leq i, j \leq \mathcal{N}_g) \quad (\text{Eq. 6.1})$$

where $q^{(k)}$ is the probability of selecting the k -th network. In this work, we assume the random walker to choose any network with equal probability ($q^{(k)} = \frac{1}{N_i}$). As an example, node g_1 in Figure 6.1 is involved in two networks, Network1 and Network2, i.e., $N_1 = 2$. The transition probability from node g_1 to node g_2 is calculated as $p(g_1 \rightarrow g_2) = \frac{1}{2} \times \frac{1}{2} + \frac{1}{2} \times 1 = 0.75$. Similarly $p(g_2 \rightarrow g_3) = \frac{1}{3} \times \frac{1}{2} + \frac{1}{3} \times \frac{1}{3} + \frac{1}{3} \times \frac{1}{2} = 0.44$. After calculation, we obtain the transition matrix M for the merged network in Figure 6.1 as

$$\begin{matrix} & g_1 & g_2 & g_3 & g_4 & g_5 \\ \begin{matrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \end{matrix} & \begin{pmatrix} 0 & 0.75 & 0.25 & 0 & 0 \\ 0.28 & 0 & 0.44 & 0.28 & 0 \\ 0.08 & 0.58 & 0 & 0.08 & 0.26 \\ 0 & 0 & 0.5 & 0 & 0.5 \\ 0 & 0.33 & 0.5 & 0.17 & 0 \end{pmatrix} \end{matrix}.$$

The pseudocode of the RWRM algorithm is shown in Algorithm 6.1.

Algorithm 6.1 The RWRM Algorithm.

Input:

$A_{(\mathcal{N}_g \times \mathcal{N}_g)}^{(1)}, A_{(\mathcal{N}_g \times \mathcal{N}_g)}^{(2)}, \dots, A_{(\mathcal{N}_g \times \mathcal{N}_g)}^{(N)}$ /* N is the number of gene networks; $\mathcal{N}_g$ is the number of nodes in each network. */
 $\mathbf{p}_0$ /* The initial probability vector ($\mathcal{N}_g \times 1$). */
 γ /*The probability of coming back to source nodes at each step in RWR. */

Output:

$\mathbf{RL}_{cdg}$ /* The rank list of candidate genes.*/;
 1: Calculate the transition matrix $M^{(k)}$ for the k -th gene network using Eq. 5.11.
 2: Calculate the transition matrix M for the merged gene network using Eq. 6.1.
 3: $\mathbf{p}_s = \mathbf{p}_0$; $\delta = 1$;
 4: **while** $\delta > 10^{-10}$ **do**
 5: $\mathbf{p}_{s+1} = (1 - \gamma)M^T \mathbf{p}_s + \gamma \mathbf{p}_0$;
 6: $\delta = \sum_{j=1}^{\mathcal{N}_g} (|\mathbf{p}_{s+1}(j) - \mathbf{p}_s(j)|)$;
 7: $\mathbf{p}_s = \mathbf{p}_{s+1}$;
 8: **end while**
 9: $\mathbf{p}_\infty = \mathbf{p}_{s+1}$;
 10: obtain $\mathbf{RL}_{cdg}$, the rank list of candidate genes, from $\mathbf{p}_\infty$. The larger $p_\infty(i)$ value, the higher is rank given for g_i .
 11: **return** $\mathbf{RL}_{cdg}$;

Here we give an example to describe how the RWRM model works. We set node g_4 as the source node and use RWR algorithm to find the most similar node in the merged network, shown in Figure 6.1(d). We expect node g_3 to be the most similar to node g_4 , because node g_3 is well connected to node g_4 in individual networks and the merged network. The initial probability $p_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T$. After running RWRM algorithm, the

stationary probability finally becomes $p_\infty = \begin{bmatrix} 0.01 & 0.04 & 0.13 & 0.71 & 0.06 \end{bmatrix}^T$. Leaving aside the source gene g_4 , therefore the rank position of g_1, g_2, g_3 and g_5 is $[4, 3, 1, 2]$. Node g_3 is found to have the highest stationary probability, therefore it is the most similar to the source node g_4 . If g_4 denotes the known disease gene and g_1, g_2, g_3 and g_5 denotes candidate genes, we may infer that g_3 possibly is a disease gene.

6.2.2 The RWRH Model

As shown in Figure 6.2, we constructed the heterogeneous network by joining three networks, namely, PPI network, phenotype network and gene-phenotype network. The way these networks are constructed is described in Chapter 3. In the heterogeneous network, nodes connecting the PPI network and phenotype network are named bridging nodes, and the other nodes are named internal nodes. In Figure 6.2, nodes $Ph2$, $Ph3$, g_2 , and g_3 are bridging nodes, the other nodes are internal nodes.

Suppose $A_{G(\mathcal{N}_g \times \mathcal{N}_g)}$, $A_{P(\mathcal{N}_{Ph} \times \mathcal{N}_{Ph})}$ and $B_{(\mathcal{N}_g \times \mathcal{N}_{Ph})}$ are adjacency matrices for gene network, phenotype network and the phenotype-gene network respectively, where $\mathcal{N}_g$ and $\mathcal{N}_{Ph}$ represent the number of genes and phenotype entities, respectively. The adjacency matrix of the heterogeneous network is represented as

$$A_H = \left[\begin{array}{c|c} A_G & B \\ \hline B^T & A_P \end{array} \right] \quad (\text{Eq. 6.2})$$

where B^T represents the transpose of B . The size of A_H is $(\mathcal{N}_g + \mathcal{N}_{Ph}) \times (\mathcal{N}_g + \mathcal{N}_{Ph})$.

Based on the above mentioned heterogeneous network, we propose the RWRH algorithm, which is an extension of RWR algorithm on the heterogeneous network. The RWRH algorithm is proposed to rank a set of candidate genes susceptible to a disease

Figure 6.2: Illustration of the heterogeneous network. Nodes connecting gene network and phenotype network are called bridging nodes. Other nodes are called internal nodes.

phenotype, which is named target phenotype. When prioritizing a set of candidate genes, the target phenotype node and associated disease gene nodes are used as source nodes. The random walker starts from any of the source nodes to run the RWRH algorithm. The set of candidate genes are ranked according to their proximity to the target phenotype. Suppose node g_1 , g_3 and g_5 denote candidate genes for phenotype node $Ph2$ in Figure 6.2, the phenotype node $Ph2$ and the gene node g_2 are used as source nodes in the RWRH algorithm.

Similar as in the RWR algorithm, the random walker moves to one of the immediate neighbors, or return to any of the source node with probability γ . However, when the random walker is on the bridging node, he may jump to another subnetwork with probability λ or move to other nodes in the current subnetwork with probability $1 - \lambda$. The parameter λ is called the jumping probability, which regulates the reinforcement between the PPI network and phenotype network. The transition from gene network

to phenotype network (vice verse) is called inter-subnetwork transition. The transition within the PPI network or the phenotype network is called intra-subnetwork transition.

The transition matrix of the heterogeneous network is represented as

$$M_H = \left[\begin{array}{c|c} (M_G)_{\mathcal{N}_g \times \mathcal{N}_g} & (M_{GP})_{\mathcal{N}_g \times \mathcal{N}_{Ph}} \\ \hline (M_{PG})_{\mathcal{N}_{Ph} \times \mathcal{N}_g} & (M_P)_{\mathcal{N}_{Ph} \times \mathcal{N}_{Ph}} \end{array} \right], \quad (\text{Eq. 6.3})$$

where M_G is the transition matrix of the gene network; M_P is the transition matrix of the phenotype network; M_{GP} is the transition matrix from the gene network to the phenotype network; M_{PG} is the transition matrix from the phenotype network to the gene network. The transition probability from node g_i to node Ph_j is given by

$$(M_{GP})_{i,j} = p(g_i \rightarrow Ph_j) = \begin{cases} \lambda \frac{B_{i,j}}{\sum_{j=1}^{\mathcal{N}_{Ph}} B_{i,j}}, & \text{if } g_i \text{ is a bridging node} \\ 0, & \text{otherwise.} \end{cases} \quad (\text{Eq. 6.4})$$

Here $(M_{GP})_{i,j}$ is the (i, j) -th element of matrix (M_{GP}) . Similarly, the transition probability from node Ph_i to node g_j is given by

$$(M_{PG})_{i,j} = p(Ph_i \rightarrow g_j) = \begin{cases} \lambda \frac{B_{j,i}}{\sum_{j=1}^{\mathcal{N}_g} B_{j,i}}, & \text{if } Ph_i \text{ is a bridging node} \\ 0, & \text{otherwise.} \end{cases} \quad (\text{Eq. 6.5})$$

The transition probability of the random walker's move from node g_i to node g_j is defined by $p(g_i \rightarrow g_j)$, which is an element of M_G at the i -th row and j -th column. It is given by

$$(M_G)_{i,j} = \begin{cases} (1 - \lambda) \frac{(A_G)_{i,j}}{\sum_{j=1}^{\mathcal{N}_g} (A_G)_{i,j}}, & \text{if node } g_i \text{ is a bridging node} \\ \frac{(A_G)_{i,j}}{\sum_{j=1}^{\mathcal{N}_g} (A_G)_{i,j}}, & \text{otherwise.} \end{cases} \quad (\text{Eq. 6.6})$$

Similarly, the element of M_P at i -th row and j -th column is the probability of the

random walker transition from node Ph_i to node Ph_j . It is given by

$$(M_P)_{i,j} = \begin{cases} (1 - \lambda) \frac{(A_P)_{i,j}}{\sum_{j=1}^{N_{Ph}} (A_P)_{i,j}}, & \text{if node } Ph_i \text{ is a bridging node} \\ \frac{(A_P)_{i,j}}{\sum_{j=1}^{N_{Ph}} (A_P)_{i,j}}, & \text{otherwise.} \end{cases} \quad (\text{Eq. 6.7})$$

The initial probability of gene network $\mathbf{u}_0$ is constructed such that equal probabilities are assigned to all the source nodes in the gene network, with the sum of the probabilities equal to 1. Similarly, the initial probability of phenotype network $\mathbf{v}_0$ is given. Suppose one disease is described as two phenotypes $Ph2$ and $Ph3$ in Figure 6.2. The RWRH algorithm can be used to predict new disease genes for this disease. In this case, $\mathbf{u}_0 = [0 \ \frac{1}{2} \ \frac{1}{2} \ 0 \ 0]^T$ and $\mathbf{v}_0 = [0 \ \frac{1}{2} \ \frac{1}{2} \ 0]^T$. The initial probability vector for heterogeneous network is represented as

$$\mathbf{p}_0 = \left[\frac{(1 - \eta)\mathbf{u}_0}{\eta\mathbf{v}_0} \right], \quad (0 < \eta < 1). \quad (\text{Eq. 6.8})$$

The parameter η , whose value is between 0 and 1, is used to weigh the importance of each sub-network. If η is 0.5, two sub-networks are equally weighted. If η is above 0.5, the random walker prefers to return to the phenotype source node, therefore the phenotype information is given more importance.

We substitute the transition matrix M and initial probability $\mathbf{p}_0$ into the iterative equation (Eq. 5.13). After some steps, the steady probability

$$\mathbf{p}_\infty = \left[\frac{(1 - \eta)\mathbf{u}_\infty}{\eta\mathbf{v}_\infty} \right] \quad (\text{Eq. 6.9})$$

is obtained. Then genes and phenotypes are ranked based on the steady probability $\mathbf{u}_\infty$ and $\mathbf{v}_\infty$, respectively. The larger the probability value, the higher is the rank position.

The pseudocode of the RWRH algorithm is shown in Algorithm 6.2.

Algorithm 6.2 The RWRH Algorithm.

