UNIVERSITY OF CALIFORNIA
RIVERSIDE

Deep Learning Methods for High-Resolution Functional Annotation and Discovery of Novel Connections Between Gene Sets

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Hao Chen

September 2021

Dissertation Committee:

Dr. Tao Jiang, Chairperson
Dr. Stefano Lonardi
Dr. Thomas Girke
Dr. Sika Zheng

The Dissertation of Hao Chen is approved:

Committee Chairperson

University of California, Riverside

Acknowledgments

I would like to acknowledge a number of people who supported me along the journey of my PhD pursuit.

First and foremost, I would like to express my sincere gratitude to my advisor, Prof. Tao Jiang, who gave me advice and support on almost every aspect of my study and life during the five years of my PhD. He always inspired me to consider problems in more detail, encouraged me to think critically and independently, and allowed me to have all the research freedom that I needed. From him, I learned how to become an independent researcher and an enjoyable life-long career. I would also like to thank Prof. Jiang for providing me with the freedom to intern in industry and research labs.

I would like to thank Prof. Stefano Lonardi, Prof. Thomas Girke, Prof. Sika Zheng, and Prof. Wenxiu Ma for serving on my PhD dissertation committee or my PhD qualifying exam committee. They not only provided insightful comments and constructive suggestions on my research projects, but also gave me invaluable support and advice concerning my career development.

I am thankful to my other mentors. I had the honor to work with Prof. Jianyang Zeng who advised me when I worked at Tsinghua University as a visiting student and introduced me to the brave new world of computational biology. I am grateful to Dr. Yingyao Zhou for being a helpful supervisor during my internship at GNF and introducing me new interesting research problems. The outcome of the collaboration with him includes very important pieces of this thesis. I sincerely thank Prof. Dongbo Bu, who provided great help in my research and very patient guidance on my scientific writing. I also thank Dr.

Meng Yi, Dr. Yan Li, and Dr. Sugato Bash at Google. Working with them gave me the unique experience to be exposed to practical problems in industry.

I thank all members of the Jiang Lab and the Algorithms and Computational Biology Lab, especially Dipan Shaw, Weihua Pan, Jianyu Zhou, Ashraful Arefeen, Qihua Liang, Dipankar Ranjan Baisya, Ei-Wen Yang, for their inspiring discussions, consistent help, and sincere sharing of their knowledge and experience.

Lastly, I owe my deepest gratitude to my family. I thank my parents, who have been giving me unconditional love and encouraging me to pursue what I want all the time. I would like to thank my wife Jiayuan. She has been a wonderful companion who understands me the most, a reliable peer who gives me insightful ideas, and a good friend who shares my outlook on life. This thesis is dedicated to them.

ABSTRACT OF THE DISSERTATION

Deep Learning Methods for High-Resolution Functional Annotation and Discovery of Novel Connections Between Gene Sets

by

Hao Chen

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, September 2021
Dr. Tao Jiang, Chairperson

Proteins are essential to life. Precise understanding of protein functions is critical in addressing many biomedical questions. Different protein isoforms can be produced from a single gene through alternative splicing, which greatly expands the diversity of proteins and the complexity of cellular functions. However, precise annotations that differentiate functions of isoforms are few. On the other hand, effective modeling of functional knowledge can empower computational methods in many biological applications. A fundamental step in such applications is the discovery of gene sets. Methods that can accurately map genotypes to phenotypes are needed for detecting novel connections between gene sets derived from different experiments, which could enable new biological discoveries.

Along with the accumulation of large-scale biological data, deep learning applications to biological data analysis are flourishing. In this dissertation, we propose three deep learning methods for the two related problems in functional genomics, *i.e.*, producing high-resolution functional annotation at the isoform level, and discovering connections between experimentally derived gene sets via functional knowledge. First, we design DIFFUSE,

which for the first time integrates isoform sequences and expression profiles to systematically predict isoform functions, by combining the power of deep learning and probabilistic graphical models. Second, to enhance the prediction of isoform functions, we propose FINER, which jointly predicts isoform functions and isoform-isoform interactions through the introduction of a unified learning objective, enabling the two tasks to benefit from each other. Finally, we develop FEGS, a representation learning approach based on hypergraph embedding, which embeds gene sets as compact features encoding functional information of gene members and facilitates gene set comparison by more sensitive detection of common phenotypes. FINER and DIFFUSE significantly outperform the existing isoform function prediction methods, and their predictions are validated by independent biological data. FEGS has been successfully applied to drug discovery and cell type identification.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proteins, the most versatile gene products, serve crucial functions in cellular systems. They provide structure to the cell, function as catalysts, transport molecules such as oxygen, provide immune protection, control cell differentiation, growth, and death, and work to synthesize new proteins. Many complex diseases are caused by the disruption of protein functions [127, 66, 32], while identifying new therapeutic targets always requires the understanding of how the target proteins function in the biological networks underlying the diseases [81, 67]. Therefore, precise functional annotation of proteins is vitally important in unraveling the molecular basis of such diseases and drug discovery.

The complexity of proteomes is much higher than that of genomes. In human, the number of human protein-coding genes is estimated to be about 20 000. However, it is estimated that more than 10^6 different proteins may exist in the human body [112]. Alternative splicing is commonly believed to play a central role in increasing the proteome diversity [76, 107]. In alternative splicing, exons of a gene may be joined together in

various ways to produce different mRNA variants which will be translated to different proteins, known as protein isoforms. In the human genome, more than 95% of the multi-exon genes undergo alternative splicing [116]. Isoforms carry specific, sometimes distinct or even opposite, biological functions [95]. Although functions and interactions at the gene level have been extensively studied and well recorded in databases [7, 73], there are very few annotations that can differentiate functions of isoforms encoded by the same gene, that is, isoforms are always treated as having the same functions in databases. Owing to the importance of precise functional annotation of proteins and the large number of isoforms, efficient computational methods that can provide high-resolution function predictions at the isoform level are in great demand.

On the other hand, it has been shown that effectively modeling of functional knowledge can empower the computational methods in many biological and medical applications [81, 103]. One of the most basic outcomes when interpreting biological data in such applications is the discovery of gene sets. For instance, the gene expression analyses identify sets of genes that are differentially expressed in different conditions. Genetic screening experiments produce sets of genes associated with a disease. A high similarity between gene sets derived from different experiments might indicate previously unrecognized connections between studied objects. Due to the hypothesis that genes in an experimentally identified gene set work coherently towards the same biological processes or functions, the designing of computational methods that can sensitively detect gene set connections needs to consider not only the numbers of shared genes between sets, but also biological functions enriched in the set of genes.

Thanks to the development of high-throughput sequencing, the genome sequences of species become largely accessible, large-scale biological data are accumulated, which depict cellular and molecular processes from different levels, such as the genome, proteome, and transcriptome. In parallel to the advances in sequencing technologies, deep learning applications to biological data analysis are flourishing due to their strength in integrating heterogeneous data, modeling complex relationships underlying the data, and generalization ability on unseen examples.

In this dissertation, we study deep learning methods for two related problems in functional genomics. Specifically, we provide new methods to produce high-resolution functional annotations at the isoform level and to effectively model functional knowledge to help discover gene set connections to enable new biological discoveries. Three deep learning methods are proposed. The details of computational challenges facing in the two problems and the proposed methods are reviewed in Sections 1.1 and 1.2 respectively.

1.1 Predicting functions and interactions of isoforms

In eukaryotic cells, through the mechanism of alternative splicing, a single gene often produces multiple protein isoforms with different sequences and thus different structures. Isoforms encoded by the same gene can carry distinct or even opposing biological functions, as illustrated in Figure 1.1. Well-studied examples include apoptosis, in which alternative splicing can act as an on/off switch for genes encoding pro-apoptotic or anti-apoptotic isoforms. For instance, two of the isoforms of BCL2L1 gene, BCL-xL, and BCL-xS, exhibit completely opposite functions: BCL-xL inhibits apoptosis while BCL-xS promotes it [149].

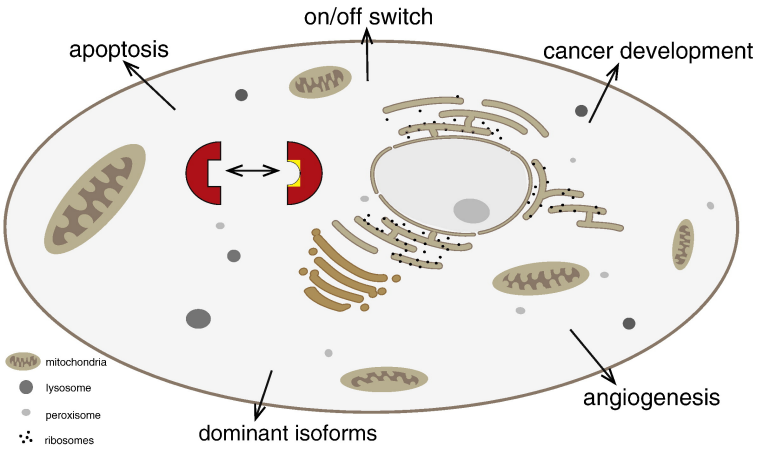


Figure 1.1: A eukaryotic cell is depicted. Alternative splicing produces protein isoforms (red half squares) by introducing protein sequences (yellow) that are encoded by alternative exons. The expression of different isoforms may change cellular processes, *e.g.*, apoptosis. Source: [135].

Although gene functions have been well studied, the specific functions of the vast majority of isoforms are still poorly understood to date. Therefore, efficient computational methods that can provide high-throughput and accurate predictions of isoform functions are in great demand, which can be a complement to traditional biological pipelines that fall short in power and efficiency.

Given the availability of annotated gene functions, supervised learning models have been successfully applied for gene function prediction [110, 83]. In contrast, the lack of isoform-level functional ground truth as the training labels makes the existing methods in training gene function prediction models not applicable to the isoform function prediction problem. Earlier isoform function prediction methods [99, 40, 102, 129] try to infer isoform functions from their expression profiles alone. The experimental results suggest that the prediction accuracy of these methods is less than desirable. Other data, such as isoform sequences and conserved domains may carry complementary functional information

to expression profiles and should be included in the predictive models.

To obtain more accurate isoform function predictions, we present a deep learning approach, named DIFFUSE [27] (Deep learning based prediction of IsoForm FUnctions from Sequences and Expression). As illustrated in Chapter 2, DIFFUSE for the first time integrates isoform sequences and expression profiles to systematically predict isoform functions. DIFFUSE combines the power of deep learning and probabilistic graphical models to model sequence information of individual isoforms and co-expression relationships of isoform pairs respectively. We designed a semi-supervised learning algorithm to overcome the lack of isoform-level functional ground truth. Experimental results show that DIFFUSE greatly outperforms earlier methods. Predicted isoform functions of DIFFUSE are further validated by their consistency with protein isoform structures (represented by contact maps) and consistency with some functional sequence features, *e.g.*, the presence or absence of certain types of binding sites in isoform sequences.

Another critical information in biological processes, protein-protein interactions (PPIs), is also often used to study gene functions. However, similar to the situation of functional annotation at the isoform level, "proteins" in the existing PPIs generally refer to "genes", which do not provide more detailed information concerning the interaction of isoforms. Therefore, directly using PPIs to predict isoform functions cannot help differentiate the functions of isoforms.

On the other hand, the isoforms of a gene can have vastly different interaction profiles. Such difference can be as great as that between proteins encoded by different genes. The interacting partners of isoforms often exhibit distinct functional characteristics

[161]. In fact, several computational methods have been developed to refine protein-protein interactions into isoform-isoform interactions (IIIs) [94, 147, 48, 72, 165]. However, the prediction of isoform functions and prediction of isoform-isoform interactions, though inherently intertwined, have so far been treated as independent computational problems in the literature. How to solve the two problems jointly and exploit the reciprocal relationship between them remains an interesting challenge.

Hence, we propose FINER (enhancing the Functional prediction of Isoforms via NEtwork Refinement on their interactions) [26], the first joint-learning method that simultaneously predicts isoform functions and refines protein-protein interactions from the gene level to the isoform level. As demonstrated in Chapter 3, jointly modeling the two tasks enables them to benefit from each other in terms of performance. We applied our method to predict tissue-specific functions and interactions of isoforms in human and found our predictions provide insights supported by biological evidence. We found our predictions are consistent with the tissue specificities of isoforms and also demonstrate the pattern of isoform interactions concerning their subcellular localization.

1.2 Discovering gene set connections via functional information of genes

Omics-based analyses are now standard practice to deconstruct the cellular and molecular processes. One of the most common outcomes when interpreting large-scale omics datasets is the discovery of gene sets. For instance, genome-wide association studies (GWAS) produce sets of genes associated with a disease. Proteomics studies produce sets

of proteins produced in different conditions. In all of these cases, the basic hypothesis is that the identified set of genes work coherently towards the same biological processes or functions. Comparison of such gene sets derived from different experiments can lead to new discoveries. For example, the L1000 dataset [136] creates a comprehensive catalog of gene expression signatures for gene knockdown/over-expression experiments and drug treatment experiments. A high similarity between gene sets derived from different signatures might indicates previously unrecognized connections, for example, the connection between a drug and its protein targets.

Routine approaches characterize similarities between gene sets based on statistics measuring the significance of the number of shared genes between two sets. Commonly used ones are the Fisher's exact test [15] and the weighted Kolmogorov-Smirnov-like statistic introduced in GSEA [137]. However, in experimentally derived gene sets, a biological process or function is usually represented by a sparse subset of its associated genes. Therefore, two sets of genes from phenotypically similar experiments may show distressingly little overlap [47], which makes the statistical methods based on gene identity counting less effective.

Based on the hypothesis that the set of genes identified from an experiment consistently function in the same biological processes or pathways, we propose a representation learning approach, FEGS (Functional Embeddings of Gene Sets), which embeds gene sets as compact features encoding information of functions enriched in the sets. The details of the method are presented in Chapter 4. The embeddings facilitate gene set comparison by more sensitively detecting shared pathways between gene sets. We successfully applied our method to high-impact applications. By representing single cells as sets of detected

genes, our method better captures cellular phenotype similarities and thus improves cell type identification. Our method also improves drug-like compound target prediction using the gene sets derived from perturbation transcriptomic signatures.

1.3 Organization of the rest of the dissertation

The rest of the dissertation consists of three chapters. In each chapter, we will introduce a computational problem with its background, and present the proposed method that attempts to address the problem. The experimental results with the conclusions will then be discussed. Specifically, DIFFUSE, the method for systematically predicting isoform functions, will be introduced in Chapter 2. FINER, the method for jointly modeling the isoform function prediction and the isoform-isoform interaction prediction, will be presented in Chapter 3. FEGS, the representation learning method for gene set embeddings, will be discussed with its applications in Chapter 4.

The publications encompassed in this dissertation are listed below:

- Hao Chen, Dipan Shaw, Jianyang Zeng, Dongbo Bu, and Tao Jiang. "DIFFUSE: predicting isoform functions from sequences and expression profiles via deep learning." *27th Conference on Intelligent Systems for Molecular Biology (ISMB/ECCB 2019)*, Basel, Switzerland, 2019. Also appears in *Bioinformatics*, 35(14): i284-i294. 2019.

- Hao Chen, Dipan Shaw, Dongbo Bu, and Tao Jiang. "FINER: enhancing the prediction of tissue-specific functions of isoforms by refining isoform interaction networks." *NAR Genomics and Bioinformatics* 3(2): lqab057. 2021.

- Hao Chen, Bin Zhou, Max W. Chang, Lars Pache, Christopher Benner, Tao Jiang, Sumit K. Chanda, and Yingyao Zhou. "Novel embeddings in functional spaces to help discover connections between gene sets." *In preparation.*

Chapter 2

DIFFUSE: Predicting Isoform Functions From Sequences and Expression Profiles via Deep Learning

2.1 Introduction

Due to alternative splicing, exons of multi-exon genes are selectively included in the transcription process, thus generating multiple isoforms from a single gene. Isoforms carry specific, sometimes distinct or even opposing, biological functions. Moreover, the expression of an isoform is often specific to tissue, developmental stage or environmental conditions, which is responsible for the diversity and adaptability of cellular activities [154, 138]. Therefore, delineating the functions of isoforms is crucial to the study of functional

complexity and diversity of genomes.

Despite their importance, the specific functions of the vast majority of isoforms are still poorly understood to date. Although many well-established databases exist [7, 73] for gene functional annotation, very few functions have been annotated at the isoform level. Owing to the large number of isoforms, systematic and global analysis of isoform functions experimentally is impractical in a short period. Therefore, efficient computational methods that can provide high-throughput and accurate predictions of isoform functions are in great demand. Given the availability of annotated gene functions, supervised learning has been successfully applied for gene function prediction [110, 83]. In contrast, the lack of isoform-level functional ground truth annotation makes isoform function prediction much more challenging.

Several methods have been proposed for isoform function prediction recently, including iMILP [99], mi-SVM [40], WLRM [102], and DeepIsoFun [129]. The basic idea of these methods is to distribute the functional annotation of a gene to all of its isoforms using techniques such as multiple instance learning (MIL) and domain adaptation (DA). However, these methods suffer from the limitation that they infer isoform functions from the information contained in expression profiles alone. The experimental results suggest that the prediction accuracy of these methods is less than desirable: the best area under the receiver operating characteristics curve (AUC) achieved by these methods is around 0.7 and the best area under the precision-recall curve (AUPRC) is around 0.3 [129].

Different types of biological data may carry complementary information of isoform functions, and hence a systematic integration of such information might lead to a

substantial improvement in prediction accuracy [97, 138]. In particular, we may divide informative biological data into the following two types. *i) Data of individual isoforms*: An isoform sequence may contain some functional sites, say active or binding sites, signal peptides and motifs. These sites, although very short, could provide strong signals about the functions of an isoform. Another source of information is (evolutionarily) conserved domains. Compared with functional sites, conserved domains are much longer, and their conservation during the evolutionary process may imply their important biological functions. Both functional sites and conserved domains could be identified from an isoform sequence, and it is well-known that the presence or absence of such sequence features can significantly influence its functions. For example, [140] studied the impact of alternative splicing on transcription factors in mouse and reported that alternative splicing can delete DNA binding domains, generating tissue-specific protein isoforms with distinct functions. *ii) Data between isoforms*: From the expression profiles of isoforms, we could easily identify the co-expression relationship between isoforms [42]. This co-expression relationship has been used to predict isoform functions in the above-mentioned methods as co-expressed isoforms tend to share similar biological functions. These two types of biological data come in different forms: the functional sites and conserved domains can be represented as strings while the co-expression relationship is usually represented as a network. How to integrate such different forms of data in isoform function prediction remains as a challenge.

In this chapter, we present a novel approach, named DIFFUSE (Deep learning based prediction of IsoForm FUnctions from Sequences and Expression), that integrates both isoform sequences and expression profiles to predict isoform functions. Our approach

goes through two stages to integrate various information into a unified predictive model. In the first stage, a deep neural network (DNN) is designed to capture features from isoform sequences and conserved domains. Taking the sequence and conserved domains of an isoform as the input, the DNN computes an initial score that measures how likely the isoform has the function under consideration. In the second stage, a conditional random field (CRF) is designed to exploit the co-expression relationship between isoforms. By combining the initial scores computed by the DNN with the co-expression relationship, the CRF assigns isoforms functional labels based on the initial scores while trying to keep highly co-expressed isoforms attaining the same labels. To overcome the lack of isoform-level training labels, we propose an iterative semi-supervised training algorithm based on the multiple instance learning (MIL) framework similar to the one in [3]. Specifically, our approach first initializes all isoforms of genes that have the function under consideration with positive labels and the other isoforms with negative labels. The initial functional labels are then used to train the model parameters. The new parameters of the model are next used to update the label of each isoform from positive genes. In each iteration, these two steps are performed alternately. Note that the isoforms of the same gene may be assigned different labels an update, which would encourage the model to capture features that can differentiate the functions of different isoforms.

To evaluate the performance of DIFFUSE, we first measure its prediction accuracy using the gene-level functional annotation in Gene Ontology (GO) as done in [99, 40, 102, 129]. DIFFUSE achieves an average AUC of 0.840 and AUPRC of 0.581 over 4,184 functional categories. We also compare DIFFUSE with the existing methods on several datasets.

Four state-of-the-art isoform function prediction methods proposed in [99, 40, 102, 129] are included in the comparison. The results demonstrate that our method significantly outperforms the others. We further analyze the divergence of the predicted functions of isoforms from the same gene. The scarcity of experimentally-verified isoform functions makes the validation of predicted functions difficult. We thus conduct a series of computational experiments to indirectly validate our predictions. More specifically, we first analyze how functional similarity is correlated with isoform sequence, expression and structural similarities. Our analysis shows that the similarity of predicted functions has higher correlation with isoform structural similarity than with sequence similarity or expression similarity, which accords previous studies [68]. The predictions are then further validated by assessing their consistency with the presence or absence of some well-studied functional sequence features followed by a targeted literature search.

2.2 Materials and methods

2.2.1 Datasets

Isoform sequences of the human genome are downloaded from the NCBI Reference Sequences database (RefSeq GRCh38.p12; [119]. To ensure sequence quality, only manually-curated RefSeq records are recruited in our computational experiments. The 'Coding DNA Sequence' (CDS) is extracted for each isoform using the RefSeq CDS annotation file. Two or more isoforms corresponding to the same CDS are treated as a single isoform. For each isoform, we search it against the NCBI Conserved Domain Database (CDD) [105] to acquire its conserved domains.

Isoform expression profile data are obtained from the literature [129]. It consists of human isoform RNA-seq data from the NCBI Reference Sequence Archive (SRA) [91] consisting of 334 studies and 1,735 experiments. Only isoforms that appear in both the sequence data and the expression data are kept. This results in a total of 39,375 isoforms from 19,303 genes consisting of 9,032 multiple isoform genes (MIGs) and 10,271 single isoform genes (SIGs).

We adopt the functional categories defined by Gene Ontology (GO), and download gene functional annotation from the UniProt Gene Ontology Annotation (UniProt-GOA) database [64]. To ensure the annotation quality, we only keep manually-curated GO terms and skip terms with the 'IEA' evidence code. Similar to [99, 129], we also ignore GO terms that are too specific or general. Finally, 4,184 GO terms associated with the numbers of genes in the range of 10 to 1,000 are considered in this study.

2.2.2 Methods overview

As mentioned before, DIFFUSE predicts isoform functions by integrating the information of isoform sequences, conserved domains and expression profiles into a unified predictive model. More specifically, we train a model for each GO term. The inference procedure of the model consists of two stages. In the first stage, taking the sequence and conserved domains of an isoform as the input, the DNN computes an initial score in the range of [0, 1] measuring how likely the isoform has the GO term. In the second stage, the CRF makes a final prediction by considering both the initial scores and the co-expression relationship among isoforms. To overcome the lack of annotated isoform functions, we de-

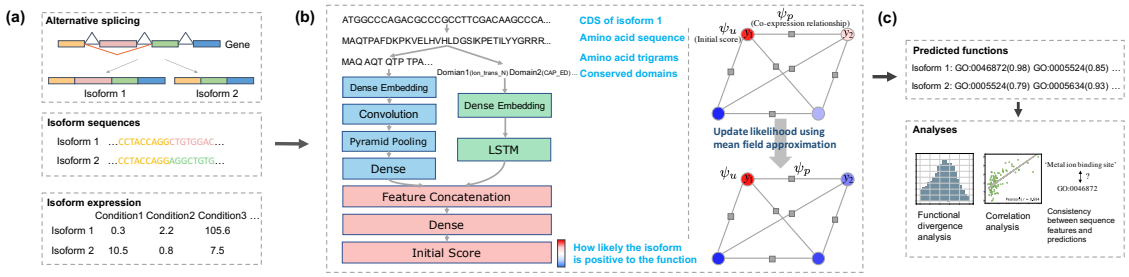


Figure 2.1: Overview of our computational pipeline. (a) Alternative splicing generates multiple isoforms from a gene with different sequences and expression profiles. (b) The DIFFUSE model contains two key components, a DNN and a CRF. The DNN consists of several layers and components including a CNN and an LSTM. Its input consists of trigrams generated from a CDS or protein sequence and conserved domains. It computes an initial score indicating how likely the output label is positive. The CRF can be represented as a complete graph G over variables y, which denote the labels of isoforms. Each unary clique or pairwise clique in G induces a unary potential or a pairwise potential denoted as ψ_u or ψ_p. The CRF makes predictions by minimizing a Gibbs energy composed of ψ_u's and ψ_p's. The initial scores are factored into ψ_u's while the co-expression relationship is factored into ψ_p's. The DNN and CRF are trained together using an iterative semi-supervised learning algorithm based on MIL, where the positive likelihood of each isoform is initialized with its initial score and then updated iteratively through the mean field approximation. (c) Several analyses are conducted in our study to support or validate our predicted isoform functions.

velop a semi-supervised algorithm following [3] to train both the DNN and CRF together iteratively. To help training the DNN, protein sequences from the SwissProt [13] database are also used as training data. In order to avoid potential information leak between the training and test data, we consider clusters of orthologous groups (COGs) and make sure that each COG is never split between the training and test data. A schematic illustration of DIFFUSE as well as the analyses to be performed in our study is given in Figure 2.1, and more details of the method are provided below.

2.2.3 Exploring sequence features using a DNN

DNNs are known to be effective in capturing biological sequence features [167, 83]. Here, we design a DNN consisting of two components (Figure 2.1b) to capture informative

features from isoform sequences and conserved domains, respectively. We use a convolutional neural network (CNN) to extract sequence features. Specifically, we first translate each isoform CDS to an amino acid sequence. Then, each sequence is represented as a series of overlapping trigrams, denoted as $s = (t_1, t_2, ..., t_m)$. Each trigram is embedded as a continuous vector by the dense embedding layer (denoted as embed$(\cdot)$) [10]. Note that the vector representations are optimized during the training process and thus are able to capture similarities between the trigrams. We then employ a one-dimensional convolutional layer with multiple convolution filters (denoted as conv$(\cdot)$) to scan the encoded sequence and detect the functional sites. After that, pooling (denoted as pool$(\cdot)$) and dense (denoted as dense$(\cdot)$) layers are used to reduce the dimensionality of the hidden features.

A big challenge here is that the lengths of isoform sequences vary a lot. Due to the fixed size of pooling window and stride, the output size of a normal pooling layer depends on the length of the input sequence, which makes connecting the pooling layer to the following dense layer impossible. To address this problem, we adopt a 'pyramid pooling' layer in our model, which is widely used in computer vision [58]. We modify it, however, as a one-dimensional pooling layer. The pyramid pooling layer can generate a fixed-length output regardless of the input sequence length. Specifically, it uses multi-level pooling bins. Pooling bins at different levels have sizes proportional to the sequence length with different ratios. The number of bins at each level is fixed. High level pooling bins capture global features while low level bins capture local features.

Conserved domains are the building blocks of proteins. Their duplication, fusion and recombination during evolution produce proteins with novel structures and functions.

In addition, the order of domains is also conserved during evolution [85]. Rearrangement of domains can influence functions of a protein. We use a recurrent neural network (RNN) to capture domain features. Domain order information is considered in the network structure design. Specifically, we order the conserved domains of an isoform as a sequence, denoted as $d = (dm_1, dm_2, ..., dm_n)$, where each domain is represented by a unique ID. Then, we use the same dense embedding technique to embed each ID into a vector representation. To capture the order information of domains, we apply the recurrent layer with long short-term memory (LSTM) units (denoted as LSTM($\cdot$)) to process the encoded domain sequence. The output of the last LSTM unit is used as the feature vector from the domain component.

Feature vectors from both the sequence and domain components are then concatenated to form a unified feature representation. Finally, the unified representation is fed into a logistic regression layer (denoted as logit($\cdot$)) to compute the initial score as follows. Formally, given isoform sequence s and sequence of domains d, the initial score computed as follows:

$$\begin{aligned} InitialScore(s, d) &= \text{logit}(\text{dense}(f_s(s), f_d(d))) \\ f_s(s) &= \text{dense}(\text{pool}(\text{conv}(\text{embed}(s)))) \\ f_d(d) &= \text{LSTM}(\text{embed}(d)). \end{aligned} \tag{2.1}$$

2.2.4 Exploring co-expression relationship using a CRF

The function of an isoform is sometimes determined by its interacting partners that are often co-expressed. To capture the co-expression relationship among isoforms, we design a CRF in the second stage (Figure 2.1b). Co-expression networks are first derived from the RNA-seq data. Specifically, we construct a co-expression network for each SRA

study using the WGCNA algorithm [89], which has been widely used in the studies of weighted correlation network analysis. To ensure the network quality, we only consider SRA studies with at least ten experiments. This results in a total of 42 networks. For each pair of isoforms, the absolute value of the Pearson correlation coefficient (PCC) between their expression profiles is assigned as the corresponding edge weight using the soft threshold method of WGCNA.

We denote the sequence, domains and expression profile of an isoform i as s_i, d_i and e_i, and use a binary scalar y_i to denote its label, indicating whether the isoform has the function under consideration or not. The CRF model aims to assign each isoform a label by minimizing a Gibbs energy function, which is defined as:

$$E(y|s,d,e) = \theta_1 \sum_i \psi_u(y_i|s_i,d_i) + \theta_2 \sum_{i<j} \psi_p(y_i,y_j|e_i,e_j). \tag{2.2}$$

The Gibbs energy is characterized by both the initial scores from the DNN and the co-expression relationship between isoforms. The unary potential $\psi_u(y_i|s_i,d_i)$ comes from the initial scores, which is defined as $\psi_u(1|s_i,d_i) = 1 - InitialScore(s_i,d_i)$ and $\psi_u(0|s_i,d_i) = 1-\psi_u(1|s_i,d_i)$. The co-expression relationship is considered in the pairwise potential, which is defined as:

$$\psi_p(y_i,y_j|e_i,e_j) = \mu(y_i,y_j) \sum_l w_l(e_i,e_j), \tag{2.3}$$

where $w_l(e_i,e_j)$ is the edge weight between isoform i and isoform j in the l-th co-expression network and $\mu(y_i,y_j)$ is a label compatibility function defined as $\mu(y_i,y_j) = [y_i \neq y_j]$ that is used to penalize highly co-expressed isoforms with differently assigned labels. The weights

θ_1 and θ_2 control the relative importance of the unary potential ψ_u and pairwise potential ψ_p in the Gibbs energy, and will be discussed more in the next section.

By finding a label assignment $\hat{y}$ that minimizes the Gibbs energy $E(\hat{y}|s,d,e)$, we aim to assign each isoform an label with low unary energy and, at the same time, ensure that highly co-expressed isoforms get the same label. Because of the computational complexity of exact inference, we apply an efficient approximation algorithm named the mean field approximation similar to [80]. Here, minimizing the Gibbs energy is formulated as maximizing the following probability:

$$P(y|s,d,e) = \frac{1}{Z}\exp(-E(y|s,d,e)), \tag{2.4}$$

where $Z = \sum_y \exp(-E(y|s,d,e))$ is a normalization constant. Instead of computing the exact distribution $P(y|s,d,e)$, the approximation algorithm computes a distribution $Q(y|s,d,e)$ that minimizes the KL-divergence $\mathbf{D}(Q||P)$, where the distribution Q is defined as a product of independent marginals:

$$Q(y|s,d,e) = \prod_i Q_i(y_i|s_i,d_i,e_i). \tag{2.5}$$

Minimizing the KL-divergence yields the following iterative update equation:

$$\begin{aligned} Q_i(y_i|s_i,d_i,e_i) = &\frac{1}{Z_i}\exp\{-\theta_1\psi_u(y_i|s_i,d_i) \\ &- \theta_2\sum_{j\neq i}\sum_l w_l(e_i,e_j)Q_j(1-y_i|s_j,d_j,e_j)\}. \end{aligned} \tag{2.6}$$

Q_i is initialized with the unary potential and updated iteratively according to Equation 2.6

Algorithm 1 Training algorithm of DIFFUSE

Initialization: Initialize the label $\hat{y}_i$ of each instance in a positive or negative bag as $\hat{y}_i = 1$ or 0, respectively. Initialize DNN parameters w and CRF parameters θ.

Parameter update: Fix instance labels and update model parameters.

Compute $\nabla_w \ell_{DNN}(w : s, d, \hat{y})$ and use SGD to update w.
Compute $\nabla_\theta \ell_{CRF}(\theta : s, d, e, \hat{y})$ and use L-BFGS-B to update θ.

Label update: Fix model parameters and update instance labels.

for each instance i in positive bags **do**
    $\hat{y}_i = \arg\max_{y_i} Q_i(y_i)$
end for
for each positive bag b **do**
    if $\max(\hat{y}_i) == 0$, for all instances i belonging to bag b **then**
        $i = \arg\max_i Q_i(1)$, for all instances i belonging to bag b
        $\hat{y}_i = 1$
    end if
end for

until convergence, which gives the final output of our model.

2.2.5 Training the model with the MIL framework

Due to the lack of ground truth isoform labels, the conventional supervised training algorithm cannot be directly applied to our model. Hence, we adopt a semi-supervisd model training algorithm under the MIL framework similar to the one in [3], which is outlined in Algorithm 1. In the MIL framework, each gene is treated as a bag, the isoforms of a gene are treated as the instances in the bag, and only the ground truth labels of the bags (*i.e.*, genes) are required. A positive bag refers to a gene that has the function under consideration. Clearly, a positive bag should contain at least one positive instance, while a negative bag should contain no positive instances. We first initialize the instances of positive bags with positive labels, and the others with negative labels. Then, the model parameters can be optimized with the initial labels in the normal supervised learning manner. In

particular, given the training instances $\{(s_i, d_i, e_i, \hat{y}_i)\}_i$, the loss function in terms of the DNN parameters w is defined as the sum of the negative log likelihoods:

$$\begin{aligned}\ell_{DNN}(w : s, d, \hat{y}) = &- \sum_i \hat{y}_i \log(InitialScore(s_i, d_i)) \\ &+ (1 - \hat{y}_i) \log(1 - InitialScore(s_i, d_i)).\end{aligned} \tag{2.7}$$

Gradients in terms of w can be computed and the stochastic gradient descent (SGD) algorithm is used to optimize w. Similarly, the CRF parameters θ are optimized by minimizing the negative log-likelihood ℓ_{CRF} using the L-BFGS-B algorithm [172], which is defined as:

$$\ell_{CRF}(\theta : s, d, e, \hat{y}) = -\log P(\hat{y}|s, d, e) + \sum_i \frac{\theta_i^2}{2\sigma^2}. \tag{2.8}$$

Here, the second term is a regularization term to reduce over-fitting, where σ^2 is a free parameter that determines how much to penalize large weights. L-BFGS-B requires to compute the gradient of ℓ_{CRF} in terms of θ. However, the number of terms in Z of $P(\hat{y}|s, d, e)$ is exponential in the number of instances, making the gradient computation intractable. We therefore use an approximate gradient algorithm given in [139], which approximates the true gradient by replacing P with the marginals Q:

$$\begin{aligned}\frac{\partial}{\partial\theta_1}\ell_{CRF}(\theta : s, d, e, \hat{y}) \approx &\sum_i Q_i(1 - \hat{y}_i|s_i, d_i, e_i)(\psi_u(\hat{y}_i|s_i, d_i) \\ &- \psi_u(1 - \hat{y}_i|s_i, d_i)) + \frac{\theta_1}{\sigma^2},\end{aligned} \tag{2.9}$$

$$\frac{\partial}{\partial \theta_2}\ell_{CRF}(\theta : s, d, e, \hat{y}) \approx \sum_i Q_i(1-\hat{y}_i|s_i, d_i, e_i)$$
$$(\sum_{j\neq i}\psi_p(\hat{y}_i, 1-\hat{y}_i|e_i, e_j) - \sum_{j\neq i}\psi_p(1-\hat{y}_i, \hat{y}_i|e_i, e_j)) + \frac{\theta_2}{\sigma^2}. \tag{2.10}$$

After updating the parameters of the model, we perform inference for each instance in positive bags using the new model. Instance labels are then updated according to the inference: $\hat{y}_i = \arg\max_{y_i} Q_i(y_i)$. For each positive bag, if all its instances are assigned with negative labels, we select the instance with the largest positive prediction score $Q_i(1)$ in the bag as positive. The parameter update step and the label update step are repeated alternately until convergence.

