Neural Inductive Matrix Completion for Predicting Disease-Gene Associations

Thesis by

Siqing Hou

In Partial Fulfillment of the Requirements

For the Degree of

Masters of Science

King Abdullah University of Science and Technology

Thuwal, Kingdom of Saudi Arabia

May, 2018

EXAMINATION COMMITTEE PAGE

The thesis of Siqing Hou is approved by the examination committee

Committee Chairperson: Xin Gao
Committee Members: Vladimir Bajic, Robert Hoehndorf

ABSTRACT

Neural Inductive Matrix Completion for Predicting Disease-Gene Associations

Siqing Hou

In silico prioritization of undiscovered associations can help find causal genes of newly discovered diseases. Some existing methods are based on known associations, and side information of diseases and genes. We exploit the possibility of using a neural network model, Neural inductive matrix completion (NIMC), in disease-gene prediction. Comparing to the state-of-the-art inductive matrix completion method, using neural networks allows us to learn latent features from non-linear functions of input features.

Previous methods use disease features only from mining text. Comparing to text mining, disease ontology is a more informative way of discovering correlation of diseases, from which we can calculate the similarities between diseases and help increase the performance of predicting disease-gene associations.

We compare the proposed method with other state-of-the-art methods for predicting associated genes for diseases from the Online Mendelian Inheritance in Man (OMIM) database. Results show that both new features and the proposed NIMC model can improve the chance of recovering an unknown associated gene in the top 100 predicted genes. Best results are obtained by using both the new features and the new model. Results also show the proposed method does better in predicting associated genes for newly discovered diseases.

ACKNOWLEDGEMENTS

I would like to thank my advisor Prof. Xin Gao for his support in overcoming obstacles I have faced through my work. I am grateful to Dr. Peng Yang for his collaboration and guidance on this topic.

I would like to thank Prof. Robert Hoehndorf and Maxat Kulmanov for providing information and data about disease ontology. In addition, I would like to express my gratitude to numerous students within the group and within the CEMSE division for their feedback.

Finally I would like to thank my committee members, Prof. Xin Gao, Prof. Vladimir Bajic and Prof. Robert Hoehndorf, for their valuable advice on this thesis.

TABLE OF CONTENTS

Examination Committee Page **2**

Copyright **3**

Abstract **4**

Acknowledgements **5**

List of Figures **8**

1 Introduction **9**

2 Background **12**

2.1 Introduction to Genes and Genetic Diseases 12

2.2 Predictions of Disease Genes 14

2.3 Cold-Start Problem 15

2.4 Human Phenotype Ontology 17

3 Related Work **18**

3.1 Katz on the Heterogeneous Network 18

3.2 CATAPULT 20

3.3 Inductive Matrix Completion 20

3.4 Neural Collaborative Filtering 22

3.5 Deep Matrix Factorization 23

4 Neural Inductive Matrix Completion **25**

4.1 Non-linear Latent Features 25

4.2 Restricting Prediction Values 25

4.3 Biased Training 26

5 Experiments **28**

5.1 Data 28

5.1.1 Features 28

5.2 Evaluation Methods 30
5.2.1 Evaluation Metrics 30
5.2.2 Parameter Settings 31
5.3 Results 31
5.3.1 3-Fold Cross-validation on the Old Dataset 31
5.3.2 3-Fold Cross-validation on the New Dataset 31
5.3.3 3-Fold Cross-validation on Singleton Diseases 33
5.3.4 Prediction of Newly Discovered Associations 35

6 Parameter Sensitivity Analysis **37**

7 Concluding Remarks **40**
7.1 Implementation 40
7.2 Future Research Work 40

References **42**

LIST OF FIGURES

3.1 Network Structure of Inductive Matrix Completion 22
3.2 Network Structure of Neural Matrix Factorization Model 23
3.3 Network Structure of Deep Matrix Factorization Model 24

4.1 Network Structure of Neural Inductive Matrix Completion 27

5.1 Top-k recall curve in 3-fold cross-validation on the old dataset. 32
5.2 Precision-recall curve in 3-fold cross-validation on the old dataset. . . 32
5.3 Top-k recall curve in 3-fold cross-validation on the new dataset. . . . 33
5.4 Precision-recall curve in 3-fold cross-validation on the new dataset. . 34
5.5 Precision-recall curve in 3-fold cross-validation on singleton diseases of the new dataset. 34
5.6 Top-k recall curve on prediction of newly discovered associations. . . 35
5.7 Precision-recall curve on prediction of newly discovered associations. . 36

6.1 Sensitivity analysis of the length of latent feature. 37
6.2 Sensitivity analysis of the regularizer coefficient. 38
6.3 Sensitivity analysis of the leaky ReLU parameter. 39

Chapter 1

Introduction

The study of relationships between genetic disorders and genes has been a major challenge in bioscience [1][2]. Knowing the causal genotype of a genetic disorder improves our understandings of the underlying molecular basis, which helps us develop treatment method of that disease. Associations between genes and genetic disorders are being continuously updated in *Online Mendelian Inheritance in Man* (OMIM) [3]. Associations in OMIM is curated by biological experiments. Most genetic diseases recorded in OMIM are associated with one and only one gene, while the other diseases are associated with two or more genes. Although biological experiments are necessary to identify an association, blind screening of all genes and diseases is too expensive and time-consuming. *In silico* predictions of disease-gene associations help select the possible unknown associations and reduce the amount of work on biological experiments. For a disease with known associations, a disease-gene prediction system should be able to prioritize other possibly related genes. A more successful system should also be able to predict possible associations for a new genetic disease without any known associations with the help of other information of that disease.

A simple idea from the design of recommender systems, such as recommending movies that a user may be interested in [4], is to recommend potential associations based on the associations we already have. Solutions like matrix completion and collaborative filtering work in this way. However, models in this category all face the cold-start problem [5], which means they do not have the ability to predict for those diseases that have no known associations. To solve the cold-start problem, side infor-

mation other than disease-gene associations is needed. Some network-based methods have been built, including PRINCE[6], RWRH[7], Katz[8] and CATAPULT[1], by exploiting the usage of the functional gene interaction network and the disease similarities network. Both networks can be joint together with the association network as a heterogeneous network. Network node similarities and random walk procedures are used in these methods to make use of side information.

Network information is not always available for new diseases. To make use of side information other than networks, inductive matrix completion(IMC) [2] was introduced as a feature-based method which can adopt any kind of disease features and gene features. The network information is easy to be translated into feature vectors by doing eigendecomposition on the adjacency matrix or similarity matrix and keeping only top eigenvectors. As a result, a network with $N \times N$ square matrix representation will be reduced to a matrix of $N \times k$, representing a feature vector of length k for each of the N entities. It is proved that IMC works effectively particularly for diseases without previously known associations. Additional features used in inductive matrix completion including microarray measurements of gene-expression levels in different tissue samples, disease-gene associations in other species and disease features based on the term-document count from the text describing the diseases. The performance of IMC showed us a great potential of feature-based methods in learning from information compared to traditional network-based methods.

In this thesis, I explored the possibility of improving the state-of-the-art of disease-gene predictions in two directions.

1. Inductive matrix completion learns latent features based on linear combination of the input features, and the output prediction is the inner product of the latent features of the disease and the latent features of the gene. However, we believe that there could be a non-linear relationship between the prediction output and the input features. Some recently proposed recommender systems use neural

network techniques in matrix completion, such as neural collaborative filtering [9] and deep matrix factorization [10]. Inspired by these methods, we propose a model called neural inductive matrix completion to improve the expressivity of inductive matrix completion. Our model uses the Leaky ReLU function and the sigmoid function to introduce non-linear factors.

2. Other than mining disease features from the text, we introduce a more informative way of discovering correlation of diseases by the semantic similarities of disease ontology. We show that by using these features, we always obtain improved results of final prediction.

Experimental results show that both strategies work well in improving the performance of disease-gene predictions.

Chapter 2

Background

In this chapter, we review the background concerning our work in disease-gene predictions, including some basic concepts, introduction to our problem and some information sources that have been used in our work.

2.1 Introduction to Genes and Genetic Diseases

The genetic information of human beings is mainly stored in 23 pairs of chromosomes. They vary largely in size and shape [11]. Twenty-two pairs of them are autosomal chromosomes which share the same morphology within one pair. The remaining pair is called the sex chromosomes, which determines our sex. Females have a pair of X chromosomes, while males have one X and one Y chromosomes. Chromosomes are made of proteins and DNA. Each DNA is a long molecular, which needs to be wrapped tightly around proteins. Genetic information is encoded in DNA in the form of base pairs. Genes are special units of DNA, which contain a number of base pairs within one DNA. To our knowledge, the 46 human chromosomes contain almost 3 billion base pairs of DNA, which may consist of about 30,000 - 40,000 protein-coding genes [11]. Only 5% of the genome base pairs are identified as protein-coding genes. This density of genes also varies from one chromosome to another [11].

The Human Genome Project [12], an international collaborated research project, has been playing an important role in our understanding of human genes. Its goal is to completely map and understand all the human genes. All of the human genes

are called the human genome. It is mainly based on sequencing of DNA molecules of human beings. Although not complete yet, it is being continuously refined and bringing us with more and more direct views of our genome.

It is also known that most organisms, especially animals and plants, also use DNA to encode their genetic information. The sequencing and analyzing of genes of some model organisms have been well studied including *Caenorhabditis elegans* [13], fruit fly (*Drosophila melanogaster*) [14], the house mouse (*Mus musculus*) [15], etc. The genomes of other species are also helpful in understanding human genes[1]. As other species may share common ancestors in simpler forms with human beings, their genomes contain genes with the same function as genes in human genome. The studies of these genes in other species can help us improve our understanding of the corresponding human genes. The sequences and annotations of genome data of human beings and some other species can be easily acquired from online databases like RefSeq [16] and Ensemble [17].

Genetic diseases refer to the diseases caused by abnormalities in the genome. Since we often refer to the genetic diseases as phenotypes, the two phrases will be used interchangeably in this thesis. While the abnormalities in the genome are presented prior to birth, the disease conditions may occur either from birth or at a later time of the life. Genetic diseases may be hereditary, passed down from their parents. They can also be caused by new mutations of DNA base pairs that originated from the patients' genome. Genetic diseases may influence near every part of the human body. Most of the genetic diseases are directly related to a mutation of a single gene. However, diseases caused by abnormalities of multiple genes also exist. Some of the common diseases like diabetes and cancer belong to this class. In these cases, not a single gene mutation can determine whether a person has a disease or not.