Input:

$(A_G)_{\mathcal{N}_g \times \mathcal{N}_g}$ /* adjacency matrix for the gene network, where $\mathcal{N}_g$ represents the number of genes. */
 $(A_P)_{\mathcal{N}_{Ph} \times \mathcal{N}_{Ph}}$ /* adjacency matrix for the phenotype network, where $\mathcal{N}_{Ph}$ represents the number phenotype entities. */
 $B_{\mathcal{N}_g \times \mathcal{N}_{Ph}}$ /* adjacency matrix for the bipartite graph. */
 $\mathbf{u}_0$ and $\mathbf{v}_0$ /* The initial probability vectors for gene network and phenotype network with size of $(\mathcal{N}_g \times 1)$ and $(\mathcal{N}_{Ph} \times 1)$, respectively. */
 λ, γ and η /*Three parameters used in the RWRH algorithm. */

Output:

$\mathbf{RL}_{cdg}$ and $\mathbf{RL}_{ph}$ /* Rank lists of candidate genes and phenotypes.*/;

- 1: using equations from Eq. 6.3 to Eq. 6.7 to calculate the transition matrix M.
- 2: obtain the initial probability $\mathbf{p}_0$ using Eq. 6.8.
- 3: $\mathbf{p}_s = \mathbf{p}_0; \delta = 1;$
- 4: **while** $\delta > 10^{-10}$ **do**
- 5: $\mathbf{p}_{s+1} = (1 - \gamma)W^T \mathbf{p}_s + \gamma \mathbf{p}_0;$
- 6: $\delta = \sum_{j=1}^{\mathcal{N}_g} (|\mathbf{p}_{s+1}(j) - \mathbf{p}_s(j)|);$
- 7: $\mathbf{p}_s = \mathbf{p}_{s+1};$
- 8: **end while**
- 9: $\mathbf{p}_\infty = \mathbf{p}_{s+1};$
- 10: obtain $\mathbf{u}_\infty$ and $\mathbf{v}_\infty$ from $\mathbf{p}_\infty$.
- 11: obtain $\mathbf{RL}_{cdg}$ and $\mathbf{RL}_{ph}$, rank lists of candidate genes and phenotypes from $\mathbf{u}_\infty$ and $\mathbf{v}_\infty$ respectively.
- 12: **return** $\mathbf{RL}_{cdg}$ and $\mathbf{RL}_{ph}$

Figure 6.3: The Complex Heterogenous Network (CHN) is constructed by connecting the phenotype network and the merged gene network. Nodes connecting two kinds of networks are named bridging nodes. Other nodes are named internal nodes.

6.2.3 The CHN Model

To improve the quality of the prioritization, here we propose a Complex Heterogeneous Network (CHN) model to integrate the multiple genomic data sources and the phenotype information. As shown in Figure 6.3, the CHN is constructed by connecting the merged gene network proposed in Section 6.2.1 and the phenotype network, using the PGRs from the OMIM database.

For a given phenotype, the CHN model prioritize a set of candidate genes associated with this phenotype. The procedure of prioritizing candidate genes is described as follows.

- i) Setp 1: Calculate the transition matrix M_H (Eq. 6.3), according to the topology of the CHN, in which M_G is calculated using Eq 6.1, M_{GP} , M_{PG} and M_P are calculated using Eq 6.4, Eq 6.5 and Eq. 6.7, respectively.

For example, when $\lambda = 0.5$, the transition matrix M is calculated as:

$$\begin{array}{c}
 \begin{matrix} g_1 & g_2 & g_3 & g_4 & g_5 & Ph_1 & Ph_2 & Ph_3 & Ph_4 \end{matrix} \\
 \begin{matrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ g_5 \\ Ph_1 \\ Ph_2 \\ Ph_3 \\ Ph_4 \end{matrix} \begin{pmatrix}
 0 & 0.75 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0.14 & 0 & 0.22 & 0.14 & 0 & 0 & 0.25 & 0.25 & 0 \\
 0.04 & 0.29 & 0 & 0.04 & 0.13 & 0 & 0 & 0.5 & 0 \\
 0 & 0 & 0.5 & 0 & 0.5 & 0 & 0 & 0 & 0 \\
 0 & 0.33 & 0.5 & 0.17 & 0 & 0 & 0 & 0 & 0 \\
 \hline
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 0 & 0.5 & 0 & 0 & 0 & 0.17 & 0 & 0.17 & 0.17 \\
 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0.5 & 0 & 0.5 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 & 0
 \end{pmatrix}
 \end{array}$$

- ii) Step 2: Construct the initial probability $\mathbf{p}_0$, as defined in Eq 6.8.
- iii) Step 3: Repeatedly calculate $\mathbf{p}_{s+1}$ using Eq 5.13, with s increasing, until the maximal difference between $\mathbf{p}_{s+1}$ and $\mathbf{p}_s$ is below 10^{-10} . Then the $\mathbf{p}_{s+1}$ is seen as the stable probability, that is $\mathbf{p}_\infty = \mathbf{p}_{s+1}$.
- iv) Step 4: Obtain $\mathbf{u}_\infty$ from the $\mathbf{p}_\infty$, according to Eq 6.9. Set of candidate genes are ranked based on the $\mathbf{u}_\infty$. The candidate gene with larger value is ranked higher.

6.3 Experiments and Results

In this section, we first compare the performance of the RWRM model and the RABI models. Then we investigate the performance of the RWRH model and use the model to predict disease genes for Alzheimer's disease. We also use the RWRH model to identify disease-disease associations. After that, we investigate the performance of CHN model and use the model to predict disease genes for Insulin-Dependent Diabetes Mellitus (IDDM).

Figure 6.4: ROC curves of RWRM and four RABI models, on the benchmark data set of 36 diseases.

6.3.1 Performance of the RWRM Model on 36 Diseases

As shown in Section 3.1, we collected the PPI data from HPRD [55] and constructed the PPI network (Dataset #2). We obtained three gene networks from BP (Dataset #5), CC (Dataset #6) and MF (Dataset #7). We merged these four networks into a single network. The merged network contains 307,823 interactions between 14,529 genes.

We used Dataset #11 (36 diseases) as benchmark data to evaluate the proposed RWRM algorithm, and compared it with four RABI models: RWR+DRS, RWR+NDOS, 1CSVM+NDOS and 1CSVM+NDOS. The performance was evaluated using LOO-CV

method, described in Section 5.2.6. We plotted the overall ROC curves, as shown in Figure 6.4. The left part of the curve corresponding to RWRM is above the others. It means that the RWRM algorithm ranked more disease genes at the top than the RAB models. The overall AUC value of RWRM is 89.4%, higher than the RAB models (Table 5.2). We also calculated the disease specific AUC value for each of the 36 diseases, shown in Table A.2 in Appendix A. We performed WSR test [91] between AUC values obtained by different integration algorithms. Results showed that AUC values obtained by the RWRM algorithm is higher than RWR+DRS ($p = 0.047$) on the benchmark data set (Dataset #11).

6.3.2 Comparison Between RWRH and CIPHER

To compare the proposed RWRH model with CIPHER, we used the same data and the same evaluation measures as Wu et al. [1]. We constructed the heterogeneous network by connecting the PPI network (Dataset #3) and the phenotype network (Dataset #14), using the PGRs (Dataset #16).

We use LOO-CV method to examine how well the algorithm predicts the PGRs, using 1,428 PGRs as benchmark data (Dataset #16). In each round of validation, we remove one gene-phenotype link, in which the gene is named held-out disease gene and the phenotype is used as target phenotype. The aim of the cross validation is to test whether or not the held-out gene will be predicted to be associated with the target phenotype. The phenotype and the rest of disease genes related to this phenotype are used as source nodes to run the RWRH algorithm.

Table 6.1: Performance of RWRH and CIPHER.

Algorithms	Number of successful predictions		
	LOO1	LOO2	<i>ab initio</i>
RWRH	814	245	201
CIPHER-SP	709	153	140
CIPHER-DN	765	165	157

LOO1

We define the candidate gene set as the held-out disease gene and the 99 nearest genes in the chromosome. We use the RWRH algorithm to rank the set of candidate genes. If the held-out disease gene is ranked as top 1, we consider it a successful prediction. We use the number of successful predictions as a measure to compare different algorithms (LOO1 in Table 6.1). For example, in Figure 6.2 if the link between node *Ph3* and node *g₂* is removed, then node *Ph3* and node *g₃* are used as source node to run the RWRH algorithm. The three parameters γ , λ and η in the RWRH algorithm were set as $\gamma = 0.7$, $\lambda = \eta = 0.5$. The RWRH algorithm successfully ranked 814 known disease genes as top 1. It is much better than CIPHER-SP and CIPHER-DN, proposed by Wu et al. [1], which obtained 709 and 765 successful predictions, respectively. These results are shown as LOO1 in Table 6.1.

LOO2

The intention of LOO1 is to test whether or not the disease gene will be ranked at the top among the set of candidate genes. But for some OMIM phenotype, no candidate genes are available. In this case, all the genes among the chromosome are ranked to find the real

disease gene. We perform LOO-CV to test the effective of the RWRH algorithm in this case. Same to the above experiment, each time we remove a known gene–phenotype link and set source nodes as the target phenotype and the rest of disease genes associated with this phenotype. But all the genes in the PPI network except the known disease genes (source nodes in the PPI network) are used as the set of candidate genes. Finally 245 disease genes are ranked top 1. In contrast, only 153 disease genes have been ranked at the top by CIPHER-SP and 165 disease genes by CIPHER-DN (LOO2 in Table 6.1).

ab initio

In experiment of LOO2, the target phenotype entry may have more than one disease genes. The target phenotype node and disease gene nodes were used as source nodes to run the RWRH algorithm. Some phenotypes do not have known disease gene (such as *Ph1* in Figure 6.2) and no candidate genes are available. Identifying causative genes for this kind of phenotype from the whole-genome is called *ab initio* prediction [1]. For this purpose, we conducted another experiment, in which we perform the LOO-CV to test the effectiveness of the RWRH algorithm on the *ab initio* prediction. For each of these 1,126 phenotype entities in the gene–phenotype graph, we remove all the links from this phenotype to disease genes and use this phenotype entity as a source node to run the random walk algorithm. For example, to find whether or not the RWRH algorithm will find the disease gene for *Ph3* in Figure 6.2, we remove the links between *Ph3* and node g_2 and g_3 , and use *Ph3* as the source node to run the RWRH algorithm. If one of the disease genes associated to the phenotype is ranked top 1 among all the 8,919 genes in the gene network, we consider it a successful prediction. As seen from Table 6.1, there

Table 6.2: Effect of λ in RWRH.

λ	Number of successful predictions		
	LOO1	LOO2	<i>ab initio</i>
0.1	789	196	192
0.3	804	217	196
0.5	814	245	201
0.7	815	257	203
0.9	811	261	203

are 201 successful predictions by our algorithm, while CIPHER-SP and CIPHER-DN successfully predicted 140 cases and 157 cases respectively.

6.3.3 Effect of Parameter Variations in RWRH

Here we study the effect of three parameters, γ , λ and η , in the proposed RWRH algorithm (Eq 6.4 - 6.9).

6.3.3.1 Variation of γ

It has been shown in Section 5.3.4 that the parameter γ only has slight effect on the results. In this thesis, the value of γ is fixed at 0.7.

6.3.3.2 Variation of λ

The parameter λ is the jumping probability between the gene network and the phenotype network. It controls the reinforcement between gene network and phenotype network. Large λ introduces more mutual dependence of rankings between genes and phenotypes. To investigate the effect of this parameter, we varied values of λ ranging from 0.1 to 0.9. The performance of the algorithm is measured using three measures, as shown

Table 6.3: Effect of η in RWRH.