2.2.6 Implementation details

A large number of manually reviewed protein sequences with annotated GO terms are available on the SwissProt [13] database. Most proteins in the database represent the canonical isoforms of genes and therefore will not help improve the model's ability to differentiate the isoform functions of the same gene. However, they are still precious resources that can help our DNN learn important functional features from sequences and domains. We download 89,459 eukaryotic (other than human) protein sequences with GO annotation from the SwissProt database. Conserved domain data are downloaded accordingly using the same method described before. The data are used to train the DNN. Specifically, given the sequence, domains and ground truth label of each protein instance, the initial score and loss of DNN are computed for the instance and then the loss is used to update the DNN parameters.

We partition our data into the training, validation and test sets with the proportions of 70%, 10% and 20%, respectively. To avoid potential information leak (*i.e.*, isoforms with very similar sequences and similar functions appear in different components of the partition), we split the data according to two criteria. First, we require that isoforms of the same gene are partitioned into the same set. Second, since our data contains proteins from different eukaryotes, we forbid orthologous genes to be split. In other words, we consider clusters of orthologous groups (COGs) [143] and require that all genes of the same COG are partitioned together. 10,308 COGs are downloaded from the EggNOG database [62]. Note that the SwissProt proteins are only used for training our model and are not involved in testing. Hyperparameters of the model are manually tuned based on the model performance on the validation data. The validation data are then merged with the training data to train a final model for each GO term before we assess its performance in terms of AUC and AUPRC.

In our computational experiments, the Adam optimizer [77] is used to optimize the DNN. The sizes of the embedding vectors for both amino acid trigrams and domain unique IDs are 32. We use 64 convolution filters with length 32 and stride 1. The pyramid pooling layer consists of pooling bins from four levels, with 1, 2, 4, and 8 bins at each level, respectively. To prevent over-fitting the model, the dropout [134] technique is adopted. The DNN model is implemented using the Keras library with TensorFlow [1] as the backend. The SciPy package is used for implementing the L-BFGS-B algorithm. To accelerate the training process, NVIDIA K80 GPUs are used.

2.3 Results

2.3.1 Prediction performance of DIFFUSE

Since annotated isoform functions are generally unavailable, following the evaluation procedure used in previous isoform function prediction studies [99, 40, 102, 129], we first evaluate the performance of our method using gene-level functional annotation. For each GO term, the maximum prediction score among the isoforms of a gene is taken to check its consistency with the gene annotation. To investigate how the prediction performance may be influenced by GO branches and the number of positive genes, we divide all the GO terms into 12 groups based on GO branch and term size, which is defined as the number of genes associated with a GO term. Specifically, we first divide GO terms into three groups based on the three main GO branches (*i.e.*, Biological Process (BP), Molecular Function (MF) and Cellular Component (CC)). Then, the terms of each group are divided into four subgroups with term sizes in the ranges of [10, 20], [21, 50], [51, 100], and [101, 1000], respectively. Both AUC and AUPRC are used to evaluate the performance for each GO term. Since the baseline for AUPRC (ratio of positive genes in the test set) is different for different GO terms, to make comparison across different groups more fair, we unify the AUPRC baseline as 0.1 for all terms by duplicating positive genes in the test set. Out of the 4,184 GO terms, 3,037 are in the BP group, 432 in CC and 715 in MF.

The (numerical, also called macro) average AUC value for BP, CC and MF are 0.829, 0.850 and 0.881, and the average AUPRC values are 0.563, 0.586 and 0.656, respectively. The distributions of AUC and AUPRC values in different groups are shown in

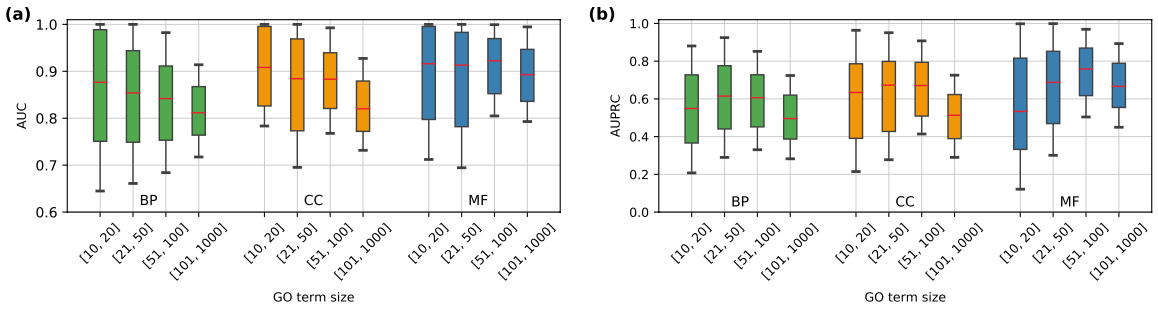


Figure 2.2: Performance evaluation in terms of AUC and AUPRC. GO terms are divided into groups based on the three main GO branches and term sizes. (a) Distributions of AUC scores over GO terms in different groups. (b) Distributions of AUPRC scores.

Figure 2.2. Interestingly, we observe that more positive genes do not yield higher performance. The groups with the largest term sizes (*i.e.*, range [101, 1000]) in fact have relatively low AUC and AUPRC values compared with the other groups. This phenomenon has been observed in several previous studies as well [129, 99]. A possible explanation is that as the term size increases, the biological features (*i.e.*, sequences and expression) of isoforms associated with a GO term become more heterogeneous and the correlation between the functional similarity and the similarities of the biological features decreases, as discussed in detail in [129].

2.3.2 Performance comparison with the existing methods

We compare DIFFUSE with four state-of-the-art isoform function prediction methods including mi-SVM [40], iMILP [99], WLRM [102], and DeepIsoFun [129]. The comparison focuses on a small set of GO terms, GO Slim [30], which provides a broad overview of the ontology content. 96 GO terms are kept after the term size filtration mentioned above. To

Table 2.1: Comparison between DIFFUSE and other isoform function prediction methods.

Method	Dataset#1		Dataset#2		Dataset#3	
	AUC	AUPRC	AUC	AUPRC	AUC	AUPRC
DIFFUSE	**0.835**	**0.585**	**0.828**	**0.537**	**0.817**	**0.524**
DeepIsoFun	0.729	0.280	0.722	0.257	0.712	0.231
WLRM	0.685	0.265	0.667	0.237	0.672	0.201
mi-SVM	0.668	0.248	0.671	0.221	0.706	0.235
iMILP *	0.678	0.317	0.662	0.292	0.639	0.288

* Since iMILP classifies an isoform into three classes rather than two classes for a given GO term (*i.e.*, positive, negative or unknown), we measure its AUC and AUPRC values using only the positive and negative classes.

make a comprehensive comparison, besides the dataset analyzed above (called Dataset#1), we include two other datasets from the literature [99, 40]. In particular, Dataset#2 contains RNA-seq data for 29,806 human isoforms of 18,923 genes, which were generated from 29 SRA human studies consisting of 455 experiments. Dataset#3 contains RNA-seq data for 16,191 mouse isoforms of 13,692 genes, which were generated from 116 SRA studies consisting of 365 experiments. The corresponding sequence, domain and annotation data are collected by following the same procedure described in the 'Materials and methods' section. The average AUC and AUPRC values are reported in Table 2.1. Note that iMILP performs a 3-class classification rather than 2-class. While all other methods treat genes without a GO annotation as negatives of this GO term, iMILP selects negative genes according a more stringent criterion and treats the others as unknowns. Here, we assess the AUC and AUPRC of iMILP based only on the positive genes and selected negative genes, which might incur some favorable bias for the method. Nonetheless, significant improvements by our method have been observed. DIFFUSE achieves improvements of 14.5%, 14.7% and 14.7% in terms of AUC and 84.5%, 83.9% and 81.9% in terms of AUPRC over the best performance of the other methods on the three datasets, respectively. Some example receiver operating

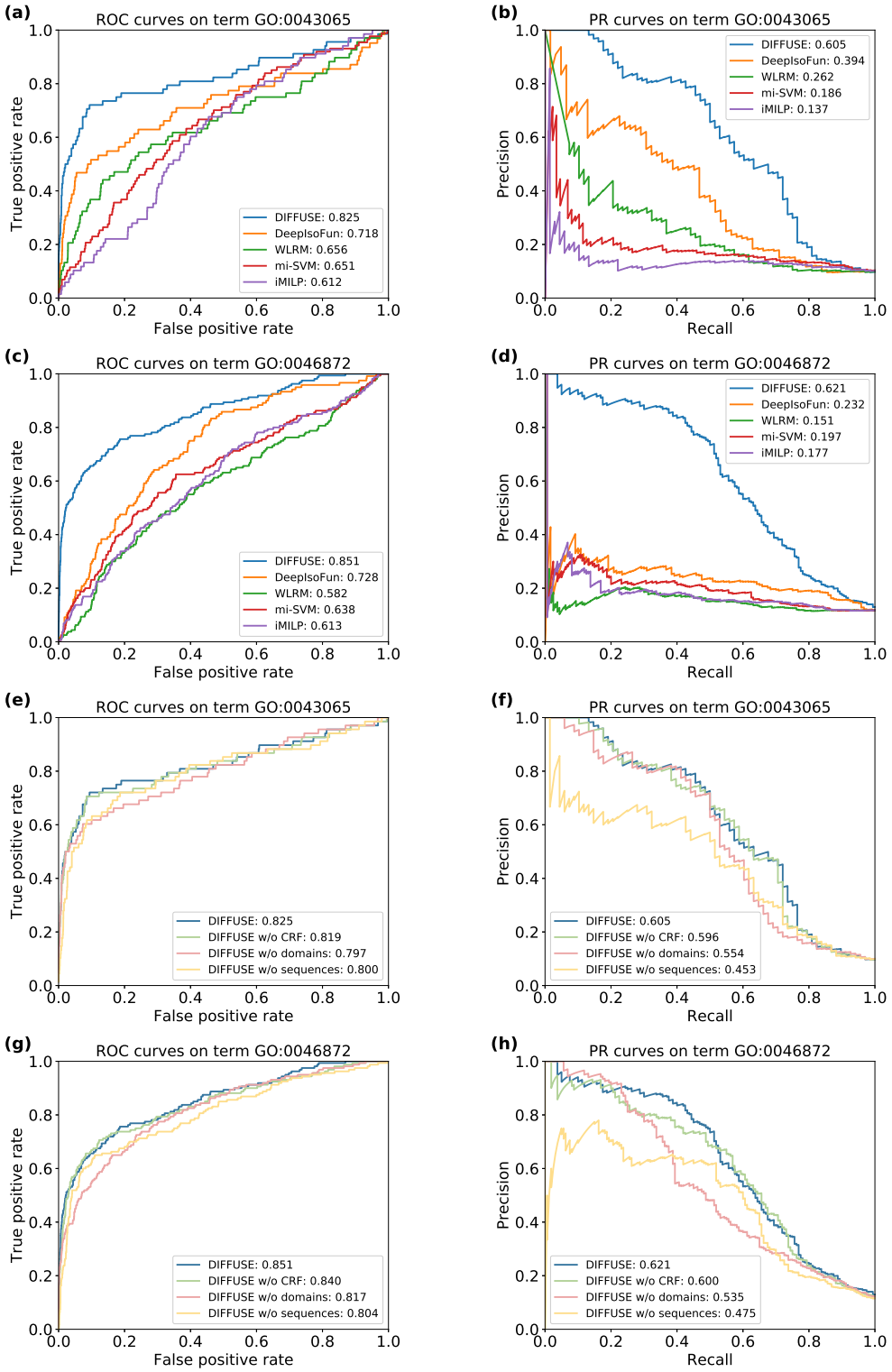

(a)
ROC curves on term GO:0043065
True positive rate
False positive rate
DIFFUSE: 0.825
DeepIsoFun: 0.718
WLRM: 0.656
mi-SVM: 0.651
iMILP: 0.612
(b)
PR curves on term GO:0043065
Precision
Recall
DIFFUSE: 0.605
DeepIsoFun: 0.394
WLRM: 0.262
mi-SVM: 0.186
iMILP: 0.137
(c)
ROC curves on term GO:0046872
True positive rate
False positive rate
DIFFUSE: 0.851
DeepIsoFun: 0.728
WLRM: 0.582
mi-SVM: 0.638
iMILP: 0.613
(d)
PR curves on term GO:0046872
Precision
Recall
DIFFUSE: 0.621
DeepIsoFun: 0.232
WLRM: 0.151
mi-SVM: 0.197
iMILP: 0.177
(e)
ROC curves on term GO:0043065
True positive rate
False positive rate
DIFFUSE: 0.825
DIFFUSE w/o CRF: 0.819
DIFFUSE w/o domains: 0.797
DIFFUSE w/o sequences: 0.800
(f)
PR curves on term GO:0043065
Precision
Recall
DIFFUSE: 0.605
DIFFUSE w/o CRF: 0.596
DIFFUSE w/o domains: 0.554
DIFFUSE w/o sequences: 0.453
(g)
ROC curves on term GO:0046872
True positive rate
False positive rate
DIFFUSE: 0.851
DIFFUSE w/o CRF: 0.840
DIFFUSE w/o domains: 0.817
DIFFUSE w/o sequences: 0.804
(h)
PR curves on term GO:0046872
Precision
Recall
DIFFUSE: 0.621
DIFFUSE w/o CRF: 0.600
DIFFUSE w/o domains: 0.535
DIFFUSE w/o sequences: 0.475

Figure 2.3: Example receiver operating characteristic (ROC) curves and precision-recall (PR) curves on terms GO:0043065 (positive regulation of apoptotic process) and GO:0046872 (metal ion binding). The plots (a-d) illustrate the ROC and PR curves achieved by the methods compared in Table 1. The curves on both GO terms demonstrate that DIFFUSE performs better than the other methods across all thresholds on false positive rate or recall. The plots (e-h) illustrate the ROC and PR curves on the same GO terms achieved by the four variants of DIFFUSE discussed in the later section 3.1.3. The PR curves on both GO terms suggest that after removing sequence features from DIFFUSE, the model predicts more false positives with high scores, which may explain why the AUPRC of DIFFUSE drops so significantly without using sequences.

Table 2.2: Comparison of the performance of DIFFUSE in training and testing on all three datasets. The average performance gaps across the three datasets are 6.3% in terms of AUC and 8.3% in terms of AUPRC. These are well within acceptable ranges reported in the literature [166, 75] and thus likely to indicate that our model was not grossly overtrained in the experiments.

	Dataset#1		Dataset#2		Dataset#3	
	AUC	AUPRC	AUC	AUPRC	AUC	AUPRC
DIFFUSE *training*	0.891	0.632	0.882	0.588	0.875	0.574
DIFFUSE *test*	0.835	0.585	0.828	0.537	0.817	0.524

characteristic curves and precision-recall curves on two GO terms achieved by the methods are illustrated in Figure 2.3a-d. The performance of DIFFUSE on the training data is given in Table 2.2 to show that the model is not grossly overtrained.

2.3.3 Analyzing the effects of model components

To evaluate the contribution of some key components and biological features used in our model, we perform an ablation study by removing these components/features from model and measuring how the performance of the model is affected. Specifically, we remove the CRF component, conserved domain features and sequence features from DIFFUSE, respectively, and test its performance on GO Slim. We observe that the average AUC drops 1.7% (from 0.835 to 0.821) and the average AUPRC drops 7.5% (from 0.585 to 0.541) without the CRF. The average AUC drops 3.7% (from 0.835 to 0.804) and the average AUPRC drops 21.2% (from 0.585 to 0.461) without using conserved domains. The average AUC drops 4.6% (from 0.835 to 0.797) and the average AUPRC drops 27.9% (from 0.585 to 0.422) without using sequences (Figure 2.4a). The results suggest that the CRF component is very effective in capturing the co-expression relationship and conserved domains contain important functional information (as known before), and both contribute significantly to

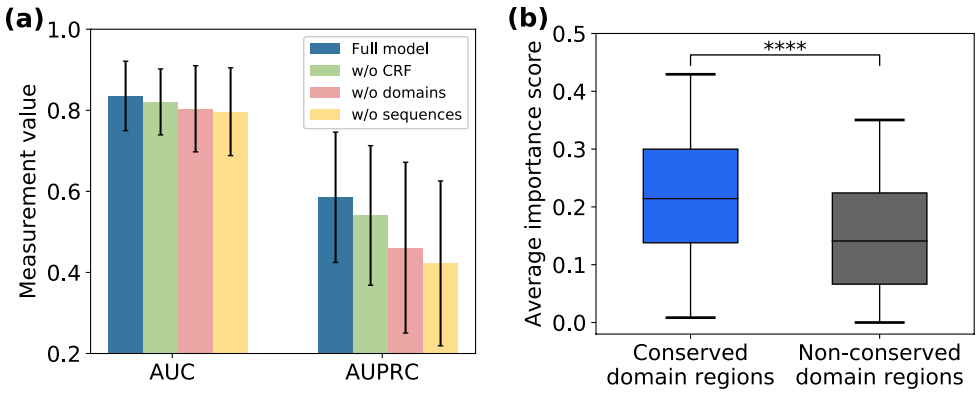


Figure 2.4: (a) The average AUC and AUPRC values over the terms in GO Slim for DIFFUSE (blue), DIFFUSE without CRF (green), DIFFUSE without using conserved domains (pink), and DIFFUSE without using sequences (yellow). (b) Average importance scores for conserved domain regions and non-conserved domain regions are calculated for each isoform-GO term pair. There is a clearly a significant difference between these two regions as supported by the one-sided Wilcoxon test (****P <0.0001).

the performance of DIFFUSE. Moreover, although conserved domains are extracted from sequences, they cannot completely replace sequences. Some example receiver operating characteristic curves and precision-recall curves on two GO terms achieved by the above four variants of DIFFUSE are illustrated in Figure 2.3e-h.

2.3.4 Importance of local sequence features in function prediction

Deep learning models are usually considered as "black boxes". In the bioinformatics domain, however, understanding the rationales behind decisions made by a model is very important to the potential users of the model. Here, we use the saliency map [132], a deep learning visualization technique, to help us understand what parts of an isoform sequence are most influential in the classification decision. Briefly, a saliency map calculates the derivative of the output of the DNN with respect to the variable at each input position,

so we can see the influence of each position of the input sequence on the output score. We denote the value of derivative at each position as its 'importance score'. The Keras-vis tool [78] is used to calculate the saliency map and the method in [88] is used to obtain the importance score of each amino acid (AA) residue of the input sequence. Since conserved domains are known to be rich in functional sites, residues inside conserved domains are expected to have higher importance scores on average.

To test this hypothesis, for each isoform-GO pair, we compute a saliency map and calculate the importance score for each AA residue of the isoform. For each saliency map, we calculate the average importance score of all AA residues inside conserved domains and that of all AA residues outside conserved domains, respectively. As expected, we observe significantly higher average importance scores in conserved domains (Figure 2.4b).

2.3.5 Analyzing the divergence of isoform functions

Delineating the specific functions of the isoforms is the ultimate goal of isoform function prediction. Hence, it would be useful to analyze the divergence of the predicted functions of the isoforms from each gene, as done in [99, 129]. We estimate the similarity of predicted functions for each pair of isoforms in terms of the semantic similarity score using GOssTo [22], again considering the three GO branches separately. The semantic dissimilarity score of two isoforms is then defined as one minus their similarity score. For each MIG, the functional divergence of its isoforms is calculated by averaging the semantic dissimilarity scores of all pairs of its isoforms sharing predicted functions in the same GO branch. Out of the 9,032 MIGs, 8,924 (5,444 or 5,521) MIGs have at least two isoforms assigned GO terms in the BP (CC or MF, respectively) branch by DIFFUSE. Among

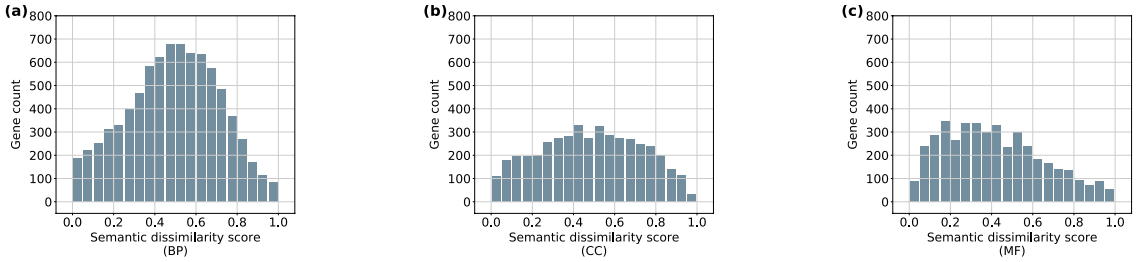


Figure 2.5: Distributions of semantic dissimilarity scores of MIGs that have functionally divergent isoforms. The range of semantic dissimilarity score [0,1] is equally divided into 20 bins. For each bin, we count how many MIGs have semantic dissimilarity scores in this range. The three GO branches are considered separately.

these MIGs, 90.3% (8,060 out of 8,924), 81.1% (4,415 out of 5,444) and 76.5% (4,222 out of 5,521) are estimated to have functional divergent isoforms (*i.e.*, semantic dissimilarity scores greater than 0) with respect to BP, CC and MF, respectively. The dissimilarity score distributions for MIGs that have functional divergent isoforms are shown in Figure 2.5, where the mean score values are 0.490, 0.482 and 0.411 for BP, CC and MF, respectively. A similar pattern of distributions was observed in a previous study [99].

As discussed above, functional divergence among isoforms of the same gene is expected. It remains unclear, however, to what extent isoforms have divergent functions. Therefore, we further investigate the functional divergence of isoforms by testing its consistency with the (protein) structural divergence of isoforms. In other words, for a gene with isoforms that share similar functions (*i.e.*, low semantic dissimilarity score), the protein structures of these isoforms are expected to be similar, and vice versa. The protein structure of an isoform can be represented as a contact map, which is a two-dimensional matrix of distance between all possible AA residue pairs and can be used to estimate protein structural similarities. Contact maps are predicted using the RaptorX [117] server. Due to

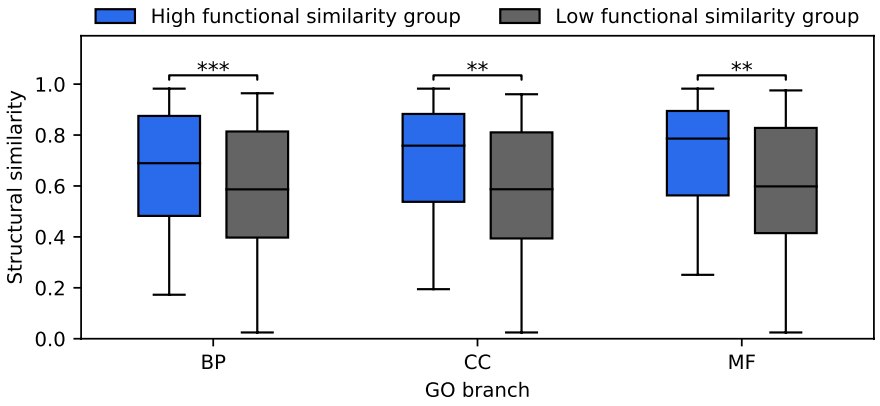


Figure 2.6: Average structural similarity between isoforms of MIGs with low or high functional similarities. Significant differences are observed in all the GO branches according to the Kruskal-Wallis tests (with P-values ***P=5.77e-04, **P=2.50e-03 and **P=3.30e-03 for BP, CC and MF, respectively). Note that the semantic dissimilarity score can only be calculated for MIGs containing two or more isoforms with GO terms in the same branch. This results in 296 (167 or 155) out of the 300 MIGs considered for the BP (CC or MF) branch.

the computational intensity of contact map prediction, we predict contact maps for isoforms of 300 randomly selected MIGs with at most 500 AAs. For each GO branch separately, we divide the genes into two groups by the median semantic dissimilarity score, resulting in a high functional similarity group and a low functional similarity group. For each gene, we calculate the average structural similarity score over all its isoform pairs, measured by the Contact Map Overlap (CMO) using the software AI-Eigen [38]. As anticipated, we observe significantly higher structural similarities between isoforms of MIGs in the high functional similarity groups for all three GO branches (Figure 2.6).

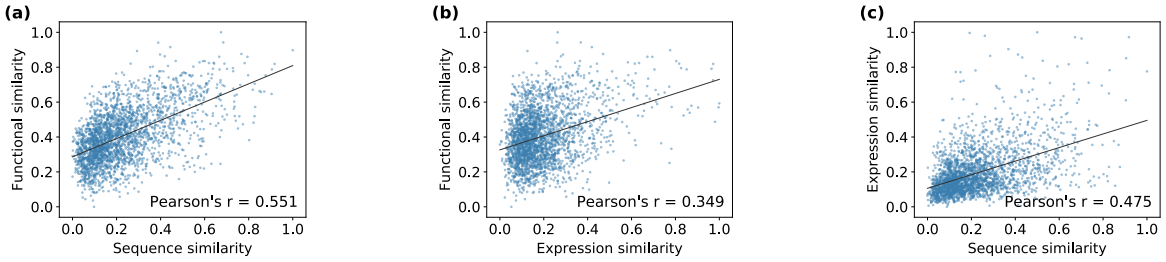


Figure 2.7: Correlations between functional similarity, sequence similarity and expression similarity. The isoforms are grouped into 2,492 clusters by hierarchical clustering. The average pairwise functional similarity, sequence similarity and expression similarity are estimated for the isoforms in each cluster. The Pearson correlation coefficient is used to measure the strength of correlation.

2.4 Validation of predicted isoform functions

The scarcity of experimentally-verified functions of isoforms raises a great a challenge to the validation of our predicted isoform functions. To address this challenge, we first indirectly validate the predicted functions by analyzing how they are correlated with isoform sequences, expression as well as protein structures. The predictions are further validated by evaluating their consistency with some well-studied UniProt sequence features related to functions. Finally, we directly validate a small set of predicted isoform functions analyzed above by a targeted literature search.

2.4.1 Correlations between functional, sequence and expression similarities

Our method is based on the assumption that isoforms with similar sequences and/or expression profiles should have similar functions. To check that our predicted functions indeed have this property, we test whether similar biological features indeed lead to

similar predictions and vice versa, as done in [129]. (Hence, this is more of a sanity check on our computational model than a proper validation of our predicted isoform functions.) We group the 39,375 isoforms into 2,492 clusters with sizes in the range of [10, 20] based on hierarchical clustering, where the bit score of BLAST [2] is used to measure the pairwise distance of isoforms. Then the average functional similarity, sequence similarity and expression similarity are estimated over all isoform pairs within each cluster. Different from the last subsection, here the functional similarity between isoforms is measured by the negative value of the Euclidean distance between their predicted functions (as two vectors). The expression similarity is measured by the PCC of two expression profiles and the sequence similarity is measured by the pairwise global alignment score of two isoform protein sequences normalized by the alignment length. Each similarity is normalized to the range of [0, 1]. Then, the PCC is used to measure the pairwise correlations between functional similarity, sequence similarity and expression similarity, as shown in Figure 2.7. Clearly, isoforms with similar sequences or expression profiles tend to have similar predicted functions. Interestingly, functional similarity seems to be more correlated with sequence similarity than with expression similarity and only a moderate correlation is found between sequence similarity and expression similarity, which explains why these two biological features can be combined to improve function prediction. To further verify these isoform-level correlations, we perform the same computational experiment at the gene level where the gene functional annotation, the longest isoform sequence of each gene and gene expression profiles are used to estimate the above three similarities. Very similar PCC values are obtained as shown in Figure 2.8.

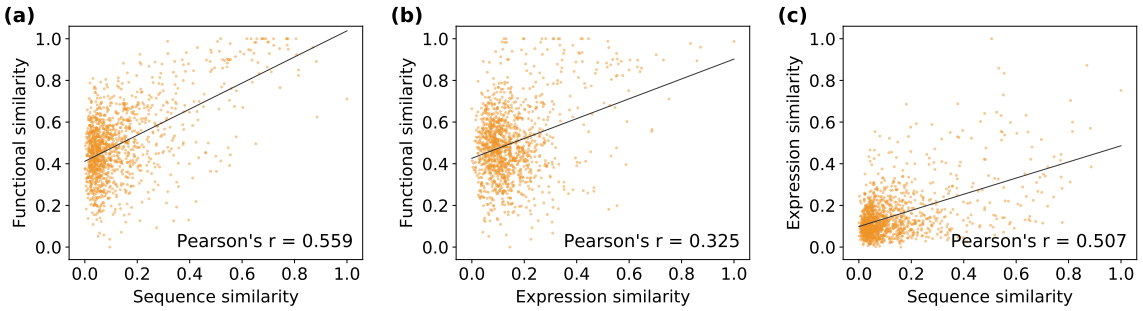


Figure 2.8: Correlations between functional similarity, sequence similarity and expression similarity at the gene level. The genes are grouped into 1254 clusters with sizes in the range of [10, 20] based on hierarchical clustering. The average pairwise functional similarity, sequence similarity and expression similarity are estimated for each cluster. The Pearson correlation coefficient (PCC) is used to measure the correlations.

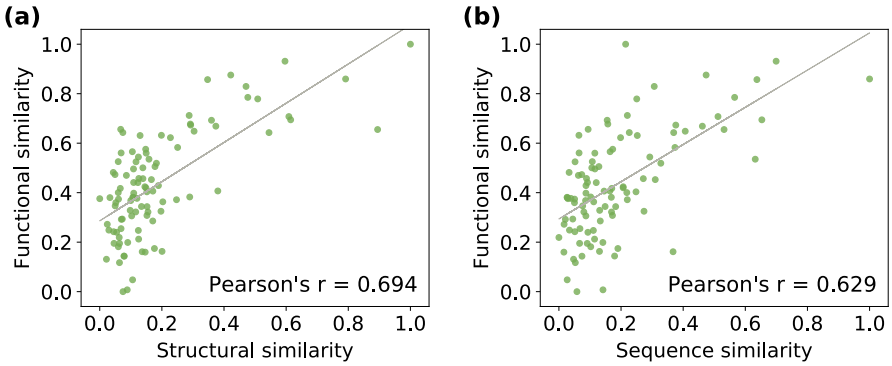


Figure 2.9: 1,500 isoforms of MIGs are grouped into 99 clusters. The average pairwise functional similarity, sequence similarity and structural similarity are estimated for each cluster. (a) The correlation between functional similarity and structural similarity. (b) The correlation between functional similarity and sequence similarity.

2.4.2 Correlation between functional and structural similarities

Previous studies [68] have shown that protein structures are more conserved than sequences. Hence, isoforms with similar functions are expected to have more similar structures than sequences. We further test how the predicted functions are correlated with protein structures represented as contact maps. Again, we download contact maps from the RaptorX server for 1,500 isoforms of MIGs. We focus on MIGs rather than SIGs in this

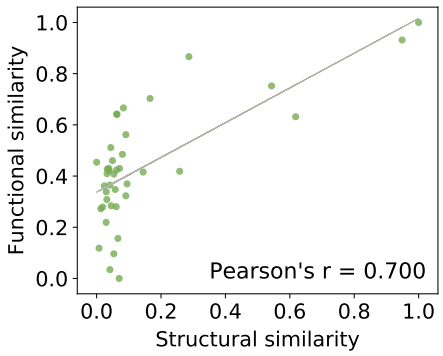


Figure 2.10: The correlation between functional similarity and structural similarity measured on 600 SIGs that are grouped into 40 clusters with sizes in the range of [10, 20]. The functional similarity is estimated based on the annotated functions of the SIGs.

test since the functions of their isoforms are currently unknown. The isoforms are grouped into 99 clusters with sizes in the range of [10, 20] and the average functional similarity, sequence similarity and structural similarity are measured for each cluster using the same methods described above. As expected, a higher PCC is found between functional similarity and structural similarity (Figure 2.9). Furthermore, we perform the same computational experiment on 600 random SIGs using their annotated functions and obtain a consistent PCC between functional similarity and structural similarity (see Figure 2.10). These analyzes indirectly support our prediction results.

2.4.3 Consistency with well-studied UniProt sequence features

A recent review [138] reported a set of function related sequence features (as defined by UniProt; [18] associated with a list of isoforms. The presence or absence of these functional sequence features can be used to infer potential isoform functions. Three of the functional sequence features can be mapped to GO terms, which are 'Metal ion binding

site' (to GO:0046872), 'ATP binding site' (to GO:0005524) and 'Nuclear localization signal' (to GO:0005634). We then map the list of isoforms reported in this review to our isoform dataset. For each GO term, we check the consistency between the presence or absence of the corresponding sequence feature in associated isoforms and the functional predictions concerning this GO term. To quantify the consistency for the three GO terms separately, the Jaccard indices are calculated as in the literature [159]. The same computational experiment is repeated for the three other methods as well in order to compare. The Jaccard indices of DIFFUSE are 0.674, 0.700 and 0.700 for GO:0046872, GO:0005524 and GO:0005634, respectively, which are significantly higher than those of DeepIsofun (0.548, 0.595 and 0.579), WLRM (0.514, 0.578 and 0.580), mi-SVM (0.534, 0.517 and 0.569), and iMILP (0.560, 0.581 and 0.521). The detailed results concerning the three GO terms are shown in Tables 2.3, 2.4, and 2.5, respectively.

2.4.4 Validation via the literature

We further perform an exhaustive literature search for experimentally verified functions of the isoforms analyzed above (*i.e.*, appearing in Tables 2.3, 2.4, and 2.5). Functions or strong functional evidence for 14 isoforms of 6 genes have been found. Out of the 14 isoforms, our method predicted correct functions for 11 of them (Table 2.6), which is significantly more accurate than the other methods. It is worth mentioning that 13 of the 14 functions reported in the literature are consistent with the presence or absence of their corresponding UniProt sequence features. This suggests that the UniProt sequence features may serve as a good benchmark to validate predicted isoform functions.

Table 2.3: Consistency between the presence or absence of sequence feature 'Metal ion binding site' and the function predictions concerning GO term GO:0046872 (metal ion binding). Note that a metal ion may have several binding sites. We treat the binding sites that correspond to the same metal ion as a group. Each isoform sequence may contain multiple metal ion binding site groups. If all metal ion binding site groups of an isoform have binding sites affected by alternative splicing, we treat the sequence feature 'Metal ion binding site' as absent in this isoform, noted by a cross. Otherwise, we treat it as present in this isoform, noted by a circle. Positive and negative predictions are represented by circles and crosses as well.