2.2 Predictions of Disease Genes

Because of the role of genes in genetic diseases, it will be helpful if we can identify the genes causing a disease. Some diseases are first observed without any knowledge of the molecular and genetic basis. It will reduce the time and cost of finding out the causal genes if we can quickly target a small set of possible causal genes instead of blindly screening the whole human genome. The predictions can be based on our existing knowledge of human genes and other diseases, as well as clinical observations of the diseases and the patients. Discovery of new causal genes is also helpful in the study of complex genetic diseases such as diabetes and cancer. The importance of this task motivates research on predicting disease genes. A model can help identify the causal genes of a disease, either by selecting a small set of possible causal genes or by ranking all genes with regards to their possibility of being the causal genes. These two approaches are actually identical and finally can help reduce the number of genes that we should conduct experiments on.

The problem can be easily formulated as a link prediction problem in a bipartite network. A bipartite network can be constructed by two kinds of nodes. One represents genes, and the other represents genetic diseases. There are links already known to us connecting pairs of diseases and their causal genes. The goal is to identify any possible links between the two parts of the network, or in a more precise way, to give a possibility that a link can occur between any pair of the two kinds of nodes.

In an equivalent way, we can also see the problem as matrix completion of the disease-gene association matrix [2]. We can construct a disease-gene association matrix. Each row represents a gene and each column represents a disease. The matrix contains values of 1s and 0s. While 1s represent there is a known connection between the gene and the disease, 0s represent the opposite. It is noticed that the opposite does not mean there is no causal connection between them, but means they are not known to have a connection. Matrix completion on the disease-gene matrix is to fill

a new value for the 0s. One solution is to fill them with 0s and 1s as the prediction of them having a connection. In practice, they are filled with a score indicating their possible connections. The higher the score, the higher the possibility that the gene serves as the cause of the disease.

We can easily notice that these two formulations are exactly equivalent, using different representations of the bipartite disease-gene network. The network representation is easy to be integrated into network-based methods like random walk and network statistics. The matrix representation can be used in numerical optimization methods such as matrix factorization.

The two formulations are also used in recommender systems. However, unlike the problem that recommender system is facing, the disease-gene association matrix does not contain any negative ratings or rating scores. The available data sources only contain confirmed associations of diseases and causal genes, but no confirmations on negative associations between any pair of disease and gene. The problem of recommender systems can be regression of the rating scores, or the binary classification of "like" or "dislike". Predicting disease-gene associations is a case of unary classification, where only positive samples are given and we are responsible to identify if a new sample, in this case, a disease-gene pair, belongs to the positive class or not. Since we do not have the availability of negative data, unobserved pairs play the role of negative data in our case. We will discuss in the following two chapters how this is done in related methods and our proposed method.

2.3 Cold-Start Problem

A major challenge in recommender systems, the cold-start problem, is also faced by disease-gene predictions. Recommender systems often face the challenge that a new user or a new item comes online, while no ratings are yet available for them. In the case of disease-gene predictions, a new disease may come without knowing its

molecular bases, or a gene is not known to cause any disease yet, but it is probably the causal gene of some diseases[2].

It is known that the cold-start problem is impossible to be effectively solved with only the associations of the two kinds of entities involved. Side information must be used for predictions in cold-start cases.

Side information can be used in different forms. A way to integrate side information is to introduce the similarity measure within each type of entity [1]. For example, a disease-disease similarity can be used as useful information of those diseases without known associations [18]. In the network formulation, it can be seen as the union of the original bipartite network and the disease-disease similarity network. The network after the union is a heterogeneous network, with different kinds of nodes, and links within the same kind of nodes and across two kinds of nodes.

Side information can also be introduced by adding a new kind of entities. Entities like proteins and drugs also play important roles in the research of genetic diseases, and highly interacts with diseases and genes. Information like drug-disease associations can also be added to the heterogeneous network. As a result, we will have a heterogeneous network with more kinds of nodes. It is easy to extend network-based method in this case.

Other than enriching the heterogeneous network, side information can also be feature vectors. Feature vectors are commonly used in matrix-based or learning-based methods. We will explain in the following two chapters how the feature vectors are used in CATAPULT[1], IMC[2] and our proposed method.

The first two kinds of side information can also be easily transformed into feature vectors by eigendecomposition or principal component analysis. We will introduce these techniques in Chapter 5.

2.4 Human Phenotype Ontology

One of our contributions is to introduce ontology as a source of side information in disease-gene predictions. An ontology is a computational representation of a domain of knowledge based upon a controlled, standardized vocabulary for describing entities and the semantic relationships between them. *Human Phenotype Ontology* (HPO) [19] is a project aiming at providing a standardized ontology system for phenotypic abnormalities in human genetic disorders. We obtained a pairwise similarity matrix of diseases from HPO and used it as an important source of side information of diseases.

Chapter 3

Related Work

In this chapter, I will introduce some methods that have been developed for predicting disease-gene associations, as well as some related methods for recommender systems that use neural network models.

3.1 Katz on the Heterogeneous Network

Katz measure [8] is a graph-based method for finding similar nodes. It is widely used in link prediction. It is firstly used for disease-gene predictions in [1] together with CATAPULT which I will introduce in the following section. Let A be the adjacency matrix of an undirected, unweighted network. $A_{ij} = 1$ if node i and node j is connected, while $A_{ij} = 0$ if they are not connected. We can count the number of walks from i to j of different lengths. It is known that $(A^l)_{ij}$ is the number of walks of length l connecting i and j. We want a single measure to summarize the number of walks of different lengths and to represent the similarity between two nodes. We could choose a series of non-negative parameters β_l and define the similarity.

$$S_{ij} = \sum_{l=1}^{k} \beta_l (A^l)_{ij}.$$

It can also be written in a matrix notation:

$$S = \sum_{l=1}^{k} \beta_l A^l.$$

If we let $k \to +\infty$ and choose $\beta_l = \beta^l$, the measure can be written as:

$$S^{katz} = \sum_{l \geq 1} \beta^l A^l = (I - \beta A)^{-1} - I,$$

where $\beta \leq \frac{1}{\|A\|_2}$. In the case of a weighted network, we can simply just replace A by a weighted adjacency matrix. In practice, walks of longer lengths have much less information than walks of shorter lengths. So we just choose a small number $k = 3$. [1]

The method uses Katz on a heterogeneous network containing nodes of genes, human diseases, and diseases of other species. Let G be the gene-gene network, let P be the bipartite network between genes and diseases. $P = [P_{H_S} P_S]$ where P_{H_S} is associations between genes and human diseases, and P_S is associations between genes and diseases of other species. Similarly, the disease-disease similarity Q can be written as

$$Q = \begin{bmatrix} Q_{H_S} & 0 \\ 0 & Q_S \end{bmatrix},$$

where Q_{H_S} is the disease-disease similarity of human diseases, while $Q_S = 0$ because we do not have information about similarities between diseases of other species.

To combine them together, we write a adjacency matrix C of the heterogeneous network:

$$C = \begin{bmatrix} G & P \\ P^\top & Q \end{bmatrix}.$$

Recall the truncated Katz measure:

$$S_{ij}^{katz} = \sum_{l=1}^{3} \beta^l (C^l)_{ij}.$$

Since we only care about the results between genes and human diseases, we can write

this part separately [1] by:

$$S_{H_S}^{katz} = \beta P_{H_S} + \beta^2(GP_{H_S} + P_{H_S}Q_{H_S}) + \beta^3(PP^\top P_{H_S} + G^2 P_{H_S} + GP_{H_S}Q_{H_S} + P_{H_S}Q_{H_S}^2).$$

3.2 CATAPULT

CATAPULT[1] is a supervised learning method to predict disease-gene associations. It is based on improving the Katz measure above by learning the coefficients of different kinds of walks. The input is a feature vector for a disease-gene pair, while the output is a score for the pair. Features are extracted from the heterogeneous network using different kinds of walks. The improvement is not only to allow coefficients to be learned but also to distinguish the different roles of different kinds of nodes in the heterogeneous network.

CATAPULT solves the unary classification problem by using a biased support vector machine [20] as well as the bagging technique [21]. First, biased SVM uses different penalty factors for false positive and false negative. Second, bagging is an iterative technique that draws a set of negative samples from the unobserved of the equal size to positive samples.

Although it is using feature vectors for learning, it is still a network-based method that extracts features of a pair of nodes from the network structure. It does not have the ability to use any disease features or gene features.

3.3 Inductive Matrix Completion

Inductive matrix completion solves the disease-gene prediction problem in a matrix factorization based method. Given a disease-gene association matrix of $P \in R^{N_g * N_d}$, where N_g is the number of genes and N_d is the number of diseases. The entries in P are 0s and 1s. Since 0s in the association matrix represent unobserved entries, which

is probably "1" if it is a true association. We want to fill the 0 entries with values representing its possibility of being a "1". A higher value means that it is more likely to be an unobserved "1".

Since it is impossible to solve the cold-start problem with only the association matrix, inductive matrix completion assumes there is additional information of each disease and gene as a feature vector. Suppose gene i has a feature vector $x_i \in R^{f_g}$ and disease j has a feature vector $y_j \in R^{f_d}$. Let $X \in R^{N_g \times f_g}$ and $Y \in R^{N_d \times f_d}$ be the matrices containing features of all genes and diseases.