η	Number of successful predictions	
	LOO1	LOO2
0.1	808	239
0.3	813	241
0.5	814	245
0.7	817	242
0.9	820	244

in Table 6.2. The performance is improved with the increasing of λ value. When λ ranging from 0.5 to 0.9, the performance becomes stable. If the λ value is too large, i.e. $\lambda > 0.9$, the random walker jumps between gene network and phenotype network based on the structure of bipartite graph. But the topological structure of gene network and phenotype network cannot be well utilized. In the extreme case, if $\lambda = 1$, the random walker will only jump between the gene network and the phenotype network, and will not reach the internal nodes. Therefore we suggest to select the λ value from 0.5 to 0.9. The performance at $\lambda < 0.5$ is comparatively poor, but still much better than CIPHER. Results suggest that the RWRH algorithm successfully captures the mutually reinforcing relationship between gene network and phenotype network.

6.3.3.3 Variation of η

The parameter η controls the impact of two kinds of source nodes, phenotype source nodes and gene source nodes. If η is 0.5, two sub-networks are equally weighted. If η is above 0.5, the random walker prefers to return to the phenotype source nodes, therefore the phenotype network plays a more important role in the prioritization of disease genes.

On the other hand, if $\eta < 0.5$, gene network is given more importance. To find the effect of η value, we performed two kinds of cross validation experiments, LOO1 and LOO2, with different η values. As seen from Table 6.3, the RWRH algorithm performs slightly better when η is above 0.5. It shows that more importance should be given to the phenotype network.

6.3.4 Comparison between RWRH and RWR on Gene Network Only

To further demonstrate the importance of phenotype network, we compared the performance of RWRH algorithm with RWR on gene network only [24]. In RWR algorithm, for one phenotype, at least two genes are required to perform leave one out cross validation. Therefore, in this experiment, only the phenotypes associated with at least two disease genes were considered. Here the data is from Dataset #16. As described in Section 3.3.3, in this dataset, there are 168 disease phenotypes associated with more than 2 disease genes. In total, there are 470 disease genes associated with these 168 disease phenotypes.

We performed LOO-CV method for each of 168 disease phenotypes. Considering one disease phenotype, in each round of cross validation, we held-out one disease gene and remove the link between this gene to the phenotype entry. The rest of disease genes and the target phenotype were used as source nodes. The held-out gene and all the 99 genes in the ALI were ranked by the RWRH algorithm. Therefore we obtained one rank list for each of the 470 disease genes. As described in Section 5.2.6, we plot the overall ROC curve based on these rank lists, shown in Figure 6.5. For comparison, we perform the

Figure 6.5: ROC curves of RWR and RWRH.

LOO-CV method for the RWR algorithm, which use the disease genes as source nodes and walk on the gene network only.

As shown in Figure 6.5, the curve of RWRH algorithm is above the curve of the RWR algorithm. It suggests that the RWRH algorithm obtained both higher sensitivity and higher specificity, therefore it is better than RWR on gene network only. The overall AUC value of the RWRH algorithm is 0.96, higher than RWR on gene network only (0.92).

6.3.5 Discovering Disease Genes for Alzheimer's Disease Using the RWRH Model

Alzheimer's disease is the most common form of progressive dementia in the elderly people. It is a genetically heterogeneous neurodegenerative disorder. There are 16 disease

Table 6.4: Top five ranked candidate genes for each phenotype of Alzheimer’s disease.

SN	MIM ID	Locus	Number of Candidate Genes	Top 5 Ranked Genes
1	602096	12p11.23-q13.12	148	RACGAP1, STK38L, PTHLH, PKP2, PPFIBP1
2	605526	10q24	122	NOLC1, GBF1, CHUK, PAX2, ZNF518
3	606187	10p13	33	VIM, CUBN, NMT2, MCM10, DCLRE1C
4	607116	20p	188	CDC25B, CSNK2A1, FOXA2, PRNP, SNAP25
5	608907	19p13.2	199	INSR, ICAM5, HNRPM, LDLR , PIN1
6	609636	7q36	87	NOS3, PAXIP1, CDK5 , CUL1, EN2
7	609790	9p22.1	8	-
8	611073	8p12-q22	241	YWHAZ, PRKDC , GEM, LYN, RIPK2
9	611152	1q21	240	CRTC2, PIK4CB, APH1A, CGN, SHC1
10	611154	1q25	74	RALGPS2, RASAL2, ABL2, IVNS1ABP, DHX9
11	611155	3q22-q24	104	CP, AGTR1, TF, PLSCR1, RBP1
12	300756	Xq21.3	9	-

phenotypes (MIM Record) for Alzheimer’s disease, 12 of which with prefix ‘%’. Description of MIM ID is provided in Section 3.1.2.

We used the proposed RWRH algorithm to prioritize candidate genes for these 12 phenotypes. Firstly, we constructed a heterogeneous network, using the PPI network, phenotype network and phenotype-gene network, shown as Dataset #2, #13 and #15. Secondly, we obtained the set of candidate genes using BioMart as described in Section 3.3.5. Then the target phenotype is used as source node to run the RWRH algorithm. After that, we obtained the ranking of candidate genes and selected out top 5 ranked candidate genes, shown in Table 6.4. The first column of Table 6.4 is the serial number of the rows in the table. The second column shows the MIM ID of the phenotype. The third column and the fourth column show the susceptible chromosome region associated with

the disease phenotype and the number of candidate genes located in this region. The last column shows names of the top five ranked candidate genes for each phenotype. Here we provide three novel prediction examples, with their evidence from available literature.

The first example is MIM 611073 (Serial No. 8 in Table 6.4) and the corresponding suspectable region is on chromosome 8p12-q22. Using the web-based query interface BioMart [79], we obtained 241 candidate genes from this region. The set of 241 candidate genes are prioritized using the RWRH algorithm, with MIM 611073 as source node. We found the second ranked gene as *PRKDC*. It is known that this gene encodes an enzyme, DNA-dependent protein kinase catalytic subunit, also known as *DNA-PKcs*. Deficits in *DNA-PKcs* render neurons vulnerable to adverse conditions of relevance to the pathogenesis of neurodegenerative disorders such as Alzheimer’s disease and stroke [106].

MIM 608907 (Serial No. 5 in Table 6.4) describes the phenotype of late onset familial Alzheimer’s disease. This phenotype was associated with the chromosome region on 19p13.2 [107]. The 199 genes in this region were candidate genes for this phenotype. We ranked the set of candidate genes using the RWRH algorithm. We found the fourth and fifth ranked genes were *LDLR* and *PIN1*, respectively. It is reported that the the ligand of *LDLR*, *ApoE* is the major genetic modifier of the age of onset of Alzheimer’s disease [108]. The gene *PIN1* has been identified as the molecular partner of *Tau* and amyloid precursor protein (*APP*), the key factors of Alzheimer’s disease [109].

MIM 609636 (Serial No. 6 in Table 6.4) describes the phenotype of early onset familial Alzheimer’s disease and the corresponding suspectable region is 7q36. Using BioMart, we obtained 87 genes in this region, which were candidate genes. We used RWRH algorithm

to prioritize the set of 87 candidate genes. We found the third ranked gene as *CDK5*. It has been proposed that relative resistance to phosphatases might be a common feature of *CDK5* substrates and could contribute to the hyperphosphorylation of *CRMP2* and *Tau* observed in Alzheimer’s disease [110].

6.3.6 Discovering Hidden Disease-Disease Associations Using the RWRH Model

As described in Section 6.2.2, the RWRH algorithm ranks genes and phenotypes at the same time. In this section, we use the RWRH algorithm to identify associations between diseases. The disease information is obtained from Dataset #12, because the disease description in this dataset is more specific than Dataset #11. There are 1,284 diseases in Dataset #12, $d_1, d_2, \dots, d_{1,284}$. Each disease is represented as a group of disease phenotypes (MIM ID).

Since the RWRH algorithm successfully captures the mutually reinforcing relationship between gene network and phenotype network, it may find some hidden associations that cannot be found by gene network or phenotype network alone. We attempt to discover the hidden associations between genes using the following steps.

- i) Step 1: We constructed the heterogeneous network, using gene network (Dataset #2), phenotype network (Dataset #13) and the PGRs (Dataset #15).
- ii) Step 2: For the i -th disease d_i , we found the disease phenotypes and disease genes.

In the heterogeneous network, we set the source nodes as the disease associated phenotypes and disease genes. We run the RWRH algorithm to rank the remaining

phenotypes in the heterogeneous network, based on the ranking score, i.e. the steady probability in $\mathbf{v}_\infty$ (Section 6.2.2).

iii) Step 3: If the top ranked phenotype is not directly connected to any phenotype of d_i in the phenotype network, we find the disease (i.e. d_j) that include the top ranked phenotype.

iv) Step 4: We define d_j as the most similar disease to d_i . There is no overlap phenotype between d_i and d_j .

We found 122 disease associations sharing no phenotype. We further removed the disease association pairs sharing common disease genes. There are 18 disease associations left, which are shown in Table 6.5. Among these 18 disease associations, 12 disease pairs have been classified in the same disease class [66]. Especially, 8 of these disease associations are metabolism diseases (Serial No. 1, 3-5, 9-11 and 17 in Table 6.5). As comparison, in the human disease network constructed by Goh et al. [66], metabolism diseases are not well connected. We can discover these relationships, because in the RWRH algorithm phenotype similarity information and gene interaction information are complementarily used. It can be seen from Table 6.5 that disease 1130 and disease 72 (Serial No. 2 in Table 6.5) are two subtypes of Oculocutaneous albinism. One is classified as Ophthalmological disease and the other is classified as dermatological disease [66]. The other example is disease 1325 and disease 315 (Serial No. 6 in Table 6.5). Disease 1325 is classified as 'multiple', and disease 315 is classified as Connective tissue disorder. In addition, there are interactions between two sets of disease genes from

Table 6.5: Predicted disease associations.

ID1	Disease 1	ID2	Disease 2	Support Information
686	HMG-CoA lyase deficiency; HMG-CoA synthase-2 deficiency	44	Acyl-CoA dehydrogenase deficiency	Metabolic [‡]
1130	Oculocutaneous albinism	72	Albinism, oculocutaneous	subtype of the same disease
140	Argininosuccinic aciduria	262	Carbamoylphosphate synthetase I deficiency	Metabolic [‡] ; urea cycle disorders
335	Citrullinemia	262	Carbamoylphosphate synthetase I deficiency	Metabolic [‡] ; urea cycle disorders
374	CPT deficiency	271	Carnitine-acylcarnitine translocase deficiency	Metabolic [‡]
1325	Rhizomelic chondrodysplasia punctata, type 1 and type 3	315	Rhizomelic chondrodysplasia punctata, type 2	subtype of the same disease; share interaction
1242	Polycythemia	505	Erythrocytosis	Hematological [‡] , share interaction
4291	Cerebral cavernous malformations	558	Fuchs endothelial corneal dystrophy	-
187	Bartter syndrome	587	Gitelman syndrome	hypokalemic metabolic alkalosis(Knoers and Levchenko, 2008)
610	Glycogen storage disease	609	Glycogenosis, hepatic	Metabolic [‡]
603	Glutaricaciduria	686	HMG-CoA lyase deficiency; HMG-CoA synthase-2 deficiency	Metabolic [‡]
1004	Methylmalonic aciduria	826	Isovaleric acidemia	Metabolic [‡]
327	Chronic granulomatous disease	889	Leukocyte adhesion deficiency	Immunological [‡] ; myeloid cell primary immune deficiencies (Malech and Hickstein, 2007)
875	Left ventricular noncompaction	1113	Noncompaction of left ventricular myocardium	Cardiovascular [‡] , subtype of the same disease
184	Bare lymphocyte syndrome	1141	Omenn syndrome	Immunological [‡]
614	GM1-gangliosidosis; GM2-gangliosidosis	1347	Sandhoff disease	Metabolic [‡] ; subtype of the same disease (Mahurana et al., 2007)
739	Hypermethioninemia	2669	Hereditary persistence of alpha-fetoprotein	-
5232	Pituitary hormone deficiency	5144	Optic nerve hypoplasia/aplasia	-

Here [‡] refers to Goh et al., 2007 [66].

disease 1325 and disease 315. Thus these two disease pairs (Serial No. 2 and No. 6 in Table 6.5) are actually subtypes of the same disease but classified into different disease

classes. The association between Bartter syndrome and Gitelman syndrome (Serial No. 9 in Table 6.5) is supported by recent literature. Type III Bartter syndrome is clinically and biochemically overlapping with Gitelman syndrome [111]. In conclusion, we found 18 novel disease associations using RWRH algorithm, which cannot be found by gene network or phenotype network only.