Gene	Isoform	Sequence feature	Predictions				
			DIFFUSE	DeepIsoFun	WLRM	mi-SVM	iMILP
ACE	P12821-1	○	○	○	○	○	○
	P12821-3	○	○	○	×	○	○
	P12821-4	○	○	○	○	○	○
ACMSD	Q8TDX5-1	○	○	○	○	○	○
	Q8TDX5-2	×	○	○	○	○	○
ADAM33	Q9BZ11-1	○	○	○	○	○	×
	Q9BZ11-2	○	○	○	○	○	○
ADAMTS13	Q76LX8-1	○	×	○	○	○	○
	Q76LX8-2	○	○	○	○	○	○
	Q76LX8-3	○	×	○	○	○	○
ALKBH2	Q6NS38-1	○	○	○	○	○	○
	Q6NS38-2	×	×	×	○	○	○
ALKBH8	Q96BT7-1	○	○	○	○	○	○
	Q96BT7-4	○	○	○	○	○	○
ALOX15B	O15296-1	○	○	○	×	×	×
	O15296-2	○	○	○	○	○	○
	O15296-4	○	○	○	○	○	○
ALOX5	P09917-1	○	○	○	×	○	×
	P09917-2	○	○	○	○	×	×
	P09917-3	○	○	○	○	○	○
AOC3	Q16853-1	○	○	○	○	○	○
	Q16853-2	×	×	○	○	○	○
	Q16853-3	○	×	○	○	×	○
APOBEC3F	Q8IUX4-1	○	×	○	○	○	○
	Q8IUX4-3	×	○	○	○	○	○
APOBEC3G	Q9HC16-1	○	○	○	○	○	○
ARG1	P05089-1	○	○	○	○	○	○
	P05089-2	○	○	○	○	○	○
ARSA	P15289-2	×	×	○	○	○	×
ASPH	Q12797-1	○	○	○	○	○	○
	Q12797-10	○	○	○	○	×	○
	Q12797-2	×	×	○	○	×	○
	Q12797-3	×	×	○	○	○	○
	Q12797-4	×	×	○	○	○	○
	Q12797-5	×	×	○	○	×	○
	Q12797-6	×	×	○	○	×	○
	Q12797-7	×	×	×	○	○	○
	Q12797-8	×	×	○	○	×	○
	Q12797-9	×	×	○	○	○	○
ATP1A1	P05023-1	○	○	○	○	○	○
	P05023-3	○	○	○	○	○	○
	P05023-4	○	○	×	○	○	○
ATP7A	Q04656-1	○	○	○	○	○	○
	Q04656-5	○	○	○	○	○	○

ATP7B	P35670-1	○	○	○	○	○	○
	P35670-3	○	○	○	○	○	○
DIS3L2	Q8IYB7-1	○	○	×	○	○	○
	Q8IYB7-3	○	×	○	○	○	○
	Q8IYB7-4	×	×	×	○	○	○
DPP3	Q9NY33-1	○	○	○	○	×	○
	Q9NY33-4	○	○	○	○	○	○
ENDOV	Q8N8Q3-1	○	○	○	○	○	○
	Q8N8Q3-2	○	○	○	○	○	○
	Q8N8Q3-3	○	○	○	○	○	○
ENOSF1	Q7L5Y1-1	○	○	○	○	○	×
	Q7L5Y1-6	○	×	×	○	○	○
EPHX2	P34913-1	○	×	○	○	○	○
	P34913-2	×	×	○	×	○	○
	P34913-3	×	○	○	○	○	○
ERAP2	Q6P179-1	○	○	○	○	×	○
	Q6P179-3	○	○	○	○	○	○
	Q6P179-4	×	×	○	○	○	×
FAN1	Q9Y2M0-1	○	○	○	×	○	○
	Q9Y2M0-2	×	○	○	○	○	○
GALNT2	Q10471-1	○	○	○	×	○	○
GALT	P07902-1	○	○	×	×	○	○
	P07902-2	○	×	○	○	○	○
GCH1	P30793-1	○	○	○	×	×	○
	P30793-2	×	×	×	×	○	×
	P30793-4	×	×	○	○	○	○
HEPH	Q9BQS7-3	○	○	○	○	○	○
	Q9BQS7-4	○	○	○	○	×	○
HMGCL	P35914-1	○	○	○	×	×	○
	P35914-2	○	○	○	○	○	○
HMGCLL1	Q8TB92-1	○	○	○	○	○	○
	Q8TB92-2	○	○	○	○	○	×
	Q8TB92-5	○	○	○	○	○	○
IDE	P14735-1	○	○	×	○	○	○
	P14735-2	×	×	○	×	×	○
IMPA1	P29218-1	○	○	○	○	×	○
	P29218-2	○	○	○	○	○	○
	P29218-3	○	○	○	○	○	○
LHPP	Q9H008-1	○	○	○	○	○	○
	Q9H008-2	×	×	○	○	×	○
MASP1	P48740-1	○	○	○	○	○	○
	P48740-2	○	○	○	○	○	○
	P48740-3	○	○	○	○	×	○
MPPE1	Q53F39-1	○	○	○	○	○	○
	Q53F39-4	○	○	○	○	○	○
NOS1	P29475-1	○	○	○	○	○	○
	P29475-3	○	○	○	○	○	○
	P29475-5	○	○	○	○	○	○
PGM1	P36871-1	○	○	○	○	○	○
	P36871-2	○	○	○	○	○	○
	P36871-3	×	×	○	×	×	○
PPM1M	Q96MI6-4	○	○	×	○	×	○
PPP3CA	Q08209-1	○	○	○	○	○	×
	Q08209-2	○	○	○	×	○	○
	Q08209-3	○	○	○	○	○	○
QPCTL	Q9NXS2-1	○	○	○	○	○	○
	Q9NXS2-3	×	×	○	○	×	○
RGN	Q15493-1	○	○	○	○	○	○
	Q15493-2	×	×	○	○	○	○

RPE	Q96AT9-1	○	○	○	○	○	×
	Q96AT9-3	×	○	○	○	×	○
	Q96AT9-4	×	○	○	○	×	×
	Q96AT9-5	×	○	○	○	○	×
SOD2	P04179-1	○	○	○	×	×	○
	P04179-2	×	○	○	○	×	○
	P04179-3	○	○	○	○	○	○
	P04179-4	○	○	○	×	○	○
SUV39H2	Q9H5I1-1	○	○	○	○	○	○
	Q9H5I1-2	○	○	×	○	○	○
	Q9H5I1-3	○	×	○	○	○	○
TET2	Q6N021-1	○	○	×	×	○	○
	Q6N021-2	×	×	○	○	○	○
THTPA	Q9BU02-1	○	○	○	○	○	○
	Q9BU02-2	×	×	○	○	○	○
USP16	Q9Y5T5-1	○	○	×	○	○	○
	Q9Y5T5-2	○	○	○	○	○	○
XPNPEP1	Q9NQW7-1	○	○	○	○	○	○
	Q9NQW7-3	○	○	○	×	○	○
	Q9NQW7-4	○	○	○	○	○	○
Jaccard index			0.674	0.548	0.514	0.534	0.560

Table 2.4: Consistency between the presence or absence of sequence feature 'ATP binding site' and the function predictions concerning GO term GO:0005524 (ATP binding). Again, note that an ATP maye have several binding sites. We treat the binding sites that correspond to the same ATP as a group. Each isoform sequence may contain multiple ATP binding site groups. If all ATP binding site groups of an isoform have binding sites affected by alternative splicing, we treat the sequence feature 'ATP binding site' as absent in this isoform. Otherwise, we treat it as present in this isoform.

Gene	Isoform	Sequence feature	Predictions				
			DIFFUSE	DeepIsoFun	WLRM	mi-SVM	iMILP
ACLY	P53396-1	○	○	○	○	×	○
	P53396-2	○	×	○	○	×	○
BRSK2	Q8IWQ3-1	○	○	○	○	○	○
	Q8IWQ3-2	○	○	○	○	○	○
	Q8IWQ3-3	○	○	○	○	○	○
	Q8IWQ3-5	○	○	○	○	×	○
	Q8IWQ3-6	×	○	○	○	×	○
CDK11A	Q9UQ88-1	○	○	○	○	○	○
	Q9UQ88-2	○	○	○	○	○	○
	Q9UQ88-4	○	○	×	○	○	○
IDE	P14735-1	○	○	×	×	○	○
	P14735-2	×	×	○	×	×	○
LATS1	O95835-1	○	○	○	○	○	○
	O95835-2	×	×	×	○	×	○
MARK1	Q9P0L2-1	○	○	○	×	○	○
	Q9P0L2-3	○	○	○	○	○	×
MTHFS	P49914-1	○	○	○	○	×	○
OAS2	P29728-1	○	○	○	○	○	○
	P29728-2	○	○	○	○	○	○
	P29728-3	×	×	×	○	×	○
PFKP	Q01813-2	×	○	○	×	×	×
	Q01813-1	○	○	○	○	○	○
SIK3	Q9Y2K2-5	○	○	○	○	○	○
	Q9Y2K2-8	○	○	○	○	○	○
STK26	Q9P289-1	○	○	○	○	○	○
	Q9P289-2	×	×	○	○	○	○
	Q9P289-3	○	×	○	○	○	○
TAOK1	Q7L7X3-1	○	×	○	○	○	○
	Q7L7X3-3	○	○	○	×	×	×
TSSK4	Q6SA08-1	○	○	○	○	○	○
	Q6SA08-2	○	○	○	×	○	○
	Q6SA08-3	×	×	○	○	○	○
UHMK1	Q8TAS1-1	○	○	○	○	○	○
	Q8TAS1-2	○	○	○	○	×	○
	Q8TAS1-3	×	×	○	×	○	○
WNK1	Q9H4A3-1	○	○	×	○	○	×
	Q9H4A3-5	○	○	○	○	○	○
	Q9H4A3-6	○	○	○	○	○	○
WNK4	Q96J92-1	○	○	○	○	×	○
Jaccard index			0.700	0.595	0.578	0.517	0.581

Table 2.5: Consistency between the presence or absence of sequence feature 'Nuclear localization signal' and the function predictions concerning GO term GO:0005634 (nucleus). Note that each isoform sequence may contain multiple nuclear localization signals. If all the nuclear localization signals of an isoform are affected by alternative splicing, we treat the sequence feature 'Nuclear localization signal' as absent in this isoform. Otherwise, we treat it as present in this isoform.

Gene	Isoform	Sequence feature	Predictions				
			DIFFUSE	DeepIsoFun	WLRM	mi-SVM	iMILP
ADK	P55263-1	○	○	○	○	○	×
	P55263-2	×	○	×	○	○	○
	P55263-3	○	○	○	×	×	×
	P55263-4	×	×	○	×	×	×
AIFM1	O95831-1	○	○	○	○	○	○
	O95831-3	○	○	○	○	×	×
	O95831-4	×	×	×	○	×	×
APTX	Q7Z2E3-1	○	○	○	○	○	○
	Q7Z2E3-10	○	○	○	○	×	○
	Q7Z2E3-11	○	○	○	○	○	○
	Q7Z2E3-3	×	×	○	○	×	○
	Q7Z2E3-5	○	○	○	○	○	○
	Q7Z2E3-7	○	○	○	○	○	○
	Q7Z2E3-9	○	○	○	○	×	○
DDX25	Q9UHL0-1	○	○	○	○	○	○
	Q9UHL0-2	×	○	○	○	×	×
DNMT1	P26358-1	○	○	○	○	○	○
	P26358-2	○	○	○	○	○	○
ERBB2	P04626-1	○	○	○	×	×	○
	P04626-4	○	○	○	○	○	○
	P04626-5	○	○	○	○	○	○
ERCC2	P18074-1	○	○	×	○	○	○
	P18074-2	×	×	○	×	×	×
HIPK2	Q9H2X6-1	○	○	○	○	○	○
	Q9H2X6-3	○	○	×	○	○	×
JMJD6	Q6NYC1-1	○	○	○	○	○	○
	Q6NYC1-3	○	○	○	○	○	○
MAPK7	Q13164-1	○	○	○	○	○	○
	Q13164-2	○	×	○	○	○	○
MDM2	Q00987-11	○	○	○	○	○	○
	Q00987-5	×	○	○	×	○	○
OGFOD1	Q8N543-2	×	×	○	○	○	×
PAPOLA	P51003-1	○	○	○	○	○	○
	P51003-2	×	×	×	×	○	×
PIAS1	O75925-1	○	○	×	○	○	○
	O75925-2	○	○	○	○	○	○
PIAS2	O75928-1	○	○	○	○	○	○
	O75928-2	○	○	○	○	○	○
	O75928-3	×	×	○	○	○	○
PIK3C2A	O00443-1	○	○	○	○	○	○
PPP1R8	Q12972-1	○	○	○	○	○	○
	Q12972-2	○	○	○	×	○	○
	Q12972-3	×	×	○	×	×	○
REV1	Q9UBZ9-2	○	○	○	○	○	×
RTEL1	Q9NZ71-1	○	○	○	○	○	○
	Q9NZ71-6	○	○	○	×	○	×
	Q9NZ71-7	○	○	○	○	×	○
	Q9NZ71-9	×	○	○	×	×	×

SPAST	Q9UBP0-1	○	○	○	○	○	×
	Q9UBP0-2	○	○	○	○	○	○
USP4	Q13107-1	○	○	×	×	○	○
	Q13107-2	○	×	○	○	○	○
	Q13107-3	×	×	×	○	○	○
WWOX	Q9NZC7-1	○	×	○	○	○	○
	Q9NZC7-3	○	○	○	○	○	×
Jaccard index			0.700	0.579	0.580	0.569	0.521

Table 2.6: Literature support for 14 isoforms of 6 genes on two GO terms. Positive and negative results are represented as circles and crosses in the table. Experimental evidence concerning relevant functions have been found for 6 genes in the literature: ACE [31], ACMSD [121], GCH1 [5], ADK [33], AIFM1 [35], and PPP1R8 [24].

GO term	Gene	Isoform	Evidence	Prediction method				
				DIFFUSE	DeepIsoFun	WLRM	mi-SVM	iMILP
GO:0046872	ACE	P12821-1	○	○	○	○	○	○
		P12821-3	○	○	○	×	○	○
	ACMSD	Q8TDX5-1	○	○	○	○	○	○
		Q8TDX5-2	×	○	○	○	○	○
	GCH1	P30793-1	○	○	○	×	×	○
		P30793-2	×	×	×	×	○	×
		P30793-4	×	×	○	○	○	○
GO:0005634	ADK	P55263-1	○	○	○	○	○	×
		P55263-2	×	○	×	○	○	○
	AIFM1	O95831-1	○	○	○	○	○	○
		O95831-3	×	○	○	○	×	×
		O95831-4	×	×	×	○	×	×
	PPP1R8	Q12972-1	○	○	○	○	○	○
		Q12972-3	×	×	○	×	×	○
Accuracy				**78.6%**	71.4%	50.0%	64.3%	64.3%

2.5 Applying DIFFUSE to predict isoform functions for some Dichocarpum species

2.5.1 Background

Dichocarpum is a genus of flowering plants belonging to the family Ranunculaceae. Various medicinal metabolites have been found in Dichocarpum species, many of which, have shown clinical utility [54]. Our collaborators have recently sequenced the full-length transcriptomes of five species of Dichocarpum, Dichocarpum basilare, Dichocarpum auriculatum, Dichocarpum fargesii, Dichocarpum lobatipetalum, and Dichocarpum malipoenense, using the PacBio sequencing technology [124]. The gene and isoform functions of these five species have not been carefully studied before. Even though the functions of their genes can be easily predicted from annotations of homologous genes by aligning sequences in the databases, the functions of isoforms can be hardly differentiated by this way since most parts of the sequences of isoforms are the same. In this section, we discuss how to apply DIFFUSE to predict isoform functions for the five Dichocarpum species. Potential applications of our predicted isoform functions include facilitating the understanding of metabolic regulations in these species, which may help medicinal studies concerning plant-based natural products.

2.5.2 Materials and methods

Unless specifically mentioned, the data collection procedures and the predictive method are the same as described in Section 2.2.

Materials

Isoform sequences: Coding sequences (CDS) parts are predicted from the transcripts using the software ANGLE [131]. Then each CDS is translated into an amino acid sequence.

Isoform expression profiles: Each species has one sequencing sample. The sequencing reads are aligned to the reference transcriptome using bowtie2 [90] and the expression levels of transcripts are quantified using RSEM [93] (measured in expected number of Fragments Per Kilobase of transcript sequence per Millions base pairs sequenced or FPKM).

Gene functional annotations: The GO annotations are first collected for each isoform by aligning its protein sequence in the Pfam database [9], after which, isoforms encoded by the same gene usually get the same electronic annotations. We then generate the functional annotations for each gene by merging the annotations of all its isoforms.

GO terms: We perform predictions on all the GO terms appearing in the electronic annotations, which includes 1900 GO terms in total. The numbers of GO terms from BP, CC, and MF branches are 807, 272, and 821 respectively.

Methods

Building isoform expression similarity networks from isoform expression profiles: Since there is no biological replicate of isoform expression for each species, the step of building isoform co-expression network cannot be performed due to the inability of calculating correlations between expression profiles. Instead, we build the networks of

isoforms based on the euclidean distance between their log transformed expression levels as below:

$$w(e_i, e_j) = \max(10 - |\log(1 + e_i) - \log(1 + e_j)|, 0), \tag{2.11}$$

where e_i and e_j are the expression levels of isoform i and isoform j, and $w(e_i, e_j)$ is the edge weight between the two isoforms in the expression similarity network.

2.5.3 Analyses of the prediction results

Divergence of isoform functions

We apply DIFFUSE to predict isoform functions for five species on all 1900 GO terms appearing in the electronic annotations. As done in Section 2.3.5, we check if DIFFUSE can differentiate functions of isoforms encoded by the same gene. We estimate the similarity of functions for each pair of isoforms within each MIG in terms of the semantic similarity score using GOssTo [22], considering the three GO branches separately. The analysis is performed for both the electronic annotations and the predicted functions of DIFFUSE. Figure 2.11 shows the distributions of GO similarity scores of all isoform pairs based on their electronic annotations on three branches separately. There are only 22.1%, 24.8%, and 19.5% isoform pairs that have differences in their functional annotations in the BP, CC, and MF branches respectively (Intermediate + Distinct). As a comparison, there are 38.7%, 50.3%, and 54.1% isoform pairs that are predicted with functional differences by DIFFUSE in the BP, CC, and MF branches respectively, as shown in Figure 2.12. The

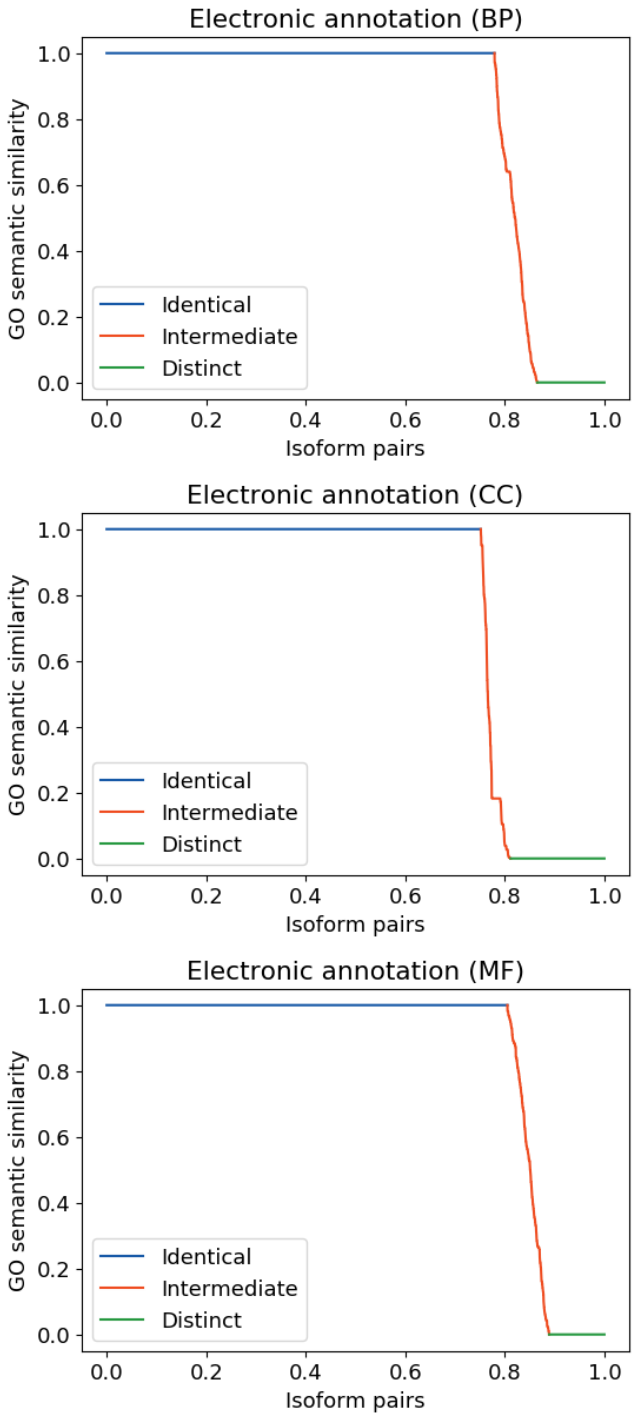


Figure 2.11: Distributions of GO similarity scores of all isoform pairs based on their electronic annotations on three branches, BP, CC, and MF respectively.

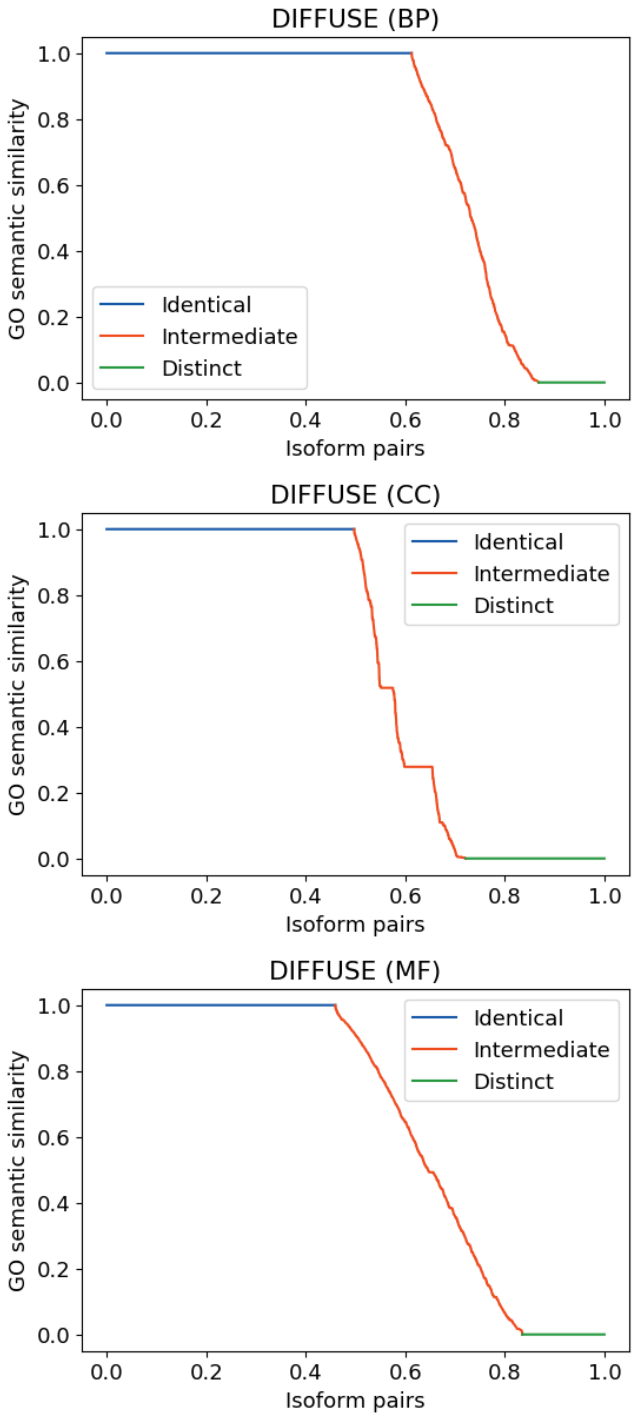


Figure 2.12: Distributions of GO similarity scores of all isoform pairs based on the predictions of DIFFUSE on three branches, BP, CC, and MF respectively.

results demonstrate that DIFFUSE is more likely to differentiate functions of isoforms comparing with the electronic annotations by sequence alignment. And the levels of functional divergence obtained by DIFFUSE are more reasonable considering our previous analytical results and those in the literature [99].

Consistency between predicted functions and important biological features

To validate the predicted functions of isoforms, we perform a similar validation to that of Section 2.4.3 to check the consistency between our predicted functions and the existence of important biological features in isoform sequences. Since some transcription factors (TFs) play important roles in controlling plant secondary metabolism [151] which are interesting in medicine studies on these plants, we check our predictions on the function of DNA binding (GO:0003677) for a set of isoforms from genes encoding TFs. Specifically, we collect 207 isoforms from genes that are predicted to encode TFs by iTAK [169], then we check the consistency between the predicted functions of these isoforms on the term GO:0003677 and the existence of DNA binding domains in their sequences. Isoforms annotated with DNA binding domains should be more likely to be predicted as having the function of DNA binding. The Jaccard index between our predictions and the existence of DNA binding domains in isoforms achieves 0.872 (Table 2.7), which can be interpreted as high consistency.

Table 2.7: Consistency between the presence or absence of DNA binding domains and the function predictions concerning GO term GO:0003677 (DNA binding) of the 207 selected isoforms.

	Have DNA binding domains	Do not have DNA binding domains
Have the function	163	10
Do not have the function	14	20
Jaccard index	0.872	

2.5.4 Conclusions

We apply DIFFUSE to predict isoform functions of five Dichocarpum species, whose gene functions and isoform functions have not been carefully studied. Our analytical results demonstrate that DIFFUSE is more likely to differentiate functions of isoforms comparing with the electronic annotations by sequence alignment. And a case study shows that our predictions on the term GO:0003677 (DNA binding) are consistent with the biological features of isoforms. We hope our predicted isoform functions can facilitate the understanding of metabolic regulations of these plant species and help the medicinal studies on them.

2.6 Discussion

As discussed in recent reviews [97, 138], the integration of various types of biological information is needed to improve isoform function prediction. In this chapter, we proposed a deep learning based method, called DIFFUSE, that integrates sequence, conserved domain and expression information into a unified predictive model. DIFFUSE greatly outperformed the existing methods in our comprehensive computatioanl experiments. However, the performance of DIFFUSE could be further improved in several aspects. First, the

co-expression networks derived from RNA-seq data are specific to different tissues and conditions, which may be correlated with specific GO terms. [99] used a search algorithm to identify the best performing subset of co-expression networks for each GO term. However, the algorithm is time-consuming. We believe that an efficient algorithm that can search for a good combination of co-expression networks specific to each GO terms could be designed and integrated into our method. Moreover, in the model training procedure, we decoupled the DNN and CRF training stages, assuming that the DNN parameters were fixed when optimizing the CRF parameters. A recent work [168] demonstrated the advantage of formulating the CRF as a layer in the DNN to enable end-to-end training with the usual back-propagation algorithm. This could further improve the performance of our model.

As demonstrated in the results (also a well-known fact), isoform functions are more correlated with protein structures than anything else. Hence, it is natural to consider incorporating protein structures in isoform function prediction [95]. However, large-scale determination of three-dimensional protein structures for isoforms accurately is computationally prohibitive. On the other hand, contact maps have been used to represent protein structures approximately and they are easier to compute (*e.g.*, [157]. We have used them in the validation of our predictions in this work and plan to explore how to incorporate them into our model in the future.

Chapter 3

FINER: Enhancing the Prediction of Tissue-Specific Functions of Isoforms by Refining Isoform Interaction Networks

3.1 Introduction

Annotating functions of gene products in complex biological systems is of fundamental importance. A large number of annotation approaches [29, 61] have been proposed and a variety of databases have been established to record functional annotation of genes [4, 41]. However, most of the existing functional annotations are at the gene level, which is coarse-grained and insufficient as a gene might have multiple products. In fact, alternative

splicing of mRNAs frequently occurs in eukaryotes, leading a single gene to often produce multiple protein isoforms [113]. The isoforms of a gene may carry different or even opposite biological functions [17]. For instance, two of the isoforms of BCL2L1 gene, BCL-xL and BCL-xS, exhibit completely opposite functions: BCL-xL inhibits programmed cell death while BCL-xS promotes it [149]. The diversity of gene products requires finer functional annotations at the isoform level instead of the gene level.

Since the experimental technologies to determine isoform functions are usually time-consuming and costly, computational approaches to predict isoform functions are highly desired. Many methods have been proposed in recent years [40, 100, 102, 130, 27, 163, 155, 98, 46]. Most of these approaches apply the multiple instance learning (MIL) technique to explore isoform features, including isoform sequence motifs, conserved domains, and expression profiles. More specifically, the MIL technique attempts to learn function-specific isoform features, *i.e.*, features that belong to at least one isoform of each gene possessing the function. The resulting function-specific features are then used to predict the functions of new (or queried) isoforms. However, these methods all suffer from the limitation that some key functional features (such as protein-protein interactions discussed below), which are proved to be effective in predicting gene functions, may not be available at the isoform level. Hence, they have prediction performance less than desirable.

Besides functional features of individual isoforms, the interactions among isoforms also form an important information source of isoform functions. The underlying ratio-

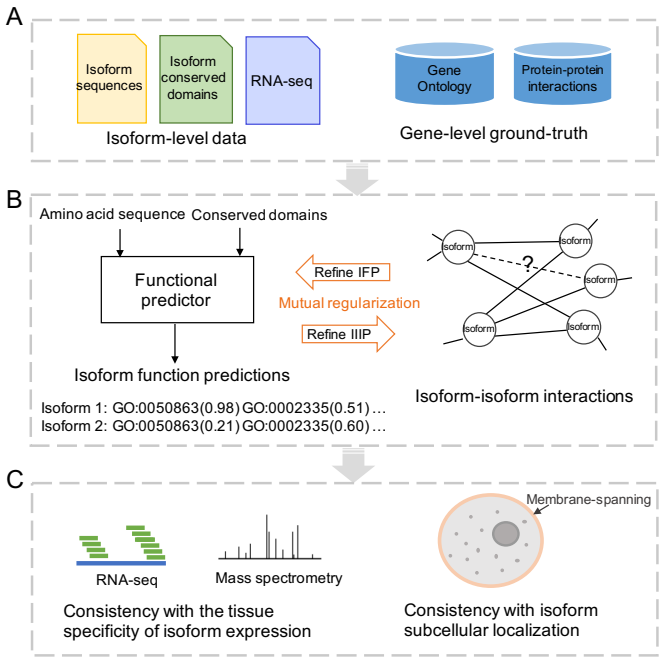


Figure 3.1: Schematic overview of the FINER workflow.

nale can be clearly demonstrated by an analogy to protein-protein interactions (PPIs): a protein usually performs specific functions through interacting with other proteins [150], thus enabling the prediction of protein functions through analyzing protein interactions. The existing PPI networks are essentially at the gene level as they exhibit only interactions among corresponding genes without providing more detailed information concerning the interaction of isoforms. In fact, the isoforms of a gene may have different interacting partners, possibly due to the difference in their interacting domains resulted from alternative splicing [140]. Thus, although PPIs have been successfully used for predicting gene functions [84, 173], they cannot be directly applied to infer fine-grained isoform functions. Recently, extensive studies have been performed to refine protein-protein interactions into isoform-isoform interactions (IIIs) [94, 147, 48, 72, 165]. Clearly, the problems of isoform

function prediction (IFP) and isoform-isoform interaction prediction (IIIP) are inherently intertwined, implying that they may not be well addressed if considered separately. Thus, how to solve the two problems jointly and exploit the reciprocal relationship between them remains an interesting challenge.

In this chapter, we present a novel approach, called FINER (*i.e.*, enhancing the Functional prediction of Isoforms via NEtwork Refinement on their interactions), that jointly solves the IFP and IIIP problems, thus allowing one problem to benefit from the other. Our approach contains three key elements as shown in Figure 3.1B: (*i*) The function prediction module predicts the functional labels of isoforms from their amino acid sequences and conserved domains. (*ii*) The interaction network refinement module identifies real interacting isoform pairs from known interacting gene pairs and denoises the existing IIIs simultaneously. (*iii*) A mutual regularizer encourages the above two modules to agree with each other, *i.e.*, isoforms with similar predicted functions will be likely connected in refined III networks and vice versa. Through the mutual regularizer, the function prediction module and interaction network refinement module exchange information and, in turn, improve their own prediction of isoform functions and interactions.

To evaluate our approach FINER, we applied it to predict tissue-specific functions and interactions of isoforms in human. Understanding tissue-specific functions of isoforms is an important but challenging task: On one hand, tissue-dependent isoform usages are pervasive across human tissues, since a gene may express various isoforms to perform different functions in different tissues [154, 50]. On the other hand, less is known about the tissue specificity of PPIs [162]. Although PPIs can be associated with tissues through the

consideration of tissue-specific expression data [8, 79], the derived interactions are perhaps less reliable, thus making the refinement on tissue-specific interactions highly desirable. In addition, diverse tissues serve as multiple sources of datasets for testing our approach. We further analyzed the relationship between our functional prediction and the subcellular localization and the tissue specificity of isoforms (Figure 3.1C). The experimental results clearly demonstrate the advantages of our approach over the state-of-the-art methods in predicting isoform functions and interactions, as well as its potential in revealing the roles of isoforms in diverse human tissues and diseases.

3.2 Materials and methods

3.2.1 Data collection

To predict isoform functions, we need gene-level functional annotation ground-truth, features of individual isoforms (including isoform sequences and conserved domains) and isoform-isoform interactions (derived from gene-level protein-protein interactions and isoform co-expression networks) (Figure 3.1A). The data used in the study are described in detail as follows.

(*i*) *Isoform sequences*: We downloaded 'Coding DNA Sequence' (CDS) of human genome (GRCh38.p13) from the NCBI Reference Sequences database (RefSeq, as of January, 2020) [120]. For each CDS, we constructed an isoform by translating it into the amino acid sequence. Two or more isoforms corresponding to the same CDS are treated as a single isoform. To ensure isoform quality, only manually-curated RefSeq records were recruited into our study. As a result, we obtained a total of 43 289 isoforms from 19

408 genes, consisting of 33 529 isoforms from 9648 multiple-isoform genes (MIGs) and 9760 single-isoform genes (SIGs).

(*ii*) *Isoform conserved domains*: For each isoform, we acquired its conserved domains by searching its amino acid sequence against the NCBI Conserved Domain Database [106].

(*iii*) *Functional annotation ground-truth of genes*: We adopted the functional terms defined by Gene Ontology (GO) [4], wherein GO terms are organized in hierarchies represented as directed acyclic graph (DAG) structures, describing functions at different levels of abstraction. For the genes used in the study, we downloaded their functional annotations from the Gene Ontology Annotation (GOA) database [65]. To ensure the annotation quality, we kept only manually-curated GO annotations and skipped electronic annotations containing the 'IEA' evidence code.

(*iv*) *Protein-protein interactions*: We used the PPI data collected by Zitnik *et al.* [173], in which, various types of physical PPIs from six reputable resources were combined [114, 125, 25, 74, 49, 108]. All the PPIs have experimental supports. The reader is referred to Zitnik *et al.* [173] for a detailed description of the data. By mapping the collected data to the genes used in our study, we acquired a total of 317 750 interactions among 19 408 genes.

(*v*) *Isoform expression profiles*: To collect expression profiles of isoforms, we first retrieved RNA-seq experiments for different types of normal human tissues from the NCBI Sequence Read Archive (SRA) database [91], where corresponding accession

numbers were obtained from the Human Protein Atlas (HPA) database [148] and the recount-brain project [122] (see Table 3.1 for a list of RNA-seq experiments). Next, we applied the tool Kallisto [14] to obtain quantified isoform expression profiles in each experiment (measured in Transcripts Per Million or TPM).

3.2.2 Construction of tissue-specific datasets

In the study, we applied FINER to predict isoform functions for 12 selected major tissues and three brain sub-tissue of human. These tissues were selected as follows. From the tissues recorded in the BRENDA Tissue Ontology [51], we first selected tissues with valid tissue-specific GO terms. Here, GO terms specifically describing cellular functions of each tissue were retrieved from Greene *et al.* [50], and only GO terms associated with at least 5 genes were recruited into our experiments (see Supplementary Table 3.2 for the lists of tissue-specific GO terms). Next, following the criterion used by Li *et al.* [100], we selected tissues associated with at least six RNA-seq experiments to guarantee the quality of co-expression networks to be constructed later. As a result, we obtained a total of 12 major tissues and three brain sub-tissues of human, which are rich enough to represent both diversity and different levels of specificity of human tissues.

Unlike isoform sequences and conserved domains that are tissue-independent, isoform expression profiles and interactions are highly tissue-specific. To construct tissue-specific PPI networks, we first selected genes with high tissue specificity, *i.e.*, the so-called

Table 3.1: Lists of RNA-seq experiments associated with different tissues, used for building tissue-specific PPIs and isoform co-expression networks.