Inductive matrix completion use a function of gene features and disease features to represent the prediction value to be filled in:

$$\hat{p}_{ij} = f(x_i, y_j).$$

The original inductive matrix completion proposed in [2] uses a linear function to represent the prediction value:

$$f_{imc}(x_i, y_j) = (W^\top x_i)^\top (H^\top y_j),$$

where $W \in R^{r \times f_g}$ and $H \in R^{r \times f_d}$. We call $W^\top x_i$ and $H^\top y_j$ the latent factors of gene i and disease j. This model is based on the assumption that the matrix P can be approximated by a low rank matrix $\hat{P} = X^\top W H^\top Y$ whose rank is at most r. The approximation can be trained by minimizing:

$$\min_{W \in R^{r \times f_g}, H \in R^{r \times f_d}} \ell(P, \hat{P}) + \frac{\lambda}{2} \left(\|W\|_F^2 + \|H\|_F^2 \right),$$

where λ is a regularization parameter, ℓ is the loss function between the original association matrix P and its approximation $\hat{P}$. We use the ℓ_2 loss function in our

experiments:

$$\ell(P, \hat{P}) = \left\| P - \hat{P} \right\|_F^2$$

IMC can also be represented as a neural network, taking the feature vector of a disease and the feature vector of a gene as input, generating the prediction value. Figure 3.1 shows the equivalent neural network model of IMC.

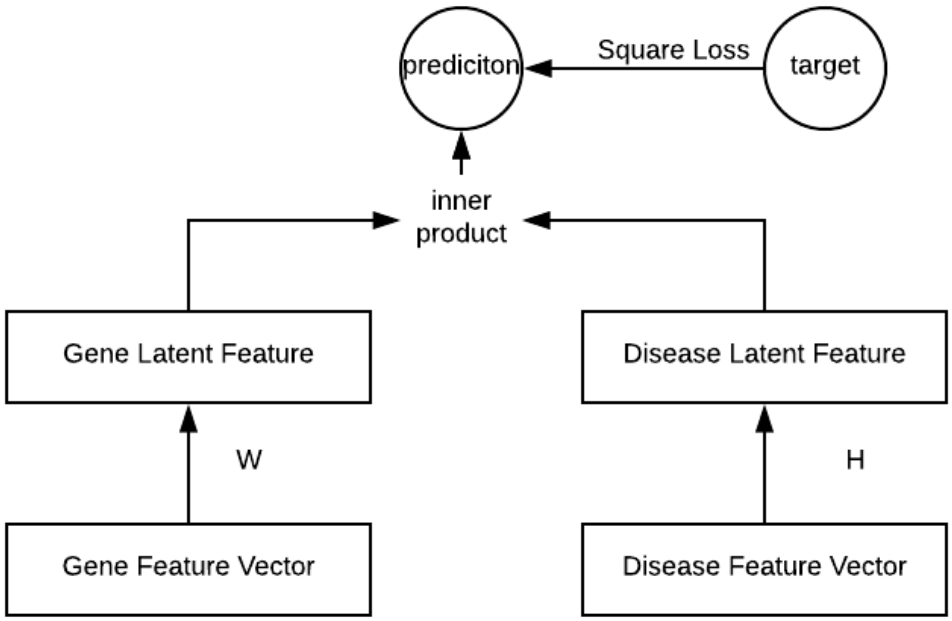

Figure 3.1: Network Structure of Inductive Matrix Completion

3.4 Neural Collaborative Filtering

Neural Collaborative Filtering (NCF) [9] is a general framework for recommender systems using neural networks. Three networks are proposed in [9] under this framework, general matrix factorization (GMF), multi-layer perceptron (MLP) and neural matrix factorization (NeuMF). NCF uses a one-hot vector as the input feature of each user and item. It does not utilize side information to solve the cold-start problem. GMF performs linear transforms to get the latent features, while MLP uses a

non-linear kernel [9]. NeuMF network is a fusion of GMF and MLP. To combine the two networks together, the last hidden layers of the two networks are concatenated together in NeuMF. Figure 3.2 [9] shows the network structure of NeuMF.

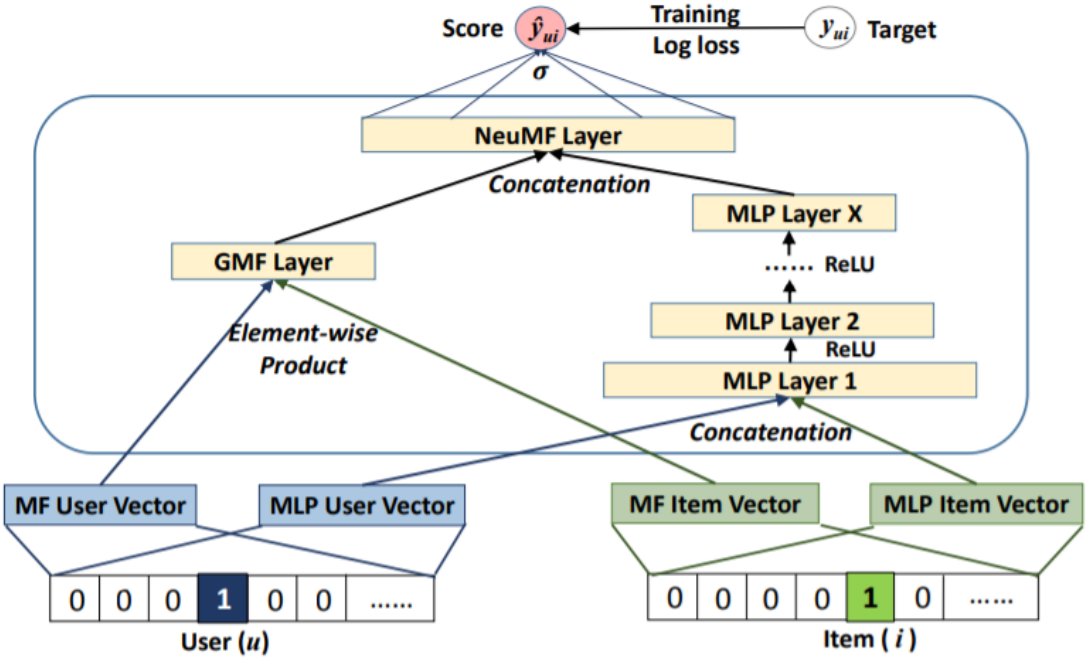

Figure 3.2: Network Structure of Neural Matrix Factorization Model

3.5 Deep Matrix Factorization

Deep Matrix Factorization (DMF) [10] is also a neural network model for recommender systems. Similar to MLP [9], it uses densely connected networks to learn latent features. But unlike MLP, which concatenates the latent features from the first hidden layer, DMF uses two densely connected networks to learn latent features separately for users and items, and the latent features are concatenated in the last hidden layer. The input vector is also different from the NCF framework. DMF uses the row vector and column vector directly from the interaction matrix. Figure 3.3 [10] shows the network structure of DMF.

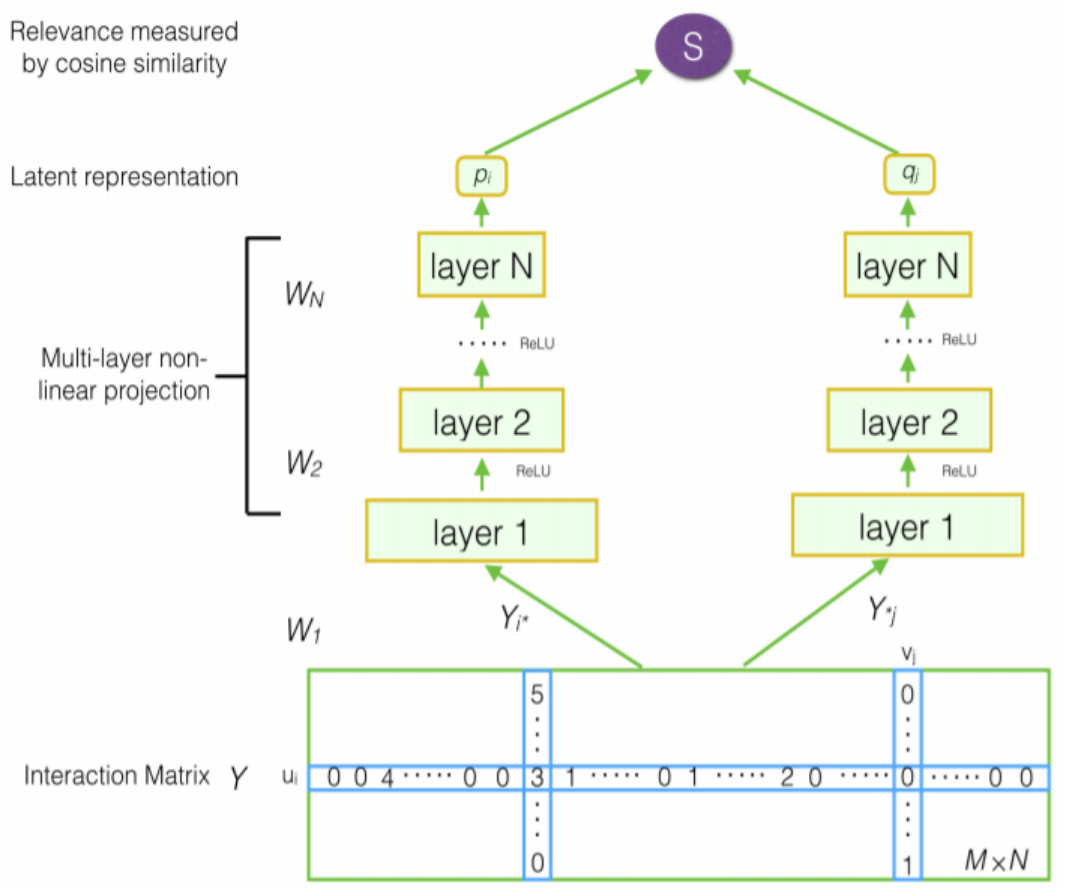

Figure 3.3: Network Structure of Deep Matrix Factorization Model

Chapter 4

Neural Inductive Matrix Completion

In this chapter, we propose a new approximation model, called neural inductive matrix completion (NIMC), based on a neural network where non-linear approaches are used for improvement.

4.1 Non-linear Latent Features

A possible improvement to the IMC model is to learn latent features from a non-linear function of the input features:

$$f_{nimc}(x_i, y_j) = \big(a(W^\top x_i)\big)^\top a(H^\top y_j),$$

where a is an activation function. We use *leaky ReLU* as the activation function in our experiments.

$$leaky\_relu_\alpha(t) = \begin{cases} t & t \geq 0 \\ \alpha t & t < 0 \end{cases}$$

where α is set to 0.1 in the experiments.