6.3.7 Performance of the CHN Model

Here we use the CHN model to integrate the phenotype network and genomic gene networks. We used Dataset #2 and Datasets #5—#7 to construct the merged gene network. Then we constructed the CHN by connecting the phenotype network (Dataset #13) and the merged gene network using PGRs (Dataset #15).

The CHN model described in Section 6.2.3 is used to prioritize a set of candidate genes. We use the LOO-CV method to evaluate this model, in which Dataset #15 is used as the benchmark data. In each round of cross validation, we remove one phenotype–gene relationship, in which the removed gene is called held-out gene and the removed disease phenotype is called target phenotype. The aim of the cross validation is to test whether or not the CHN model will successfully predict the relationship between the held-out gene and the target phenotype. We collected 99 genes in the nearest chromosome region (ALI) of the held-out disease gene. The set of test genes constitute these 99 genes and the held-out gene. We use the CHN model to prioritize the set of test genes, in which the target phenotype and the rest of the associated disease genes are set as source nodes. The parameters are $\gamma = 0.7$, $\lambda = 0.5$ and $\eta = 0.5$. If the held-out disease gene is ranked at the top, it is [considered](#) a successful prediction. The number of successful predictions

Table 6.6: Number of successful predictions in LOO-CV.

Integration Methods	Number of Successful Predictions
CHN	2,105
NDOS+RWRH	2,048
DRS+RWRH	2,008

is used as the performance measure. Results are shown in Table 6.6. It can be seen that the CHN model successfully identified 2,105 PGRs.

We evaluate two RABI models, RWRH+NDOS and RWRH+DRS, on the same benchmark data (Dataset #15). We constructed a heterogeneous network by connecting the phenotype network and the PPI network, and run the RWRH algorithm to prioritize the set of test genes, and thus we obtained the rank list of test genes. Then we constructed a heterogeneous network by connecting the phenotype network and the gene network obtained from BP and run the RWRH algorithm to prioritize the set of test genes and obtained the rank list of test genes. Similarly we obtained the rank lists of test genes using the heterogeneous networks corresponding to CC and MF respectively. Finally separate rank lists are merged into one list using the NDOS algorithm or DRS algorithm. The number of successful predictions are shown in Table 6.6. It can be seen that the RWRH+NDOS and RWRH+DRS successfully predicted 2,048 and 2,008 PGRs, which are smaller than the CHN model. In conclusion, the CHN model performed better than the RWRH+NDOS and RWRH+DRS models.

6.3.8 Discovering Disease Genes of Insulin-Dependent Diabetes Mellitus Using the CHN Model

Insulin-Dependent Diabetes Mellitus (IDDM) is also known as Type 1 diabetes mellitus, which is a consequence of autoimmune destruction of insulin-producing pancreatic beta cells. It is a genetically heterogeneous autoimmune disease, with multiple genes and loci involved in the pathogenesis [112]. There are 14 susceptible loci without association to any disease gene in the OMIM [5], listed in Table 6.7. The second column shows the MIM ID of the disease phenotype. The third column is the associated locus. And the fourth column shows the number of candidate genes in the susceptible locus.

Considering one phenotype, we obtained the set of candidate genes using BioMart. We use the proposed CHN model to prioritize the set of candidate genes. The parameters for the CHN model are $\gamma = 0.7$, $\lambda = 0.5$ and $\eta = 0.5$. Top five candidate genes are shown in the fifth column of Table 6.7, for the reference of biologists. Some of our predictions have got literature support.

The fifth ranked gene for MIM 600318 (Serial No. 1 in Table 6.7) is NR2F2 (also called COUP-TFII). It has been found to contribute to the control of insulin secretion through the complex HNF4 transcription factor network operating in chicken pancreatic beta cell [113]. Future studies should therefore include the search for mutations in the human NR2F2 gene in patients with IDDM.

The first ranked gene for MIM 600321 (Serial No. 4 in Table 6.7) is NEUROD1. Cinek et al. [114] has found the association between NEUROD1 and IDDM in Czech children. They compared 285 children with IDDM diagnosed under the age of 15 years

Table 6.7: Top five ranked candidate genes for each phenotype of IDDM.

SN	MIM ID	Locus	Number of Candidate Genes	Top 5 Candidate Genes
1	600318	15q26	72	BLM, IGF1R, FURIN, RHCG, NR2F2
2	600319	11q13	263	PPME1, RELA, STIP1, GSTP1, MEN1
3	601941	18q21	88	BCL2, TXNL1, MYO5B, SMAD4, TNFRSF11A
4	600321	2q31	81	NEUROD1 , ABCB11, WIPF1, LRP2, CCDC141
5	600883	6q25-q27	38	ESR1, PLG, TBP, VIL2, IGF2R
6	601208	14q24.3-q31	85	TSHR, FOS, EIF2B2, NEK9, PTPN21
7	601318	2q34	19	IDH1, ERBB4, PIP5K3, LANCL1, PTHR2
8	601666	6q21	51	FYN, CDC40, ATG5, CD164, TUBE1
9	603266	10q25	55	TCF7L2 , GFRA1, VTI1A, ACSL5, ADRA2A
10	605598	5q31.1-q33.1	241	SPINK1, NR3C1 , VDAC1, HSPA9, IL3
11	612520	12q24	216	SH2B3, TCF1, OAS1, PTPN11, OAS2
12	612521	6q25	70	ESR1, SYNE1, VIL2, LATS1, OPRM1
13	612622	4q27	17	CCNA2, IL2, MAD2L1, FGF2, TRPC3
14	613006	10q23.31	26	PTEN, ACTA2, PANK1, ANKRD1, MPHOSPH1

with 289 non-diabetic control children. Their results confirm that NEUROD1 polymorphism Ala45Thr is associated with IDDM.

The locus of MIM 603266 (Serial No. 9 in Table 6.7) is overlap with loci of type 2 diabetes (T2D), therefore genes in this region may explain the mechanisms of both types of diabetes. The first ranked gene TCF7L2 is known T2D causing gene and the fifth ranked gene ADRA2A is also assumed to increase T2D risk [115]. A single-nucleotide polymorphism (SNP) in the human ADRA2A gene has been found responsible to reduced insulin secretion [115]. It is possible to find associations between IDDM and TCF7L2 or ADRA2A through further mutation analysis.

NR3C1 is also known as GR (Glucocorticoid Receptor). It is ranked the second

among the set of 241 candidate genes for MIM 605598 (Serial No. 10 in Table 6.7). Rosmond and Holm [116] performed a five-year follow-up study of three polymorphisms in this gene in 163 unrelated Swedish men. They found a significant increase in body weight, body mass index, abdominal obesity, fasting glucose, insulin, and homeostasis model assessment over the 5-year follow-up among homozygotes for the rare BclI allele. Syed et al. [117] reported an SNP in the human GR gene (rs2918419) to be linked with insulin resistance in men. These results suggest that NR3C1 is likely to be associated with IDDM18.

6.4 Summary

In this chapter, we proposed three NBI models for candidate gene prioritization, the RWRM model, the RWRH model and the CHN model.

The RWRM model integrates four genomic data sources, the PPI network and gene networks from BP, CC and MF. We compared the performance of RWRM with the RABI models proposed in Chapter 5. As shown in Figure 6.5, the RWRM model achieved both higher sensitivity and higher specificity for the prioritization of candidate genes. We performed WSR test [91] between AUC values obtained by RWRM and the two RABI models. Results showed that RWRM performed significantly better than the RWR+DRS model ($p = 0.047$) on the benchmark data set (Dataset #11).

The RWRH model integrates the PPI network and the phenotype network derived from the OMIM database. We constructed a heterogeneous network by connecting the two networks. We used LOO-CV method to compare the performance of RWRH with

CIPHER [1] on the benchmark data (Dataset #16). Results showed that RWRH successfully predicted more disease genes than CIPHER (Table 6.1), indicating better performance of the RWRH model. We then compared the RWRH algorithm with RWR on PPI network only. The experiment results showed that RWRH performed much better than RWR (Figure 6.3). The proposed RWRH algorithm was used to predict new disease genes of Alzheimer’s Disease. Some of the predictions were supported by existing literature. In addition, we used the RWRH to identify the relationship between diseases. We identified 18 disease associations that cannot be found from phenotype data or gene network only.

In the last part of this chapter, we proposed a CHN model, which integrates multiple genomic data sources and the phenotype information. We used Dataset #16 as benchmark data containing 3,871 PGRs. Using the proposed CHN model, we successfully identified 2,105 of 3,871 PGRs, whereas the DRS-based and NDOS-based integration methods, respectively, could identify 2,008 and 2,048 number of PGRs. This shows the better performance of our proposed CHN model.

Chapter 7

Conclusions and Future Directions

In this chapter, we summarize the research contributions presented in this dissertation.

We also give some possible directions for future work.

7.1 Conclusions and Discussion

In our research work, we focus on data integration schemes for prioritization of a set of candidate genes. Different genomic data sources and phenotype data obtained from the OMIM database have been integrated for this task. In order to find effective and efficient integration algorithms, we investigated rank aggregation based integration (RABI) models and network based integration (NBI) models.

In Chapter 4, we investigated five topological features derived from the protein-protein interaction (PPI) network for use in the characterization and classification of cancer genes. Statistical significance of these features indicates that cancer gene tends to interact with other cancer genes.

We investigated all the combinations of the five features and found, that the best feature combination is $\{Feature1, Feature3, Feature5\}$. Among the three features, *Feature1* measures centrality of a node in the graph, *Feature3* measures proportion

of cancer genes in its 2-neighborhood, and *Features5* measures the communication efficiency of cancer genes. We found that *Feature1* and *Feature5* are highly correlated, therefore the performance of two-feature combinations $\{Feature1, Feature3\}$ and $\{Feature3, Feature5\}$ are similar to $\{Feature1, Feature3, Feature5\}$. We also found that *Feature3*, 2N-index is the most predictive feature, that is, the proportion of cancer genes around one gene's 2-neighborhood is an important indicator for prediction. Using this feature alone, we achieved the accuracy as high as 70.5%. We compared two kinds of classification algorithms, and found, that SVM performed slightly better than MLP. Using the feature combination of $\{Feature1, Feature3, Feature5\}$, the accuracy for SVM classifier was 73.9%

In Chapter 5, we attempted to improve the two stages of the RABI model, the ranking procedure and the combination procedure, for candidate gene prioritization. We introduced two ranking algorithms into the first stage, 1CSVM and RWR, and proposed two models 1CSVM+NDOS and RWR+NDOS. We integrated four data sources, BP, CC, MF and the PPI network to prioritize candidate genes. In the 1CSVM+NDOS model, we used 1CSVM to prioritize candidate genes based on each of the four data sources and used NDOS to combine multiple rank lists. In the RWR+NDOS model, we ranked set of candidate genes using the RWR algorithm and used NDOS to combine multiple rank lists. These two integration models were compared with MKL [34], a kernel based integration algorithm. On the benchmark data of 36 disease genes, 1CSVM+NDOS and RWR+NDOS models achieved AUC values at 84.4% and 85.1%, respectively, whereas the MKL algorithm achieved AUC value at 79.3%. We conclude that these two RABI models are more suitable for the problem of candidate gene prioritization. In addition,

we found the RWR+NDOS model performed slightly better than the 1CSVM+NDOS model.