Datasets	Tissue	# of experiments	SRA IDs
Major tissues	Adipose tissue	7	ERR579122, ERR315342, ERR315343, ERR579146, ERR315332, ERR315431, ERR315378
	Bone marrow	8	ERR315396, ERR315395, ERR315469, ERR315425, ERR315333, ERR315406, ERR315404, ERR315486
	Heart	9	ERR315413, ERR315384, ERR315389, ERR315367, ERR315356, ERR315331, ERR315435, ERR315430, ERR315328
	Intestine	30	ERR315462, ERR579129, ERR315357, ERR315348, ERR579148, ERR315403, ERR315400, ERR315484, ERR315445, ERR315442, ERR315457, ERR315461, ERR579127, ERR579151, ERR579140, ERR579147, ERR315419, ERR315381, ERR315388, ERR315364, ERR315344, ERR315423, ERR315409, ERR315408, ERR315465, ERR315366, ERR315454, ERR315345, ERR315437, ERR315481
	Lung	8	ERR315444, ERR315353, ERR315341, ERR315346, ERR315424, ERR315439, ERR315326, ERR315487
	Lymphocyte	13	ERR315371, ERR315493, ERR315393, ERR315390, ERR315373, ERR315387, ERR315475, ERR315440, ERR315441, ERR315426, ERR315471, ERR315329, ERR315488
	Placenta	7	ERR315399, ERR315375, ERR315374, ERR315377, ERR315476, ERR315336, ERR315478
	Skeletal muscle	6	ERR579130, ERR579152, ERR579149, ERR579142, ERR579143, ERR579141
	Skin	6	ERR315460, ERR315464, ERR315372, ERR315376, ERR315339, ERR315401
	Testis	8	ERR315415, ERR315492, ERR315391, ERR315446, ERR315352, ERR315351, ERR315350, ERR315456
	Thyroid gland	9	ERR315412, ERR315491, ERR315397, ERR315363, ERR315358, ERR315428, ERR315422, ERR315337, ERR315483
	Uterine endometrium	9	ERR315495, ERR315490, ERR579138, ERR315386, ERR315368, ERR579123, ERR315438, ERR315433, ERR315361
Brain sub-tissues	Cerebellum	10	SRR1222586, SRR2070736, SRR1927055, SRR1927057, SRR1927059, SRR1927061, SRR1927063, SRR1927065, SRR1927067, SRR1927069

Brain sub-tissues	Cerebral cortex	35	SRR1747144, SRR2015714, SRR1047838, SRR1047839, SRR1047840, SRR1047841, SRR1047842, SRR1047843, SRR1047844, SRR1047845, SRR1047846, SRR1047847, SRR1047848, SRR1047849, SRR1047850, SRR1047851, SRR1047852, SRR1047853, SRR1047854, SRR1047855, SRR1047856, SRR1047857, SRR1047858, SRR1047859, SRR1047860, SRR1047861, SRR1047862, SRR2557113, SRR2557114, SRR2557117, SRR2557119, SRR2557124, SRR2557125, SRR2557126, SRR835931
	Spinal cord	10	SRR1042023, SRR1042024, SRR1042025, SRR1042026, SRR1042027, SRR1042028, SRR1042029, SRR1042030, SRR1042031, SRR1042032

Table 3.2: Lists of GO terms specifically describing the cellular functions of different tissues that are included in our experiments.

Tissue	# of GO terms	GO terms
Adipose tissue	13	GO:0060612, GO:0045444, GO:0044321, GO:0044320, GO:0045600, GO:0070344, GO:0070345, GO:0050872, GO:0050873, GO:0090335, GO:0090336, GO:0045599, GO:0045598
Bone marrow	3	GO:0071863, GO:0071864, GO:0048539
Heart	131	GO:0003151, GO:0003157, GO:0045823, GO:0045822, GO:0010881, GO:0060307, GO:0003205, GO:0003206, GO:0003203, GO:0003209, GO:0060914, GO:0010002, GO:0003231, GO:0055003, GO:0055006, GO:0055007, GO:0055008, GO:0055009, GO:0061343, GO:0060347, GO:0060973, GO:0060976, GO:0060977, GO:0010459, GO:0003279, GO:0003272, GO:0003015, GO:0003160, GO:0003161, GO:0003281, GO:0003283, GO:0061311, GO:0061314, GO:0060379, GO:0060372, GO:0060373, GO:0060371, GO:0003230, GO:0001947, GO:0003307, GO:0003300, GO:0002026, GO:0002027, GO:0003207, GO:0003128, GO:0061371, GO:0048738, GO:0048739, GO:0055017, GO:0055015, GO:0055013, GO:0055012, GO:0010667, GO:0010665, GO:0060039, GO:0060038, GO:0035051, GO:0035050, GO:0010612, GO:0007507, GO:0010611, GO:0010614, GO:0003208, GO:0003348, GO:0003344, GO:0060911, GO:0003007, GO:0055119, GO:0055117, GO:0003179, GO:0003171, GO:0003170, GO:0003177, GO:0003176, GO:0003175, GO:0003174, GO:0055026, GO:0055024, GO:0055025, GO:0055022, GO:0055023, GO:0055021, GO:0007512, GO:0003186, GO:0003184, GO:0003183, GO:0003180, GO:0003181, GO:0003188, GO:0060045, GO:0060047, GO:0003228, GO:0060956, GO:0060048, GO:0003222, GO:0055010, GO:0003256, GO:0003139, GO:0090381, GO:0061337, GO:0060043, GO:0003143, GO:0003148, GO:0003149, GO:0061384, GO:0051891, GO:0051890, GO:2000136, GO:2000138, GO:0060317, GO:0003214, GO:0003215, GO:0003211, GO:0060419, GO:0060413, GO:0060412, GO:0060411, GO:0003229, GO:0003223, GO:0008016, GO:0003062, GO:0003197, GO:0003190, GO:0003198, GO:0060420, GO:0060421, GO:0010613, GO:0061117, GO:0060452, GO:0010882, GO:0010460
Intestine	6	GO:0030300, GO:0042572, GO:0042573, GO:0001523, GO:0030299, GO:0030277
Lung	7	GO:0030324, GO:0060441, GO:0060425, GO:0060428, GO:0060487, GO:0060479, GO:0048286

Lymphocyte	165	GO:0002707, GO:0002706, GO:0002709, GO:0002708, GO:2000319, GO:0051133, GO:0046651, GO:2000401, GO:2000402, GO:2000403, GO:2000404, GO:0033077, GO:0072676, GO:0072678, GO:0002293, GO:0002292, GO:0002295, GO:0002294, GO:0030889, GO:0030888, GO:0045628, GO:0045621, GO:0045622, GO:0045625, GO:0045624, GO:2000564, GO:0050870, GO:0050871, GO:0070229, GO:0070228, GO:0010820, GO:0042129, GO:0070232, GO:0070233, GO:0070230, GO:0070231, GO:0070234, GO:0002456, GO:0002455, GO:0002260, GO:0030217, GO:0002327, GO:0045620, GO:0051023, GO:0051024, GO:0050855, GO:0042093, GO:0042098, GO:0002889, GO:0016064, GO:0051249, GO:0001782, GO:2000516, GO:2000514, GO:2000515, GO:0045061, GO:0045619, GO:0002335, GO:0050869, GO:0050868, GO:0050864, GO:0050861, GO:0050860, GO:0050863, GO:0050862, GO:0002725, GO:0002724, GO:0002726, GO:0010818, GO:0030183, GO:0042130, GO:2000321, GO:2000320, GO:0033089, GO:0002381, GO:0043029, GO:2000551, GO:0045577, GO:0070227, GO:0033085, GO:0001916, GO:0001914, GO:0046006, GO:0046007, GO:0043380, GO:0048247, GO:0002903, GO:0045058, GO:0002902, GO:0002208, GO:0002204, GO:0001771, GO:0006958, GO:0043372, GO:2000562, GO:2000561, GO:0035710, GO:0045580, GO:0045581, GO:0045582, GO:0045589, GO:0002891, GO:0002890, GO:0043367, GO:0051251, GO:0051250, GO:0050670, GO:0050671, GO:0050672, GO:0031294, GO:0042113, GO:0042110, GO:0031295, GO:0050852, GO:0050853, GO:0050856, GO:0002710, GO:0002711, GO:0002712, GO:0002713, GO:0002714, GO:0072539, GO:0046639, GO:0046638, GO:0046637, GO:0046636, GO:0046635, GO:0046634, GO:0046632, GO:0046631, GO:0002664, GO:0045830, GO:0019724, GO:0046649, GO:0046642, GO:0046640, GO:0046641, GO:0010819, GO:0045191, GO:0002286, GO:0002287, GO:0002285, GO:0002360, GO:0002363, GO:0051135, GO:0030890, GO:0033081, GO:2000406, GO:0043368, GO:0043369, GO:0045190, GO:0045591, GO:0045623, GO:0030098, GO:0043371, GO:0043370, GO:0043373, GO:0042104, GO:0042100, GO:0042102, GO:0002449, GO:0045579, GO:0002315, GO:0002312, GO:0002313
Placenta	8	GO:0060711, GO:0060713, GO:0060706, GO:0060669, GO:0060674, GO:0001892, GO:0001893, GO:0001890
Skeletal muscle	49	GO:0051154, GO:0061337, GO:0010832, GO:0048742, GO:0045843, GO:0010830, GO:0030240, GO:0006941, GO:0060538, GO:0014857, GO:0055002, GO:0014888, GO:0014897, GO:0048641, GO:0060297, GO:0014855, GO:0030239, GO:0045844, GO:0003009, GO:0014722, GO:0048741, GO:0051145, GO:0007528, GO:0014902, GO:0048743, GO:0014819, GO:0007519, GO:0071688, GO:0043501, GO:0055003, GO:0030241, GO:0045988, GO:0045989, GO:0006942, GO:0016202, GO:0045214, GO:0014866, GO:0048643, GO:0014706, GO:0051155, GO:0014733, GO:0048642, GO:0051146, GO:0043403, GO:0010831, GO:0014904, GO:0010664, GO:0051153, GO:0010662

Skin	29	GO:0060088, GO:0051797, GO:0031069, GO:0045682, GO:0060113, GO:0035315, GO:0008544, GO:0042633, GO:0042635, GO:0042634, GO:0001942, GO:0060119, GO:0060117, GO:0033561, GO:0031424, GO:0042491, GO:0048730, GO:0045683, GO:0022405, GO:0045684, GO:0045606, GO:0045605, GO:0045604, GO:0043588, GO:0043589, GO:0070268, GO:0060122, GO:0002093, GO:0009913
Testis	4	GO:2000018, GO:0060008, GO:0008584, GO:2000020
Thyroid gland	3	GO:0042403, GO:0030878, GO:0006590
Uterine endometrium	3	GO:0022602, GO:0046697, GO:0042698
Cerebellum	14	GO:0021983, GO:0021707, GO:0021702, GO:0021681, GO:0021696, GO:0021697, GO:0021694, GO:0021695, GO:0021692, GO:0021549, GO:0021587, GO:0021684, GO:0021680, GO:0021683
Cerebral cortex	7	GO:0021762, GO:0021987, GO:0021799, GO:0021756, GO:0021895, GO:0021795, GO:0021801
Spinal cord	29	GO:0021515, GO:0021517, GO:0021516, GO:0021511, GO:0021510, GO:0021513, GO:0021522, GO:0034351, GO:0060251, GO:0060253, GO:0034350, GO:0022030, GO:0048485, GO:0014014, GO:0014015, GO:0014013, GO:0060019, GO:0021932, GO:0042063, GO:0014009, GO:0001774, GO:0021782, GO:0021801, GO:0045687, GO:0045686, GO:0045685, GO:0010001, GO:0060252, GO:0008347

"tissue enhanced genes" [148]. Specifically, for each of the 12 major tissues, we selected genes that have at least four-fold higher mRNA levels over the average levels in the other major tissues. For the three brain sub-tissues, we relaxed the above threshold to two-fold due to the smaller differences between sub-tissues. Next, a subnetwork was extracted from the global PPI network for each tissue as the tissue-specific PPI network, in which each edge from the global PPI network was included if at least one of the two genes connected by the edge is tissue enhanced. The underlying rationale is that tissue enhanced genes are likely to perform functions specific to the involved tissues, while their interacting partners, if not tissue enhanced, are likely ubiquitously expressed genes that perform tissue-specific functions only when interacting with tissue enhanced genes [12, 50].

We further constructed isoform co-expression networks by measuring expression correlations of isoform pairs across all RNA-seq experiments associated with the tissue, wherein only isoforms of genes appearing in the corresponding tissue-specific PPI network were considered. Expression correlation coefficients as edge weights were computed by the absolute value of the leave-one-out Pearson correlation coefficients [101], which is robust against single experimental outliers. To retain reliable co-expression edges but avoid noisy ones, we only kept the top five percent edges with the largest weights in each co-expression network.

3.2.3 The framework of FINER

The architecture of FINER consists of three key modules, namely, the function prediction module that predicts isoform functions (denoted as GO terms) for the input

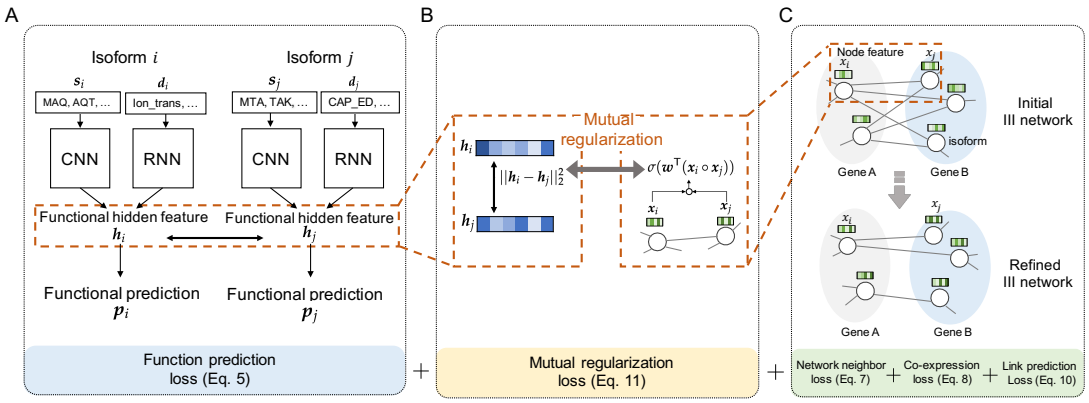


Figure 3.2: Schematic illustration of the architecture of FINER, which consists of three modules: (**A**) a neural network based function prediction module, (**C**) an III refinement module for iteratively updating III networks and (**B**) a mutual regularization module that is introduced to enable the previous two modules to exchange information with each other. That is, module B encourages isoforms with similar predicted functions to be more likely connected in the refined III networks, and vice versa. See the 'Materials and methods' section for more details.

isoforms from their sequences and domains, the III refinement module that iteratively refines the gene-level PPI network to the isoform-isoform interaction network by taking into account isoform co-expression relationship, and a mutual regularization module that enables the exchange of information between the above two modules. A schematic illustration of the architecture is provided in Figure 3.2. The details of the three modules, together with their training procedures, are described below.

3.2.4 Function prediction module and its learning objective

We constructed the function prediction module on the basis of DIFFUSE [27] with extensions (Figure 3.2A). The backbone of DIFFUSE is a deep neural network designed for predicting isoform functions based on isoform sequences and conserved domains. Specifically, the neural network contains two components: (i) A convolutional neural network

(CNN) component is used to extract sequence features of isoforms, in which the amino acid sequence of each isoform is encoded as a series of overlapping tri-grams $\boldsymbol{s}$. Each tri-gram is encoded as a continuous vector by the dense embedding layer [10]. A one-dimensional convolutional layer with multiple convolution filters is then employed to detect functional sites by scanning the encoded sequence and represent the extracted information into a sequence hidden feature vector $\boldsymbol{h}_s$. A pyramid pooling layer was designed in the CNN component to deal with isoform sequences of different lengths. (ii) The other component is a recurrent neural network (RNN) with long short-term memory (LSTM) units [60]. In the RNN component, each type of conserved domain is represented as a unique token. Domains of each isoform are ordered as a sequence of tokens $\boldsymbol{d}$, which are encoded by the same dense embedding technique and then input to the LSTM units successively. Content of tokens with their order information for each isoform are thus captured and again represented as a domain hidden feature vector $\boldsymbol{h}_d$.

The two types of hidden features, $\boldsymbol{h}_s$ and $\boldsymbol{h}_d$, are concatenated and then fused as a unified functional feature vector $\boldsymbol{h}$ through a fully connected layer. The feature extraction and fusion process are formally defined as:

$$\boldsymbol{h} = Dense([\boldsymbol{h}_s, \boldsymbol{h}_d]) = Dense([CNN(\boldsymbol{s}), RNN(\boldsymbol{d})]), \tag{3.1}$$

where $Dense(\cdot)$ denotes the fully connected layer and $[\cdot, \cdot]$ denotes the concatenation of two vectors.

Unlike DIFFUSE, which produces a binary prediction on each individual GO term, FINER produces a multi-label prediction on all the GO terms specific to a given tissue

simultaneously, thus making the entire training process more efficient and allowing common knowledge to be shared across all GO terms. Specifically, we used a fully connected layer to map a functional feature vector $\boldsymbol{h}$ to an output vector $\boldsymbol{o}$:

$$\boldsymbol{o} = Dense(\boldsymbol{h}). \tag{3.2}$$

Here, the output vector $\boldsymbol{o}$ has T dimensions, where T denotes the number of GO terms specific to a tissue. The sigmoid function is applied on each dimension of the output to normalize the prediction on each GO term to a score in the range [0, 1], indicating how likely the input isoform performs the corresponding function.

Because of the hierarchical nature of GO, an isoform is automatically labeled with a GO term if any of its child terms are labeled on the isoform. To ensure consistent prediction on all GO terms, we designed a hierarchical prediction layer as done in Kulmanov *et al.* [84]. For each term in the set of T GO terms, we created a binary mask vector, denoted as $\boldsymbol{c}_t$ (where $t = 1, 2, ..., T$), wherein the bits corresponding to the GO term and its children are set as 1. The maximum score from the element-wise product of the output vector and the mask vector is set as the GO term's prediction, which is formally denoted as:

$$a_t = max(\boldsymbol{c}_t \circ \boldsymbol{o}) \quad \text{for} \quad t = 1, 2, ..., T. \tag{3.3}$$

Finally, the prediction results on all T terms are merged as the functional prediction of the input isoform $\boldsymbol{p} = Hierarchical(\boldsymbol{o}) = (a_1, a_2, ..., a_T)$.

To overcome the difficulty of lack of ground-truth isoform function annotations,

we applied the multiple instance learning (MIL) technique, following the previous work on isoform function prediction in [27, 40]. Specifically, each gene is treated as a bag and the isoforms of a gene are treated as the instances of the bag. For a given function, positive bags refers to genes associated with the function. Clearly, a positive bag should contain at least one positive instance but may also have some negative instances, while a negative bag should contain no positive instances. We initialize all instances of positive bags with positive labels, and the others with negative labels. Given an isoform i and its initial label on GO term t, we can define the following "binary cross entropy loss":

$$l_{i,t} = -(y_{i,t} \log(p_{i,t}) + (1 - y_{i,t}) \log(1 - p_{i,t})), \tag{3.4}$$

where $y_{i,t}$ is a one-hot indicator for the label of isoform i on GO term t, and $p_{i,t}$ is the corresponding prediction score. To characterize the above bag instance relationship, we weighted each "binary cross entropy loss" by the corresponding prediction score, so that significant punishment would be applied on large prediction scores with negative labels but not on small prediction scores with positive labels.

Given a set of K isoforms, the learning objective for the function prediction module is to minimize the following "function prediction loss" defined by the following weighted binary cross entropy [59]:

$$L_{fp} = -\sum_{i}^{K} \sum_{t}^{T} \hat{p}_{i,t} l_{i,t}, \tag{3.5}$$

where $\hat{p}_{i,t}$ is a constant assigned by $p_{i,t}$ to avoid direct minimization of the prediction score.

The isoform labels are recalculated after each training iteration under the MIL constraints, which will be described in more detail in the sections below.

3.2.5 Isoform-isoform interaction refinement module and its learning objective

For a given tissue, we iteratively refine its isoform-isoform interaction network initialized as the tissue-specific PPI network (Figure 3.2C). The III network contains the isoforms of genes that appear in the tissue-specific PPI network. Initially, we connect isoforms if and only if their genes have interactions in the tissue-specific PPI network. We formally define an III network as a undirected graph $G_{III} = (\boldsymbol{V}, \boldsymbol{E})$, in which isoforms are represented as a set of nodes $\boldsymbol{V} = \{v_i\}_{i=1}^{|\boldsymbol{V}|}$, and their interactions are represented as edges $\boldsymbol{E}$ between nodes. Our goal is to produce a refined III network $G'_{III} = (\boldsymbol{V}, \boldsymbol{E}')$ on the same set of nodes but with a new set of edges $\boldsymbol{E}'$, reflecting real interactions among isoforms.

We refine the III network according to isoforms' neighbors in the current III network and the isoform co-expression relationship. For each isoform $v_i \in \boldsymbol{V}$, these two types of information are represented as a node feature vector $\boldsymbol{x}_i$. The details of this representation are described as follows.

(i) *Isoform neighborhood*: The neighborhood of node v_i is defined as a set of nodes visited by a series of random walks starting from v_i, denoted as $\boldsymbol{N}_i$ [52]. The isoforms with similar neighborhoods should share similar node feature vectors as they have similar interacting partners. To characterize this relationship, we specify the following objective function: For a node v_i, the objective seeks to correctly predict $\boldsymbol{N}_i$ from their

node feature vectors. As the neighborhood relationship is not certainly bidirectional based on its definition, we use a context vector $\boldsymbol{x}'_i$ to represent each node when it is treated as the prediction target. Thus, predicting the neighborhood is modeled as the conditional likelihood given by a softmax unit parameterized by the products of node vectors. The objective is to minimize the following negative log likelihood through the updating of node feature and context vectors:

$$\begin{aligned} L_{nb} &= -\sum_{i=1}^{|\boldsymbol{V}|}\sum_{j\in\boldsymbol{N}_i}\log p(v_j|v_i) \\ &= -\sum_{i=1}^{|\boldsymbol{V}|}\sum_{j\in\boldsymbol{N}_i}\log\frac{\exp(\boldsymbol{x}'^{\mathsf{T}}_j\boldsymbol{x}_i)}{\sum_{v_k\in\boldsymbol{V}}\exp(\boldsymbol{x}'^{\mathsf{T}}_k\boldsymbol{x}_i)}. \end{aligned} \tag{3.6}$$

As the computation of the full softmax is expensive, we approximate the objective using negative sampling [109]. For each node v_j in the neighborhood of node v_i, we sample a set of non-neighborhood nodes, $\boldsymbol{R}_{ij} \subseteq \boldsymbol{V} - \boldsymbol{N}_i$. Thus, the task becomes to distinguish node v_j from nodes in $\boldsymbol{R}_{ij}$. Then, the above objective can be formulated as the following "network neighborhood loss":

$$L_{nb} = -\sum_{i=1}^{|\boldsymbol{V}|}\sum_{j\in\boldsymbol{N}_i}(\log\sigma(\boldsymbol{x}'^{\mathsf{T}}_j\boldsymbol{x}_i) - \sum_{k\in\boldsymbol{R}_{ij}}\log\sigma(\boldsymbol{x}'^{\mathsf{T}}_k\boldsymbol{x}_i)). \tag{3.7}$$

(*ii*) *Co-expression relationship:* Co-expressed isoforms are usually those involved in common biological processes and thus may have common interacting partners [126]. As introduced in the 'Construction of tissue-specific datasets' section, the tissue-specific co-expression network $G_{EXP} = (\boldsymbol{V}, \boldsymbol{R})$ is constructed on the same set of nodes $\boldsymbol{V}$ as the tissue-specific III network, with a set of weighted edges $\boldsymbol{R}$ where the weight of edge

r_{ij} between nodes (v_i, v_j) reflects the expression correlation between two corresponding isoforms. Then, the following "co-expression loss" introduces a regularization for node feature vectors under the squared euclidean distance, weighted by the edge weights of the co-expression network, which encourages similar node feature vectors to be shared by co-expressed isoforms:

$$L_{coe} = \sum_{i=1}^{|\boldsymbol{V}|} \sum_{j=1}^{|\boldsymbol{V}|} r_{ij} ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2. \tag{3.8}$$

To predict interactions from node features, we built a binary classifier. Specifically, for a pair of nodes (v_i, v_j), we first combine their feature vectors using the element-wise multiplication, *i.e.*, $\boldsymbol{x}_i \circ \boldsymbol{x}_j$, which is a commonly used operation in modeling the symmetric relations from vector representations [28, 70, 57]. Then, the sigmoid function is applied on the weighted summation of the combined representation's dimensions, which outputs a score in the range [0, 1], indicating how likely the interaction happens between the two corresponding isoforms:

$$z_{ij} = \sigma(\boldsymbol{w}^{\mathsf{T}}(\boldsymbol{x}_i \circ \boldsymbol{x}_j)), \tag{3.9}$$

where $\boldsymbol{w}$ is a vector of trainable parameters which learns to weight the contribution of different dimensions of node feature vectors. To train the weight vector, we treat links in the current III network as labels and apply the same weighted cross entropy introduced in Equation 3.5 as the "link prediction loss":

$$L_{lp} = -\sum_{i=1}^{|\boldsymbol{V}|}\sum_{j=1}^{|\boldsymbol{V}|}\hat{z}_{ij}(e_{ij}\log(z_{ij}) + (1-e_{ij})\log(1-z_{ij})), \tag{3.10}$$

where e_{ij} is a binary indicator for the link between node i and node j in the current III network and $\hat{z}_{ij}$ is a constant assigned by z_{ij}. Feature vectors are also adjusted to adapt to the weight vector based on the objective function, which facilitates the training process.

3.2.6 Mutual regularization and the joint learning objective

The key idea, which forms the cornerstone of this work, is to establish the connection between the two tasks, IFP and IIIP. Inspired by the graph regularizer [20] recently proposed for training neural networks with the help of static graphs, we propose a mutual regularizer for both modules (Figure 3.2B) that uses edges in the current III network to regularize the learning process of the functional predictor and also encourages dynamic adjustments in the III network consistent with the prediction results made by the functional module:

$$L_{mut} = -\sum_{i=1}^{|\boldsymbol{V}|}\sum_{j=1}^{|\boldsymbol{V}|}z_{ij}(m - ||\boldsymbol{h}_i - \boldsymbol{h}_j||_2^2), \tag{3.11}$$

where $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$ are functional feature vectors of the corresponding isoforms of node i and j, defined in Equation 3.1, and m is a predefined margin. Intuitively, this "mutual regularization loss" encourages the functional predictor to learn similar functional feature vectors for two isoforms if they are connected in the current III network. On the other hand, if two isoforms have similar functional feature vectors, *i.e.*, the squared euclidean distance over them is smaller than the predefined margin m, a larger prediction score of

Algorithm 2 Learning algorithm of FINER

Initialization: Isoform sequences, $\boldsymbol{s}$; Conserved domains, $\boldsymbol{d}$; Initial functional labels, $\boldsymbol{y}$; Initial III network, $G_{III} = (\boldsymbol{V}, \boldsymbol{E})$; Co-expression network, $G_{EXP} = (\boldsymbol{V}, \boldsymbol{R})$.

Output: Functional predictor with parameters Θ; Refined III network $G'_{III} = (\boldsymbol{V}, \boldsymbol{E}')$.

Initialize parameters $\boldsymbol{\Theta}$, $\boldsymbol{x}$, $\boldsymbol{x}'$ and $\boldsymbol{w}$, $G'_{III} = G_{III}$
while not converged **do**
    Sample batches for functional predictor by $\boldsymbol{E}'$
    for each batch **do**
        Update $\boldsymbol{\Theta}$ by Equation 3.13
    end for
    for each isoform i **do**
        Make inference on $\boldsymbol{h}_i$ and $\boldsymbol{p}_i$
    end for
    Update $\boldsymbol{y}$ under the MIL constraints
    Sample batches for III refinement module by Node2vecWalk(G'_{III})
    for each batch **do**
        Update $\boldsymbol{x}$, $\boldsymbol{x}'$, and $\boldsymbol{w}$ by Equation 3.14
    end for
    for each node pair (v_i, v_j) **do**
        Make inference on z_{ij}
    end for
    Update $\boldsymbol{E}'$
end while

their interaction is encouraged.

To sum up, the joint objective of FINER is to minimize the following loss function:

$$L = \lambda_1 L_{fp} + \lambda_2 L_{mut} + \lambda_3 L_{nb} + \lambda_4 L_{coe} + \lambda_5 L_{lp}, \tag{3.12}$$

where λ_1, λ_2, λ_3, λ_4, and λ_5 are the balancing hyper-parameters.

3.2.7 Training procedure of FINER

To learn general functional knowledge from sequences and domains, we first pre-train the function prediction module using a large number of proteins retrieved from the

SwissProt database [11] as done in DIFFUSE [27]. In this study, we collected 98 400 eukaryotic (other than human) protein sequences with GO annotations from the SwissProt database. Conserved domain data were retrieved accordingly using the same method described before. The "binary cross entropy loss" defined in Equation 3.4 is used to pretrain the functional predictor.

Next, the function prediction module and the III module are alternately trained with the isoform data, until convergence. The pseudocode for the learning algorithm is given in Algorithm 2, and its basic ideas are sketched below.

(i) *Training the function prediction module*: In the functional module training phase, parameters of the functional predictor $\boldsymbol{\Theta}$ are updated by minimizing the weighted summation of two components in the loss function with the stochastic gradient descent method:

$$\min_{\boldsymbol{\Theta}} \lambda_1 L_{fp} + \lambda_2 L_{mut}. \tag{3.13}$$

To speedup learning, isoforms connected in the current III network are sampled into the same training batch. After each training phase of the functional module, the inference is performed for all isoforms on their functional feature vectors $\boldsymbol{h}$ and functional predictions $\boldsymbol{p}$. Under the MIL setting, for each GO term, the labels of all instances in positive bags are updated according to the following criteria: (i) Instances with prediction scores above the predefined threshold are assigned with positive labels, while the others are assigned with negative labels. (ii) For each positive bag, if all its instances are assigned with negative labels, we select the instance with the largest positive prediction score in the bag as positive. The updated labels are used for training

in subsequent iterations.

(*ii*) *Training the III network refinement module*: In the III module training phase, node vectors and weight parameter $\boldsymbol{w}$ are updated by minimizing the weighted summation of four components in the loss function with the stochastic gradient descent method:

$$\min_{\boldsymbol{x},\boldsymbol{x}',\boldsymbol{w}} \lambda_2 L_{mut} + \lambda_3 L_{nb} + \lambda_4 L_{coe} + \lambda_5 L_{lp}, \tag{3.14}$$

After each training phase of the III module, the inference is performed for each node pair (v_i, v_j) on the link prediction z_{ij}, based on which, edges in the current III network are updated to obtain a refined III network.

Due to the noisy nature of tissue-specific PPIs, we would like to denoise the existing interactions while discovering *de novo* interactions. Therefore, unlike the label update procedure in the functional module, edge update here does not consider bag-instance constraints. The following criteria are considered when updating edges instead: (*i*) In the refined III network, edges are set between nodes if the corresponding link prediction scores are above the predefined threshold. (*ii*) To guarantee the inclusion of interaction information for each isoform, the top 10 edges with the largest link prediction scores associated with each node are also included in the refined III network. Edges in the refined network are then used for regularizing the functional module in subsequent iterations.

3.3 RESULTS

3.3.1 Prediction of tissue-specific isoform functions

We applied FINER to predict tissue-specific functions of isoforms on the human tissue datasets, including 12 major tissues and three brain sub-tissues. The prediction procedure, together with the calculation of prediction accuracy, are described below:

(*i*) *Dataset partition*: For each tissue, we randomly partition its isoforms into training, validation, and test sets with the proportions of 70%, 10%, and 20%, respectively. Hyper-parameters of the models are manually tuned based on model performance on the validation data (see Table 3.3 for the calibrated hyper-parameter values). The validation data are finally merged with the training data to train a model for performance evaluation on the test data. To avoid potential information leak (*i.e.*, different components of the partition share isoforms with very similar sequences and thus similar functions), we first require that isoforms of the same gene are partitioned into the same set. In addition, since the function prediction module is pretrained with the SwissProt protein sequences from different eukaryotes and there are closely related paralogous genes in the human genome, we consider clusters of orthologous groups (COGs) defined in the EggNOG database up to the level of eukaryotes [63] (note that such COGs also include many paralogous genes) to prevent closely related homologous genes from being split among different sets. In other words, all genes of the same COG are required to be partitioned together. In addition, all (non-human) SwissProt proteins belonging to COGs that contain (human) genes in the test set are excluded from the pretraining phase.

Table 3.3: The calibrated hyper-parameter values of FINER models.

Hyper-parameter	Value
Dimensionality of the amino acid tri-gram embedding	8
Dimensionality of the conserved domain token embedding	8
Convolutional layer kernel number	64
Convolutional kernel size	64
Dimensionality of the sequence hidden feature vector $\boldsymbol{h}_s$	128
Dimensionality of the domain hidden feature vector $\boldsymbol{h}_d$	128
Dimensionality of the functional feature vector $\boldsymbol{h}$	128
Dimensionality of the node feature vector $\boldsymbol{x}$	16
Return parameter of random walk in the Node2Vec algorithm [52]	4
In-out parameter of random walk in the Node2Vec algorithm	1
Margin in the mutual regularization loss m	0.1
Optimizers of both modules	Adam [77]

The balancing of the hyper-parameters of the loss function and learning rates were tuned specifically for the model of each tissue.

(*ii*) *Prediction accuracy evaluation*: As the ground-truth of isoform functions is generally unavailable, we adopt the widely-used alternative evaluation strategy at the gene level [27, 40, 100, 130], with the rationale that if the functions of isoforms are correctly predicted, their gene functions should be correctly predicted automatically. Hence, for each GO term, a prediction score for each gene is generated by taking the maximum prediction score among its isoforms, and the performance is measured by comparing the gene-level prediction with the ground-truth. Both the area under the receiver operating characteristics curve (AUC) and the area under the precision-recall curve (AUPRC) are used to evaluate the performance for each GO term. To make comparisons across different datasets fairly, we unify the AUPRC baseline as 0.1 for all GO terms as done in [130, 27].

To evaluate the effect of III refinement on functional prediction, we compare the

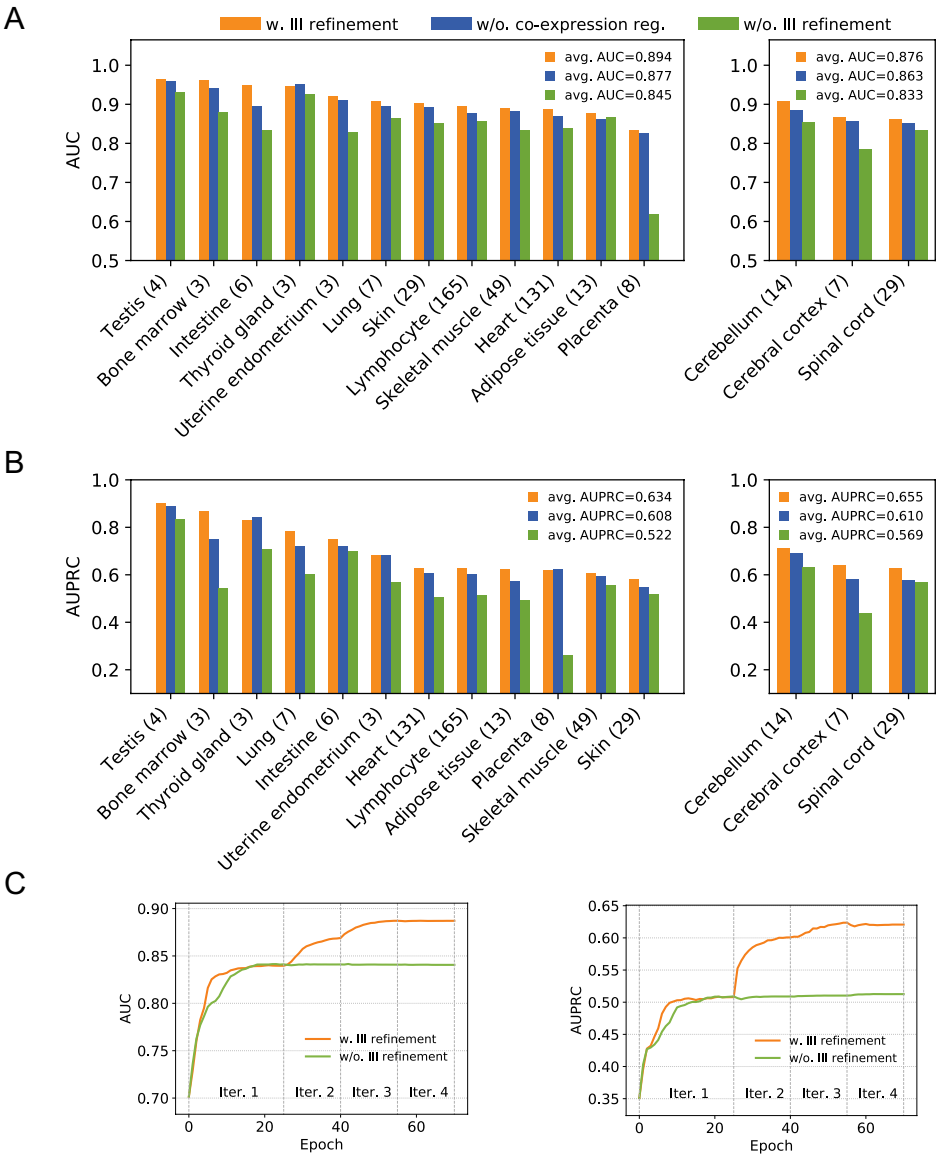


Figure 3.3: (**A**) Comparison of functional prediction performance measured by the average AUC over GO terms on each dataset, between FINER (orange), FINER without co-expression regularization (blue), and FINER without III refinement (green). The number of GO terms associated with each tissue is noted after the name of the tissue. (**B**) Comparison of functional prediction performance measured by the average AUPRC. (**C**) Learning curves of the function prediction module with (orange) and without (green) III refinement on the Heart tissue dataset in terms of both AUC and AUPRC.