4.2 Restricting Prediction Values

As in IMC, prediction values can be any real number. However, suppose a prediction of a positive sample is greater than 1 during the training case, based on the ℓ_2-norm we used, it will be punished and descended towards 1. However, we believe a greater

value is a strong evidence that it is positive, thus not to be punished in this case. In opposite, prediction of negative samples smaller than 0 should not be ascended to 0.

To achieve this goal, the prediction value is restricted between 0 and 1 by sigmoid function.

$$\hat{p}_{ij} = S(f_{nimc}(x_i, y_j) - \beta),$$

where

$$S(t) = \frac{e^t}{1 + e^t}.$$

4.3 Biased Training

In the case of IMC, it treats the unobserved pairs just as negative pairs. However, this will lead to an extremely unbalanced case where the number of negative pairs is around 10,000 times bigger than the number of positive pairs. Instead of using bagging technique like CATAPULT, we choose only a biased approach to deal with negative samples. we revise the objective function to minimize as:

$$\min_{W \in R^{r \times f_g}, H \in R^{r \times f_d}} \left(\sum_{i,j} c_{ij} \ell(p_{ij}, \hat{p}_{ij}) \right) + \frac{\lambda}{2} \left(\|W\|_F^2 + \|H\|_F^2 \right), f$$

where c_{ij} is the entry weight given to the matrix entry of gene i and disease j. This allows us to set higher weights for important entries. While all known associations are confirmed, all unknown entries are undecided. It is believed that false negative should be punished more than false positive in this situation. Thus we set $c_{ij} = 1$ when ij is a known association, otherwise we set $c_{ij} = c_{negative}$ where $c_{negative} < 1$.

We do not use bagging because using all entries in the association matrix in the same iteration makes it easier to implement the optimization in a tensor representation, thus the optimization will be more efficient when accelerated by GPU.

Following IMC, the objective function we used is the scalar square loss:

$$\ell(p_{ij}, \hat{p}_{ij}) = (p_{ij} - \hat{p}_{ij})^2.$$

Figure 4.1 shows the network structure of NIMC.

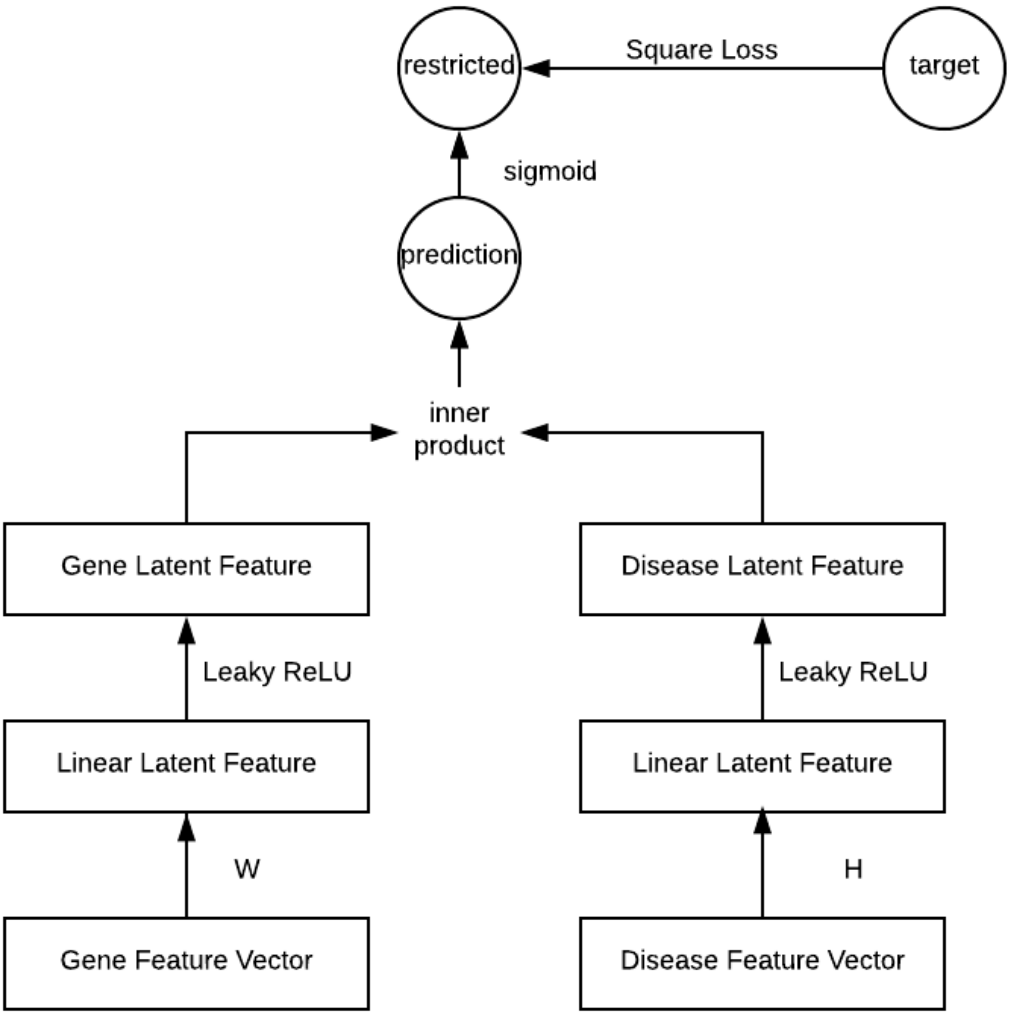

Figure 4.1: Network Structure of Neural Inductive Matrix Completion

Chapter 5

Experiments

In this chapter, we conduct comprehensive experiments that are done to prove that our proposed method has achieved a competitive performance.

5.1 Data

We have prepared two datasets for experiments. The first dataset is collected by [1] from OMIM on August 11, 2011, which is the dataset commonly used in evaluating disease-gene associations. This dataset contains 3209 diseases, 12331 genes, and 3954 disease-gene associations. Most diseases have only one known association. However, the OMIM database is continuously growing and the number of diseases, as well as the number of known associations, has increased a lot. We collected a new dataset from OMIM on November 26, 2017. We kept the same set of 12331 genes and collected 4990 diseases and 5359 disease-gene associations. We call these two datasets namely the old dataset and the new dataset.

5.1.1 Features

Original features from biological databases used in our model are high dimensional. The methods we used to reduce the dimensionality include principal component analysis (PCA) [22] and singular value decomposition (SVD). Some of the features are homogeneous networks or similarities, which instead we use eigendecomposition to extract eigenvectors. As a result, we extract a vector of length 100 from each of the

data sources and concatenate them together to form a longer vector.

We follow IMC[2] to obtain three sets of gene features and two sets of disease features. In addition, we add a set of disease features based on disease ontology. These include:

- Gene Features

    1. Microarray measurements data of gene-expression levels in different tissue samples. It is obtained from BioGPS [23] and Connectivity Map [24]. Principal component analysis is used to reduce the dimensionality to 100.
    2. A gene network representing functional interactions between genes called HumanNet [25] is used. We obtain the top 100 eigenvectors as features.
    3. Gene-disease associations in other species than human are used to extract features. Due to gene orthology, same gene may behave similarly in different species. We follow [1] to collect disease-gene associations from 8 different species: plant (*Arabidopsis thaliana*) from TAIR [26], worm (*Caenorhabditis elegans*) from WormBase [13] [27], fruit fly (*Drosophila melanogaster*) from FlyBase [14], mouse (*Mus musculus*) from MGD [15], yeast (*Saccharomyces cerevisiae* from [28],[29],[30] and [31], *Escherichia coli* from [32], zebrafish *Danio rerio* from ZFIN [33] and chicken *Gallus gallus* from GEISHA [34]. The top 100 singular vectors are used as feature vectors.

- Disease Features

    1. Disease similarity network MimMiner [18]. MimMiner computes disease similarities based on text mining. The top 100 eigenvectors are selected as feature vectors.
    2. We follow [2] to use clinical text from OMIM webpages. We apply the TF-IDF scheme on the Clinical Features and Clinical Management sections

after removing most rare words and most common words. We take top 100 principal components as the feature vectors.

3. Disease Ontology Similarities. We obtain a similarity matrix of OMIM diseases using Resnik pairwise similarity [35] with the best-match average (BMA)[36] strategy. Then the top 100 eigenvectors are used as feature vectors.

5.2 Evaluation Methods

We evaluate the performance using our proposed model and enriched features compared to inductive matrix completion [2], Katz on the heterogeneous network [1] and CATAPULT [1].

5.2.1 Evaluation Metrics

On either the old and the new dataset, we split all known disease-gene associations randomly into three groups of equal size. We hide all associations from one group and train a model based on remained associations. The experiments are repeated three times for each group. After a model is trained in each experiment, for each disease, we order all the negative pairs (either hidden or unknown) by the descending order of prediction scores. Given a positive integer k, we check for each hidden disease-gene pair whether it is in the top-k pairs for that disease. We call the portion of hidden pairs in the top-k pairs the top-k recall, which is non-decreasing by k. We record the top-k recall for k between 1 and 100 and take an average of all three experiments.

At the meantime, by setting each k, we can compute a pair of precision and recall. The recall is calculated the same way as above, and the precision is the number of recovered hidden pairs among all predicted pairs (top-k pairs of all diseases).

5.2.2 Parameter Settings

We choose the best parameters based on the 3-fold cross-validation on the old dataset by grid-search. The parameters we choose for our experiments are:

1. The length of latent feature $r = 1000$.
2. The regularizer coefficient $\lambda = 0.01$.
3. The parameter of leaky ReLU function $\beta = 0.1$.
4. The penalty coefficient of false negative $c_{negative} = 0.0001$.

5.3 Results

5.3.1 3-Fold Cross-validation on the Old Dataset

On the old dataset, we compare IMC, Katz on the heterogeneous network, CATAPULT and our proposed model NIMC in a 3-fold cross-validation setting. For NIMC, only the original two sets of features from IMC are used, because the disease ontology features are published after the old dataset was created, which may lead to bias on associations discovered after OMIM associations are collected.

All known disease-gene associations are randomly split into three groups of equal size.

We plot the results in Figure 5.1 and Figure 5.2. we observe that NIMC outperforms all competitors when $k > 5$. Katz slightly outperforms NIMC when k is small and recall is small.