In order to improve the combination procedure of the RABI model (Figure 5.1), we proposed a new rank aggregation algorithm, named DRS. The DRS-based integration showed improved performance over the NDOS-based integration. On the same benchmark data (Dataset #11), the 1CSVM+DRS and RWR+DRS models achieved AUC values at 85.4% and 87.7%, respectively. Furthermore, the DRS algorithm is computationally more efficient than the NDOS algorithm. When the number of data sources increases to 20, the execution time of DRS is more than 180 times faster than NDOS. Therefore, we conclude that the proposed DRS algorithm is an effective and powerful rank aggregation algorithm, and the RWR+DRS model is better than the other three RABI models (1CSVM+DRS, RWR+NDOS and 1CSVM+NDOS) and the MKL algorithm, for the problem of candidate gene prioritization.

In Chapter 6, we proposed three NBI models for candidate gene prioritization, the RWRM model, the RWRH model and the CHN model. The RWRM model integrates four genomic data sources, BP, CC, MF and the PPI network. We compared the performance of RWRM and with the RABI models proposed in Chapter 5. We have shown that the RWRM model achieved both higher sensitivity and higher specificity. We performed WSR test [91] between AUC values obtained by RWRM and the two RABI models. We showed that RWRM performed significantly better than the RWR+DRS model on the benchmark dataset (Dataset #11), with p value as $p = 0.047$.

The RWRH model integrates the PPI network and the phenotype network derived from the OMIM database. We constructed the heterogeneous network by connecting the

phenotype network and the PPI network using PGRs, and applied the RWRH algorithm, which is the extension of the RWR algorithm on the heterogeneous network. We used LOO-CV to compare the performance of RWRH with CIPHER [1] on the benchmark data. Results showed that RWRH performed much better than CIPHER (Table 6.1). We then compared the RWRH algorithm with RWR algorithm using PPI network only. We showed that RWRH performed much better than RWR, suggesting the effectiveness of the phenotype information. Next the proposed RWRH algorithm was used to predict new disease genes of Alzheimer’s Disease (Dataset #17). Some of the predictions were supported by the available literature. In addition, using the RWRH model, we discovered 18 disease associations that cannot be found from phenotype data or gene network only, shown in Table 6.5.

The CHN model integrates multiple genomic data sources and the phenotype information. Each genomic data source is represented as a gene network, and these gene networks are merged into one network . The merged gene network and the phenotype network is connected using the PGRs to obtain the complex heterogeneous network. We extend the RWR algorithm on the complex heterogeneous network to prioritize set of candidate genes. Using the 3,871 PGRs (Dataset #15) as benchmark data, we successfully identified 2,105 PGRs, whereas the DRS-based and NDOS-based integration methods, respectively, could identify 2,008 and 2,048 number of PGRs. This shows the superior performance of our proposed CHN model. We used the CHN model to predict disease genes for IDDM, shown in Table 6.7.

7.2 Future Directions

Here we provide some possible directions for future study in the area of disease gene prediction.

7.2.1 Cancer Gene Identification

In this thesis, we proposed five topological features obtained from PPI network to predict cancer genes. However, this is just a general prediction. We did not make prediction for a specific cancer, such as breast cancer. Another problem is that only the PPI network is used in this work to predict cancer genes. Integration of more data sources might increase the quality of the prediction. A possible direction for cancer gene prediction is to apply the integration models proposed in Chapter 5 and Chapter 6 on cancer datasets, to predict disease genes for specific cancer.

7.2.2 Integration of More Data Sources

In this thesis, we integrated the PPI network, the phenotype network and gene networks obtained from BP, CC and MF. It will be better to include more data sources, such as domain interaction, gene expression, protein sequence data and literature mining data. Disease affected pathway, derived from differentially expressed genes [118], would be helpful in the prioritization of candidate genes. The phenotype data also need to be improved. In this thesis, the similarity between two phenotypes entities are calculated based on the text description in OMIM [5]. But OMIM does not use a controlled vocabulary and is heavily underannotated [119]. Recently, the ontological description of OMIM phenotypes has been proposed [45]. With the availability of well annotated phenotype

data, a better quality phenotype network might be obtained by using suitable ontological similarity measure, which will improve the performance of the proposed RWRH model.

7.2.3 Assignment of Weights to Different Data Sources

In this thesis, in both the RABI and NBI models, different genomic data sources are equally weighted. However, different data sources, due to their quality of information content, have different predictive power for different diseases. It is necessary to give a systemically weighting strategy based on the information content of the database. If the weights are proper, the integration algorithm will perform better for the prioritization of candidate genes.

7.2.4 Prioritization of SNPs from GWAS Data

Recently, genome wide association studies (GWAS) have been generally used to detect allelic variations that affect susceptibility to complex diseases [120]. A number of bioinformatics algorithms have been proposed to identify disease related SNP (single-nucleotide polymorphism) from GWAS data, including gene-set based [121], text-based [122] and pathway based [123] approach. The prioritization algorithms investigated in this thesis can be used to prioritize candidate genes obtained from GWAS data. We start from the selected candidate SNPs. Candidate genes are seemed as the neighboring genes of selected candidate SNPs. After prioritization, both the disease gene and the corresponding SNP can be obtained.

References

- [1] X. Wu, R. Jiang, M. Q. Zhang, and S. Li, “Network-based global inference of human disease genes.” *Molecular Systems Biology*, vol. 4, no. 1, May 2008, Article Number 189.
- [2] [Online]. Available: http://images.nigms.nih.gov/imageRepository/2539/Chromosome_inside_Nucleus.jpg
- [3] F. Yamamoto, H. Clausen, T. White, J. Marken, and S. Hakomori, “Molecular genetic basis of the histo-blood group ABO system.” *Nature*, vol. 345, no. 6272, pp. 229–233, 1990.
- [4] V. McKusick, *Mendelian Inheritance in Man; A Catalog of Human Genes and Genetic Disorders*. The Johns Hopkins University Press., 1998.
- [5] A. Hamosh, A. F. Scott, J. S. Amberger, C. A. Bocchini, and V. A. McKusick, “Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders.” *Nucleic Acids Research*, vol. 33 Database Issue, pp. D514–D517, January 2005.
- [6] A. Balmain, J. Gray, and B. Ponder, “The genetics and genomics of cancer.” *Nature Genetics*, vol. 33, no. Suppl, pp. 238–244, 2003.
- [7] E. Lander and S. NJ., “Genetic dissection of complex traits.” *Science*, vol. 265, no. 5181, pp. 2037–2048, 1994.

- [8] M. G. Anne, H. N. Joseph, and J. A. Timothy, “Finding genes that underlie complex traits,” *Science*, vol. 298, no. 5602, pp. 2345 – 2349, December 2002.
- [9] D. Botstein and N. Risch, “Discovering genotypes underlying human phenotypes: past successes for Mendelian disease, future approaches for complex disease.” *Nature Genetics*, vol. 33 Suppl, pp. 228–237, March 2003.
- [10] S. G. Davey, S. Ebrahim, S. Lewis, A. Hansell, L. Palmer, and B. PR, “Genetic epidemiology and public health: hope, hype, and future prospects.” *Lancet*, vol. 366, no. 9495, pp. 1484–1498, 2005.
- [11] [Online]. Available: <http://www.who.int/mediacentre/factsheets/fs297/en/>
- [12] P. A. Futreal, L. Coin, M. Marshall, and T. Down, “A census of human cancer genes.” *Nature Reviews Cancer*, vol. 4, pp. 177–183, March 2004. [Online]. Available: <http://www.sanger.ac.uk/genetics/CGP/Census/>
- [13] R. L. Strausberg, A. J. G. Simpson, and R. Wooster, “Sequence-based cancer genomics: progress, lessons and opportunities,” *Nature Reviews Genetics*, vol. 4, pp. 409–418, June 2003.
- [14] L. D. Stein, “Human genome: End of the beginning.” *Nature*, vol. 431, no. 7011, pp. 915–916, 2004.
- [15] F. S. Collins, “Positional cloning moves from perditional to traditional.” *Nature Genetics*, vol. 9, no. 4, pp. 347–350, 1995.
- [16] G. D. Schuler, M. S. Boguski, E. A. Stewart, L. D. Stein, G. Gyapay, K. Rice, and R. E. White, “A gene map of the human genome.” *Science*, vol. 274, no. 5287, pp. 540–546, 1996.

- [17] D. Botstein, R. L. White, M. Skolnick, and R. W. Davis, "Construction of a genetic linkage map in man using restriction fragment length polymorphisms." *American Journal of Human Genetics*, vol. 32, no. 3, pp. 314–331, 1980.
- [18] F. Turner, D. Clutterbuck, and C. Semple, "POCUS: mining genomic sequence annotation to predict disease genes," *Genome Biology*, vol. 4, no. 11, 2003.
- [19] C. Perez-Iratxeta, P. Bork, and M. A. Andrade, "Association of genes to genetically inherited diseases using data mining." *Nature Genetics*, vol. 31, no. 3, July 2002.
- [20] J. Freudenberg and P. Propping, "A similarity-based method for genome-wide prediction of disease-relevant human genes." *Bioinformatics*, vol. 18 Suppl 2, 2002.
- [21] N. López-Bigas and C. A. Ouzounis, "Genome-wide identification of genes likely to be involved in human genetic disease." *Nucleic Acids Research*, vol. 32, no. 10, pp. 3108–3114, 2004.
- [22] E. A. Adie, R. R. Adams, K. L. Evans, D. J. Porteous, and B. S. Pickard, "Speeding disease gene discovery by sequence based candidate prioritization." *BMC Bioinformatics*, vol. 6, no. 1, March 2005, Article Number 55.
- [23] N. Tiffin, J. F. Kelso, A. R. Powell, H. Pan, V. B. Bajic, and W. A. Hide, "Integration of text- and data-mining using ontologies successfully selects disease gene candidates," *Nucleic Acids Research*, vol. 33, no. 5, pp. 1544–1552, 2005.
- [24] S. Köhler, S. Bauer, D. Horn, and P. N. Robinson, "Walking the interactome for prioritization of candidate disease genes," *The American Journal of Human Genetics*, vol. 82, pp. 949–958, April 2008.
- [25] J. Xu and Y. Li, "Discovering disease-genes by topological features in human protein-protein interaction network," *Bioinformatics*, vol. 22, pp. 2800–2805, 2006.

- [26] M. Oti, B. Snel, M. A. Huynen, and H. G. Brunner, “Predicting disease genes using protein-protein interactions.” *Journal of Medical Genetics*, vol. 43, no. 8, pp. 691–698, August 2006.
- [27] P. Bork, L. Jensen, C. von Mering, A. Ramani, I. Lee, and E. Marcotte, “Protein interaction networks from yeast to human.” *Current Opinion in Structural Biology*, vol. 14, no. 3, pp. 292–299, 2004.
- [28] M. A. Harris, J. Clark, A. Ireland, and J. Lomax, “The Gene Ontology (GO) database and informatics resource,” *Nucleic Acids Research*, vol. 32, January 2004.
- [29] E. A. Adie, R. R. Adams, K. L. Evans, D. J. Porteous, and B. S. Pickard, “Suspects: enabling fast and effective prioritization of positional candidates.” *Bioinformatics*, vol. 22, no. 6, pp. 773–774, March 2006.
- [30] K. J. Gaulton, K. L. Mohlke, and T. J. Vision, “A computational system to select candidate genes for complex human traits.” *Bioinformatics*, vol. 23, no. 9, pp. 1132–1140, May 2007.
- [31] L. Franke, H. Bakel, L. Fokkens, E. D. de Jong, M. Egmont-Petersen, and C. Wijmenga, “Reconstruction of a functional human gene network, with an application for prioritizing positional candidate genes.” *American Journal of Human Genetics*, vol. 78, no. 6, pp. 1011–1025, June 2006.
- [32] S. Rossi, D. Masotti, C. Nardini, E. Bonora, G. Romeo, E. Macii, L. Benini, and S. Volinia, “TOM: a web-based integrated approach for identification of candidate disease genes.” *Nucleic Acids Research*, vol. 34, no. Web Server issue, pp. W285–W292, July 2006.