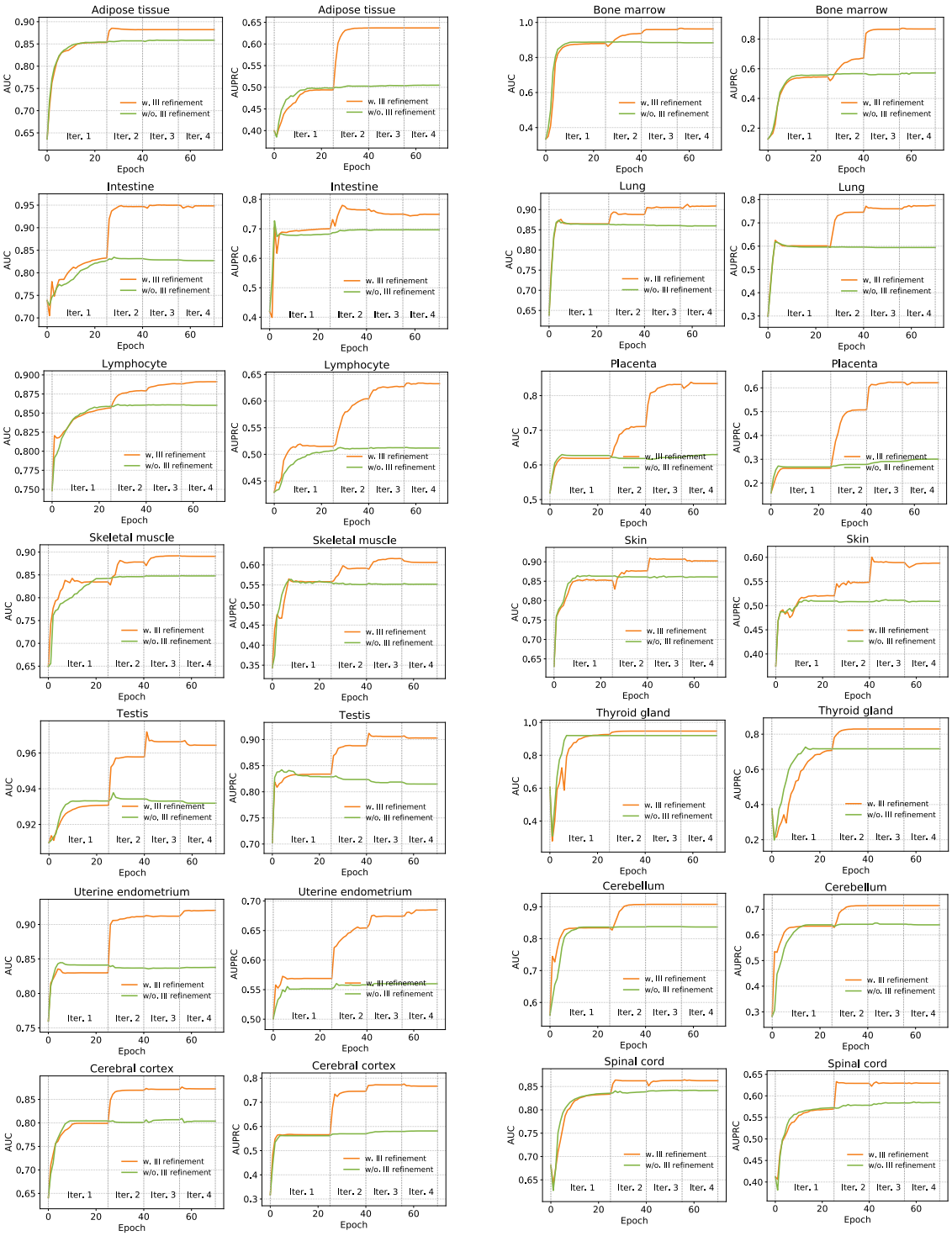


Figure 3.4: Learning curves of FINER on all tissue-specific datasets other than Heart.

performance of FINER with that of FINER without III refinement as well as with FINER without co-expression regularization in the III refinement module. Figure 3.3 summarizes the average AUC and AUPRC values over all the GO terms in each tissue. On average, FINER achieves improvements of 5.80% and 21.5% over FINER without III refinement in terms of AUC and AUPRC, respectively, on the major tissue datasets, as well as improvements of 5.16% and 15.1% in terms of AUC and AUPRC, respectively, on the brain datasets. In addition, FINER achieves improvements of 1.94% and 4.28% over FINER without co-expression regularization in terms of AUC and AUPRC, respectively, on the major tissue datasets, as well as improvements of 1.51% and 7.37% in terms of AUC and AUPRC, respectively, on the brain datasets. The learning curves of the function prediction module in Figure 3.3C clearly demonstrate that the performance of the module benefited from the refinement of III networks, *i.e.*, the performance of the function prediction module clearly gets better after each III network update, until convergence (see Figure 3.4 for learning curves on all the other tissues).

A concrete example is shown in Figure 3.5. Isoform NM_000660 is the single isoform of gene TGFB1. According to GO annotations, TGFB1 is labeled as having the heart-specific function of cardiac chamber development (GO:0003205). Without applying III refinement, NM_000660 is predicted to have the function of GO:0003205 with a score of only 0.571, which is just at the boundary between having or not having the function. Meanwhile, most of its interacting partners in the initial III network are predicted as not having the function. In contrast, when applying III refinement, NM_000660 is predicted to be interacting with isoforms that are predicted as having the function, and NM_000660

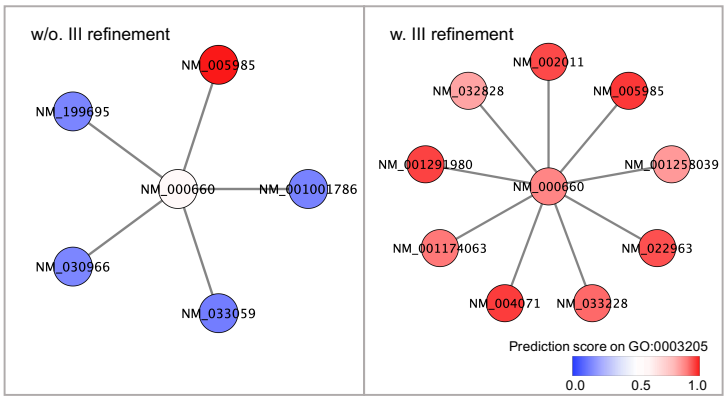


Figure 3.5: Illustration of the interactions and functional prediction scores on the term GO:0003205 of isoform NM_000660 in both the initial III network and the refined III network. Red nodes represent isoforms predicted as having the function, while blue nodes represent isoforms predicted as not having the function.

itself is predicted as having the function with a high score of 0.870.

3.3.2 Comparison with the existing methods

We further make comprehensive comparisons between the functional prediction performance of FINER and that of several state-of-the-art methods with different objectives, including two recent isoform function prediction methods DIFFUSE [27] and DisoFun [155], a tissue-specific protein function prediction method OhmNet [173] and a general biological network refinement method NE [152]. Note that three isoform function prediction methods, DisoFun, ISOGO [46] and IsoResolve [98] have been published in the literature after DIFFUSE. Although these methods have not been compared with DIFFUSE directly on the same dataset, their reported overall performance all seem to be worse than that of DIFFUSE. Among the methods, DisoFun adopted a more strict evaluation metric in its performance evaluation. Moreover, it also considers PPI information similar to FINER. We

Table 3.4: Comparison of functional prediction performance between FINER and some existing state-of-the-art methods.

Method	Major Tissue Datasets		Brain Datasets	
	AUC (SD)	AUPRC (SD)	AUC (SD)	AUPRC (SD)
OhmNet	0.751 (0.099)	0.431 (0.196)	0.743 (0.123)	0.423 (0.223)
DisoFun	0.805 (0.188)	0.460 (0.236)	0.770 (0.214)	0.419 (0.242)
DIFFUSE	0.836 (0.134)	0.568 (0.208)	0.822 (0.190)	0.541 (0.256)
$\text{FINER}_{\text{fixed}}$+RAW	0.845 (0.102)	0.522 (0.185)	0.833 (0.149)	0.569 (0.254)
$\text{FINER}_{\text{fixed}}$+NE	0.859 (0.104)	0.540 (0.178)	0.841 (0.131)	0.577 (0.242)
FINER	**0.894** (0.080)	**0.634** (0.158)	**0.876** (0.115)	**0.655** (0.234)

therefore choose to include DisoFun in the comparison here.

DisoFun predicts isoform functions using a matrix factorization approach based on isoform expression profiles, where PPIs are used to perform a gene-level regularization. OhmNet first learns protein embeddings from different tissue-specific PPI networks, taking into consideration the dependence between tissues. Independent function classifiers are then trained to predict tissue-specific protein functions at the gene level. NE has been successfully used to denoise tissue-specific PPIs in the literature [152]. In the study, we apply NE to denoise initial III networks in our datasets. To compare the effect of their results on enhancing isoform function prediction with that of our refined III networks, we provide FINER with NE's denoised III networks and keep them fixed throughout functional prediction. We denote this model as $\text{FINER}_{\text{fixed}}$+NE. To make the comparisons more clear, we include the performance of FINER without III refinement here, denoted as $\text{FINER}_{\text{fixed}}$+RAW. As shown in Table 3.4, FINER improved over the best performance of the three isoform/protein function prediction methods (*i.e.*, OhmNet, DisoFun, and DIFFUSE) on both the major tissue and brain datasets by 6.94% and 6.57%, respectively, in terms of average AUC, as well as 11.62% and 21.1%, respectively, in terms of average AUPRC. We observe that the

standard deviations (SDs) in FINER's performance across different tissues are generally smaller than the other methods. Moreover, a comparison among FINER, $\text{FINER}_{\text{fixed}}$+NE and $\text{FINER}_{\text{fixed}}$+RAW demonstrates that FINER acquires larger performance gains from our iteratively refined III networks than the denoised III networks of NE, even though they are still better than the initial networks.

Due to the lack of tissue-specific interaction ground-truth, we measure the consistency between our refined III networks and the results of a state-of-the-art III prediction method. TENSION [72] is compared here as it is the most recent tissue-specific III prediction method. For each tissue, a core subnetwork is extracted from the predicted III network of each method, which is induced by the set of isoforms whose genes are associated with the tissue-specific functions. The Jaccard index is used to measure the similarity between the subnetworks generated by the two methods for each tissue. As shown in Table 3.5, the average of Jaccard indexes across all tissues is 0.332, and they are all significantly larger than the expected ones if two networks are randomly (and independently) generated with the same sets of nodes and number of edges as in the networks predicted by FINER and TENSION (under the column E[Jaccard index] in Table 3.5). The moderate similarity between the core parts of III networks on most of the tissues suggests that the III predictions made by the two methods are perhaps informative.

Table 3.5: Similarity between the core subnetworks of tissue-specific III networks predicted by FINER and those predicted by TENSION in each tissue, where each subnetwork is induced by the set of isoform nodes from genes associated with the corresponding tissue-specific functions. P-values from Fisher's exact tests are used to demonstrate the significance of the difference between the Jaccard indexes calculated and the expected ones if two networks are randomly (and independently) generated with the same sets of nodes and number of edges as in the networks predicted by FINER and TENSION.

Tissue	Jaccard index	E[Jaccard index]	P-value	# of isoforms in subnetwork
Uterine endometrium	0.594	0.102	1.40E-26	24
Lung	0.553	0.066	6.49E-79	48
Bone marrow	0.517	0.077	1.96E-71	47
Testis	0.500	0.039	3.01E-48	45
Cerebral cortex	0.427	0.009	2.57E-170	147
Adipose tissue	0.333	0.031	1.14E-220	138
Cerebellum	0.317	0.035	1.19E-27	48
Thyroid gland	0.311	0.147	1.62E-5	21
Spinal cord	0.308	0.008	2.24E-171	191
Intestine	0.297	0.032	2.91E-50	70
Skin	0.242	0.011	0.0	337
Heart	0.194	0.008	0.0	729
Placenta	0.154	0.050	7.14E-38	111
Skeletal muscle	0.141	0.014	0.0	795
Lymphocyte	0.086	0.009	0.0	1106
Average	0.332			

3.3.3 Consistency between the predicted functions of isoforms and their tissue specificity

We validate our isoform-level predictions by investigating their consistency with the tissue specificity of isoforms. It is well-known that the expression of genes is usually tissue-specific. Previous studies have shown that in a certain tissue, the highly-expressed genes are usually associated with functions specific to the tissue [45]. For example, genes with elevated expression in skin are associated with functions related to the barrier function, skin pigmentation, and hair development, while genes elevated in liver are associated with metabolic processes and glycogen storage [148]. As isoforms are actual function carriers,

we expect that this relationship also remains true at the isoform level. That is, the set of isoforms elevated in a tissue should be enriched with the corresponding tissue-specific functions. Thus, we quantify the expression specificity of each isoform in a given tissue by the fold change of its mRNA level in the tissue over the average level in other tissues. For each tissue, a set of "tissue enhanced isoforms" are selected from the test set based on the "tissue enhanced" criteria same as those in the "Materials and methods" section. To generate functional annotations of isoforms on each GO term, we binarize the corresponding prediction scores by applying the threshold that optimizes the F1 score with respect to the gene-level ground-truth. Then, Fisher's exact test is performed to test each tissue-specific GO term's enrichment in the set of tissue enhanced isoforms. The multiple testing correction with false discovery rate (FDR) controlling is applied to the P values. Figure 3.6A shows the fractions of GO terms that are enriched in the tissue enhanced isoform sets of each tissue. Enrichment (*i.e.*, $P(\text{corrected}) \leq 0.05$) is found in 91.4% (385 out of 421) of the GO terms on the major tissue datasets and 84.0% (42 out of 50) on the brain datasets. These results confirm that the consistency between (predicted) functions and tissue-specific expressions remains at the isoform-level.

We further investigate whether our functional predictions differentiate tissue enhanced isoforms from non-tissue enhanced ones in functional genes. Specifically, for each tissue-specific GO term, we consider only the genes that are associated with the term, and divide isoforms of these genes into two sets, namely, a set of "tissue enhanced isoforms" and a set of "non-tissue enhanced isoforms" based on the same criteria as before. Note that either the tissue enhanced isoform set or the non-tissue enhanced isoform set could

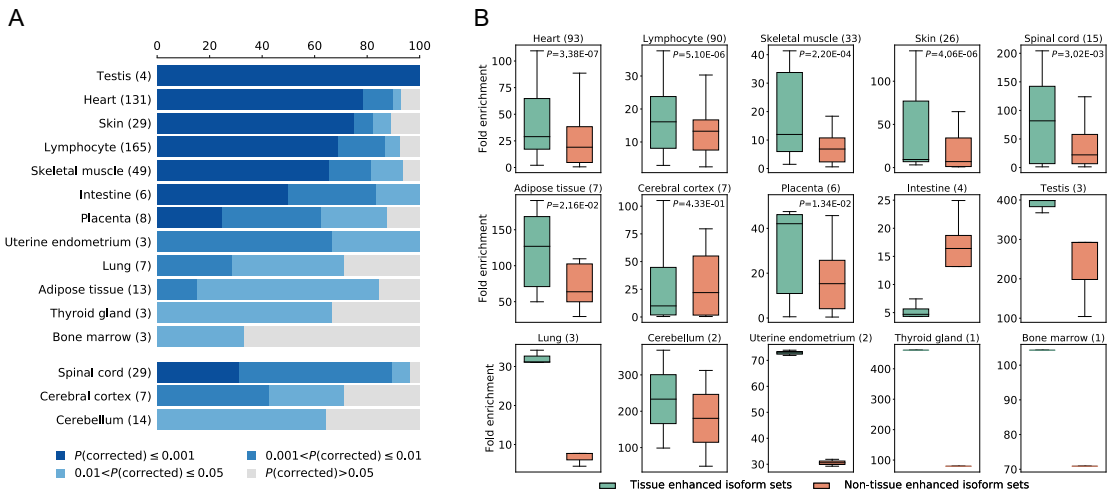


Figure 3.6: (**A**) The fractions of GO terms that are enriched in the set of tissue enhanced isoforms of each tissue. Different levels of enrichment are colored differently. (**B**) Fold enrichment of GO terms in sets of tissue enhanced isoforms (green) and sets of non-tissue enhanced isoforms (orange), where for each GO term, only isoforms of genes associated with the term are considered. The one-sided Wilcoxon test is performed on the results of each tissue with at least 5 GO terms (numbers of GO terms are noted in the titles) included in this analysis to test the significance of the difference in GO enrichment between tissue enhanced and non-tissue enhanced isoform sets.

be empty for a GO term. If this happens, the corresponding GO term is then ignored in the analysis. We compare the fold enrichment of a GO term in both sets. The higher the fold is, the more significant enrichment is found in a set. As shown in Figure 3.6B, the one-sided Wilcoxon test exhibits significant differences of GO enrichment between such two sets of tissue enhanced and non-tissue enhanced isoforms. The results suggest that FINER was able to identify tissue-enhanced isoforms from genes with tissue-specific functions and assign these functions to such isoforms.

3.3.4 Consistency between the highest connected isoforms and isoform protein-level expression

Previous studies have found that in a given tissue, the isoform of each gene with the most interacting partners usually shows a higher expression level than other isoforms of the same gene, and is more likely to play functional roles in the tissue. This observation is consistent across a variety of tissues at both the transcript level and the protein level [94, 96]. To check the validity of this observation in our refined III networks, we identify the highest connected isoform (HCI) of each MIG in different tissues, where the HCI is defined as the isoform of each MIG that has the highest degree in the III network of a given tissue. An independent dataset for tissue-specific protein-level expression of isoforms was then collected from Wang *et al.* [153]. For each tissue, the dataset lists a set of isoforms that are detected at the protein level by mass spectrometry. Due to its low sequence coverage, most genes have only one detected isoform in each tissue, which usually is the highest expressed isoform at the protein level. Ideally, the HCIs of each MIG in different tissues should be the isoforms that have protein expression evidence in the corresponding tissues. As shown in Table 3.6, the numbers of MIGs whose HCIs in a tissue are detected at the protein level, denoted as N_{FINER}, are significantly higher than the expected numbers of MIGs (N_{chance}) if their HCIs in the tissues are randomly chosen and detected at the protein level. We repeat the same experiment on the III predictions of TENSION. The numbers of MIGs with HCIs in the III networks predicted by TENSION that are detected at the protein level, denoted as N_{TENSION}, are not as significantly different from N_{chance} as the ones of FINER. These results confirm the above observation in our refined III networks.

Table 3.6: The numbers of MIGs whose HCIs are detected at the protein level in each tissue. Comparisons are made between HCIs of III networks predicted by FINER and those predicted by TENSION.

Tissue	# of MIGs	N_{chance}	N_{FINER} (P-value)	N_{TENSION} (P-value)
Heart	316	121	217 (2.35E-14)	177 (9.12E-14)
Lung	272	103	150 (1.72E-13)	121 (1.91E-03)
Lymphocyte	399	157	252 (4.33E-14)	196 (5.04E-06)
Placenta	599	228	345 (5.95E-14)	236 (1.29E-01)
Testis	754	287	421 (5.80E-14)	313 (5.40E-03)
Thyroid gland	466	180	287 (3.46E-14)	196 (2.37E-02)
Uterine endometrium	235	91	147 (1.98E-14)	124 (8.47E-07)

Table 3.7: The numbers of MIGs with their 2nd HCIs or HCIs detected at the protein level in each tissue. Comparisons are made between the III networks predicted by FINER and those predicted by TENSION.

Tissue	# of MIGs	N_{FINER}		N_{TENSION}	
		2nd HCI	HCI	2nd HCI	HCI
Heart	316	85	217	90	177
Lung	272	94	150	98	121
Lymphocyte	399	111	252	146	196
Placenta	599	186	345	242	236
Testis	754	249	421	303	313
Thyroid gland	466	129	287	202	196
Uterine endometrium	235	67	147	76	124

We also consider more isoforms of each MIG that have high degrees in the predicted IIIs, and found that the numbers of MIGs whose third highest connected isoforms (3rd HCIs), second highest connected isoforms (2nd HCIs) or HCIs obtained by FINER are detected at the protein level monotonically increase in all tissues (Tables 3.7 and 3.8). This suggests that the isoforms detected at the protein levels tend to have higher degrees in the III networks predicted by FINER. However, this monotonicity property does not always hold in the III networks predicted by TENSION.

Table 3.8: The numbers of MIGs with their 3rd HCIs, 2nd HCIs or HCIs detected at the protein level in each tissue. Here, MIGs with at least three isoforms are considered. Comparisons are made between the III networks predicted by FINER and those predicted by TENSION.

Tissue	# of MIGs	N_{FINER}			N_{TENSION}		
		3rd HCI	2nd HCI	HCI	3rd HCI	2nd HCI	HCI
Heart	177	21	57	102	49	45	77
Lung	149	23	54	65	41	41	52
Lymphocyte	189	34	57	93	55	50	62
Placenta	316	65	103	144	93	100	95
Testis	389	66	128	172	98	123	103
Thyroid gland	241	39	66	122	50	98	70
Uterine endometrium	120	15	34	63	28	37	45

3.3.5 Consistency between interactions of isoforms and their subcellular localization

Subcellular localization of isoforms determines the environments where they operate. Therefore, subcellular localization plays a significant role in controlling the availability of interacting partners of isoforms and further influencing their functions [164]. Thul *et al.* [144] also discovered that interactions among proteins within the same or connected cell organelles are more likely to happen compared to isoforms between disconnected organelles. Inspired by this finding, we collected some data of isoform subcellular localization from Uhlén *et al.* [148], in which isoforms are annotated with locations predicted from their sequences: soluble (intracellular isoforms), membrane-spanning, or secreted. We then examine the enrichment of interactions among isoforms in the same or between different subcellular locations. Figures 3.7A and 3.7C show that, when considering isoforms of SIGs alone, a significant enrichment of interactions is always found between isoforms within the

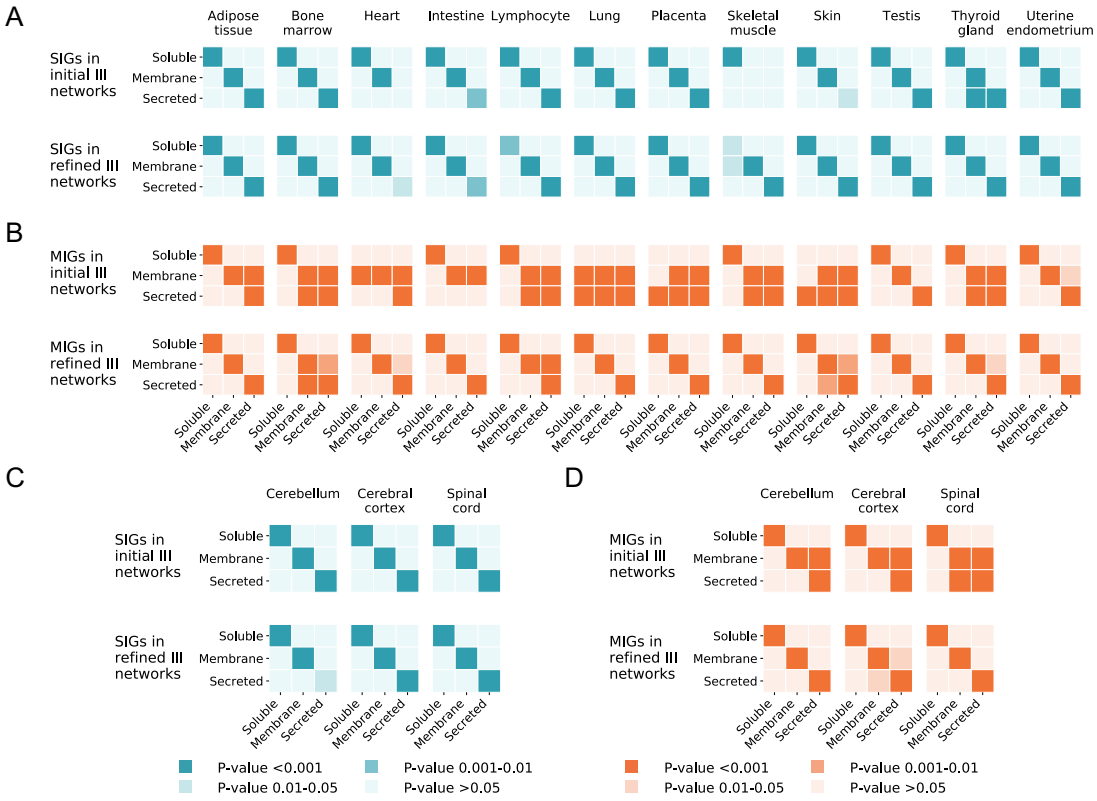


Figure 3.7: Heat maps describe the probability (measured by the FDR-corrected P value for the binomial test) of observing at least as many isoforms in a given location (y axis) by chance, given the location of each isoform's interaction partner (x axis). (**A**) Comparison of the above probabilities between the isoforms of SIGs in the initial III networks and those in the refined III networks for the 12 major tissues. (**B**) The same comparison for the isoforms of MIGs in the 12 major tissue datasets. (**C**) The same comparison for the isoforms of SIGs in the 3 brain sub-tissue datasets. (**D**) The same comparison for the isoforms of MIGs in the 3 brain sub-tissue datasets.

same subcellular location but rarely found between those in different locations, no matter the initial or refined III networks are used. On the other hand, an enhancement of this trend (*i.e.*, enrichment of interactions between isoforms within the same subcellular location) can be seen in refined III networks compared with the initial ones in the Heart, Skeletal muscle, Skin, and Thyroid gland tissues. In contrast, Figures 3.7B and 3.7D show that, when considering only isoforms in MIGs, more enrichment of interactions between isoforms in different locations is found in the initial III networks, but the above trend observed in SIGs still remains true in the refined III networks. A plausible conclusion from these observations is that our results concerning the isoforms of SIGs show consistency with the previous findings [144]. In other words, even though isoforms at the same subcellular location may not belong to the same or connected organelles, it is conceivable that interactions could be more likely to happen between these isoforms compared to isoforms in different locations, as found consistently in our observations. However, since different isoforms of MIGs can be localized differently, initializing III networks based on PPIs may introduce many false interactions between isoforms from different locations. Through III network refinement, real interactions are revealed and thus the expected trend is recovered.

3.3.6 Differentiating functions of isoforms with different localization

It is commonly found that a single gene can encode isoforms with different subcellular localization [148], which suggests the potential functional differences between them. We test if FINER can correctly differentiate the functions of isoforms from the same gene, measured in terms of consistency with their localization. We focus on a set of subcellular location enriched GO terms. Specifically, for each subcellular location, we consider the set

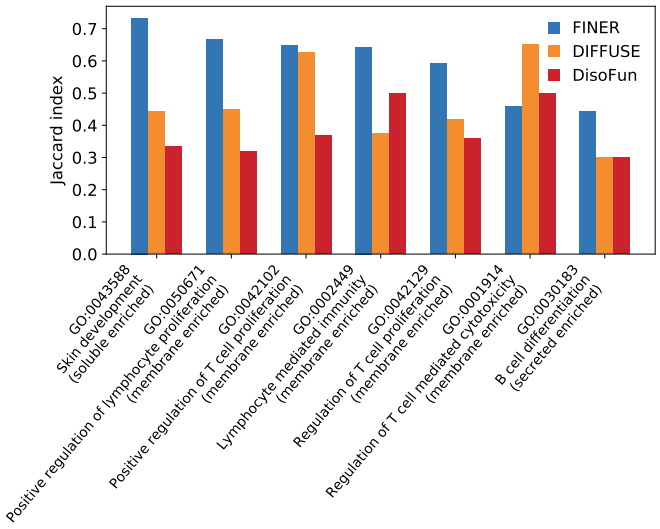


Figure 3.8: Comparison between FINER (blue), DIFFUSE (yellow) and DisoFun (Red) in terms of consistency between their predictions on location enriched GO terms and subcellular localization of isoforms, where the consistency is measured by the Jaccard index.

of genes that encode isoforms located there. Then, GO terms that are enriched in the gene set are selected as the location enriched terms through GO enrichment analysis. For each selected GO term and the corresponding subcellular location, we consider MIGs that are associated with the GO term and encode isoforms with different localization containing at least one isoform in the considered location. Isoforms with prediction scores greater than the background of their genes are annotated with the GO term, where the background of a gene is defined as the average prediction score of all its isoforms. The Jaccard index is used to quantify the agreement that isoforms annotated with a GO term are also located in its corresponding subcellular location.

Figure 3.8 shows that the predictions of FINER achieve a higher consistency with isoform subcellular localization than those of DIFFUSE and DisoFun in 6 out of the 7

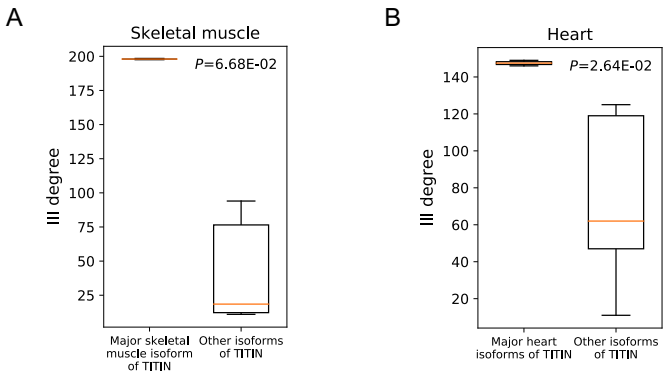


Figure 3.9: The degrees of TITIN isoforms in the refined tissue-specific III networks of skeletal muscle and heart, respectively. (**A**) N2A is the major skeletal muscle isoform of gene TITIN, who has a higher degree compared with the other TITIN isoforms in the refined III network of skeletal muscle. (**B**) N2B and N2BA are the two major heart isoforms of gene TITIN, who have higher degrees compared with the other TITIN isoforms in the refined III network of heart. The Kruskal-Wallis test is performed to test the significance of the degree difference between two groups of isoforms in each tissue.

considered GO terms, while DIFFUSE generally outperforms DisoFun. This result suggests that isoform localization information resides in the refined IIIs and isoform sequences may help FINER differentiate the functions of isoforms with different localization.

3.3.7 Case studies with literature support

We finally perform a literature search for experimental evidence to support the predictions of FINER. In particular, some evidence concerning the tissue specificity of isoforms and their functions is collected from the literature for three genes. The first gene FYN encodes isoforms FynB and FynT. Whereas FynB accumulates highly in the brain, FynT is expressed predominantly in lymphocytes. Accordingly, FynT but not FynB serves a tissue-specific function in T cell activation [34]. This evidence is consistent with the relationship between function and tissue specificity analyzed in Figure 3.6. FINER cor-

rectly predicted the tissue-specific functions of both isoforms. For the lymphocyte-specific GO term "Regulation of T cell activation (GO:0050863)", FynT has a prediction score 1.3 times the background score of its gene, while FynB only has a score 0.6 times the background. The second gene PPARG involves two isoforms with different tissue specificity. The expression of PPARG2 is restricted mainly to the adipose tissue, whereas PPARG1 is expressed in the adipose tissue and many other tissues. PPARG2 can stimulate the formation of adipocytes (fat cells). However, evidence shows that PPARG1 has no or reduced ability to induce adipogenesis [111, 123]. Our predictions on the GO term "Fat cell differentiation (GO:0045444)" accord with the experimental observation. That is, PPARG2 has a prediction score 1.2 times the background, while the score of PPARG1 is 0.8 times the background. The last example concerns three isoforms encoded by gene TITIN. While the isoform N2A is the major isoform of TITIN expressed in skeletal muscles, N2B and N2BA are major TITIN isoforms expressed in the heart, whose expression ratio is related to human heart diseases [104]. The III predictions of FINER show that N2A is the highest connected isoform in the refined III network of skeletal muscles, while isoforms N2B and N2BA are the highest connected ones in that of the heart (Figure 3.9), consistent with the relationship analyzed in Table 3.6.

In addition, we are able to find some experimental evidence that indirectly supports the predictions of FINER concerning the tissue-specific functions of isoforms in four genes. The evidence is collected by the following procedure. For each tissue, among all the MIGs associated with at least one GO term specific to the given tissue, the MIGs whose HCIs have the top five highest degrees (among all HCIs) are selected. An exhaustive literature search is

Table 3.9: Functional prediction cases of FINER that are supported by experimental evidence from the literature.

Tissue	GO term	Gene	Isoform	Fold change of prediction score
Adipose tissue	GO:0045444 Fat cell differentiation	PPARG	PPARG2 PPARG1	1.2 0.8
Lymphocyte	GO:0050863 Regulation of T cell activation	FYN	FynT FynB	1.3 0.6
Adipose tissue	GO:0070345 Negative regulation of fat cell proliferation	GATA2	NM_032638 NM_001145662	1.2 0.8
Lymphocyte	GO:0045579 Positive regulation of B cell differentiation	CD40	NM_001250 NM_152854	1.2 0.9
Testis	GO:0008584 Male gonad development	WT1	NM_001198551 NM_024424 NM_000378 NM_024426 NM_001198552 NM_001367854	1.3 1.2 1.0 0.9 0.8 0.7
Thyroid gland	GO:0030878 Thyroid gland development	NKX2-5	NM_004387 NM_001166176 NM_001166175	1.2 0.9 0.9

then performed against the selected MIGs. Information about tissue-specific functions and corresponding predictions of FINER concerning the isoforms in these MIGs is listed in Table 3.9 (along with the cases discussed in the previous paragraph). Details of the functional evidence are discussed below. The gene CD40 plays an important signal transduction role in the pathway responsible for B cell growth and differentiation. Compared with the isoform NM_001250 encoded by CD40, isoform NM_152854 lacks the transmembrane domain, which makes it signal-nontransducible [145]. Consistently, for the lymphocyte-specific GO term "Positive regulation of B cell differentiation (GO:0045579)", FINER predicts NM_001250 to have a score 1.2 times the background score of its gene, while NM_152854 has a score

0.9 times the background. The gene WT1 regulates gonad development through activating the expression of the gene SF1. However, the presence of the KTS motif in WT1 isoforms hinders their interaction with the SF1 promoter [158]. Accordingly, among the six isoforms encoded by WT1, FINER gives the three isoforms lacking the KTS motif (NM_001198551, NM_024424, NM_000378) higher scores on the testis-specific GO term "Male gonad development (GO:0008584)" than the other three isoforms with the KTS motif, as shown in Table 3.9. In the adipose tissue, the gene GATA2 acts as a negative regulator of adipocyte proliferation through interaction with FOG proteins, where the interaction relies on the contact of their zinc fingers [69]. Between the two isoforms encoded by GATA2, NM_001145662 lacks a zinc finger compared with NM_032638. Accordingly, FINER predicts NM_032638 to have a score 1.2 times the background on the GO term "Negative regulation of fat cell proliferation (GO:0070345)", while the score of NM_001145662 is 0.8 times the background. The gene NKX2-5 acts as a transcription factor during the thyroid gland development [37]. Among the three isoforms NM_004387, NM_001166175 and NM_001166176 of NKX2-5, the DNA binding domain is missing in NM_001166175 and NM_001166176 due to alternative splicing. Correspondingly, on the thyroid gland-specific GO term "Thyroid gland development (GO:0030878)", the isoform with the DNA binding domain (NM_004387) is predicted with a higher score than the other two as shown in Table 3.9.

3.4 DISCUSSION

Isoform function prediction (IFP) and isoform-isoform interaction prediction (IIIP) are two important problems in studying the diversity of gene products. The close ties

between functions and interactions of protein isoforms make the IFP and IIIP problems inherently intertwined. In this work, we presented FINER, a unified framework for solving the two problems jointly. FINER establishes the connection between IFP and IIIP by introducing a joint learning objective, which enables both tasks to benefit from each other. We apply FINER to predict tissue-specific isoform functions and interactions on two datasets, which contain 12 major tissues and three brain sub-tissues of human, respectively. FINER outperforms the state-of-the-art methods across different tissue datasets, and provides isoform function and interaction predictions that accord with other biological evidence, including isoform tissue specificity and isoform subcellular localization. These results suggest FINER's potential in facilitating the functional exploration of (individual) isoforms and their roles in diverse human tissues and diseases.