5.3.2 3-Fold Cross-validation on the New Dataset

On the new dataset, we compare IMC, Katz on the heterogeneous network, CATAPULT and NIMC in a 3-fold cross-validation setting. For NIMC, we test it in both the case with the disease ontology features and the case without disease ontology

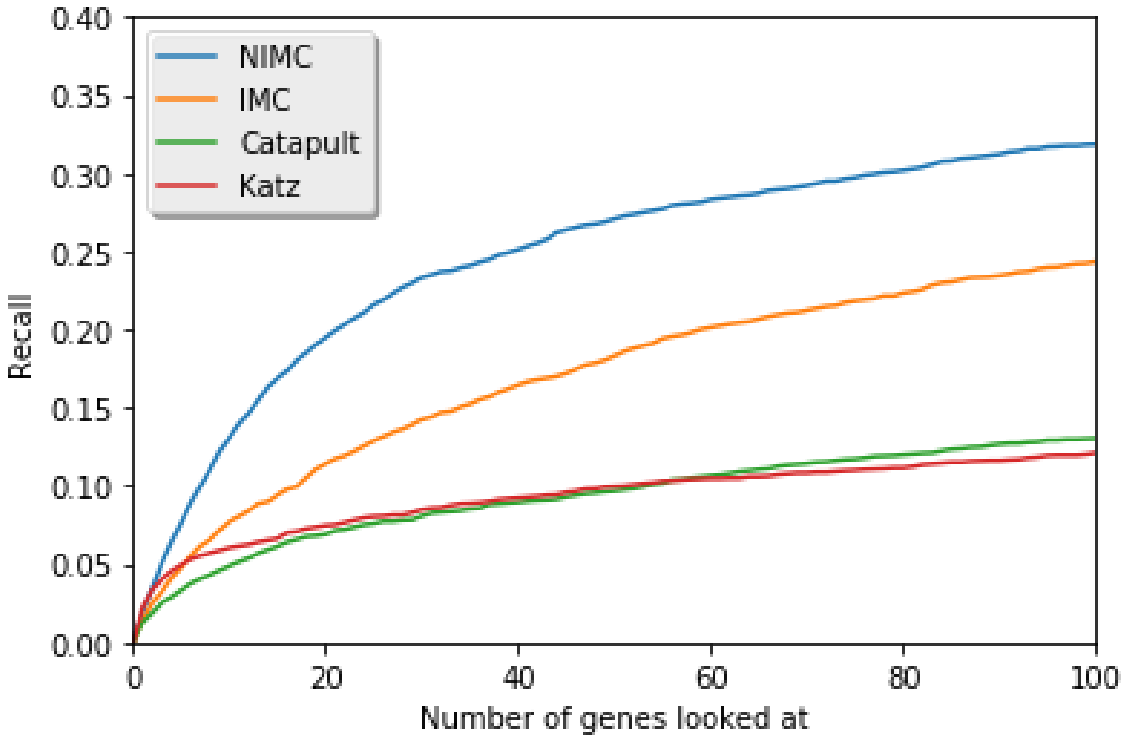

Figure 5.1: Top-k recall curve in 3-fold cross-validation on the old dataset.

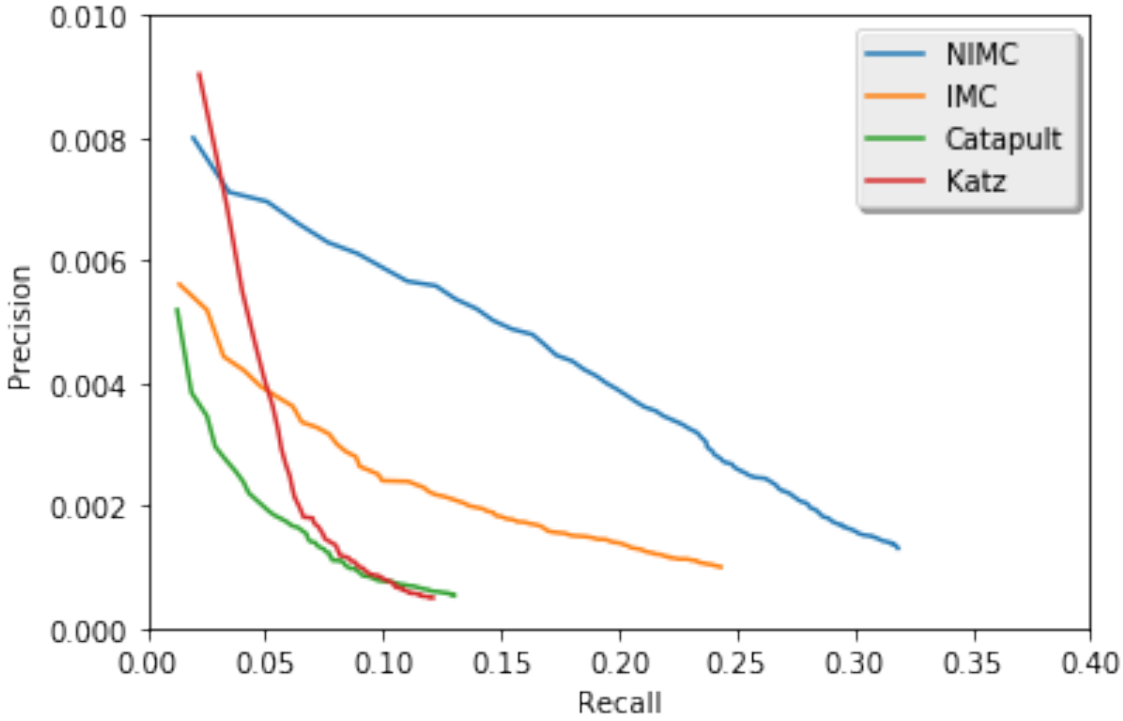

Figure 5.2: Precision-recall curve in 3-fold cross-validation on the old dataset.

features. The top-k recall curve and precision-recall curve are plotted in Figure 5.3 and Figure 5.4.

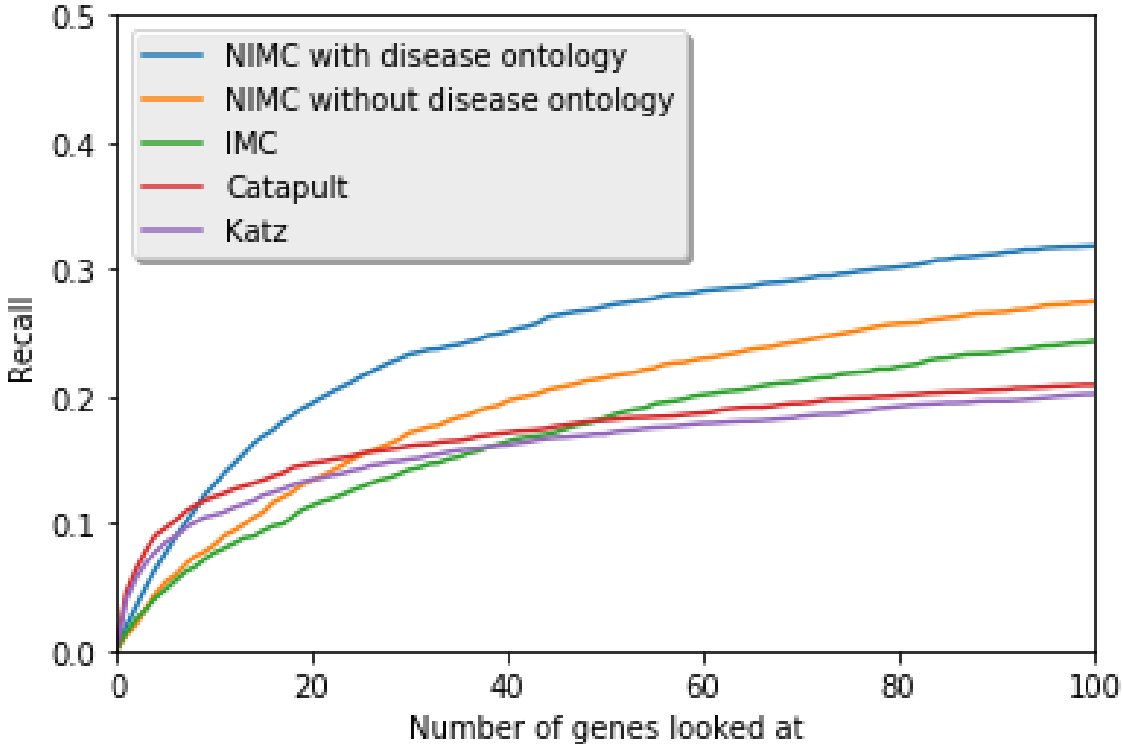

Figure 5.3: Top-k recall curve in 3-fold cross-validation on the new dataset.

5.3.3 3-Fold Cross-validation on Singleton Diseases

We also did an experiment by splitting all singleton diseases (diseases that have only one known association) into three equal folds. In the evaluation on each fold, we hide the associations of singleton diseases within this fold, and use other two folds and all other non-singleton diseases for training. This is an *ab initio* setting where we can test the ability of solving the cold-start problem. The recall curve is plot in Figure 5.5.

We observe our method still gets a higher recall than others within the top-10 to -100 range.