- [33] S. Aerts, D. Lambrechts, S. Maity, P. Van Loo, and B. Coessens, “Gene prioritization through genomic data fusion,” *Nature Biotechnology*, vol. 24, no. 5, pp. 537–544, May 2006. [Online]. Available: <http://www.esat.kuleuven.be/endeavourweb>
- [34] T. De Bie, L. C. Tranchevent, L. M. van Oeffelen, and Y. Moreau, “Kernel-based data fusion for gene prioritization.” *Bioinformatics*, vol. 23, no. 13, pp. i125–i132, July 2007.
- [35] J. M. Stuart, E. Segal, D. Koller, and S. K. Kim, “A gene-coexpression network for global discovery of conserved genetic modules,” *Science*, vol. 302, no. 5643, pp. 249–255, October 2003.
- [36] K. Lage, E. O. Karlberg, Z. M. Størling, P. I. Olason, A. G. Pedersen, O. Rigina, A. M. Hinsby, Z. Tümer, F. Pociot, N. Tommerup, Y. Moreau, and S. Brunak, “A human phenome-interactome network of protein complexes implicated in genetic disorders.” *Nature Biotechnology*, vol. 25, no. 3, pp. 309–316, March 2007.
- [37] M. Oti and H. G. Brunner, “The modular nature of genetic diseases,” *Clinical Genetics*, vol. 71, no. 1, pp. 1–11, January 2007.
- [38] U. de Lichtenberg, L. J. Jensen, S. Brunak, and P. Bork, “Dynamic complex formation during the yeast cell cycle.” *Science*, vol. 307, no. 5710, pp. 724–727, 2005.
- [39] K. Järvelin and J. Kekäläinen, “IR evaluation methods for retrieving highly relevant documents,” in *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: ACM Press, 2000, pp. 41–48.

- [40] U. Ala, R. M. Piro, E. Grassi, C. Damasco, L. Silengo, M. Oti, P. Provero, and F. Di Cunto, “Prediction of human disease genes by human-mouse conserved co-expression analysis.” *PLoS Computational Biology*, vol. 4, no. 3, March 2008.
- [41] M. Oti, J. van Reeuwijk, M. A. Huynen, and H. G. Brunner, “Conserved co-expression for candidate disease gene prioritization,” *BMC Bioinformatics*, vol. 9, April 2008, Article Number 208.
- [42] J. B. Bard and S. Y. Rhee, “Ontologies in biology: design, applications and future challenges.” *Nature Reviews Genetics*, vol. 5, no. 3, pp. 213–222, March 2004.
- [43] J. Kelso, J. Visagie, G. Theiler, A. Christoffels, S. Bardien, D. Smedley, D. Otgaar, G. Greyling, C. V. Jongeneel, M. I. McCarthy, T. Hide, and W. Hide, “eVOC: a controlled vocabulary for unifying gene expression data.” *Genome Research*, vol. 13, no. 6A, pp. 1222–1230, June 2003.
- [44] C. L. Smith, C. A. Goldsmith, and J. T. Eppig, “The mammalian phenotype ontology as a tool for annotating, analyzing and comparing phenotypic information.” *Genome Biol*, vol. 6, no. 1, 2005, Article Number R7.
- [45] P. Robinson, S. Köhler, S. Bauer, D. Seelow, D. Horn, and S. Mundlos, “The Human Phenotype Ontology: a tool for annotating and analyzing human hereditary disease.” *The American Journal of Human Genetics*, vol. 83, no. 5, pp. 610–615, November 2008.
- [46] T. K. B. Gandhi, J. Zhong, S. Mathivanan, and L. Karthick, “Analysis of the human protein interactome and comparison with yeast, worm and fly interaction datasets,” *Nature Genetics*, vol. 38, no. 3, pp. 285–293, February 2006.
- [47] M. Kanehisa and P. Bork, “Bioinformatics in the post-sequence era,” *Nature Genetics*, vol. 33, no. Suppl, pp. 305–310, 2003.

- [48] L. D. Stein, “Integrating biological databases.” *Nature Reviews Genetics*, vol. 4, no. 5, pp. 337–345, May 2003.
- [49] A. I. Su, M. P. Cooke, K. A. Ching, and Y. Hakak, “Large-scale analysis of the human and mouse transcriptomes.” *Proceedings of the National Academy of Sciences*, vol. 99, no. 7, pp. 4465–4470, April 2002.
- [50] S. Hunter, R. Apweiler, T. K. K. Attwood, and A. Bairoch, “Interpro: the integrative protein signature database.” *Nucleic Acids Research*, pp. D211–D215, October 2008.
- [51] P. W. Lord, R. D. Stevens, A. Brass, and C. A. Goble, “Investigating semantic similarity measures across the gene ontology: the relationship between sequence and annotation,” *Bioinformatics*, vol. 19, no. 10, pp. 1275–1283, July 2003.
- [52] M. Kanehisa, M. Araki, S. Goto, M. Hattori, and M. Hirakawa, “KEGG for linking genomes to life and the environment,” *Nucleic Acids Research*, vol. 36, pp. D480–D484, January 2008. [Online]. Available: <http://www.genome.jp/kegg/>
- [53] G. D. Bader, D. Betel, and C. W. Hogue, “BIND: the biomolecular interaction network database.” *Nucleic Acids Research*, vol. 31, no. 1, pp. 248–250, January 2003.
- [54] G. Joshi-Tope, M. Gillespie, I. Vastrik, P. D’Eustachio, and E. Schmidt, “Reactome: a knowledgebase of biological pathways.” *Nucleic Acids Research*, vol. 33, no. Database issue, pp. D428–D432, January 2005.
- [55] S. Peri, J. D. Navarro, R. Amanchy, and T. Z. Kristiansen, “Development of human protein reference database as an initial platform for approaching systems biology in humans,” *Genome Research*, vol. 13, no. 10, pp. 2363–2371, October 2003. [Online]. Available: <http://www.hprd.org/download>

- [56] M. A. van Driel, J. Bruggeman, G. Vriend, H. G. Brunner, and J. A. Leunissen, “A text-mining analysis of the human phenome.” *European Journal of Human Genetics*, vol. 14, no. 5, pp. 535–542, May 2006.
- [57] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, “Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.” *Nucleic Acids Research*, vol. 25, no. 17, pp. 3389–3402, 1997.
- [58] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, “Estimating the support of a high-dimensional distribution,” *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, July 2001.
- [59] G. R. Lanckriet, T. De Bie, N. Cristianini, M. I. Jordan, and W. S. Noble, “A statistical framework for genomic data fusion.” *Bioinformatics*, vol. 20, no. 16, pp. 2626–2635, November 2004.
- [60] A. L. Barabási and Z. N. Oltvai, “Network biology: understanding the cell’s functional organization.” *Nature Reviews Genetics*, vol. 5, no. 2, pp. 101–113, February 2004.
- [61] J.-H. Wang, “Protein recognition by cell surface receptors: physiological receptors versus virus interactions.” *Trends in Biochemical Sciences*, vol. 27, no. 3, pp. 122–126, March 2002.
- [62] J. Battey, E. Jordan, D. Cox, and W. Dove, “An action plan for mouse genomics,” *Nature Genetics*, vol. 21, pp. 73–75, 1999.
- [63] E. Birnery, A. Bateman, M. Clamp, and T. Hubbard, “Mining the draft human genome,” *Nature*, vol. 409, pp. 827–828, 2001.

- [64] P. Uetz, L. Giot, G. Cagney, and T. A. Mansfield, “A comprehensive analysis of protein-protein interactions in *saccharomyces cerevisiae*,” *Nature*, vol. 403, no. 6770, pp. 623–627, February 2000.
- [65] Z. Tu, L. Wang, M. Xu, and X. Zhou, “Further understanding human disease genes by comparing with housekeeping genes and other genes,” *BMC Genomics*, vol. 7, 2006.
- [66] K.-I. Goh, M. E. Cusick, D. Valle, B. Childs, M. Vidal, and A. L. Barabási, “The human disease network.” *Proceedings of the National Academy of Sciences*, vol. 104, no. 21, pp. 8685–8690, May 2007.
- [67] P. Flicek, B. L. L. Aken, K. Beal, B. Ballester, and M. Caccamo, “Ensembl 2008.” *Nucleic Acids Research*, vol. 36, no. Database issue, pp. D707–D714, November 2008.
- [68] H. Jeong, S. P. Mason, A. L. Barabasi, and Z. N. Oltvai, “Lethality and centrality in protein networks,” *Nature*, vol. 411, no. 6833, pp. 41–42, May 2001.
- [69] J. Chen and B. Yuan, “Detecting functional modules in the yeast protein–protein interaction network,” *Bioinformatics*, vol. 22, pp. 2283–2290, 2006.
- [70] Z. Guo, Y. Li, X. Gong, C. Yao, W. Ma, and D. Wang, “Edge-based scoring and searching method for identifying condition-responsive protein-protein interaction sub-network.” *Bioinformatics*, vol. 23, no. 16, pp. 2121–2128, June 2007.
- [71] R. Sharan, I. Ulitsky, and R. Shamir, “Network-based prediction of protein function,” *Molecular Systems Biology*, vol. 3, March 2007.
- [72] R. I. Kondor and J. Lafferty, “Diffusion kernels on graphs and other discrete structures,” in *Proceedings of the International Conference on Machine Learning*, 2002, pp. 315–322.

- [73] P. Hu, G. Bader, D. A. Wigle, and A. Emili, “Computational prediction of cancer-gene function,” *Nature Reviews Cancer*, vol. 7, no. 1, pp. 23–34, December 2006.
- [74] J. J. Jiang and D. W. Conrath, “Semantic similarity based on corpus statistics and lexical taxonomy,” in *Proceedings of International Conference Research on Computational Linguistics*, September 1997, pp. 9008–9022.
- [75] D. Lin, “An information-theoretic definition of similarity,” in *Proceedings of the International Conference on Machine Learning*. Morgan Kaufmann, San Francisco, CA, 1998, pp. 296–304.
- [76] P. Resnik, “Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language,” *Journal of Artificial Intelligence Research*, vol. 11, pp. 95–130, 1999.
- [77] M. Mistry and P. Pavlidis, “Gene Ontology term overlap as a measure of gene functional similarity,” *BMC Bioinformatics*, vol. 9, pp. 327–338, August 2008.
- [78] H. J. Lowe and G. O. Barnett, “Understanding and using the medical subject headings (MeSH) vocabulary to perform literature searches.” *JAMA*, vol. 271, no. 14, pp. 1103–1108, April 1994.
- [79] D. Smedley, S. Haider, B. Ballester, R. Holland, D. London, G. Thorisson, and A. Kasprzyk, “BioMart–biological queries made easy.” *BMC Genomics*, vol. 10, no. 1, 2009, Article Number 22.
- [80] S. J. Furney, D. G. Higgins, C. A. Ouzounis, and N. López-Bigas, “Structural and functional properties of genes involved in human cancer,” *BMC Genomics*, vol. 7, 2006, Article Number 3.