There are several directions for future work. First, the relationship between tissues is not considered in FINER. The reason is that the tissues studied in this work are relatively independent from each other. If tissue-specific functional terms and well-characterized RNA-seq data are available for a wider range of tissues in the future, the dependence between tissues can be considered and transferring functional knowledge between closely related tissues can be explored in FINER. In addition, FINER converges quickly in practice with several rounds of alternately training the function prediction module and the III refinement module, although we do not have a theoretical proof for its convergence yet. We hope to perform more theoretical analysis in the future. Moreover, although this work focuses on the fundamental problem of isoform function prediction, it would be interesting to see whether FINER can be directly applied to predict isoform–disease associations effectively.

Chapter 4

Novel Embeddings in Functional Spaces to Help Discover Connections Between Gene Sets

4.1 Introduction

Omics-based analyses are now standard practice to deconstruct the molecular mechanisms underlying complex biological systems. One of the most common outcomes when interpreting large-scale omics datasets is the discovery of gene sets. For instance, the gene expression studies measure expression levels of thousands of genes in different conditions, which are further used to identify a set of genes that are differentially expressed. Genetic screening studies identify sets of important genes associated with a disease state or other phenotypes. Critically, comparison of such experimentally derived gene sets usually

enable new discoveries. For example, the L1000 dataset [136] creates a comprehensive catalog of genetic perturbagen- or drug-induced gene expression signatures. A high similarity between gene sets derived from different signatures might indicates previously unrecognized connections (e.g., between a drug and its potential protein targets, or between two structurally dissimilar drugs but targeting the same proteins). As the omics data accumulates and becomes largely available, it is critically important to develop tools that can help scientists to compare gene sets from their data with those from public datasets to uncover associations that lead to new findings.

Routine approaches characterizing the similarity between two gene sets rely on statistics to measure the significance of the number of shared genes between two sets. Among the numerous methods have been developed, commonly used ones are the Fisher's exact test [15] and the weighted Kolmogorov-Smirnov-like statistic introduced in GSEA [137]. The hypothesis of these approaches is that a significant overlap between two sets of genes indicates they are probably involved in the same biological functions, pathways, or regulations. However, in experimentally derived gene sets, a causal pathway is usually represented by a sparse subset of its members. Therefore, two sets of genes from independent experiments studying the same biological system or phenotype may show distressingly little overlap [47]. Previous studies also reported that overlaps between experimentally derived gene sets are more readily apparent at the level of pathways than at the level of gene identities [146, 43].

To facilitate functional analysis of genes and gene sets, many web portals, [171, 36, 82], have been developed. The core functionality of such web portals is the enrichment

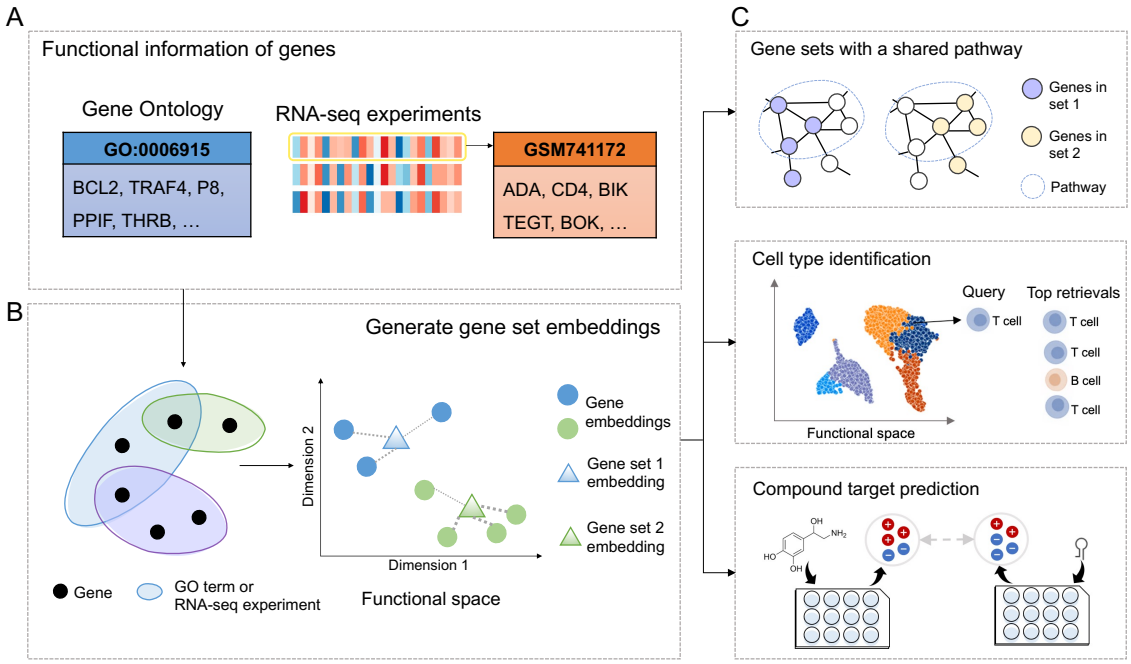


Figure 4.1: Schematic overview of the FEGS workflow. (**A**) FEGS considers two types of functional information of genes, GO annotations and that derived from RNA-seq data. (**B**) FEGS model gene-GO association or gene-RNA-seq sample association as hypergraphs, and then embed genes as vector representations with the help of random walks in the hypergraphs. A consensus embedding for an input gene set is generated by a linear combination of the gene embeddings. (**C**) The gene set embeddings are shown to be sensitive in detecting gene sets with shared pathways and can be applied to multiple applications, such as cell type identification, and compound target prediction.

analysis, in which the input gene set is compared to thousands of groups of genes predefined as functional classes by their involvement in known biological processes, protein localization, pathways, or other features. Functional classes whose members are significantly overrepresented in the input gene set are reported to users to describe the underlying functions of the gene set. These tools, however, do not provide a quantitative way to characterize similarities between different input gene sets on a large scale. On the other hand, machine learning methods have been widely applied to omics data analyses [115, 160]. Leveraging gene set to make new biological discoveries with the help of machine learning usually re-

quires converting gene sets into compact vector representations as the input of available models.

In this chapter, we present a novel computational method, FEGS (Functional Embeddings of Gene Sets), for obtaining high-quality, compact vector representations of gene sets. FEGS is able to embed any user input gene set as a vector representation, where the direction of the vector encodes biological functions enriched in the set of genes. The embeddings generated by FEGS can thus be easily used to quantify similarities between input gene sets with standard similarity measures of vectors, for example, the cosine similarity. In addition, the embeddings of gene sets can also serve as input of available machine learning models for different applications. Specifically, FEGS first pre-trains embeddings for individual genes encoding their comprehensive functional information, in which, both known gene functions recorded in the knowledgebase and functional information concealed in large-scale experimental data are considered (Figure 4.1A). We propose a novel hypergraph based sampling algorithm in the embedding process to encourage genes sharing similar functions to have similar embeddings (Figure 4.1B). Then for an arbitrary input gene set, a consensus gene set embedding is computed by a linear combination of embeddings of genes in the set, in which genes are given different weights to boost the signal-to-noise ratio.

We demonstrate that FEGS offers greater sensitivity to enable the detection of shared pathways between gene sets, and thus in principle can better identify phenotypically similar experiments (Figure 4.1C). We further show the utility of FEGS in high-impact applications. By representing cells as sets of detected genes, the embeddings generated by FEGS better capture phenotype similarities between cells and thus improve cell type

identification across different read depths and tissue types. In addition, the benefits of FEGS embeddings can also be combined with the power of deep learning for more challenging tasks. We use FEGS to generate embeddings for gene sets derived from compound and genetic perturbagen signatures. Serving as the inputs, embeddings of FEGS greatly help improve the performance of an independent deep learning model for compound target prediction.

4.2 Results

4.2.1 Overview of FEGS and other embedding methods for comparison

We consider human genes in this study. FEGS embeds input gene set as vector representations with two steps (Figure 4.1B). (i) FEGS first pre-trains two kinds of embeddings for individual genes in two separate functional spaces independently. The first functional space summarizes known functions of genes recorded in the Gene Ontology database [30], which is one of the largest sources that defines tens of thousands functional classes (GO terms) to describe gene functions. The dimension of the functional space is much smaller than the number of GO terms to reduce the redundancies inherently in the GO hierarchy. On the other hand, functions that haven't been well characterized in the knowledgebase could also be interesting, and such functional information is usually concealed in gene expression patterns under different conditions. Therefore, as a complement, the other low-dimensional functional space summarizes functional information of genes derived from massive RNA-seq data, retrieved from the ARCHS4 database [86]. We adopt the learning objective of the Skip-gram model [109] in FEGS, which is the basis for many well-known word and network embedding methods [118, 52, 52, 141, 142], and propose a novel hyper-

graph based sampling algorithm to encourage genes sharing similar functions to be aligned in the functional spaces. Two kinds of embeddings are then concatenated as the joint vector representation of each gene. (*ii*) Then for an arbitrary input gene sets, a consensus gene set embedding is computed by a linear combination of the pre-trained embeddings of genes in the set, in which weights of gene embeddings are determined by their similarities with other embeddings in the set to reduce the impact of outlier genes in the set.

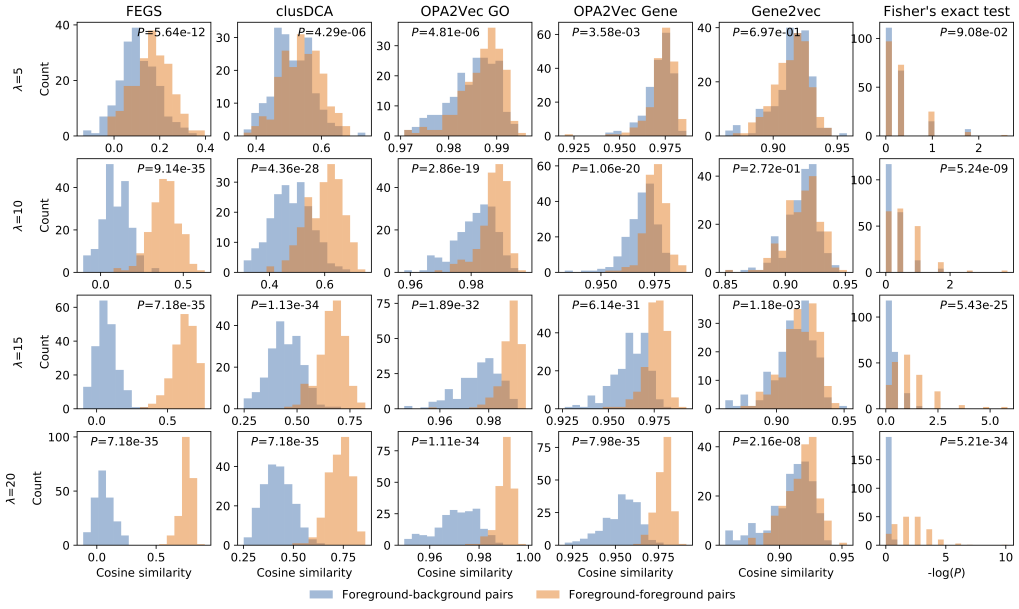
We demonstrated the superiority of our method with a simulation study and then showed its utility in two practical applications, including cell type identification and compound target prediction. In each experiment, we evaluate the embeddings generated by FEGS against those obtained from other state-of-the-art embedding methods that also consider GO or gene expression information: OPA2Vec [133], Gene2vec [39], and clusDCA [156]. OPA2Vec first generates sentences from GO annotations and metadata to form a corpus, in which genes and GO terms are treated as words. It then applies a Word2Vec [109] model to jointly generate embeddings for genes and GO terms. The PubMed [21] abstracts are used as a corpus by OPA2Vec for pretraining. Using OPA2Vec's gene embeddings, an input gene set can be represented by averaging embeddings of individual genes in the set, denoted as 'OPA2Vec Gene'. For completeness, we also generate gene set embeddings using OPA2Vec's GO embeddings, by first adding up embeddings of GO terms associated with each gene as the gene embedding and then averaging gene embeddings in the set, denoted as 'OPA2Vec GO'. Gene2vec generate gene embeddings utilizing transcriptome-wide gene co-expression patterns derived from large-scale GEO datasets. clusDCA generates embeddings of GO terms considering the directed acyclic graph structures of GO hierarchies. Embeddings

of Gene2vec and clusDCA can also be used to generate gene set embeddings as described above.

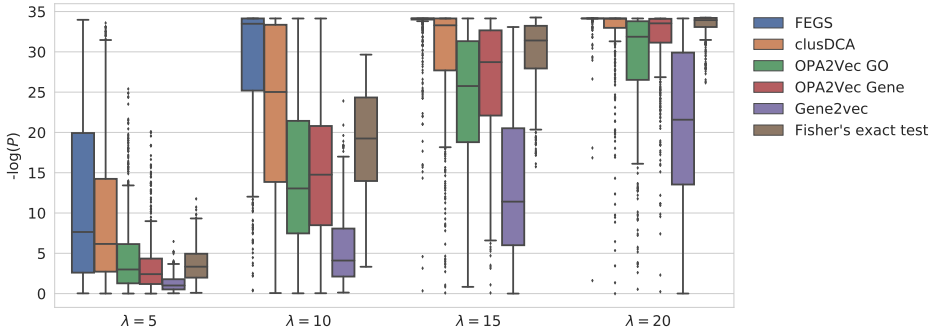
4.2.2 FEGS improves the sensitivity of detecting common pathways between gene sets

We first examine the sensitivity of FEGS embeddings in detecting gene sets with shared pathways. Current practice in gene set comparison applies statistical metrics based on gene identity overlap, where any two distinct genes are considered orthogonal and share no similarity. In experimentally derived gene sets, a causal pathway associated with a specific phenotype is usually represented by a sparse subset of its members, which therefore makes the statistical tests fails to detect the connection between gene sets derived from phenotypically similar experiments.

We test our hypothesis that by effectively encoding function enrichment information in the gene set embeddings, FEGS can offer greater sensitivity in this task. We collected pathway membership information of genes from the Reactome database [44]. Considering each pathway, genes are separated as pathway associated genes and non-pathway associated genes. We generate three gene sets, S_{fg}, S'_{fg}, and S_{bg}, each with 100 genes. S_{fg} and S'_{fg} are randomly generated foreground gene sets, in which λ genes are randomly sampled from pathway associated genes, and the remaining from non-pathway associated genes. S_{fg} and S'_{fg} simulate experimentally derived gene sets with a common pathway present, where λ controls the level of pathway enrichment. S_{bg} is a randomly generated background gene set from non-pathway associated genes. After generating embeddings for the gene sets, the



(A)



(B)

Figure 4.2: (**A**) Similarity score distributions for foreground-foreground gene set pairs and foreground-background gene set pairs obtained using different methods, under different pathway enrichment level. The one-sided Wilcoxon signed-rank test is used to test the significance of their differentiation, and the P value is noted in each subplot. (**B**) The distributions of the negative logrithm of the above P values from the Wilcoxon test for 460 pathways obtained using different methods with different λ.

foreground-foreground similarity between S_{fg} and S'_{fg}, and foreground-background similarity between S_{fg} and S_{bg} are measured by the cosine similarities of their embeddings. As a baseline, we also use the Fisher's exact test to test the significance of the overlap between these gene sets, and use the negative logrithm of the P value to characterize their similarities. The above sampling process is repeated 200 times and the resulting similarity score distributions for foreground-foreground pairs and foreground-background pairs are compared. The one-sided Wilcoxon signed-rank test is used to test how significantly are the similarity scores of the former larger than those of the latter.

The results on the pathway R-HSA-5576891 (Cardiac conduction) (Figure 4.2A) exemplify the superiority of our method. When pathway enrichment signal is week ($\lambda = 5$, the first row), the Fisher's exact test (the last column) fails to distinguish between foreground-foreground pairs and foreground-background pairs, while our method (the first column) distinguishes two kinds of pairs, that is, the similarities measured by our embeddings of foreground-foreground pairs are significantly larger than those of foreground-background pairs (P value = 5.64e-12, one-sided Wilcoxon test). As λ increases, the differentiation between two kinds of gene set pairs becomes clearer, while our method always best distinguishes them.

More generally, we consider all the pathways of human in the Reactome database with the number of associated genes in the range of 50 to 200, which results in 460 pathways. We performed the above simulation experiment for each pathway under different enrichment levels ($\lambda = 5, 10, 15, 20$), and use the negative logrithm of the above mentioned P values from the Wilcoxon test to measure the levels of differentiation between two kinds of gene set

pairs. The larger the value, the clearer the difference. The distributions of the measurements for 460 pathways obtained using different methods are shown in Figure 4.2B. When the enrichment signal is week ($\lambda = 5$), most embedding methods outperform the Fisher's exact test, which comes into play as λ increases. However, our method still more effectively distinguish gene set pairs with or without common pathways than the Fisher's exact test under all the enrichment levels.

The observations demonstrate that our embeddings can more sensitively detect gene sets with shared pathways and thus may better characterize the underlying phenotype similarities.

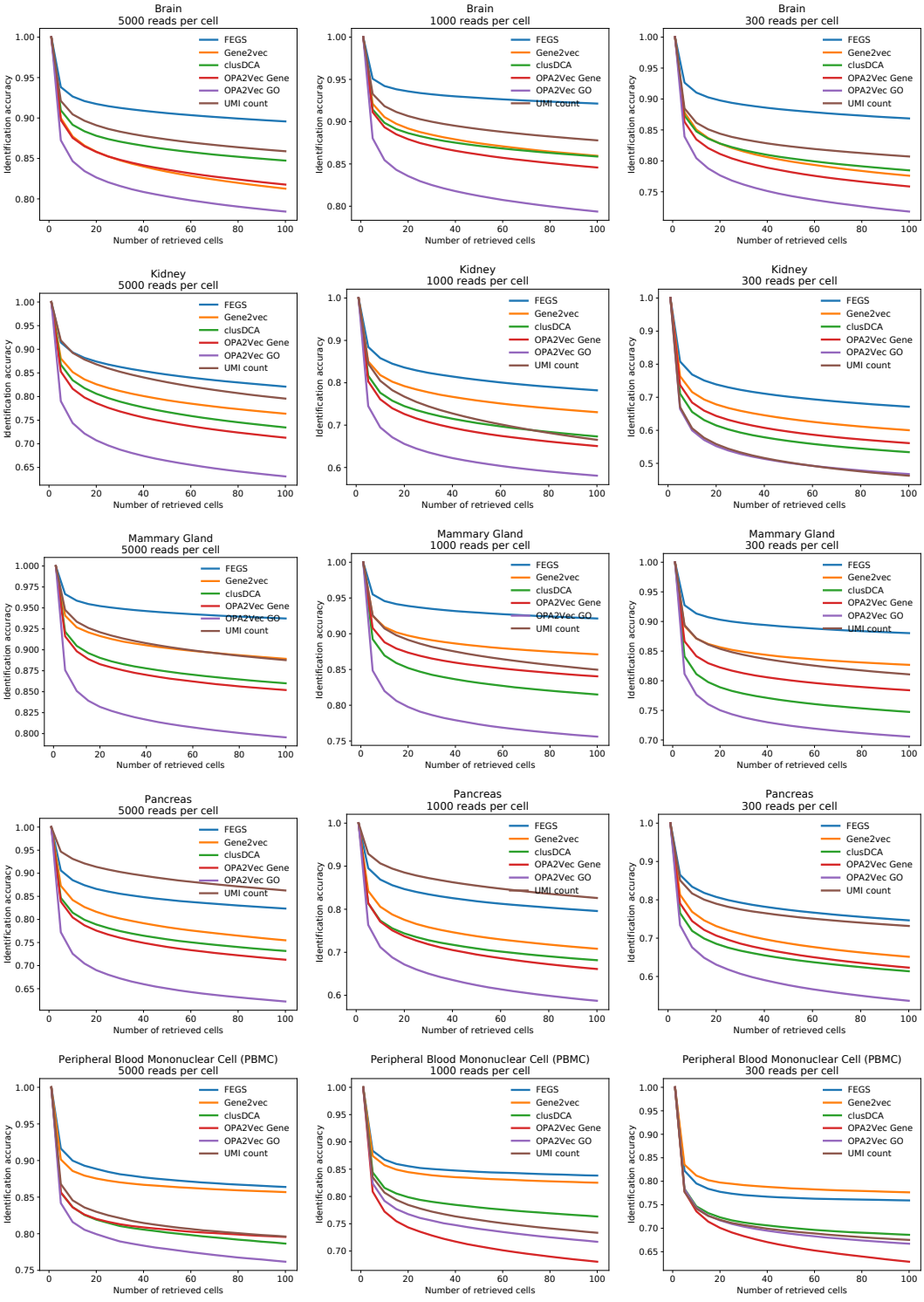
4.2.3 FEGS improves cell type identification

The emergence of single-cell technologies enables the high-definition dissection of cellular heterogeneity in unprecedented detail. Single-cell experiments (e.g. single cell RNA-seq, scRNA-seq) can measure tens of thousands of cells in different phenotypic states, leading to a large number of potential comparisons. However, single-cell measurements also suffer from limited capture efficiency, leading to the lack of detection for many truly expressed genes (i.e. dropouts) [56]. Due to the limitation, phenotypically similar cells are likely to generate profiles that subsample genes functioning in activated pathways, which increases noise when comparing individual cells using their gene expression profiles.

We hypothesize that by embedding single cells as vectors encoding activated pathway information, FEGS can better capture cellular phenotype similarities. We apply FEGS in the task of cell type identification. We collected the scRNA-seq dataset of human Peripheral Blood Mononuclear Cells (PBMC) from the 10X Genomics Chromium platform

and six other scRNA-seq datasets of different human tissues from the Animal Cell Atlas (ACA) database [23], each with at least two different types of cells. For each dataset, we convert each cell in the dataset as a set of genes that meet the two criteria: (*i*) The gene is detected in the cell. (*ii*) The gene is within the top 1500 'variable genes' that exhibit the highest cell-to-cell variation in the dataset, recognized by Seurat [55], which is a standard pre-processing step and has been proved to help highlight biological signals [128, 16]. Then embeddings of each cell are generated using different methods. Similarities between cells are measured by the cosine similarity of their embeddings. To make a comparison, we also represent cells as vectors of their gene expression levels measured by unique molecular identified (UMI) counts, and compute cell-cell similarities using the negative value of L2 distance over the vectors, which is commonly used for cell clustering [128, 92]. For each cell as query, we retrieve the top k most similar cells based on cell-cell similarities of each method, and then compute the identification accuracy, which is a standard metric that quantifies the average numbers of correct retrievals (retrieved cells of the same type as the query) given any query of interests [152] (described in the Methods section).

We compare performance of different methods at multiple read depths by down sampling the overall number of scRNA-seq reads per cell to three levels: 5000 reads per cell, 1000 reads per cell, and 300 reads per cell. Detailed comparisons between FEGS and other methods in terms of identification accuracy with varied numbers of retrieved cells (Figure 4.3) shows that FEGS improves cell type identification across different datasets under different read depths. Specifically, in terms of average identification accuracy when considering


Brain
5000 reads per cell
Brain
1000 reads per cell
Brain
300 reads per cell
Kidney
5000 reads per cell
Kidney
1000 reads per cell
Kidney
300 reads per cell
Mammary Gland
5000 reads per cell
Mammary Gland
1000 reads per cell
Mammary Gland
300 reads per cell
Pancreas
5000 reads per cell
Pancreas
1000 reads per cell
Pancreas
300 reads per cell
Peripheral Blood Mononuclear Cell (PBMC)
5000 reads per cell
Peripheral Blood Mononuclear Cell (PBMC)
1000 reads per cell
Peripheral Blood Mononuclear Cell (PBMC)
300 reads per cell
FEGS
Gene2vec
clusDCA
OPA2Vec Gene
OPA2Vec GO
UMI count
Identification accuracy
Number of retrieved cells

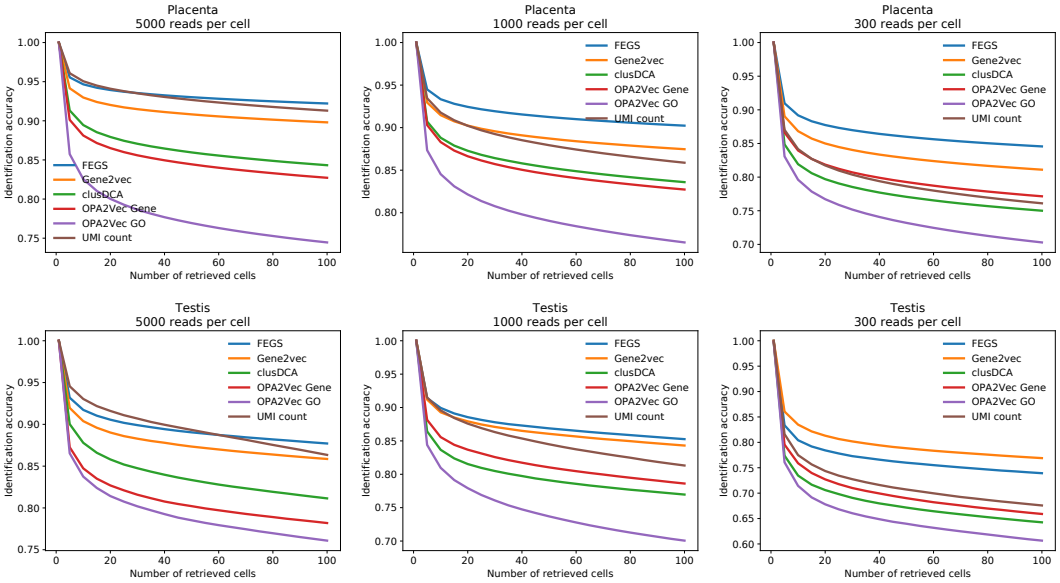


Figure 4.3: Cell type identification accuracy when varying the number of retrieved cells.

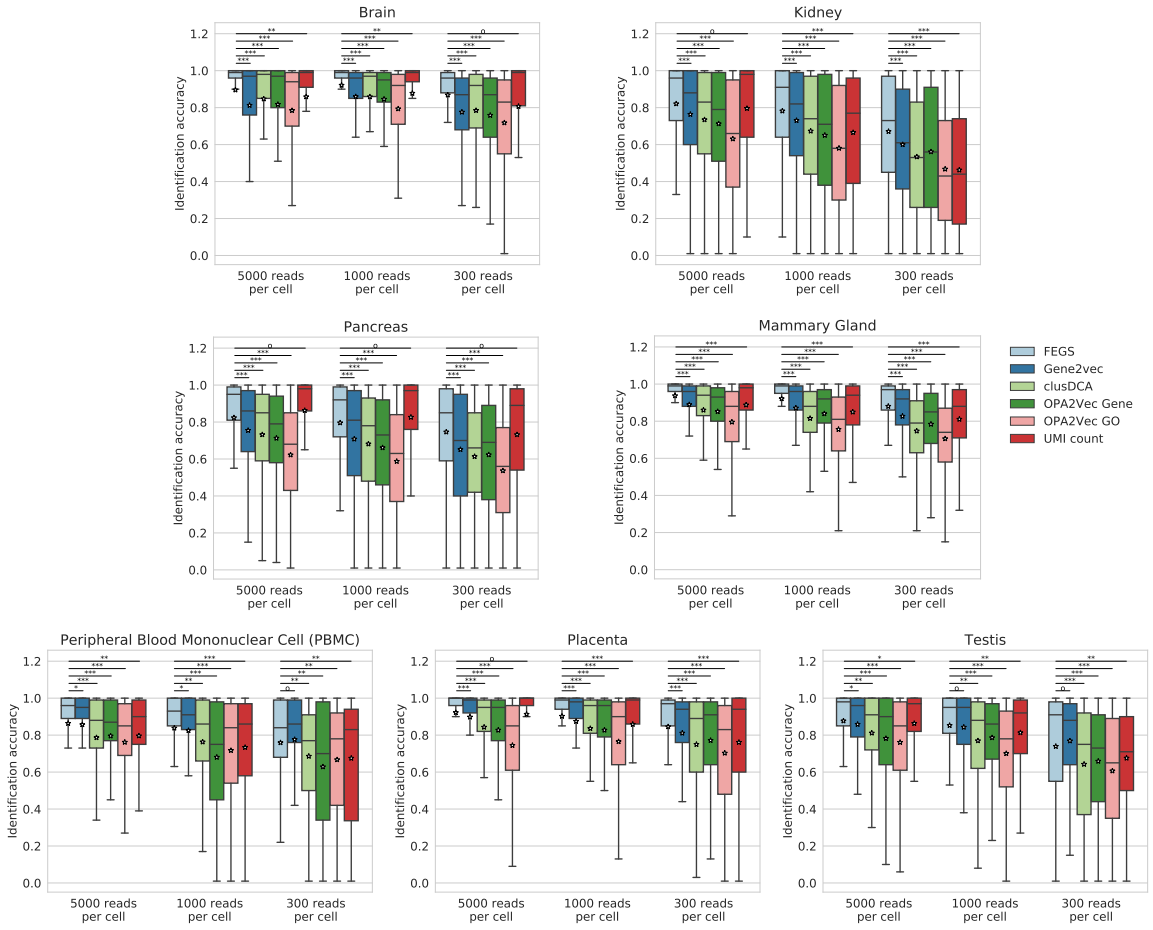
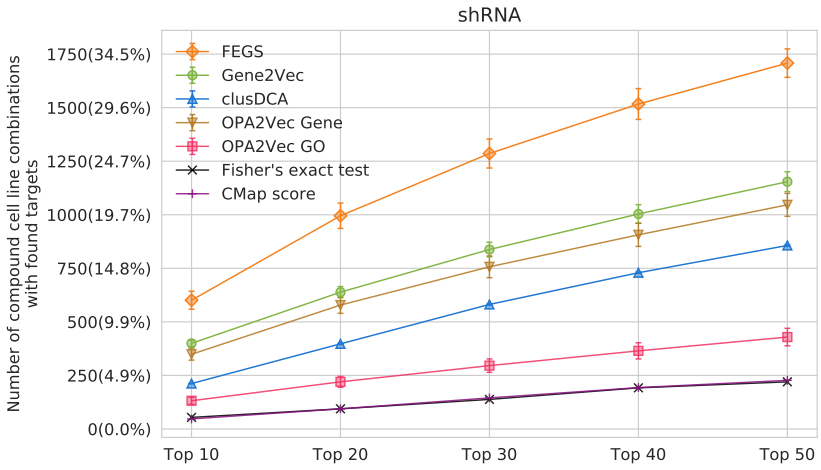


Figure 4.4: Distribution of cell type identification accuracy of methods in different datasets with different read depths, considering 100 top retrievals per cell query. Stars show the means of distributions. The one-sided Mann-Whitney rank test is used to test if the accuracy of FEGS is significant higher than that of the other compared methods (∘: $P \geq 0.05$, *: 1e−10 $\leq P <$ 0.05, **: 1e−100 $\leq P <$ 1e−10, ***: $P <$ 1e−100).

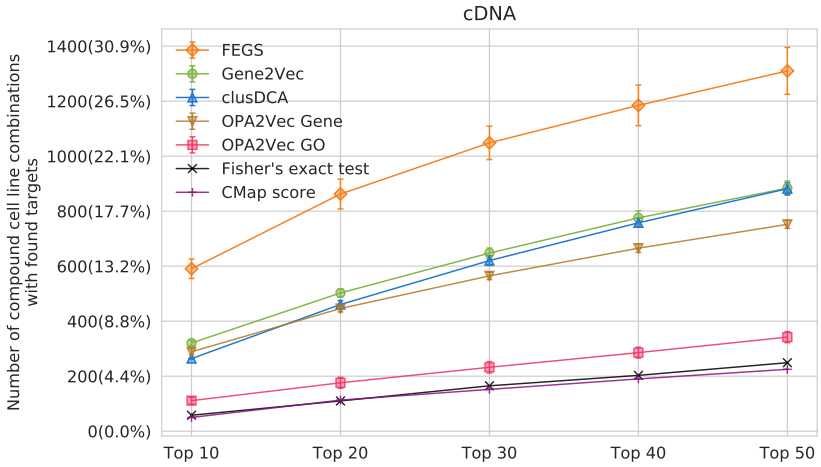
100 top retrievals per cell, our method improves over the best of the other embedding methods by 5.41%, 5.47%, and 6.14% on average across datasets, for read depths of 5000, 1000, and 300 reads per cell respectively (Figure 4.3). More interestingly, the improvement of our method over the L2 distance of UMI counts gets more obvious as read depth decreases (by 2.80%, 7.36%, and 13.75% on average across datasets for three read depths respectively). The observation accords to our hypothesis that as read depth decreases, genes detected in the activated pathways become sparser, which makes the expression profile based method less effective. Our method, however, is able to recover the activated pathway signals and thus maintains the performance. In addition, the P values of the one-sided Mann-Whitney rank test demonstrate that these improvements on different datasets and different read depths are generally significant (Figure 4.4). The results indicates that FEGS can reliably and better capture phenotype similarities between single cells.

4.2.4 FEGS improves compound target prediction

New paradigm to predict molecular interactions using cellular gene expression profiles offer promise for genome-wide screens of drug targets [87]. The NIH Library of Integrated Cellular Signatures (LINCS L1000) program [136] creates a comprehensive catalog of cellular gene expression signatures from perturbagens corresponding to over 25,000 biological entities, including treatments with drug-like compound and gene over-expression (cDNA) or knockdown (shRNA) experiments, in around 80 cell lines. Based on the hypothesis that compounds that inhibit (activate) their targets should yield similar perturbed gene expression signatures to silencing (over-expressing) the target genes, we test the abil-



(A)



(B)

Figure 4.5: (**A**) Top k accuracy of different methods on the shRNA data, which measure in how many gene rank lists of different compounds in different cell lines, there is at least one known target of the compound ranked in the top k. (**B**) Top k accuracy of different methods on the cDNA data. Results of the embedding methods are the average accuracy under three hyperparameter settings of the neural network (described in the Methods section). Error bars represent the standard error of the mean.

ity of FEGS in predicting compound targets using the gene sets derived from the genetic perturbagen- or compound-induced gene expression signatures.

From each compound treatment, gene over-expression or gene knockdown experiment in each cell line, we identify a set of upregulated genes (UP) and a set of downregulated genes (DN) from the differential gene expression values, measured by a z-scoring procedure described in Subramanian *et al.* [136], which can be retrieved from the L1000 platform. Specifically, we rank all the genes measured in each experiment by their z-scores and take the genes ranked in the top 100 with the absolute values of z-scores greater than 2 as the upregulated gene set (UP), and take those ranked in the bottom 100 with the absolute values of z-scores greater than 2 as the downregulated gene set (DN). Then for a pair of compound treatment and gene perturbation (shRNA or cDNA) experiments, we can make four gene set comparisons to study the underlying connection between the compound and the gene, using the gene sets in two directions of each (UP:UP, UP:DN, DN:UP, DN:DN).

We design a Siamese deep learning model [19] (described in the Methods section) to predict whether a gene encodes a protein of a compound. The deep learning model takes two embeddings of a pair of gene sets as input and outputs a score in the range of [0, 1]. The maximum score of the aforementioned four gene set pairs is taken to indicate how likely is the input gene to be the target of the input compound. To evaluate performance of different methods, we perform cross-validation on a group of compounds with known targets, recorded in the Broad's database. Data are partitioned based on compounds, that is, all the gene set pairs associated with the same compound are partitioned into the same set, so the trained model is used to make predictions on unseen compound for different cell

lines. For a given compound in a given cell line, we can rank all the genes based on their probabilities to be the targets of the compound predicted by the model. Models are trained and evaluated separately for cDNA and shRNA data.

For completeness, we also include the Fisher's exact test and the default signature comparing method used by L1000, CMap score, into the comparison. For a compound gene pair, P values from the Fisher's exact test of four gene set pair comparisons are computed, and the maximum negative logrithm of the them is kept for the compound gene pair, which is used to generate the gene rank list for the compound. The CMap score is also computed for each compound gene pair and used to rank genes, which is based on GSEA [137] and introduced in the original paper of L1000 [136].