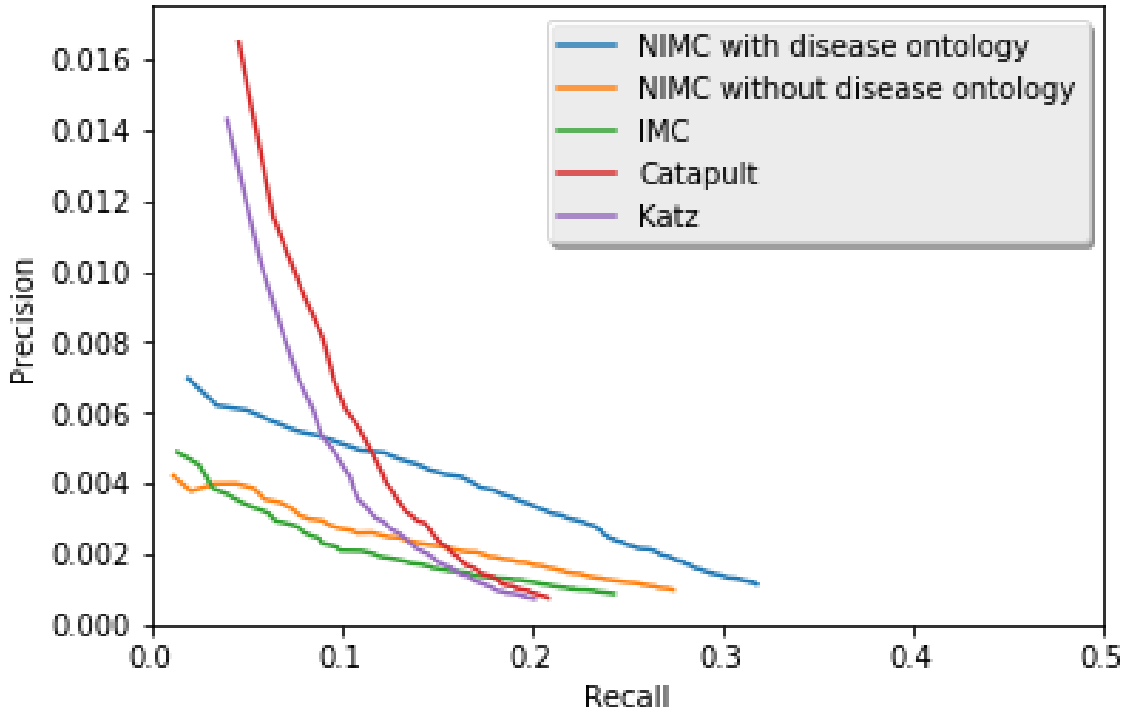

Figure 5.4: Precision-recall curve in 3-fold cross-validation on the new dataset.

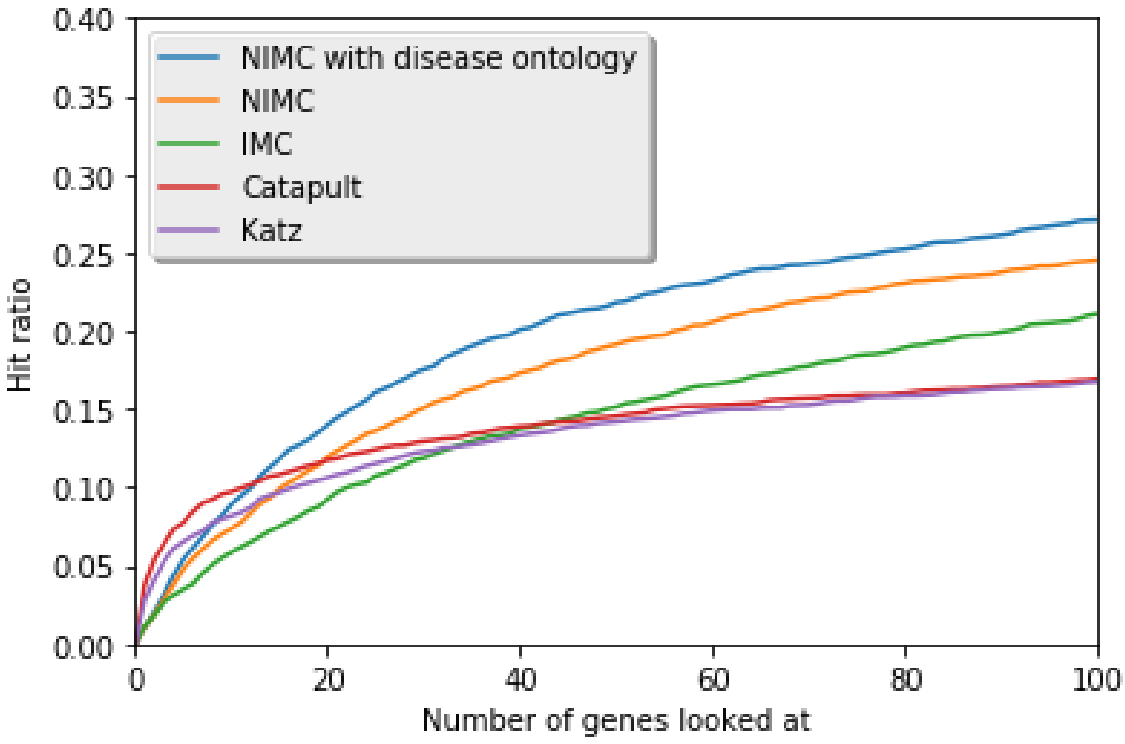

Figure 5.5: Precision-recall curve in 3-fold cross-validation on singleton diseases of the new dataset.

5.3.4 Prediction of Newly Discovered Associations

The goal of disease-gene prediction is to predict unknown associations based on information we currently have. To simulate such a scenario, we train a model on 3954 associations in the old dataset and evaluate the prediction of newly reported associations in the new dataset. The top-k recall curve and the precision-recall curve is plotted in Figure 5.6 and Figure 5.7.

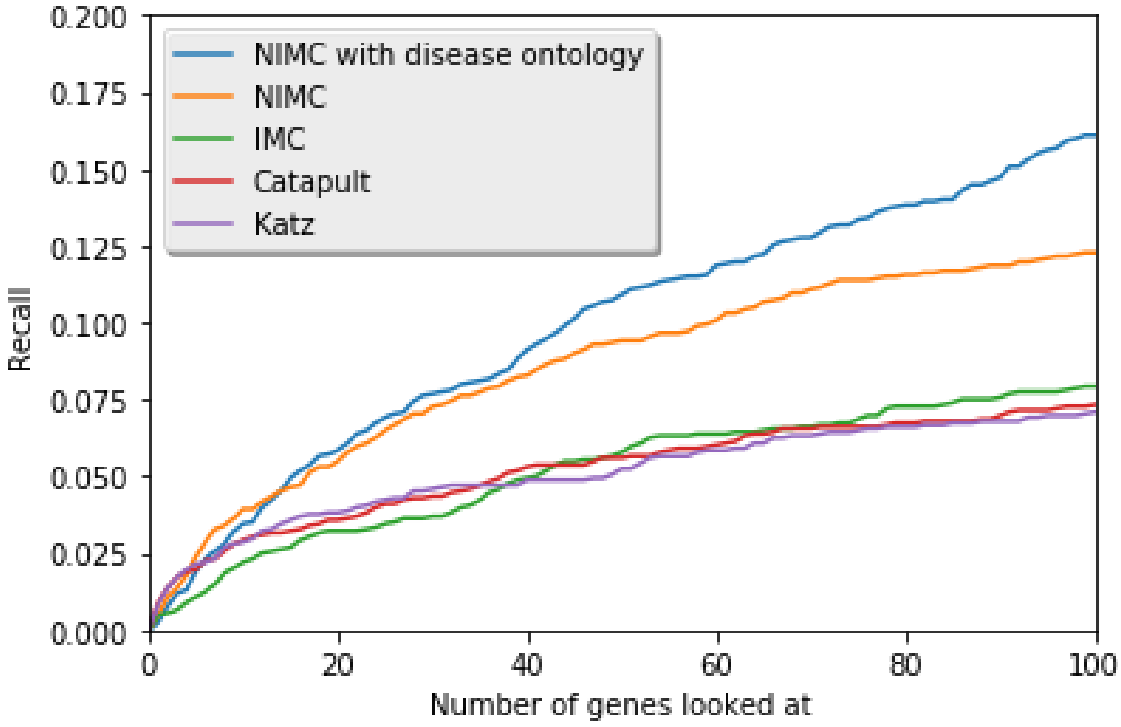

Figure 5.6: Top-k recall curve on prediction of newly discovered associations.

We observe that NIMC with disease ontology outperforms all competitors when $k > 10$. It loses a little to Katz and Catapult when k is small and recall is small. Compared to NIMC without disease ontology, it is obvious that disease ontology helps increase the efficiency of the method.

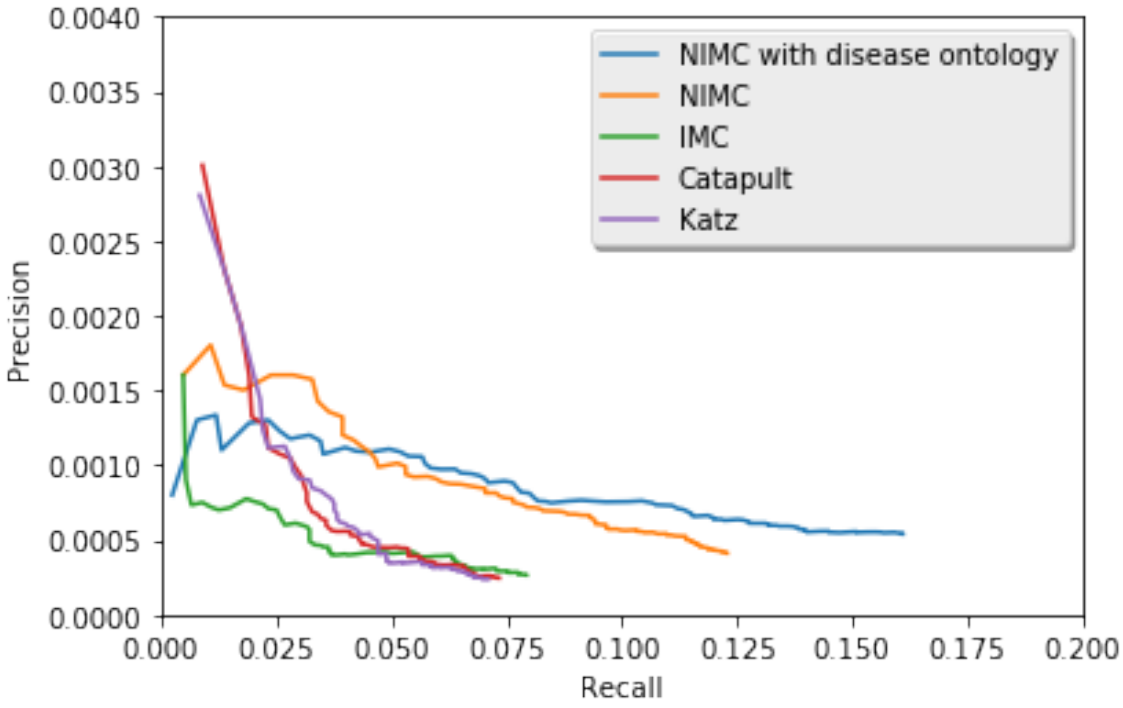

Figure 5.7: Precision-recall curve on prediction of newly discovered associations.