- [81] D. S. Goldberg and F. P. Roth, “Assessing experimentally derived interactions in a small world,” *Proceedings of the National Academy of Sciences*, vol. 100, pp. 4372–4376, April 2003.
- [82] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, pp. 273–297, September 1995.
- [83] C.-C. Chang and C.-J. Lin, *LIBSVM: A library for support vector machines*, 2001. [Online]. Available: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [84] I. T. Nabney, *Netlab: Algorithms for Pattern Recognition*, 2004. [Online]. Available: <http://www.ncrg.aston.ac.uk/netlab>
- [85] P. F. Jonsson and P. A. Bates, “Global topological features of cancer proteins in the human interactome,” *Bioinformatics*, vol. 22, pp. 2291–2297, 2006.
- [86] E. Segal, N. Friedman, D. Koller, and A. Regev, “A module map showing conditional activity of expression modules in cancer,” *Nature Genetics*, vol. 36, pp. 1090–1098, Oct 2004.
- [87] L. Lovasz, “Random walks on graphs: A survey,” *Royal Society Mathematical Studies*, vol. 2, pp. 1–46, 1993.
- [88] C. R. MacCluer, “The many proofs and applications of Perron’s theorem.” *SIAM Review*, vol. 42, no. 3, pp. 487–498, 2000.
- [89] [Online]. Available: <http://sedumi.ie.lehigh.edu>
- [90] R. Kohavi, “A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection,” in *Proceedings of the International Joint Conferences on Artificial Intelligence*, 1995, pp. 1137–1145.

- [91] J. D. Gibbons and S. Chakraborti, *Nonparametric Statistical Inference*, 4th ed. CRC, 2003.
- [92] L.-C. Tranchevent, R. Barriot, S. Yu, and S. V. Vooren, “ENDEAVOUR update: a web resource for gene prioritization in multiple species,” *Nucleic Acids Research*, pp. 377–384, May 2008.
- [93] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, and G. K. Gerber, “Transcriptional regulatory networks in *saccharomyces cerevisiae*.” *Science*, vol. 298, no. 5594, pp. 799–804, 2002.
- [94] N. Luscombe, M. Babu, H. Yu, M. Snyder, S. Teichmann, and M. Gerstein, “Genomic analysis of regulatory network dynamics reveals large topological changes.” *Nature*, vol. 431, no. 7006, pp. 308–312, 2004.
- [95] J. Förster, I. Famili, P. Fu, B. Ø. Palsson, and J. Nielsen, “Genome-scale reconstruction of the *saccharomyces cerevisiae* metabolic network.” *Genome Research*, vol. 13, no. 2, pp. 244–253, 2003.
- [96] N. C. Duarte, S. A. Becker, N. Jamshidi, I. Thiele, M. L. Mo, T. D. Vo, R. Srivas, and B. Ø. Palsson, “Global reconstruction of the human metabolic network based on genomic and bibliomic data.” *Proceedings of the National Academy of Sciences*, vol. 104, no. 6, pp. 1777–1782, 2007.
- [97] B. Zhang and S. Horvath, “A general framework for weighted gene co-expression network analysis.” *Statistical Applications in Genetics and Molecular Biology*, vol. 4, no. 1, 2005, Article Number 17.
- [98] X. Zhou, M. Kao, H. Huang, A. Wong, J. Nunez-Iglesias, M. Primig, O. Aparicio, C. Finch, T. Morgan, and W. Wong, “Functional annotation and network recon-

- struction through cross-platform integration of microarray data.” *Nature Biotechnology*, vol. 23, no. 2, pp. 238–243, 2005.
- [99] Y. Tao, J. Li, C. Friedman, and Y. A. Lussier, “Information theory applied to the sparse gene ontology annotation network to predict novel gene function.” *Bioinformatics*, vol. 23, no. 13, pp. i52–i538, 2007.
- [100] V. A. McKusick, “On lumpers and splitters, or the nosology of genetic disease.” *Perspectives in Biology and Medicine*, vol. 12, pp. 298–312, 1969.
- [101] E. I. Traboulsi, R. Koenekoop, and E. M. Stone, “Lumpers or splitters? the role of molecular diagnosis in leber congenital amaurosis,” *Ophthalmic Genetics*, vol. 27, no. 4, pp. 113–115, 2006.
- [102] D. Zhou, S. Orshanskiy, H. Zha, and C. L. Giles, “Co-ranking authors and documents in a heterogeneous network,” in *Proceedings of International Conference on Data Mining (ICDM)*. Omaha, Nebraska, USA, October 2007, pp. 739–744.
- [103] A. Rzhetsky, D. Wajngurt, N. Park, and T. Zheng, “Probing genetic overlap among complex human phenotypes,” *Proceedings of the National Academy of Sciences*, vol. 104, no. 28, pp. 11 694–11 699, July 2007.
- [104] L. Biesecker, “Mapping phenotypes to language: a proposal to organize and standardize the clinical descriptions of malformations.” *Clinical Genetics*, vol. 68, no. 4, pp. 320–326, Sep 2005.
- [105] A. K. Ramani, R. C. Bunescu, R. J. Mooney, and E. M. Marcotte1, “Consolidating the set of known human protein-protein interactions in preparation for large-scale mapping of the human interactome.” *Genome Biology*, vol. 6, no. 5, 2005, Article Number R40.

- [106] P. Zhang, C. Dilley, and M. Mattson, “DNA damage responses in neural cells: Focus on the telomere.” *Neuroscience*, vol. 145, no. 4, pp. 1439–1448, April 2007.
- [107] E. M. Wijsman, E. W. Daw, C.-E. Yu, H. Payami, E. J. Steinbart, D. Nochlin, E. M. Conlon, T. D. Bird, and G. D. Schellenberg, “Evidence for a novel late-onset Alzheimer disease locus on chromosome 19p13.2.” *American Journal of Human Genetics*, vol. 75, pp. 398–409, January 2004.
- [108] J. Herz, “Apolipoprotein E receptors in the nervous system.” *Current Opinion in Lipidology*, vol. 20, no. 3, pp. 190–196, June 2009.
- [109] K. Takahashi, C. Uchida, R. Shin, K. Shimazaki, and T. Uchida, “Prolyl isomerase, Pin1: new findings of post-translational modifications and physiological substrates in cancer, asthma and Alzheimer’s disease.” *Cellular and Molecular Life Sciences*, vol. 65, no. 3, pp. 359–375, February 2008.
- [110] A. Cole, M. Soutar, M. Rembutsu, L. van Aalten, C. Hastie, H. McLauchlan, M. Pegg, M. Balastik, K. Lu, and C. Sutherland, “Relative resistance of CDK5-phosphorylated CRMP2 to dephosphorylation.” *Journal of Biological Chemistry*, vol. 283, no. 26, pp. 359–375, June 2008.
- [111] N. V. Knoers and E. N. Levtchenko, “Gitelman syndrome,” *Orphanet Journal of Rare Diseases*, vol. 3, July 2008, Article Number 22.
- [112] F. Pociot and M. McDermott, “Genetics of type 1 diabetes mellitus.” *Genes and Immunity*, vol. 3, no. 5, pp. 235–249, February 2002.
- [113] A. Perilhou, C. Turrel-Cuzin, P. Zhang, I. Kharroubi, H. Wang, V. Fauveau, D. K. Scott, C. B. Wollheim, and M. Vasseur-Cognet, “The MODY1 gene for hepatocyte nuclear factor 4 α and a feedback loop control COUP-TFII expression in pancreatic

- beta cells.” *Molecular and Cellular Biology*, vol. 28, no. 14, pp. 4588–4597, July 2008.
- [114] O. Cinek, P. Drevinek, Z. Sumnik, and B. Bendlova, “NEUROD polymorphism Ala45Thr is associated with Type 1 diabetes mellitus in Czech children.” *Diabetes Research and Clinical Practice*, vol. 60, no. 1, pp. 49–56, April 2003.
- [115] A. H. Rosengren, R. Jokubka, D. Tojjar, C. Granhall, O. Hansson, D.-Q. Li, V. Nagaraj, T. M. Reinbothe, J. Tuncel, L. Eliasson, L. Groop, P. Rorsman, A. Salehi, V. Lyssenko, H. Luthman, and E. Renström, “Overexpression of Alpha2A-Adrenergic receptors contributes to type 2 diabetes.” *Science*, vol. 327, no. 5962, pp. 217–220, January 2010.
- [116] R. Rosmond and G. Holm, “A 5-year follow-up study of 3 polymorphisms in the human glucocorticoid receptor gene in relation to obesity, hypertension, and diabetes.” *Journal of the CardioMetabolic Syndrome*, vol. 3, no. 3, pp. 132–135, 2008.
- [117] A. A. Syed, C. G. Halpin, J. A. E. Irving, N. C. Unwin, M. White, R. S. Bhopal, C. P. F. Redfern, and J. U. Weaver, “A common intron 2 polymorphism of the glucocorticoid receptor gene is associated with insulin resistance in men.” *Clinical Endocrinology*, vol. 68, no. 6, pp. 879–884, 2008.
- [118] M. Liu, A. Liberzon, S. W. Kong, W. R. Lai, P. J. Park, I. S. Kohane, and S. Kasif, “Network-Based Analysis of Affected Biological Processes in Type 2 Diabetes Models,” *PLoS Genet*, vol. 3, no. 6, June 2007, Article Number e96.
- [119] M. Oti, M. A. Huynen, and H. G. Brunner, “The biological coherence of human phenome databases.” *The American Journal of Human Genetics*, vol. 85, no. 6, pp. 801–808, December 2009.

- [120] W. G. Feero and A. E. Guttmacher, “Genomewide association studies and assessment of the risk of disease.” *The New England Journal of Medicine*, vol. 363, no. 2, pp. 166–176, 2010.
- [121] K. Wang, M. Li, and M. Bucan, “Pathway-based approaches for analysis of genomewide association studies.” *The American Journal of Human Genetics*, vol. 81, no. 6, pp. 1278–1283, December 2007.
- [122] S. Raychaudhuri, R. M. Plenge, E. J. Rossin, A. C. Y. Ng, S. M. Purcell, P. Sklar, E. M. Scolnick, R. J. Xavier, D. Altshuler, M. J. Daly, and I. S. Consortium, “Identifying relationships among genomic disease regions: Predicting genes at pathogenic SNP associations and rare deletions.” *PLoS Genetics*, vol. 5, no. 6, pp. 359–375, June 2009.
- [123] H. Eleftherohorinou, V. Wright, C. Hoggart, A. Hartikainen, M. Jarvelin, D. Balding, L. Coin, and M. Levin, “Pathway analysis of GWAS provides new insights into genetic susceptibility to 3 inflammatory diseases.” *PLoS One*, vol. 4, no. 11, pp. 359–375, November 2009.
- [124] O. Spjuth, T. Helmus, E. L. Willighagen, S. Kuhn, M. Eklund, J. Wagener, P. Murray-Rust, C. Steinbeck, and J. E. Wikberg, *BMC Bioinformatics*. [Online]. Available: <http://www.bioclipse.net>
- [125] T. H. Ola Spjuth, E. L. Willighagen, S. Kuhn, M. Eklund, J. Wagener, P. Murray-Rust, C. Steinbeck, and J. E. Wikberg, *BMC Bioinformatics*.