We use the top-k accuracy to evaluate the performance of different methods, which is standard for the compound target screening task [115]. That is, considering all the gene rank lists for different compounds in different cell lines, which results in 5068 rank lists for shRNA and 4531 rank lists for cDNA, we count in how many of them there is at least one known target of the compound ranked in the top 10 (20, 30, 40, and 50). As shown in Figure 4.5, our method greatly outperforms the other methods, especially the default signature comparing method of L1000, CMap score, for both shRNA and cDNA data. Specifically, on the shRNA data, our method improved the top-10 (top-20/top-30/top-40/top-50) accuracy by 50.4% (55.8%/53.5%/51.1%/48.0%) over the best among other compared methods, while on the cDNA data, our method improve the top-10 (top-20/top-30/top-40/top-50) accuracy by 84.2% (71.6%/61.9%/52.7%/48.2%) over the best among compared methods. Such improvements in this task indicate the huge potential of our method in helping drug

discovery.

4.2.5 Analysing the effects of different components of FEGS

We next evaluate the contribution of major components of FEGS. We perform ablation studies by removing components from FEGS and measuring how the performance of the method is affected in different tasks. We first evaluate the contribution of two kinds of functional information respectively by generating gene set embeddings using only the GO information, denoted as FEGS GO, and using only the RNA-seq information, denoted as FEGS RNA-seq. We also evaluate the contribution of the gene weighting step when generating the consensus embedding of a gene set from gene embeddings, by replacing this step with simply averaging individual gene embeddings, denoted as FEGS w/o. gene weighting.

We first compare FEGS with FEGS GO and FEGS RNA-seq. In the simulation study, FEGS GO outperforms FEGS and FEGS RNA-seq (Figure 4.6). We believe this result is because sampling genes from known pathways favors to FEGS GO, which also considers known functions of genes. In the cell type identification task (Figure 4.8) and the compound target prediction task (Figure 4.11 and 4.12), FEGS clear outperforms FEGS GO and FEGS RNA-seq, which demonstrate the benefits of combining two complementary functional information in our method.

We then compare FEGS with FEGS w/o. gene weighting. FEGS clearly outperforms FEGS w/o. gene weighting in the simulation study (Figure 4.7) and the compound target prediction task (Figure 4.13 and 4.14). In the cell type identification task, FEGS shows higher average identification accuracy over FEGS w/o. gene weighting with statisti-

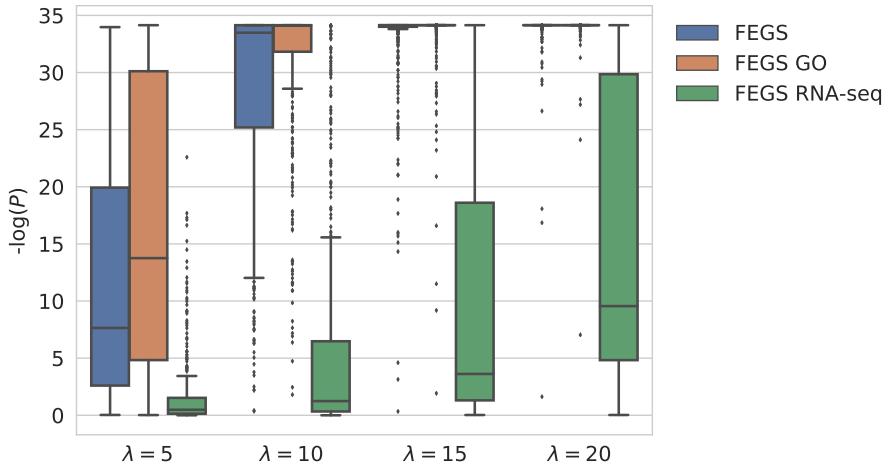


Figure 4.6: The distributions of separations between foreground-foreground pairs and foreground-background pairs in 460 pathways. Comparisons are made among FEGS, FEGS built on only GO information, FEGS built on only RNA-seq information.

cal significance in 9 out of 21 comparisons (Figure 4.9). The explanation is that the gene weighting step is designed to reduce the impact of outlier gene in the consensus embedding, however, following the standard of analyzing scRNA-seq data, the gene set generation step for single cells results in 'denoised' gene sets already by considering only highly 'variable genes' of each dataset, which makes our denoising step not showing its effect. To prove this hypothesis, we relax the gene set generation criteria to consider the top 5000 'variable genes' in each dataset, which might introduce more noise in the generated gene sets. In this experiment, FEGS gets clearer improvements over FEGS w/o. gene weighting, that is, FEGS shows higher average identification accuracy over FEGS w/o. gene weighting with statistical significance in 15 out of 21 comparisons (Figure 4.10). The results demonstrate the gene weighting step effectively boosts the signal-to-noise ratio in the gene set embeddings.

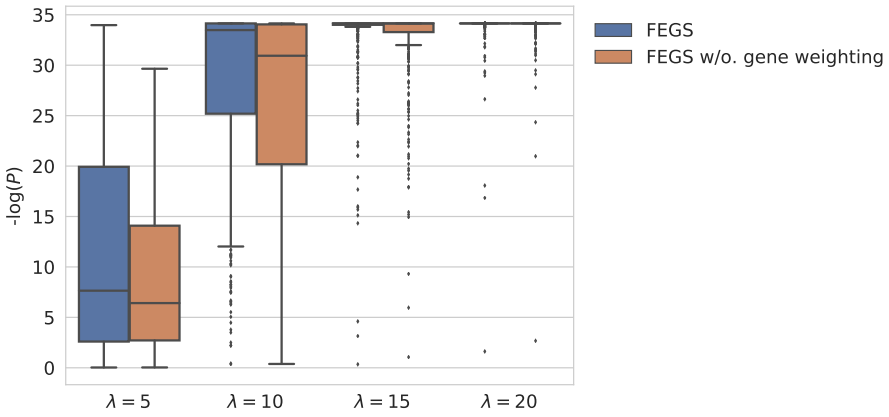


Figure 4.7: The distributions of separations between foreground-foreground pairs and foreground-background pairs in 460 pathways. Comparisons are made between FEGS and FEGS without gene weighting when generating gene set embeddings.

4.3 Methods

4.3.1 Functional information of individual genes

FEGS pre-trains two kinds of embeddings for individual genes considering functional information from two sources separately: (*i*) Gene Ontology and (*ii*) RNA-seq data. Gene Ontology database [30] is one of the largest sources that defines tens of thousands functional classes (GO terms) to describe gene functions. We consider all the biological process terms that associate with less than or equal to 200 genes to focus on those relatively more specific functions, which results in 15,047 GO terms.

As complementary information to GO, we downloaded all the 238,522 RNA-seq samples of human collected by the ARCHS4 [86] database (as of December, 2020). Gene counts for each sample are quantified by ARCHS4 against the GRCh38 human reference genome using Kallisto [14]. Following the data processing procedure of ARCHS4, we apply the log2 transformation then quantile normalization on the gene counts. The gene expression

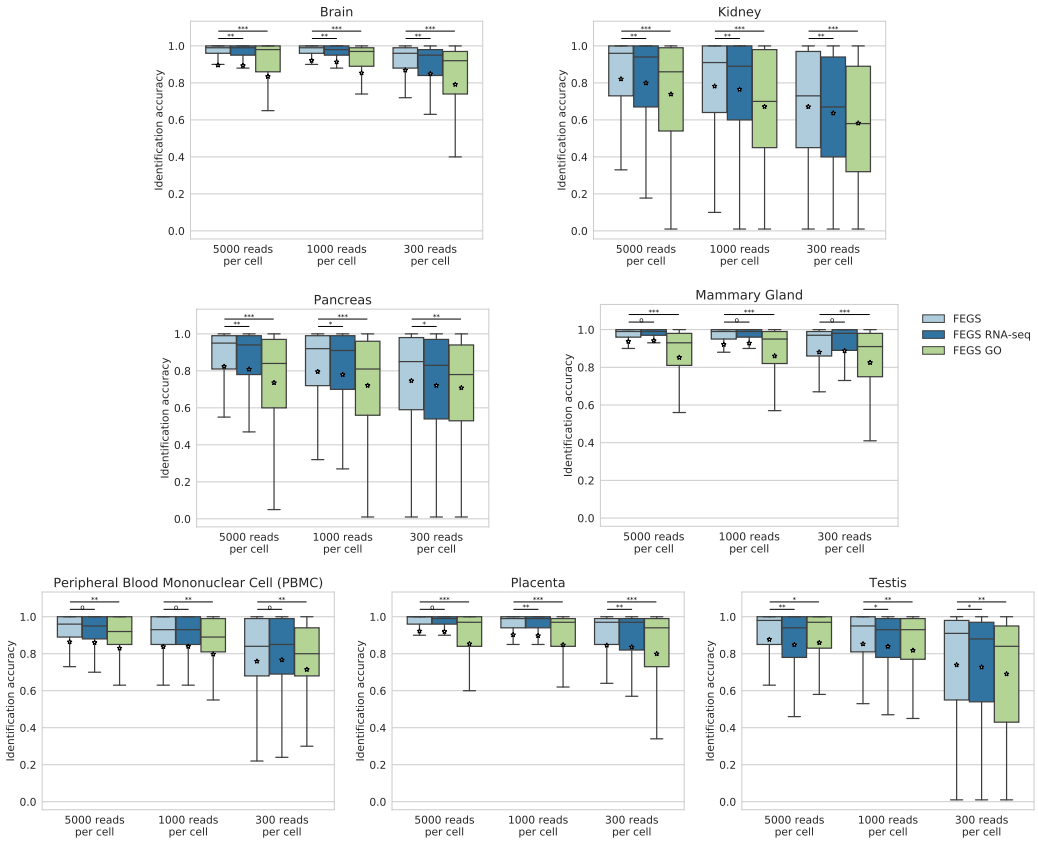


Figure 4.8: Distribution of cell type identification accuracy when considering 100 top retrievals per cell query. Stars show the means of distributions. One-sided Mann-Whitney rank test is used to test if the accuracy of FEGS is significant higher than that of FEGS RNA-seq and FEGS GO (∘: $P \geq 0.05$, *: 1e−10 $\leq P <$ 0.05, **: 1e−100 $\leq P <$ 1e−10, ***: $P <$ 1e−100).

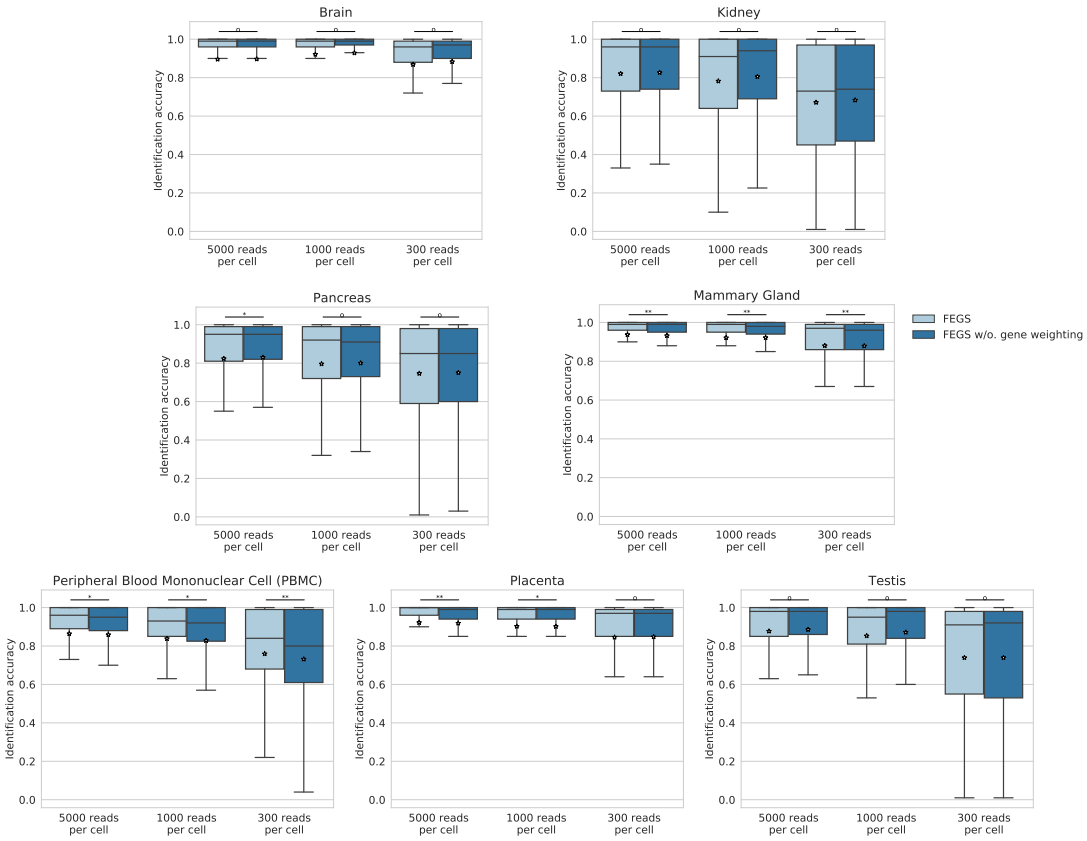


Figure 4.9: Distribution of cell type identification accuracy when considering 100 top retrievals per cell query. Stars show the means of distributions. One-sided Mann-Whitney rank test is used to test if the accuracy of FEGS is significant higher than that of FEGS without gene weighting when generating gene set embeddings. ($\circ$: $P \geq 0.05$, *: 1e$-$10 $\leq P <$ 0.05, **: 1e$-$100 $\leq P <$ 1e$-$10, ***: $P <$ 1e$-$100). The top 1500 'variable genes' that exhibit the highest cell-to-cell variation of each dataset are considered when generating gene sets of single cells.

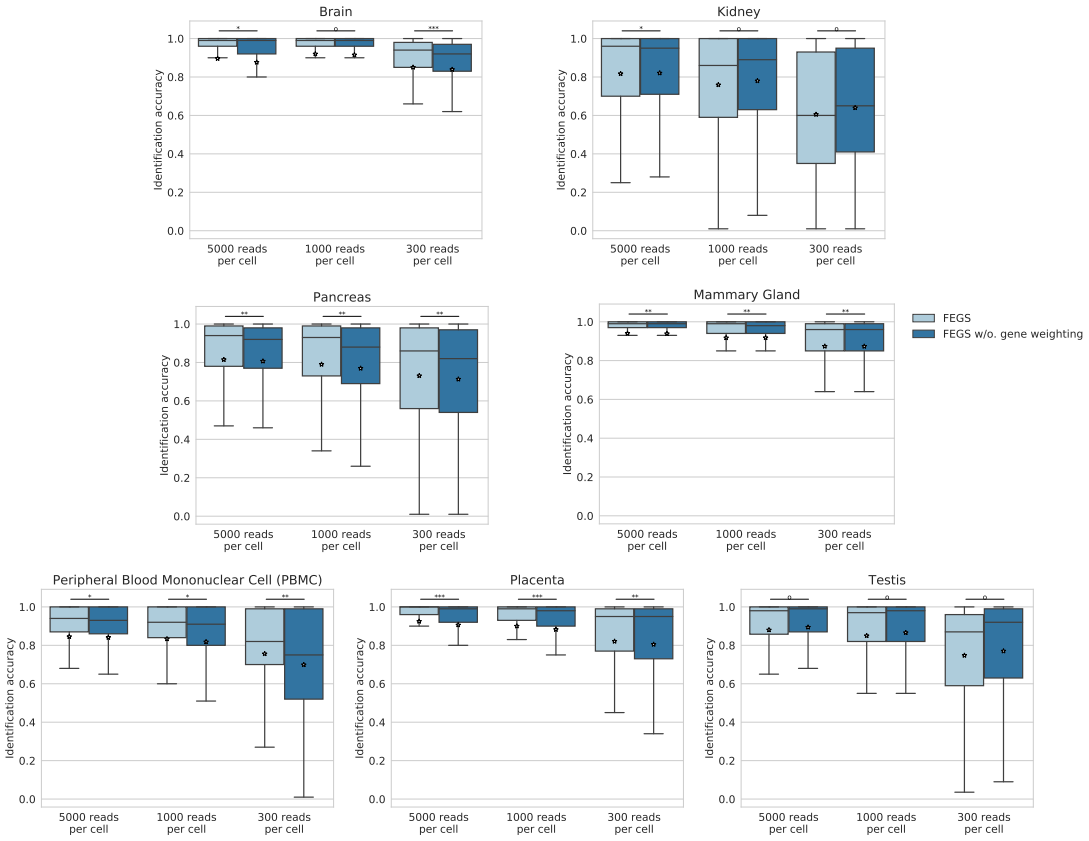


Figure 4.10: The same comparison as Figure 4.9, while the scope of the 'variable genes' of each dataset is increased to the top 5000 genes that exhibit the highest cell-to-cell variation.

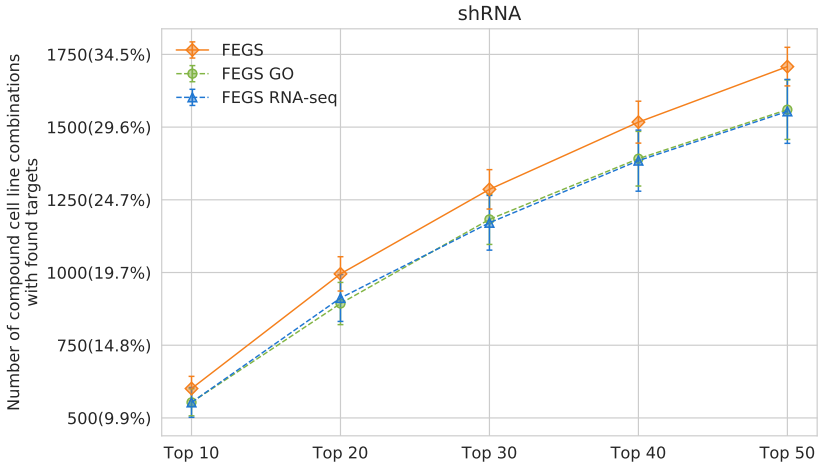


Figure 4.11: Top k accuracy on the shRNA data. Comparisons are made among FEGS, FEGS GO, and FEGS RNA-seq.

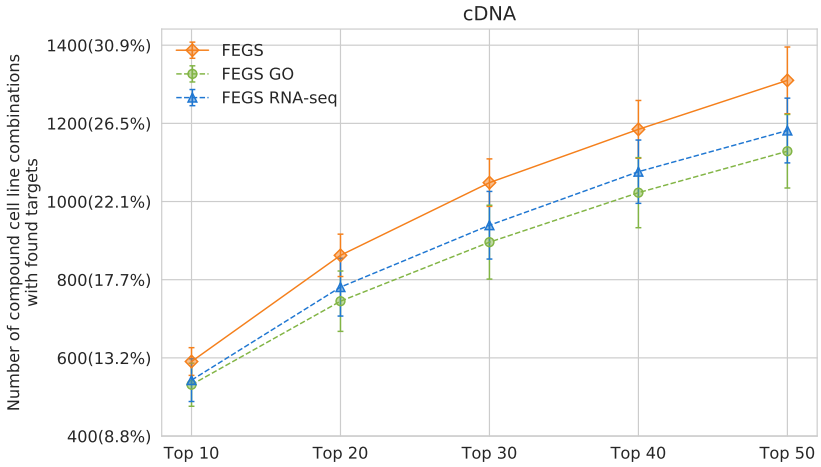


Figure 4.12: Top k accuracy on the cDNA data. Comparisons are made among FEGS, FEGS GO, and FEGS RNA-seq.

Figure 4.13: Top k accuracy on the shRNA data. Comparisons are made between FEGS and FEGS without gene weighting.

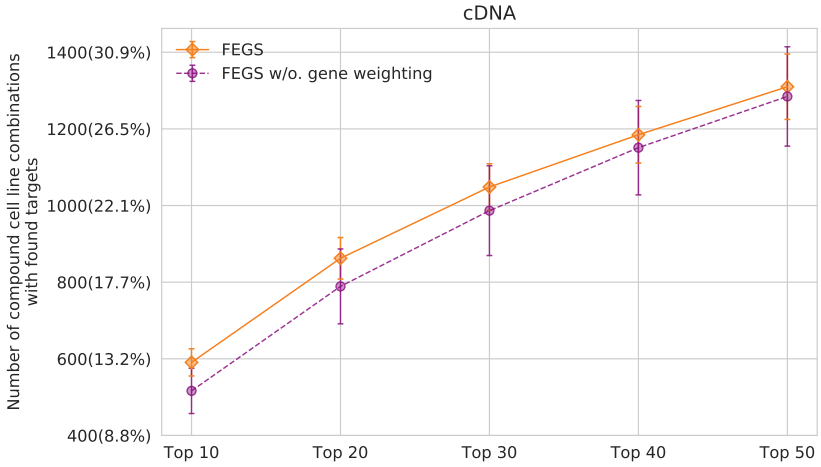


Figure 4.14: Top k accuracy on the cDNA data. Comparisons are made between FEGS and FEGS without gene weighting.

is then z-score normalized across samples to identify the relative gene expression. For each sample, after ranking genes based on their z-scores, we identify a set of regulated genes that meet the following two criteria at the same time: (*i*) Genes whose absolute values of the z-scores are greater than 2. (*ii*) Genes ranked in the top 100 or the bottom 100. The set of identified genes are assumed to work coherently towards the same biological processes or functions.

Finally, there are 25,084 human genes that are involved in at least one GO term or the regulated gene set of one RNA-seq sample, which are kept in this study.

4.3.2 Embedding individual genes into functional spaces with a novel sampling strategy

FEGS creates two d dimensional functional spaces separately for GO and RNA-seq, and uses the same algorithm to embed each gene as a vector representation of length d in each space, such that functionally similar genes should be embedded closely together, otherwise they should be separated from each other in the space. To achieve this goal, we specify the following learning objective. Given a gene u in the functional space of GO (RNA-seq), the objective function seeks to predict, which genes commonly appear in the same GOs (RNA-seq samples) with u, denoted as $N(u)$, based on their learned embeddings. More formally, the objective is to maximize a conditional likelihood that is the product of softmax units parameterized by the dot product of genes' embeddings:

$$p(N(u)|u) = \prod_{v \in N(u)} \frac{\exp(x_v^{\mathsf{T}} x_u)}{\sum_{n \in V} \exp(x_n^{\mathsf{T}} x_u)}, \tag{4.1}$$

where x are embeddings of genes, u and v is a pair of genes with similar functions, and gene z is from the set of all the genes in the genome, denoted as V.

A challenge is how to effectively sample such gene pairs with similar functions. Unlike word embedding and network node embedding, which usually adopt similar learning objectives [118, 52, 52, 141, 142], the data in our task don't have the sequence or graph structures that can be used to sample data pairs. We propose to convert gene-GO (gene-RNA-seq) associations into a hypergraph (Figure 4.1B), in which, each gene is represented as a node and each GO term (RNA-seq sample) is represented as a hyperedge e which connects all the genes associated with the GO term (regulated gene set of the RNA-seq sample). We then perform random walk with restart (RWR) in the hypergraph to identify genes that lie close to each other in the graph. To pay more attention to specific functions, we assign each hyperedge e a weight by its information content, defined as:

$$w(e) = -\log_2 \frac{f(e)}{\sum_{e' \in E} |e'|}, \tag{4.2}$$

where E is the set of all the edges in the hypergraph, $|e|$ is the number of nodes connected by the hyperedge e, and $f(e)$ is the frequency of the edge. In the gene-GO association hypergraph, the frequency of an edge e is defined as $\sum_{c \in C(e)} |c|$, where $C(e)$ includes hyperedges corresponding to all the children GO terms of the term represented by e and the term itself, while in the gene-RNA-seq association hypergraph, the frequency of an edge e is defined as $|e|$. The smaller the size of an edge, the greater its weight.

The random walk in a hypergraph can be interpreted as, given the current node $u \in V$, first choose a hyperedge e over all the hyperedges incident to u with the probability

proportional to $w(e)$, and then randomly choose a node $v \in e$ uniformly [170]. Let B denote the transition probability matrix of random walk in the hypergraph, each entry of B is thus computed as:

$$B(u,v) = \sum_{e \in E} w(e) \frac{H(u,e)}{d(u)} \frac{H(v,e)}{|e|}, \tag{4.3}$$

where H is a $|V| \times |E|$ matrix with entries $H(u,e) = 1$ if $u \in e$ and 0 otherwise, $d(u)$ is the degree of gene u in the hypergraph, which is defined as $d(u) = \sum_{\{e \in E | u \in e\}} w(e)$. The RWR from a node u is then defined as follows in matrix notation:

$$s_u^{t+1} = (1-q) B s_u^t + q a_u, \tag{4.4}$$

where q is the probability of restart, a_u represents the initial state, which is a V-dimensional vector with one on the u-th element and zeros elsewhere, s_u^t is a V-dimensional distribution vector which holds the probability of each nodes being visited after t steps starting from node u. The distribution vectors of all the nodes form as a matrix S^t. After iterative update, we get the stationary distribution matrix $S = S^\infty$ when the Frobenius norm of the difference between S^{t+1} and S^t is smaller than a predefined threshold. A higher probability in the stationary distribution matrix indicates two corresponding genes lie closer in the hypergraph, which suggests that they share more and even more specific functions with each other comparing to with other genes in the graph.

For each gene u, we sample $N(u)$ from other genes in the genome with probabilities proportional to the stationary distribution s_u. As the computation of the full softmax in Equation 4.1 is expensive, we approximate the objective using negative sampling [53, 109].

Specifically, for each gene u, we sample a set of negative samples $R(u)$ from other genes in the genome with probabilities inversely proportional to its stationary distribution s_u. In our experiment, the ratio between the sizes of $N(u)$ and $R(u)$ is 1:5. Then the overall learning objective is to minimize the following negative loglikelihood:

$$-\sum_{u\in V}(\sum_{v\in N(u)}\log\sigma(x_v^{\mathsf{T}}x_u)-\sum_{z\in R(u)}\log\sigma(x_z^{\mathsf{T}}x_u)), \tag{4.5}$$

where σ is the sigmoid function. The task is thus formulated as distinguishing the functionally similar gene pairs (u, v) from functionally dissimilar gene pairs (u, z) though optimizing their embeddings.

The method is applied to two kinds of functional information to get embeddings of each gene in two functional spaces. For a given gene u, we concatenate its two embeddings, $x_u^{(GO)}$ and $x_u^{(RNA-seq)}$, as a joint vector representation $x_u = [x_u^{(GO)}, x_u^{(RNA-seq)}]$.

4.3.3 Generating embeddings for arbitrary gene sets from gene embeddings

For an arbitrary input gene set, we generate an consensus gene set embedding from the pretrained gene embeddings. Due the technical noise in omics data [16] and different choices of parameters in data preprocessing steps [137], outlier genes that do not work in the pathways underlying the phenotype of interest may be introduced in the discovered gene set. We thus compute the consensus gene set by a linear combination of the embeddings of genes in the set, who are given different weights. Specifically, for a given gene u in an

input gene set G, we compute its average similarities with other genes in the set based on two kinds of embeddings, denoted as:

$$r_u^{(GO)} = \frac{1}{|G|} \sum_{v \in G} \cos(x_u^{(GO)}, x_v^{(GO)}), \tag{4.6}$$

$$r_u^{(RNA-seq)} = \frac{1}{|G|} \sum_{v \in G} \cos(x_u^{(RNA-seq)}, x_v^{(RNA-seq)}), \tag{4.7}$$

where cos is the cosine similarity. $r_u^{(GO)}$ and $r_u^{(RNA-seq)}$ are then z-score normalized against the corresponding similarity distribution observed between the gene u and all the other genes in the genome, denoted as $z_u^{(GO)}$ and $z_u^{(RNA-seq)}$.

The weight of gene u in the gene set G is then determined by

$$w_u = \min(\max(\max(z_u^{(GO)}, z_u^{(RNA-seq)}), 0), 1), \tag{4.8}$$

and the consensus gene set embedding is computed as $x_G = \frac{1}{|G|} \sum_{u \in G} w_u * x_u$, by which the impact of outlier genes in the set will be reduced.

4.3.4 Cell type identification accuracy

We use retrieval accuracy for evaluation of the cell type identification. For a query single cell q, the accuracy on its top k retrievals is defined as:

$$acc(q, k) = \frac{\text{\# of correct retrievals}}{\min(k, T_q)}, \tag{4.9}$$

where '# of correct retrievals' is the number of retrieved cells with the same type of q, and

T_q is the number of cells in the dataset with the same type of q. The average accuracy over all the M cells in the dataset is defined as:

$$Acc = \frac{1}{M} \sum_{i=1}^{M} acc(q_i, T_{q_i}). \tag{4.10}$$

4.3.5 Neural network architecture for compound target prediction

We design a neural network based binary classifier which takes compound gene set embedding and shRNA (cDNA) gene set embedding as input and outputs how likely does the gene of the shRNA (cDNA) to be the target of the input compound. The neural network contains two components descipbed as follows.

Siamese feature extraction component. Due to of the symmetric relation of two gene sets when comparing them, we design a Siamese neural network component [19] that uses the same weights to process and extract hidden features from the compound gene set embedding and shRNA (cDNA) gene set embedding. The component is an one-layer fully connected neural network, denoted as:

$$D(x) = Dense_1(x), \tag{4.11}$$

where $Dense$ denotes a fully connected layer, and x is one of the input gene set embeddings. Given a pair embeddings of gene sets from compound treatment and genetic pertebagen (shRNA or cDNA), (x_{cpd}, x_{gene}), the Siamese component is applied to obtain hidden features $D(x_{cpd})$ and $D(x_{gene})$ from both embeddings. Both hidden features are then combined using the element-wise multiplication, $D(x_{cpd}) \odot D(x_{gene})$, which is a commonly used operation

for modeling the symmetric relations of two inputs [28, 57, 70].

Classification component. We build a two-layer fully connected neural network as the classifier and apply it on the combined hidden feature of the input pair. The sigmoid activation function is applied in the last layer, whose output is thus a scalar in the range [0, 1]. The whole neural network can then be denoted as:

$$o = Dense_3(Dense_2(D(x_{cpd}) \odot D(x_{gene}))), \tag{4.12}$$

We use the cross entropy loss for the training of the neural network. Considering all the training sample pairs, each with a binary label y_i, indicating whether or not the input gene is the target of the input compound, the learning objective is to minimize the following cross-entropy loss:

$$-\sum_i (y_i \log o_i + (1 - y_i) \log(1 - o_i)) \tag{4.13}$$

In the experiment, we examine the influence of neuron numbers in each layer on the performance. Since the last layer always has one neuron, we tried three combinations for the number of neurons in $[Dense1, Dense2]$: [256, 64], [128, 32], and [56, 16]. The results (Figure 4) demonstrate that the performance is stable on different settings.

4.4 Discussion

In this work, we propose a novel embedding methods that can embed any input gene set as a fixed-length vector representation, which encodes both database-recorded and

data-derived functional information enriched in the gene set. Through a simulation study, we proved our hypothesis that our function based embedding can greatly facilitates gene set comparison by more sensitive detection of shared pathways between gene sets. We have shown how FEGS can be applied to high-impact applications. We demonstrated that the embeddings of FEGS can help improve cell type identification through better capturing cellular phenotype similarities. In addition, the embeddings of FEGS from perturbagen signatures can be combined with the power of deep learning to greatly improve compound target prediction in genome wide. With this concept, our method should have other useful applications, like disease subtype identification.

As here we want to develop a general method for users to help comparing gene sets derived from omics data, two types of prior functional knowledge from the GO database and that derived from RNA-seq data are considered in our embedding. However, using the same strategy, other more specific information can be easily incorporated into the embedding, and thus our method can be transformed as one that is specifically targeted at a certain task. For example, in the cell type or state identification task, epigenomic information, like methylation, may be incorporated into the embedding and thus help capture cell phenotypes.

In our experiments, similarity scores are compared across gene set pairs within the same datasets or same cell lines. However, when there are needs to compare similarity scores of gene set pairs between different datasets or cell lines, the similarity scores can be easily normalized as P values by comparing each with the similarity score distribution within the dataset.

We expect to integrate FEGS into exiting gene set analysis portals, *e.g.*, our previous developed web portal Metascape [171], for users to easily and better compare gene sets derived from their data and perform meta-analysis taking the advantage of massive publicly available data sets that continue to accumulate.

Chapter 5

Conclusions

A more fine-grained understanding of the functions of proteins can enhance our understanding of biological processes, which can help to uncover mechanisms of complex diseases. In this dissertation, we explore two problems in functional genomics using deep learning: (*i*) Refining functional annotations and interactions of proteins from the gene level to a higher resolution at the isoform level. (*ii*) Exploiting functional knowledge to discover connections within omics data.

We propose three methods: DIFFUSE, FINER, and FEGS. DIFFUSE for the first time integrates isoform sequences and expression profiles to systematically predict isoform functions, by combining the power of deep learning and probabilistic graphical models. FINER models isoform function prediction and isoform-isoform interaction prediction jointly. By introducing a mutual regularization term, two learning tasks are unified into one single learning objective, enabling both tasks to benefit from each other. FEGS embeds gene sets as compact features encoding their functional enrichment information through a novel

hypergraph embedding method, which facilitates gene set comparison by more sensitive detection of shared pathways. FINER and DIFFUSE significantly outperform earlier isoform function prediction methods and their predictions are validated by independent biological data. FEGS has been successfully applied in several high-impact biological applications.

The recent advances in protein structure prediction methods such as AlphaGO [71] and RoseTTAFold [6] offer great promise for understanding protein functions. These highly accurate protein structure prediction methods may be able to find structural differences between isoforms, which can facilitate the understanding of their functional differences. Better isoform function prediction and validation methods can be therefore developed using the predicted structures. On the other hand, alternative splicing plays an important role in some diseases. New computational methods that can accurately predict aberrant splicing events and their functional impacts will be the key to bridge the gaps between splicing, isoform functions, and disease phenotypes.

Bibliography

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. TensorFlow: A System for Large-Scale Machine Learning. In *OSDI*, volume 16, pages 265–283, 2016.

[2] Stephen F Altschul, Thomas L Madden, Alejandro A Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic acids research*, 25(17):3389–3402, 1997.

[3] Stuart Andrews, Thomas Hofmann, and Ioannis Tsochantaridis. Multiple instance learning with generalized support vector machines. In *AAAI/IAAI*, pages 943–944, 2002.

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

[5] Günter Auerbach, Anja Herrmann, Andreas Bracher, Gerd Bader, Markus Gütlich, Markus Fischer, Martin Neukamm, Marta Garrido-Franco, John Richardson, Herbert Nar, et al. Zinc plays a key role in human and bacterial GTP cyclohydrolase I. *Proceedings of the National Academy of Sciences*, 97(25):13567–13572, 2000.

[6] Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 2021.

[7] Amos Bairoch, Rolf Apweiler, Cathy H Wu, Winona C Barker, Brigitte Boeckmann, Serenella Ferro, Elisabeth Gasteiger, Hongzhan Huang, Rodrigo Lopez, Michele Magrane, et al. The universal protein resource (UniProt). *Nucleic acids research*, 33(suppl_1):D154–D159, 2005.

[8] Omer Basha, Ruth Barshir, Moran Sharon, Eugene Lerman, Binyamin F Kirson, Idan Hekselman, and Esti Yeger-Lotem. The tissuenet v. 2 database: A quantitative view of protein-protein interactions across human tissues. *Nucleic acids research*, 45(D1):D427–D431, 2017.

[9] Alex Bateman, Lachlan Coin, Richard Durbin, Robert D Finn, Volker Hollich, Sam Griffiths-Jones, Ajay Khanna, Mhairi Marshall, Simon Moxon, Erik LL Sonnhammer, et al. The pfam protein families database. *Nucleic acids research*, 32(suppl_1):D138–D141, 2004.

[10] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[11] Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J Martin, Karine Michoud, Claire O'Donovan, Isabelle Phan, et al. The swiss-prot protein knowledgebase and its supplement trembl in 2003. *Nucleic acids research*, 31(1):365–370, 2003.

[12] Alice Bossi and Ben Lehner. Tissue specificity and the human protein interaction network. *Molecular systems biology*, 5(1):260, 2009.

[13] Emmanuel Boutet, Damien Lieberherr, Michael Tognolli, Michel Schneider, Parit Bansal, Alan J Bridge, Sylvain Poux, Lydie Bougueleret, and Ioannis Xenarios. UniProtKB/Swiss-Prot, the manually annotated section of the UniProt KnowledgeBase: how to use the entry view. In *Plant Bioinformatics*, pages 23–54. Springer, 2016.