Chapter 6

Parameter Sensitivity Analysis

In this chapter, we analyze the sensitivity of the hyperparameters in NIMC, including

1. The length of latent feature r.

2. The regularizer coefficient *lambda*.

3. The parameter of *leaky ReLU* function.

We plot all results in the form of top-k recall curve.

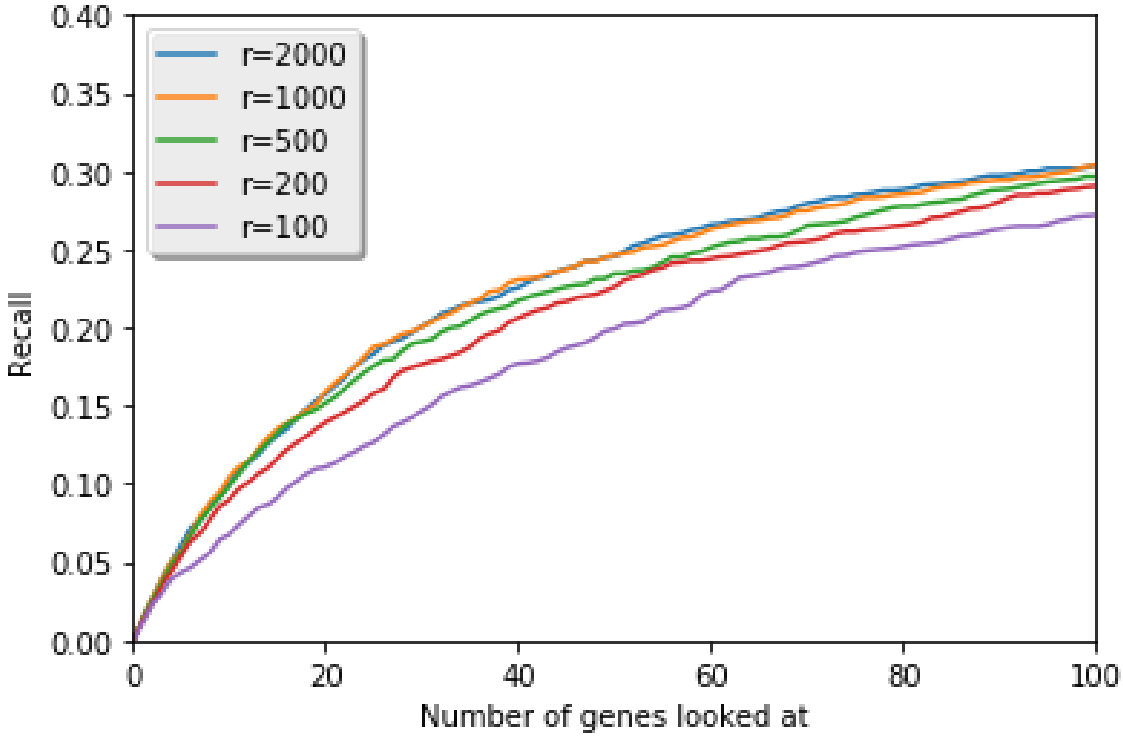

Figure 6.1: Sensitivity analysis of the length of latent feature.

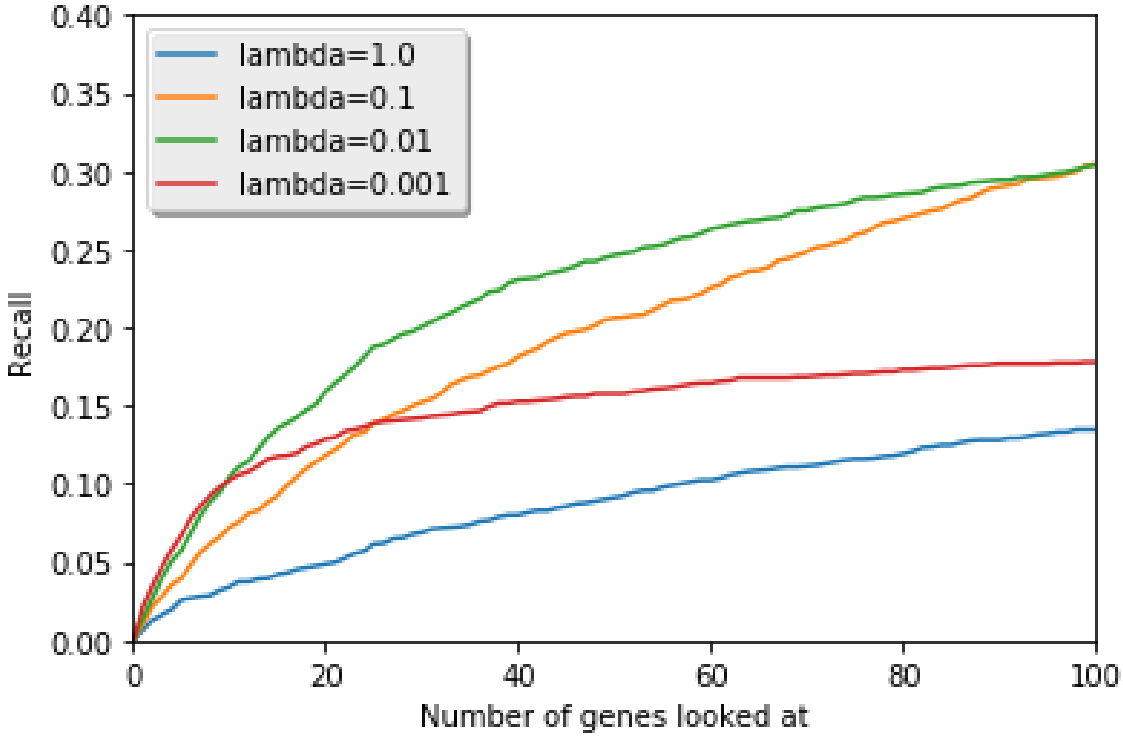

Figure 6.2: Sensitivity analysis of the regularizer coefficient.

Figure 6.1 shows that on different settings of the length of latent feature r, the performance increases when increasing r to a larger number. This can be explained as the model is more expressive when r is large. However, performance does not change much when r changes in a very large range. The curves when $r = 1000$ and $r = 2000$ almost coincide with each other.

Figure 6.2 shows that results are best when $lambda = 0.01$, when $\lambda = 0.1$, the results are worse than the best case in general, but the recall when $k = 100$ is same as the best case.

Figure 6.3 shows that different settings of the leaky ReLU parameter do not influence the results too much. When $\alpha = 0$, leaky ReLU becomes ReLU, the curve shows using ReLU instead of leaky ReLU can increase the recall when k is around 100, but lower the performance when k is around 25. The curves in other cases almost coincide.

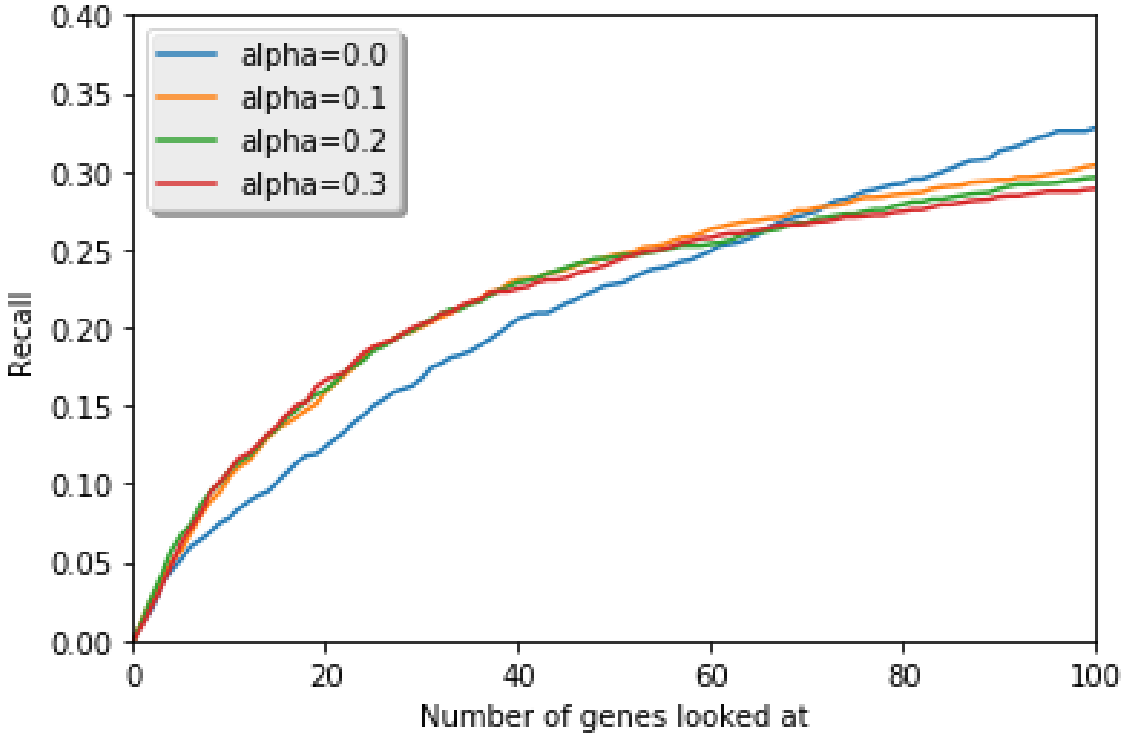

Figure 6.3: Sensitivity analysis of the leaky ReLU parameter.

All figures show that NIMC is not sensitive to hyperparameters. The most sensitive parameter is the regularization coefficient λ, but it is still easy to find a good solution by grid search.

Chapter 7

Concluding Remarks

In this thesis, we proposed a new model for predicting disease-gene association called NIMC. NIMC outperforms the state-of-the-art methods in finding unknown associations in top-100 ranked genes for each disease. Adding disease ontology as a source of information contributes significantly to the performance of NIMC. The sensitivity analysis shows that NIMC is not sensitive to hyperparameters.