Author's Publications

1. **Yongjin Li** and Jagdish C. Patra, "Genome-Wide Inferring Gene-Phenotype Relationship by Walking on the Heterogeneous Network," *Bioinformatics*, vol. 26, no. 9, pp. 2121–2128, 2010.
2. **Yongjin Li** and Jagdish C. Patra, "Integration of Multiple Data Sources to Prioritize Candidate Genes by Using a Discounted Rating System," *BMC Bioinformatics*, vol. 11, no. Suppl 1, S20, 2010.
3. **Yongjin Li** and Jagdish C. Patra, "Network-based Data Integration for Disease Genes Discovery", *IEEE Transactions on Computational Biology and Bioinformatics*, (under revision).
4. **Yongjin Li**, Jagdish C. Patra and Jiabao Sun, "A New Method to Effectively Combine Heterogeneous Data Sources for Disease Gene Prioritization," *IEEE International Conference on Bioinformatics and Bioengineering, BIBE 2009*, pp. 123–129, Taiwan, China.
5. Jiabao Sun, Jagdish C. Patra and **Yongjin Li**, "Functional Link Artificial Neural Network-based disease gene prediction", *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2009*, pp. 3003–3010, Atlanta, USA.
6. **Yongjin Li** and Jagdish C. Patra, "Selection of Features from Protein-Protein Interaction Network for Identifying Cancer Genes," *The IEEE International Conference on Systems, Man, and Cybernetics, SMC 2008*, pp. 1707–1712, Singapore.

Appendix A

Detailed Results on 36 Diseases

In this Appendix, we show the results of integration models using Dataset #11 as benchmark data. Table A.1 shows the results of MKL algorithm with different parameters $lb = 0, 0.25, 0.5, 0.75$ and 1. Table A.2 shows the results of RWRM and four RABI models.

Table A.1: The performance of MKL with different lb values, evaluated on each of the 36 diseases, measured by AUC values (%).

SN	Disease Name	$\mathcal{N}_{dg}$	AUC Value(%)				
			$lb = 0$	$lb = 0.25$	$lb = 0.5$	$lb = 0.75$	$lb = 1$
1	Amyotrophic lateral sclerosis	8	78.9	77.9	77.3	75.6	77.3
2	Age-Related Macular Degeneration	12	52.3	54.9	57.5	59.6	57.5
3	Bardet-Biedly Syndrome	13	64.4	65.1	66.1	66.9	66.1
4	Breast Cancer	15	59.5	60.9	61.8	62.8	61.8
5	Chondrodysplasia punctata	14	98.2	98.6	98.9	98.9	98.9
6	Noonan Syndrome, Costello syndrome, Cardiofaciocutaneous Syndrome	9	79.8	82.5	84.9	87.0	84.9
7	Hypertrophic cardiomyopathy	15	78.2	85.0	86.0	86.4	86.0
8	Congenital myasthenic syndromes	10	87.8	89.2	99.4	99.8	99.4
9	Charcot Marie Tooth Disease	24	53.5	53.4	51.7	50.8	51.7
10	Cataract	11	77.8	81.6	85.0	87.1	85.0
11	Dilated cardiomyopathy	20	62.5	67.0	70.5	72.2	70.5
12	Epidermolysis bullosa	16	92.7	93.4	95.8	97.7	95.8
13	Ehlers Danlos syndrome	8	93.3	96.4	96.1	96.5	96.1
14	Esophageal carcinoma	10	74.2	72.4	69.8	65.4	69.8
15	Essential hypertension	12	70.2	65.1	64.3	63.8	64.3
16	Fanconi anemia	12	79.4	79.6	79.9	84.7	79.9
17	Glioma of brain	21	72.7	72.8	71.5	72.4	71.5
18	Hereditary nonpolyposis colorectal cancer	7	93.9	94.6	94.6	94.6	94.6
19	Hermansky-Pudlak syndrome	8	87.8	82.8	84.0	84.5	84.0
20	Inflammatory Bowel Disease	8	74.1	73.6	71.8	66.3	71.8
21	Leber congenital amaurosis	9	95.4	95.8	94.2	86.6	94.2
22	Limb-Girdle Muscle Dystrophy	14	55.2	59.0	64.4	67.3	64.4
23	Long QT Syndrome	9	67.7	72.7	75.2	77.3	75.2
24	Leigh Syndrome	24	89.0	91.2	90.7	89.7	90.7
25	Microphthalmia	9	67.3	70.1	71.7	74.0	71.7
26	Mitochondrial complex I deficiency disorders	16	99.8	99.8	99.8	99.8	99.8
27	Maturity-Onset Diabetes of the Young	19	42.8	43.7	44.2	46.3	44.2
28	Nonsyndromic hearing loss	47	93.9	94.6	94.8	93.9	94.8
29	Obesity	13	85.2	86.8	88.2	91.2	88.2
30	Parkinson	8	72.9	72.8	75.0	78.5	75.0
31	Pheochromocytoma	10	72.3	72.0	68.5	67.8	68.5
32	Retinitis Pigmentosa	23	99.1	96.4	95.9	94.2	95.9
33	Spinocerebellar Ataxia	13	90.6	89.7	89.7	88.0	89.7
34	Thyroid carcinoma	8	50.6	51.3	55.4	60.0	55.4
35	Xeroderma Pigmentosum	8	95.5	96.5	97.5	97.6	97.5
36	Prostate Cancer	14	42.0	49.8	51.0	52.1	51.0

Table A.2: The comparison of several integration methods used in this thesis, on each of 36 diseases, measured by AUC values (%).

SN	Disease Name	$\mathcal{N}_{dg}$	AUC Value(%)					
			MKL	1CSVM+NDOS	RWR+NDOS	1CSVM+DRS	RWR+DRS	RWRM
1	Amyotrophic lateral sclerosis	8	77.3	77.1	75.1	90.6	87.9	96.1
2	Age-Related Macular Degeneration	12	57.5	72.3	72.4	77.2	78.1	88.8
3	Bardet-Biedly Syndrome	13	66.1	83.6	82.2	62.8	83.6	76.2
4	Breast Cancer	15	61.8	75.5	74.3	80.3	82.4	83.2
5	Chondrodysplasia punctata	14	98.9	95.2	95.9	91	92.1	97
6	Noonan Syndrome, Costello syndrome, Cardiofaciocutaneous Syndrome	9	84.9	96.4	96.5	94.6	90	98.8
7	Hypertrophic cardiomyopathy	15	86	88.6	88.6	96.8	95.7	94.6
8	Congenital myasthenic syndromes	10	99.4	100	100	99.6	99.6	100
9	Charcot Marie Tooth Disease	24	51.7	68.2	70.4	86.5	90.9	90.8
10	Cataract	11	85	87.3	90.7	76.6	83.9	90
11	Dilated cardiomyopathy	20	70.5	85.4	86.6	82.3	88.7	93.9
12	Epidermolysis bullosa	16	95.8	99.5	99.3	97	96.2	99.7
13	Ehlers Danlos syndrome	8	96.1	97.3	97.4	95.1	93.6	96.9
14	Esophageal carcinoma	10	69.8	67	71.3	75.2	73.8	70.4
15	Essential hypertension	12	64.3	56.2	53.5	75.1	77.7	76
16	Fanconi anemia	12	79.9	97.2	99.7	88.3	92.3	99.8
17	Glioma of brain	21	71.5	85.3	85.6	84.4	86	83.7
18	Hereditary nonpolyposis colorectal cancer	7	94.6	96.9	98.3	98.4	98.1	97.9
19	Hermansky-Pudlak syndrome	8	84	87.5	87.9	85.8	91.1	99.1
20	Inflammatory Bowel Disease	8	71.8	69.6	70.9	75.5	83.3	72.1
21	Leber congenital amaurosis	9	94.2	79.7	89	66.1	77.9	85.7
22	Limb-Girdle Muscle Dystrophy	14	64.4	86.6	88.8	81.4	90.3	96.4
23	Long QT Syndrome	9	75.2	88.1	85.9	84.3	90.1	91.9
24	Leigh Syndrome	24	90.7	91.7	91.1	88.6	92.1	88.7
25	Microphthalmia	9	71.7	80.7	82.4	78.8	78.6	74.8
26	Mitochondrial complex I deficiency disorders	16	99.8	99.7	99.8	99.8	99.8	99.7
27	Maturity-Onset Diabetes of the Young	19	44.2	55.8	55.1	80.5	75.3	83.1
28	Nonsyndromic hearing loss	47	94.8	93.4	96.1	87.8	93.2	91.1
29	Obesity	13	88.2	93.4	93.5	94.8	94.8	98.5
30	Parkinson	8	75	93.9	92.4	94.3	91.1	88.9
31	Pheochromocytoma	10	68.5	79.7	75.8	94.8	92.7	89.2
32	Retinitis Pigmentosa	23	95.9	88.9	93.7	71.7	79.5	86.7
33	Spinocerebellar Ataxia	13	89.7	79.4	81.8	79.9	79.3	78.1
34	Thyroid carcinoma	8	55.4	76.6	77.9	81.9	74.9	80.9
35	Xeroderma Pigmentosum	8	97.5	100	99.9	99.5	99.3	99.1
36	Prostate Cancer	14	51	75.5	79.2	84.1	84.5	84

Appendix B

Softwares Used in This Work

In this Appendix, we provided the usage of two softwares used in thesis thesis, MimMiner [56] and BioMart [79].

B.1 MimMiner

MimMiner [56] is not a public available software. It can be obtained by requesting from the author, Prof. H.G. Brunner (h.brunner@antrg.umcn.nl). Below is a brief description of the software. Detailed information is available at the manuscript of the software.

MimMiner was written in Perl. These are the Perl scripts required to parse the OMIM diseases database using Medical Subject Heading (MeSH) terms. The OMIM disease records are converted to MeSH term vectors. This enables the phenotypic similarity between the diseases to be systematically determined.

Usage:

- Place all Perl scripts in the same directory.
- To find out what arguments to use for a Perl script run it without arguments.
- Procedure:
 - 1) Download OMIM text file and MeSH XML file.

- 2) Convert MeSH XML file to MeSH synonyms file and MeSH tree file.
- 3) Filter the MeSH synonyms and tree files for those term categories you wish to use for the text mining.
- 4) Parse the OMIM text file using the chosen MeSH term categories.
- 5) Expand term vectors using the MeSH hierarchy information if desired.
- 6) Adjust term weights if desired.
- 7) Calculate the OMIM record similarity/distance matrix.

B.2 BioMart

BioMart is a query-oriented data management system developed jointly by the Ontario Institute for Cancer Research (OICR) and the European Bioinformatics Institute (EBI). This system provides a interface for querying different biological data sources in an integrated manner. One can cross query most of the major biological databases using this system. Users can submit the query through the web interface MartView (<http://www.biomart.org/biomart/martview>), or through a Perl API as well as RESTful and SOAP oriented web services.

BioMart is completely Open Source, licensed under the LGPL, and freely available to anyone without restrictions. A number of third party softwares include the BioMart Plugin, such as Bioclipse [124, 125], biomaRt, Cytoscape, Galaxy, Gitools, Ruby API, Taverna, WebLab. We used the biomaRt package, which is a R package.

The biomaRt package is an add-on package for R (<http://www.r-project.org/>) that provides query ability to BioMart databases. In this work, we used the biomaRt package to extract the gene-phenotype information from OMIM [5] and to find the gene location information in the chromosome. Below are some scripts used in this research:

1) To install this package, start R and enter:

```
source("http://bioconductor.org/biocLite.R")  
biocLite("biomaRt")
```

2) load the biomaRt package

```
library("biomaRt")
```

3) to select the Ensembl database and the hsapiens dataset in this database.

```
ensembl = useMart("ensembl", dataset = "hsapiens_gene_ensembl")
```

4) Extract all the gene-phenotype relationships from OMIM

```
getBM(attributes = c("entrezgene", "mim_gene_accession", "mim_morbid_accession"),  
mart=ensembl)
```

5) to find the gene location information in the chromosome

```
allGenes = getBM(c("ensembl_gene_id", "chromosome_name", "start_position", "end_position",  
"strand"), mart=mart), then use the allGenes data.frame to lookup the 100 genes closest  
to the gene of interest.
```