7.1 Implementation

Neural inductive matrix completion is implemented by us in Python using TensorFlow framework [37]. We use the Adam Optimizer [38] implemented by TensorFlow using 0.1 as the learning rate. Training NIMC takes a few minutes on a machine with 2 NVidia Titan X Pascal GPU cards.

7.2 Future Research Work

Future extensions of this work can be done in various directions. From the point of view of data, the recent abundance of biological databases has provided us more information that can be integrated for *in silico* experiments. Besides disease ontology, it can be an extension to utilize gene ontology as gene features. It is also possible to consider the structures of proteins which are transcription and translation product of human genes.

From the point of view of machine learning, the model can be improved with

special consideration of this special task. Unlike the case of recommender systems where we see users as individuals that are not well connected, genes and diseases are highly correlated. Human genes, proteins and other chemicals are all working together within a complicated human body, while biological pathways are a good tool to describe the internal relation of them. A machine learning model can be applied to this task if it has the potential to learn from the biological pathways, which is a heterogeneous directed graph. In addition, other work can be done to improve the model, including adding more layers to the model, using different strategies to hybrid the features, etc.

REFERENCES

[1] U. M. Singh-Blom, N. Natarajan, A. Tewari, J. O. Woods, I. S. Dhillon, and E. M. Marcotte, "Prediction and validation of gene-disease associations using methods inspired by social network analyses," *PloS one*, vol. 8, no. 5, p. e58977, 2013.

[2] N. Natarajan and I. S. Dhillon, "Inductive matrix completion for predicting genedisease associations," *Bioinformatics*, vol. 30, no. 12, pp. i60–i68, 2014. [Online]. Available: +http://dx.doi.org/10.1093/bioinformatics/btu269

[3] McKusick-Nathans Institute of Genetic Medicine, "Omim - online mendelian inheritance in man," https://omim.org/, 3 2018, (Accessed on 03/27/2018).

[4] J. Bennett, S. Lanning *et al.*, "The netflix prize," in *Proceedings of KDD cup and workshop*, vol. 2007. New York, NY, USA, 2007, p. 35.

[5] V. Klema and A. Laub, "The singular value decomposition: Its computation and some applications," *IEEE Transactions on automatic control*, vol. 25, no. 2, pp. 164–176, 1980.

[6] O. Vanunu, O. Magger, E. Ruppin, T. Shlomi, and R. Sharan, "Associating genes and protein complexes with disease via network propagation," *PLoS computational biology*, vol. 6, no. 1, p. e1000641, 2010.

[7] Y. Li and J. C. Patra, "Genome-wide inferring gene–phenotype relationship by walking on the heterogeneous network," *Bioinformatics*, vol. 26, no. 9, pp. 1219–1224, 2010.

[8] L. Katz, "A new status index derived from sociometric analysis," *Psychometrika*, vol. 18, no. 1, pp. 39–43, 1953.

[9] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 173–182.

[10] H.-J. Xue, X.-Y. Dai, J. Zhang, S. Huang, and J. Chen, "Deep matrix factorization models for recommender systems," *static. ijcai. org*, 2017.

[11] Bethesda(MD), "Genes and diseases," National Center for Biotechnology Information (US). Genes and Disease [Internet].

[12] National Institutes of Health *et al.*, "An overview of the human genome project," 2005.

[13] N. Chen, T. W. Harris, I. Antoshechkin, C. Bastiani, T. Bieri, D. Blasiar, K. Bradnam, P. Canaran, J. Chan, C.-K. Chen *et al.*, "Wormbase: a comprehensive data resource for caenorhabditis biology and genomics," *Nucleic acids research*, vol. 33, no. suppl_1, pp. D383–D389, 2005.

[14] S. Tweedie, M. Ashburner, K. Falls, P. Leyland, P. McQuilton, S. Marygold, G. Millburn, D. Osumi-Sutherland, A. Schroeder, R. Seal *et al.*, "Flybase: enhancing drosophila gene ontology annotations," *Nucleic acids research*, vol. 37, no. suppl_1, pp. D555–D559, 2008.

[15] J. T. Eppig, J. A. Blake, C. J. Bult, J. A. Kadin, J. E. Richardson, and M. G. D. Group, "The mouse genome database (mgd): new features facilitating a model system," *Nucleic acids research*, vol. 35, no. suppl_1, pp. D630–D637, 2006.

[16] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, B. Smith-White, D. Ako-Adjei *et al.*, "Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation," *Nucleic acids research*, vol. 44, no. D1, pp. D733–D745, 2015.

[17] D. R. Zerbino, P. Achuthan, W. Akanni, M. Amode, D. Barrell, J. Bhai, K. Billis, C. Cummins, A. Gall, C. G. Girn, L. Gil, L. Gordon, L. Haggerty, E. Haskell, T. Hourlier, O. G. Izuogu, S. H. Janacek, T. Juettemann, J. K. To, M. R. Laird, I. Lavidas, Z. Liu, J. E. Loveland, T. Maurel, W. McLaren, B. Moore, J. Mudge, D. N. Murphy, V. Newman, M. Nuhn, D. Ogeh, C. K. Ong, A. Parker, M. Patricio, H. S. Riat, H. Schuilenburg, D. Sheppard, H. Sparrow, K. Taylor, A. Thormann, A. Vullo, B. Walts, A. Zadissa, A. Frankish, S. E. Hunt, M. Kostadima, N. Langridge, F. J. Martin, M. Muffato, E. Perry, M. Ruffier, D. M. Staines, S. J. Trevanion, B. L. Aken, F. Cunningham, A. Yates, and P. Flicek, "Ensembl 2018," *Nucleic Acids Research*, vol. 46, no. D1, pp. D754–D761, 2018. [Online]. Available: http://dx.doi.org/10.1093/nar/gkx1098

[18] 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, p. 535, 2006.

[19] S. Köhler, N. A. Vasilevsky, M. Engelstad, E. Foster, J. McMurry, S. Aymé, G. Baynam, S. M. Bello, C. F. Boerkoel, K. M. Boycott *et al.*, "The human phenotype ontology in 2017," *Nucleic acids research*, vol. 45, no. D1, pp. D865–D876, 2016.

[20] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *European conference on machine learning*. Springer, 2004, pp. 39–50.

[21] F. Mordelet and J.-P. Vert, "A bagging svm to learn from positive and unlabeled examples," *arXiv preprint arXiv:1010.0772*, 2010.

[22] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[23] "Biogps - your gene portal system," http://biogps.org/, (Accessed on 03/27/2018).

[24] "Connectivity map," https://portals.broadinstitute.org/cmap/, (Accessed on 03/27/2018).

[25] I. Lee, U. M. Blom, P. I. Wang, J. E. Shim, and E. M. Marcotte, "Prioritizing candidate disease genes by network-based boosting of genome-wide association data," *Genome research*, vol. 21, no. 7, pp. 1109–1121, 2011.

[26] D. Swarbreck, C. Wilks, P. Lamesch, T. Z. Berardini, M. Garcia-Hernandez, H. Foerster, D. Li, T. Meyer, R. Muller, L. Ploetz *et al.*, "The arabidopsis information resource (tair): gene structure and function annotation," *Nucleic acids research*, vol. 36, no. suppl_1, pp. D1009–D1014, 2007.

[27] R. A. Green, H.-L. Kao, A. Audhya, S. Arur, J. R. Mayers, H. N. Fridolfsson, M. Schulman, S. Schloissnig, S. Niessen, K. Laband *et al.*, "A high-resolution c. elegans essential gene network based on phenotypic profiling of a complex tissue," *Cell*, vol. 145, no. 3, pp. 470–482, 2011.

[28] S. S. Dwight, M. A. Harris, K. Dolinski, C. A. Ball, G. Binkley, K. R. Christie, D. G. Fisk, L. Issel-Tarver, M. Schroeder, G. Sherlock *et al.*, "Saccharomyces genome database (sgd) provides secondary gene annotation using the gene ontology (go)," *Nucleic acids research*, vol. 30, no. 1, pp. 69–72, 2002.

[29] T. L. Saito, M. Ohtani, H. Sawai, F. Sano, A. Saka, D. Watanabe, M. Yukawa, Y. Ohya, and S. Morishita, "Scmd: Saccharomyces cerevisiae morphological database," *Nucleic acids research*, vol. 32, no. suppl_1, pp. D319–D322, 2004.

[30] K. L. McGary, I. Lee, and E. M. Marcotte, "Broad network-based predictability of saccharomyces cerevisiae gene loss-of-function phenotypes," *Genome biology*, vol. 8, no. 12, p. R258, 2007.

[31] M. E. Hillenmeyer, E. Fung, J. Wildenhain, S. E. Pierce, S. Hoon, W. Lee, M. Proctor, R. P. S. Onge, M. Tyers, D. Koller *et al.*, "The chemical genomic portrait of yeast: uncovering a phenotype for all genes," *Science*, vol. 320, no. 5874, pp. 362–365, 2008.

[32] R. J. Nichols, S. Sen, Y. J. Choo, P. Beltrao, M. Zietek, R. Chaba, S. Lee, K. M. Kazmierczak, K. J. Lee, A. Wong *et al.*, "Phenotypic landscape of a bacterial cell," *Cell*, vol. 144, no. 1, pp. 143–156, 2011.

[33] J. Sprague, L. Bayraktaroglu, D. Clements, T. Conlin, D. Fashena, K. Frazer, M. Haendel, D. G. Howe, P. Mani, S. Ramachandran *et al.*, "The zebrafish information network: the zebrafish model organism database," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D581–D585, 2006.

[34] G. W. Bell, T. A. Yatskievych, and P. B. Antin, "Geisha, a whole-mount in situ hybridization gene expression screen in chicken embryos," *Developmental dynamics*, vol. 229, no. 3, pp. 677–687, 2004.

[35] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," *arXiv preprint cmp-lg/9511007*, 1995.

[36] J. Z. Wang, Z. Du, R. Payattakool, P. S. Yu, and C.-F. Chen, "A new method to measure the semantic similarity of go terms," *Bioinformatics*, vol. 23, no. 10, pp. 1274–1281, 2007.

[37] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.