[14] Nicolas L Bray, Harold Pimentel, Páll Melsted, and Lior Pachter. Near-optimal probabilistic rna-seq quantification. *Nature biotechnology*, 34(5):525–527, 2016.

[15] Rainer Breitling, Anna Amtmann, and Pawel Herzyk. Iterative group analysis (iga): a simple tool to enhance sensitivity and facilitate interpretation of microarray experiments. *BMC bioinformatics*, 5(1):1–8, 2004.

[16] Philip Brennecke, Simon Anders, Jong Kyoung Kim, Aleksandra A Kołodziejczyk, Xiuwei Zhang, Valentina Proserpio, Bianka Baying, Vladimir Benes, Sarah A Teichmann, John C Marioni, et al. Accounting for technical noise in single-cell rna-seq experiments. *Nature methods*, 10(11):1093, 2013.

[17] David Brett, Heike Pospisil, Juan Valcárcel, Jens Reich, and Peer Bork. Alternative splicing and genome complexity. *Nature genetics*, 30(1):29–30, 2002.

[18] Lionel Breuza, Sylvain Poux, Anne Estreicher, Maria Livia Famiglietti, Michele Magrane, Michael Tognolli, Alan Bridge, Delphine Baratin, and Nicole Redaschi. The UniProtKB guide to the human proteome. *Database*, 2016, 2016.

[19] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. *Advances in neural information processing systems*, 6:737–744, 1993.

[20] Thang D. Bui, Sujith Ravi, and Vivek Ramavajjala. Neural graph learning: Training neural networks using graphs. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, WSDM '18, page 64–71, New York, NY, USA, 2018. Association for Computing Machinery.

[21] Kathi Canese and Sarah Weis. Pubmed: the bibliographic database. In *The NCBI Handbook [Internet]. 2nd edition.* National Center for Biotechnology Information (US), 2013.

[22] Horacio Caniza, Alfonso E Romero, Samuel Heron, Haixuan Yang, Alessandra Devoto, Marco Frasca, Marco Mesiti, Giorgio Valentini, and Alberto Paccanaro. GOssTo: a stand-alone application and a web tool for calculating semantic similarities on the Gene Ontology. *Bioinformatics*, 30(15):2235–2236, 2014.

[23] Zhi-Jie Cao, Lin Wei, Shen Lu, De-Chang Yang, and Ge Gao. Searching large-scale scrna-seq databases via unbiased cell embedding with cell blast. *Nature communications*, 11(1):1–13, 2020.

[24] Annie CY Chang, Björn Sohlberg, Laura Trinkle-Mulcahy, Felix Claverie-Martin, Philip Cohen, and Stanley N Cohen. Alternative splicing regulates the production of ARD-1 endoribonuclease and NIPP-1, an inhibitor of protein phosphatase-1, as isoforms encoded by the same gene. *Gene*, 240(1):45–55, 1999.

[25] Andrew Chatr-Aryamontri, Rose Oughtred, Lorrie Boucher, Jennifer Rust, Christie Chang, Nadine K Kolas, Lara O'Donnell, Sara Oster, Chandra Theesfeld, Adnane Sellam, et al. The biogrid interaction database: 2017 update. *Nucleic acids research*, 45(D1):D369–D379, 2017.

[26] Hao Chen, Dipan Shaw, Dongbo Bu, and Tao Jiang. Finer: enhancing the prediction of tissue-specific functions of isoforms by refining isoform interaction networks. *NAR Genomics and Bioinformatics*, 3(2):lqab057, 2021.

[27] Hao Chen, Dipan Shaw, Jianyang Zeng, Dongbo Bu, and Tao Jiang. Diffuse: predicting isoform functions from sequences and expression profiles via deep learning. *Bioinformatics*, 35(14):i284–i294, 2019.

[28] Muhao Chen, Chelsea J-T Ju, Guangyu Zhou, Xuelu Chen, Tianran Zhang, Kai-Wei Chang, Carlo Zaniolo, and Wei Wang. Multifaceted protein–protein interaction prediction based on siamese residual rcnn. *Bioinformatics*, 35(14):i305–i314, 2019.

[29] Ana Conesa, Stefan Götz, Juan Miguel García-Gómez, Javier Terol, Manuel Talón, and Montserrat Robles. Blast2go: a universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics*, 21(18):3674–3676, 2005.

[30] Gene Ontology Consortium. The Gene Ontology (GO) database and informatics resource. *Nucleic acids research*, 32(suppl_1):D258–D261, 2004.

[31] Hazel R Corradi, Sylva LU Schwager, Aloysius T Nchinda, Edward D Sturrock, and K Ravi Acharya. Crystal structure of the N domain of human somatic angiotensin I-converting enzyme provides a structural basis for domain-specific inhibitor design. *Journal of molecular biology*, 357(3):964–974, 2006.

[32] Pau Creixell, Jüri Reimand, Syed Haider, Guanming Wu, Tatsuhiro Shibata, Miguel Vazquez, Ville Mustonen, Abel Gonzalez-Perez, John Pearson, Chris Sander, et al. Pathway and network analysis of cancer genomes. *Nature methods*, 12(7):615, 2015.

[33] Xianying Amy Cui, Bhag Singh, Jae Park, and Radhey S Gupta. Subcellular localization of adenosine kinase in mammalian cells: The long isoform of AdK is localized in the nucleus. *Biochemical and biophysical research communications*, 388(1):46–50, 2009.

[34] D Davidson, LM Chow, M Fournel, and A Veillette. Differential regulation of t cell antigen responsiveness by isoforms of the src-related tyrosine protein kinase p59fyn. *The Journal of experimental medicine*, 175(6):1483–1492, 1992.

[35] Cécile Delettre, Victor J Yuste, Rana S Moubarak, Marlène Bras, Nadine Robert, and Santos A Susin. Identification and characterization of AIFsh2, a mitochondrial apoptosis-inducing factor (AIF) isoform with NADH oxidase activity. *Journal of Biological Chemistry*, 281(27):18507–18518, 2006.

[36] Glynn Dennis, Brad T Sherman, Douglas A Hosack, Jun Yang, Wei Gao, H Clifford Lane, and Richard A Lempicki. David: database for annotation, visualization, and integrated discovery. *Genome biology*, 4(9):1–11, 2003.

[37] Monica Dentice, Viviana Cordeddu, Annamaria Rosica, Alfonso Massimiliano Ferrara, Libero Santarpia, Domenico Salvatore, Luca Chiovato, Anna Perri, Lidia Moschini, Cristina Fazzini, et al. Missense mutation in the transcription factor nkx2–5: a novel molecular event in the pathogenesis of thyroid dysgenesis. *The Journal of Clinical Endocrinology & Metabolism*, 91(4):1428–1433, 2006.

[38] Pietro Di Lena, Piero Fariselli, Luciano Margara, Marco Vassura, and Rita Casadio. Fast overlapping of protein contact maps by alignment of eigenvectors. *Bioinformatics*, 26(18):2250–2258, 2010.

[39] Jingcheng Du, Peilin Jia, Yulin Dai, Cui Tao, Zhongming Zhao, and Degui Zhi. Gene2vec: distributed representation of genes based on co-expression. *BMC genomics*, 20(1):7–15, 2019.

[40] Ridvan Eksi, Hong-Dong Li, Rajasree Menon, Yuchen Wen, Gilbert S Omenn, Matthias Kretzler, and Yuanfang Guan. Systematically differentiating functions for alternatively spliced isoforms through integrating rna-seq data. *PLoS Comput Biol*, 9(11):e1003314, 2013.

[41] Sara El-Gebali, Jaina Mistry, Alex Bateman, Sean R Eddy, Aurélien Luciani, Simon C Potter, Matloob Qureshi, Lorna J Richardson, Gustavo A Salazar, Alfredo Smart, et al. The pfam protein families database in 2019. *Nucleic acids research*, 47(D1):D427–D432, 2019.

[42] Jonathan D Ellis, Miriam Barrios-Rodiles, Recep Çolak, Manuel Irimia, TaeHyung Kim, John A Calarco, Xinchen Wang, Qun Pan, Dave O'Hanlon, Philip M Kim, et al. Tissue-specific alternative splicing remodels protein-protein interaction networks. *Molecular cell*, 46(6):884–892, 2012.

[43] Evangelos Evangelou and John PA Ioannidis. Meta-analysis methods for genome-wide association studies and beyond. *Nature Reviews Genetics*, 14(6):379–389, 2013.

[44] Antonio Fabregat, Steven Jupe, Lisa Matthews, Konstantinos Sidiropoulos, Marc Gillespie, Phani Garapati, Robin Haw, Bijay Jassal, Florian Korninger, Bruce May, et al. The reactome pathway knowledgebase. *Nucleic acids research*, 46(D1):D649–D655, 2018.

[45] Linn Fagerberg, Björn M Hallström, Per Oksvold, Caroline Kampf, Dijana Djureinovic, Jacob Odeberg, Masato Habuka, Simin Tahmasebpoor, Angelika Danielsson, Karolina Edlund, et al. Analysis of the human tissue-specific expression by genome-wide integration of transcriptomics and antibody-based proteomics. *Molecular & Cellular Proteomics*, 13(2):397–406, 2014.

[46] Juan A Ferrer-Bonsoms, Ignacio Cassol, Pablo Fernández-Acín, Carlos Castilla, Fernando Carazo, and Angel Rubio. Isogo: Functional annotation of protein-coding splice variants. *Scientific reports*, 10(1):1–11, 2020.

[47] Nicolas O Fortunel, Hasan H Otu, Huck-Hui Ng, Jinhui Chen, Xiuqian Mu, Timothy Chevassut, Xiaoyu Li, Marie Joseph, Charles Bailey, Jacques A Hatzfeld, et al. Comment on"'stemness': transcriptional profiling of embryonic and adult stem cells" and" a stem cell molecular signature". *Science (New York, NY)*, 302(5644):393, 2003.

[48] Mohamed Ali Ghadie, Luke Lambourne, Marc Vidal, and Yu Xia. Domain-based prediction of the human isoform interactome provides insights into the functional impact of alternative splicing. *PLoS computational biology*, 13(8):e1005717, 2017.

[49] Madalina Giurgiu, Julian Reinhard, Barbara Brauner, Irmtraud Dunger-Kaltenbach, Gisela Fobo, Goar Frishman, Corinna Montrone, and Andreas Ruepp. Corum: the comprehensive resource of mammalian protein complexes—2019. *Nucleic acids research*, 47(D1):D559–D563, 2019.

[50] Casey S Greene, Arjun Krishnan, Aaron K Wong, Emanuela Ricciotti, Rene A Zelaya, Daniel S Himmelstein, Ran Zhang, Boris M Hartmann, Elena Zaslavsky, Stuart C Sealfon, et al. Understanding multicellular function and disease with human tissue-specific networks. *Nature genetics*, 47(6):569–576, 2015.

[51] Marion Gremse, Antje Chang, Ida Schomburg, Andreas Grote, Maurice Scheer, Christian Ebeling, and Dietmar Schomburg. The brenda tissue ontology (bto): the first all-integrating ontology of all organisms for enzyme sources. *Nucleic acids research*, 39(suppl_1):D507–D513, 2010.

[52] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[53] Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(2), 2012.

[54] Da-Cheng Hao, Guang-Bo Ge, Pei-Gen Xiao, Ping Wang, and Ling Yang. Drug metabolism and pharmacokinetic diversity of ranunculaceae medicinal compounds. *Current drug metabolism*, 16(4):294–321, 2015.

[55] Yuhan Hao, Stephanie Hao, Erica Andersen-Nissen, William M Mauck III, Shiwei Zheng, Andrew Butler, Maddie J Lee, Aaron J Wilk, Charlotte Darby, Michael Zager, et al. Integrated analysis of multimodal single-cell data. *Cell*, 2021.

[56] Ashraful Haque, Jessica Engel, Sarah A Teichmann, and Tapio Lönnberg. A practical guide to single-cell rna-sequencing for biomedical research and clinical applications. *Genome medicine*, 9(1):1–12, 2017.

[57] Somaye Hashemifar, Behnam Neyshabur, Aly A Khan, and Jinbo Xu. Predicting protein–protein interactions through sequence-based deep learning. *Bioinformatics*, 34(17):i802–i810, 2018.

[58] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *european conference on computer vision*, pages 346–361. Springer, 2014.

[59] Yi He, Jiayuan Shi, Chuan Wang, Haibin Huang, Jiaming Liu, Guanbin Li, Risheng Liu, and Jue Wang. Semi-supervised skin detection by network with mutual guidance. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2111–2120, 2019.

[60] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[61] Da Wei Huang, Brad T Sherman, Qina Tan, Joseph Kir, David Liu, David Bryant, Yongjian Guo, Robert Stephens, Michael W Baseler, H Clifford Lane, et al. David bioinformatics resources: expanded annotation database and novel algorithms to better extract biology from large gene lists. *Nucleic acids research*, 35(suppl_2):W169–W175, 2007.

[62] Jaime Huerta-Cepas, Damian Szklarczyk, Kristoffer Forslund, Helen Cook, Davide Heller, Mathias C Walter, Thomas Rattei, Daniel R Mende, Shinichi Sunagawa, Michael Kuhn, et al. eggNOG 4.5: a hierarchical orthology framework with improved functional annotations for eukaryotic, prokaryotic and viral sequences. *Nucleic acids research*, 44(D1):D286–D293, 2015.

[63] Jaime Huerta-Cepas, Damian Szklarczyk, Davide Heller, Ana Hernández-Plaza, Sofia K Forslund, Helen Cook, Daniel R Mende, Ivica Letunic, Thomas Rattei, Lars J Jensen, et al. eggnog 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. *Nucleic acids research*, 47(D1):D309–D314, 2019.

[64] Rachael P Huntley, Tony Sawford, Prudence Mutowo-Meullenet, Aleksandra Shypitsyna, Carlos Bonilla, Maria J Martin, and Claire O'donovan. The GOA database: gene ontology annotation updates for 2015. *Nucleic acids research*, 43(D1):D1057–D1063, 2014.

[65] Rachael P Huntley, Tony Sawford, Prudence Mutowo-Meullenet, Aleksandra Shypitsyna, Carlos Bonilla, Maria J Martin, and Claire O'Donovan. The goa database: gene ontology annotation updates for 2015. *Nucleic acids research*, 43(D1):D1057–D1063, 2015.

[66] Edward L Huttlin, Raphael J Bruckner, Joao A Paulo, Joe R Cannon, Lily Ting, Kurt Baltier, Greg Colby, Fana Gebreab, Melanie P Gygi, Hannah Parzen, et al. Architecture of the human interactome defines protein communities and disease networks. *Nature*, 545(7655):505–509, 2017.

[67] Trey Ideker and Roded Sharan. Protein networks in disease. *Genome research*, 18(4):644–652, 2008.

[68] Kristoffer Illergård, David H Ardell, and Arne Elofsson. Structure is three to ten times more conserved than sequence—a study of structural response in protein cores. *Proteins: Structure, Function, and Bioinformatics*, 77(3):499–508, 2009.

[69] Briony HA Jack and Merlin Crossley. Gata proteins work together with friend of gata (fog) and c-terminal binding protein (ctbp) co-regulators to control adipogenesis. *Journal of Biological Chemistry*, 285(42):32405–32414, 2010.

[70] Jyun-Yu Jiang, Francine Chen, Yan-Ying Chen, and Wei Wang. Learning to disentangle interleaved conversational threads with a siamese hierarchical network and similarity ranking. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1812–1822, 2018.

[71] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, pages 1–11, 2021.

[72] Gaurav Kandoi and Julie A Dickerson. Tissue-specific mouse mrna isoform networks. *Scientific reports*, 9(1):1–24, 2019.

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

[74] T tempspacetempspaceS Keshava Prasad, Renu Goel, Kumaran Kandasamy, Shivakumar Keerthikumar, Sameer Kumar, Suresh Mathivanan, Deepthi Telikicherla, Rajesh Raju, Beema Shafreen, Abhilash Venugopal, et al. Human protein reference database—2009 update. *Nucleic acids research*, 37(suppl_1):D767–D772, 2009.

[75] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, et al. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

[76] Eddo Kim, Amir Goren, and Gil Ast. Alternative splicing: current perspectives. *Bioessays*, 30(1):38–47, 2008.

[77] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[78] Kotikalapudi, Raghavendra and contributors. keras-vis. https://github.com/raghakot/keras-vis, 2017.

[79] Max Kotlyar, Chiara Pastrello, Zara Malik, and Igor Jurisica. Iid 2018 update: context-specific physical protein–protein interactions in human, model organisms and domesticated species. *Nucleic acids research*, 47(D1):D581–D589, 2019.

[80] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[81] Brent M Kuenzi, Jisoo Park, Samson H Fong, Kyle S Sanchez, John Lee, Jason F Kreisberg, Jianzhu Ma, and Trey Ideker. Predicting drug response and synergy using a deep learning model of human cancer cells. *Cancer cell*, 38(5):672–684, 2020.

[82] Maxim V Kuleshov, Matthew R Jones, Andrew D Rouillard, Nicolas F Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L Jenkins, Kathleen M Jagodnik, Alexander Lachmann, et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic acids research*, 44(W1):W90–W97, 2016.

[83] Maxat Kulmanov, Mohammed Asif Khan, and Robert Hoehndorf. DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics*, 2017.

[84] Maxat Kulmanov, Mohammed Asif Khan, and Robert Hoehndorf. Deepgo: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics*, 34(4):660–668, 2018.

[85] Sarah K Kummerfeld and Sarah A Teichmann. Protein domain organisation: adding order. *BMC bioinformatics*, 10(1):39, 2009.

[86] Alexander Lachmann, Denis Torre, Alexandra B Keenan, Kathleen M Jagodnik, Hoyjin J Lee, Lily Wang, Moshe C Silverstein, and Avi Ma'ayan. Massive mining of publicly available rna-seq data from human and mouse. *Nature communications*, 9(1):1–10, 2018.

[87] Justin Lamb. The connectivity map: a new tool for biomedical research. *Nature reviews cancer*, 7(1):54–60, 2007.

[88] Jack Lanchantin, Ritambhara Singh, Beilun Wang, and Yanjun Qi. Deep motif dashboard: Visualizing and understanding genomic sequences using deep neural networks. In *PACIFIC SYMPOSIUM ON BIOCOMPUTING 2017*, pages 254–265. World Scientific, 2017.

[89] Peter Langfelder and Steve Horvath. WGCNA: an R package for weighted correlation network analysis. *BMC bioinformatics*, 9(1):559, 2008.

[90] Ben Langmead and Steven L Salzberg. Fast gapped-read alignment with bowtie 2. *Nature methods*, 9(4):357–359, 2012.

[91] Rasko Leinonen, Hideaki Sugawara, Martin Shumway, and International Nucleotide Sequence Database Collaboration. The sequence read archive. *Nucleic acids research*, 39(suppl_1):D19–D21, 2010.

[92] Jacob H Levine, Erin F Simonds, Sean C Bendall, Kara L Davis, D Amir El-ad, Michelle D Tadmor, Oren Litvin, Harris G Fienberg, Astraea Jager, Eli R Zunder, et al. Data-driven phenotypic dissection of aml reveals progenitor-like cells that correlate with prognosis. *Cell*, 162(1):184–197, 2015.

[93] Bo Li and Colin N Dewey. Rsem: accurate transcript quantification from rna-seq data with or without a reference genome. *BMC bioinformatics*, 12(1):1–16, 2011.

[94] Hong-Dong Li, Rajasree Menon, Brandon Govindarajoo, Bharat Panwar, Yang Zhang, Gilbert S Omenn, and Yuanfang Guan. Functional networks of highest-connected splice isoforms: from the chromosome 17 human proteome project. *Journal of proteome research*, 14(9):3484–3491, 2015.

[95] Hong-Dong Li, Rajasree Menon, Gilbert S Omenn, and Yuanfang Guan. The emerging era of genomic data integration for analyzing splice isoform function. *Trends in Genetics*, 30(8):340–347, 2014.

[96] Hong-Dong Li, Rajasree Menon, Gilbert S Omenn, and Yuanfang Guan. Revisiting the identification of canonical splice isoforms through integration of functional genomics and proteomics evidence. *Proteomics*, 14(23-24):2709–2718, 2014.

[97] Hong-Dong Li, Gilbert S Omenn, and Yuanfang Guan. A proteogenomic approach to understand splice isoform functions through sequence and expression-based computational modeling. *Briefings in bioinformatics*, page bbv109, 2016.

[98] Hong-Dong Li, Changhuo Yang, Zhimin Zhang, Mengyun Yang, Fang-Xiang Wu, Gilbert S Omenn, and Jianxin Wang. Isoresolve: predicting splice isoform functions by integrating gene and isoform-level features with domain adaptation. *Bioinformatics*, 2020.

[99] Wenyuan Li, Shuli Kang, Chun-Chi Liu, Shihua Zhang, Yi Shi, Yan Liu, and Xianghong Jasmine Zhou. High-resolution functional annotation of human transcriptome: predicting isoform functions by a novel multiple instance-based label propagation method. *Nucleic acids research*, 42(6):e39–e39, 2013.

[100] Wenyuan Li, Shuli Kang, Chun-Chi Liu, Shihua Zhang, Yi Shi, Yan Liu, and Xianghong Jasmine Zhou. High-resolution functional annotation of human transcriptome: predicting isoform functions by a novel multiple instance-based label propagation method. *Nucleic acids research*, 42(6):e39–e39, 2014.

[101] Wenyuan Li, Chun-Chi Liu, Tong Zhang, Haifeng Li, Michael S Waterman, and Xianghong Jasmine Zhou. Integrative analysis of many weighted co-expression networks using tensor computation. *PLoS Comput Biol*, 7(6):e1001106, 2011.

[102] Tingjin Luo, Weizhong Zhang, Shang Qiu, Yang Yang, Dongyun Yi, Guangtao Wang, Jieping Ye, and Jie Wang. Functional annotation of human protein coding isoforms via non-convex multi-instance learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 345–354, 2017.

[103] Jianzhu Ma, Michael Ku Yu, Samson Fong, Keiichiro Ono, Eric Sage, Barry Demchak, Roded Sharan, and Trey Ideker. Using deep learning to model the hierarchical structure and function of a cell. *Nature methods*, 15(4):290–298, 2018.

[104] I Makarenko, CA Opitz, MC Leake, C Neagoe, M Kulke, JK Gwathmey, F Del Monte, RJ Hajjar, and WA Linke. Passive stiffness changes caused by upregulation of compliant titin isoforms in human dilated cardiomyopathy hearts. *Circulation research*, 95(7):708–716, 2004.

[105] Aron Marchler-Bauer, Myra K Derbyshire, Noreen R Gonzales, Shennan Lu, Farideh Chitsaz, Lewis Y Geer, Renata C Geer, Jane He, Marc Gwadz, David I Hurwitz, et al. CDD: NCBI's conserved domain database. *Nucleic acids research*, 43(D1):D222–D226, 2014.

[106] Aron Marchler-Bauer, Myra K Derbyshire, Noreen R Gonzales, Shennan Lu, Farideh Chitsaz, Lewis Y Geer, Renata C Geer, Jane He, Marc Gwadz, David I Hurwitz, et al. Cdd: Ncbi's conserved domain database. *Nucleic acids research*, 43(D1):D222–D226, 2015.

[107] Arianne J Matlin, Francis Clark, and Christopher WJ Smith. Understanding alternative splicing: towards a cellular code. *Nature reviews Molecular cell biology*, 6(5):386–398, 2005.

[108] Jörg Menche, Amitabh Sharma, Maksim Kitsak, Susan Dina Ghiassian, Marc Vidal, Joseph Loscalzo, and Albert-László Barabási. Uncovering disease-disease relationships through the incomplete interactome. *Science*, 347(6224), 2015.

[109] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[110] Sara Mostafavi, Debajyoti Ray, David Warde-Farley, Chris Grouios, and Quaid Morris. GeneMANIA: a real-time multiple association network integration algorithm for predicting gene function. *Genome biology*, 9(1):S4, 2008.

[111] Elisabetta Mueller, Stavit Drori, Anita Aiyer, Junming Yie, Pasha Sarraf, Hong Chen, Stefanie Hauser, Evan D Rosen, Kai Ge, Robert G Roeder, et al. Genetic analysis of adipogenesis through peroxisome proliferator-activated receptor γ isoforms. *Journal of Biological Chemistry*, 277(44):41925–41930, 2002.

[112] Javier Munoz and Albert JR Heck. From the human genome to the human proteome. *Angewandte Chemie International Edition*, 53(41):10864–10866, 2014.

[113] Timothy W Nilsen and Brenton R Graveley. Expansion of the eukaryotic proteome by alternative splicing. *Nature*, 463(7280):457–463, 2010.

[114] Sandra Orchard, Mais Ammari, Bruno Aranda, Lionel Breuza, Leonardo Briganti, Fiona Broackes-Carter, Nancy H Campbell, Gayatri Chavali, Carol Chen, Noemi Del-Toro, et al. The mintact project—intact as a common curation platform for 11 molecular interaction databases. *Nucleic acids research*, 42(D1):D358–D363, 2014.

[115] Nicolas A Pabon, Yan Xia, Samuel K Estabrooks, Zhaofeng Ye, Amanda K Herbrand, Evelyn Süß, Ricardo M Biondi, Victoria A Assimon, Jason E Gestwicki, Jeffrey L Brodsky, et al. Predicting protein targets for drug-like compounds using transcriptomics. *PLoS computational biology*, 14(12):e1006651, 2018.

[116] Qun Pan, Ofer Shai, Leo J Lee, Brendan J Frey, and Benjamin J Blencowe. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing. *Nature genetics*, 40(12):1413–1415, 2008.

[117] Jian Peng and Jinbo Xu. RaptorX: exploiting structure information for protein alignment by statistical inference. *Proteins: Structure, Function, and Bioinformatics*, 79(S10):161–171, 2011.

[118] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[119] Kim D Pruitt, Tatiana Tatusova, Garth R Brown, and Donna R Maglott. NCBI Reference Sequences (RefSeq): current status, new features and genome annotation policy. *Nucleic acids research*, 40(D1):D130–D135, 2011.

[120] Kim D Pruitt, Tatiana Tatusova, and Donna R Maglott. Ncbi reference sequences (refseq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic acids research*, 35(suppl_1):D61–D65, 2007.

[121] Lisa Pucci, Silvia Perozzi, Flavio Cimadamore, Giuseppe Orsomando, and Nadia Raffaelli. Tissue expression and biochemical characterization of human 2-amino 3-carboxymuconate 6-semialdehyde decarboxylase, a key enzyme in tryptophan catabolism. *The FEBS journal*, 274(3):827–840, 2007.

[122] Ashkaun Razmara, Shannon E Ellis, Dustin J Sokolowski, Sean Davis, Michael D Wilson, Jeffrey T Leek, Andrew E Jaffe, and Leonardo Collado-Torres. recountbrain: a curated repository of human brain rna-seq datasets metadata. *BioRxiv*, page 618025, 2019.

[123] Delin Ren, Trevor N Collingwood, Edward J Rebar, Alan P Wolffe, and Heidi S Camp. Pparγ knockdown by engineered transcription factors: exogenous pparγ2 but not pparγ1 reactivates adipogenesis. *Genes & development*, 16(1):27–32, 2002.

[124] Anthony Rhoads and Kin Fai Au. Pacbio sequencing and its applications. *Genomics, proteomics & bioinformatics*, 13(5):278–289, 2015.

[125] Thomas Rolland, Murat Taşan, Benoit Charloteaux, Samuel J Pevzner, Quan Zhong, Nidhi Sahni, Song Yi, Irma Lemmens, Celia Fontanillo, Roberto Mosca, et al. A proteome-scale map of the human interactome network. *Cell*, 159(5):1212–1226, 2014.

[126] Swarup Roy, Dhruba K Bhattacharyya, and Jugal K Kalita. Reconstruction of gene co-expression network from microarray data using local expression patterns. *BMC bioinformatics*, 15(S7):S10, 2014.

[127] Camilo Ruiz, Marinka Zitnik, and Jure Leskovec. Identification of disease treatment mechanisms through the multiscale interactome. *Nature communications*, 12(1):1–15, 2021.

[128] Rahul Satija, Jeffrey A Farrell, David Gennert, Alexander F Schier, and Aviv Regev. Spatial reconstruction of single-cell gene expression data. *Nature biotechnology*, 33(5):495–502, 2015.

[129] Dipan Shaw, Hao Chen, and Tao Jiang. DeepIsoFun: a deep domain adaptation approach to predict isoform functions. *Bioinformatics*, page bty1017, 2018.

[130] Dipan Shaw, Hao Chen, and Tao Jiang. Deepisofun: a deep domain adaptation approach to predict isoform functions. *Bioinformatics*, 35(15):2535–2544, 2019.

[131] Kana Shimizu, Jun Adachi, and Yoichi Muraoka. Angle: a sequencing errors resistant program for predicting protein coding regions in unfinished cdna. *Journal of Bioinformatics and Computational Biology*, 4(03):649–664, 2006.

[132] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[133] Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Opa2vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics*, 35(12):2133–2140, 2019.

[134] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[135] Stefan Stamm, Shani Ben-Ari, Ilona Rafalska, Yesheng Tang, Zhaiyi Zhang, Debra Toiber, TA Thanaraj, and Hermona Soreq. Function of alternative splicing. *Gene*, 344:1–20, 2005.

[136] Aravind Subramanian, Rajiv Narayan, Steven M Corsello, David D Peck, Ted E Natoli, Xiaodong Lu, Joshua Gould, John F Davis, Andrew A Tubelli, Jacob K Asiedu, et al. A next generation connectivity map: L1000 platform and the first 1,000,000 profiles. *Cell*, 171(6):1437–1452, 2017.

[137] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

[138] Dinanath Sulakhe, Mark D'Souza, Sheng Wang, Sandhya Balasubramanian, Prashanth Athri, Bingqing Xie, Stefan Canzar, Gady Agam, T Conrad Gilliam, and Natalia Maltsev. Exploring the functional impact of alternative splicing on human protein isoforms using available annotation sources. *Briefings in bioinformatics*, 2018.

[139] Charles Sutton, Andrew McCallum, et al. An introduction to conditional random fields. *Foundations and Trends® in Machine Learning*, 4(4):267–373, 2012.

[140] Bahar Taneri, Ben Snyder, Alexey Novoradovsky, and Terry Gaasterland. Alternative splicing of mouse transcription factors affects their dna-binding domain architecture and is tissue specific. *Genome Biology*, 5(10):R75, 2004.

[141] Jian Tang, Meng Qu, and Qiaozhu Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1165–1174, 2015.

[142] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077, 2015.

[143] Roman L Tatusov, Michael Y Galperin, Darren A Natale, and Eugene V Koonin. The COG database: a tool for genome-scale analysis of protein functions and evolution. *Nucleic acids research*, 28(1):33–36, 2000.

[144] Peter J Thul, Lovisa Åkesson, Mikaela Wiking, Diana Mahdessian, Aikaterini Geladaki, Hammou Ait Blal, Tove Alm, Anna Asplund, Lars Björk, Lisa M Breckels, et al. A subcellular map of the human proteome. *Science*, 356(6340), 2017.

[145] Masahide Tone, Yukiko Tone, Paul J Fairchild, Michelle Wykes, and Herman Waldmann. Regulation of cd40 function by its isoforms generated through alternative splicing. *Proceedings of the National Academy of Sciences*, 98(4):1751–1756, 2001.

[146] Shashank Tripathi, Marie O Pohl, Yingyao Zhou, Ariel Rodriguez-Frandsen, Guojun Wang, David A Stein, Hong M Moulton, Paul DeJesus, Jianwei Che, Lubbertus CF Mulder, et al. Meta-and orthogonal integration of influenza "omics" data defines a role for ubr4 in virus budding. *Cell host & microbe*, 18(6):723–735, 2015.

[147] Yu-Ting Tseng, Wenyuan Li, Ching-Hsien Chen, Shihua Zhang, Jeremy JW Chen, Xianghong Jasmine Zhou, and Chun-Chi Liu. Iiidb: a database for isoform-isoform interactions and isoform network modules. In *BMC genomics*, volume 16, page S10. Springer, 2015.

[148] Mathias Uhlén, Linn Fagerberg, Björn M Hallström, Cecilia Lindskog, Per Oksvold, Adil Mardinoglu, Åsa Sivertsson, Caroline Kampf, Evelina Sjöstedt, Anna Asplund, et al. Tissue-based map of the human proteome. *Science*, 347(6220), 2015.

[149] Laura M Urbanski, Nathan Leclair, and Olga Anczuków. Alternative-splicing defects in cancer: splicing regulators and their downstream targets, guiding the way to novel cancer therapeutics. *Wiley Interdisciplinary Reviews: RNA*, 9(4):e1476, 2018.

[150] Alexei Vazquez, Alessandro Flammini, Amos Maritan, and Alessandro Vespignani. Global protein function prediction from protein-protein interaction networks. *Nature biotechnology*, 21(6):697–700, 2003.

[151] Debora Vom Endt, Jan W Kijne, and Johan Memelink. Transcription factors controlling plant secondary metabolism: what regulates the regulators? *Phytochemistry*, 61(2):107–114, 2002.

[152] Bo Wang, Armin Pourshafeie, Marinka Zitnik, Junjie Zhu, Carlos D Bustamante, Serafim Batzoglou, and Jure Leskovec. Network enhancement as a general method to denoise weighted biological networks. *Nature communications*, 9(1):1–8, 2018.

[153] Dongxue Wang, Basak Eraslan, Thomas Wieland, Björn Hallström, Thomas Hopf, Daniel Paul Zolg, Jana Zecha, Anna Asplund, Li-hua Li, Chen Meng, et al. A deep proteome and transcriptome abundance atlas of 29 healthy human tissues. *Molecular systems biology*, 15(2):e8503, 2019.

[154] Eric T Wang, Rickard Sandberg, Shujun Luo, Irina Khrebtukova, Lu Zhang, Christine Mayr, Stephen F Kingsmore, Gary P Schroth, and Christopher B Burge. Alternative isoform regulation in human tissue transcriptomes. *Nature*, 456(7221):470–476, 2008.

[155] Keyao Wang, Jun Wang, Carlotta Domeniconi, Xiangliang Zhang, and Guoxian Yu. Differentiating isoform functions with collaborative matrix factorization. *Bioinformatics*, 36(6):1864–1871, 2020.

[156] Sheng Wang, Hyunghoon Cho, ChengXiang Zhai, Bonnie Berger, and Jian Peng. Exploiting ontology graph for predicting sparsely annotated gene function. *Bioinformatics*, 31(12):i357–i364, 2015.

[157] Sheng Wang, Siqi Sun, Zhen Li, Renyu Zhang, and Jinbo Xu. Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLoS computational biology*, 13(1):e1005324, 2017.

[158] Dagmar Wilhelm and Christoph Englert. The wilms tumor suppressor wt1 regulates early gonad development by activation of sf1. *Genes & development*, 16(14):1839–1851, 2002.

[159] Ei-Wen Yang and Tao Jiang. SDEAP: a splice graph based differential transcript expression analysis tool for population data. *Bioinformatics*, 32(23):3593–3602, 2016.

[160] Lin Yang, Yuqing Zhu, Hua Yu, Xiaolong Cheng, Sitong Chen, Yulan Chu, He Huang, Jin Zhang, and Wei Li. scmageck links genotypes with multiple phenotypes in single-cell crispr screens. *Genome biology*, 21(1):1–14, 2020.

[161] Xinping Yang, Jasmin Coulombe-Huntington, Shuli Kang, Gloria M Sheynkman, Tong Hao, Aaron Richardson, Song Sun, Fan Yang, Yun A Shen, Ryan R Murray, et al. Widespread expansion of protein interaction capabilities by alternative splicing. *Cell*, 164(4):805–817, 2016.

[162] Esti Yeger-Lotem and Roded Sharan. Human protein interaction networks across tissues and diseases. *Frontiers in genetics*, 6:257, 2015.

[163] Guoxian Yu, Keyao Wang, Carlotta Domeniconi, Maozu Guo, and Jun Wang. Isoform function prediction based on bi-random walks on a heterogeneous network. *Bioinformatics*, 36(1):303–310, 2020.

[164] Nancy Y Yu, James R Wagner, Matthew R Laird, Gabor Melli, Sébastien Rey, Raymond Lo, Phuong Dao, S Cenk Sahinalp, Martin Ester, Leonard J Foster, et al. Psortb 3.0: improved protein subcellular localization prediction with refined localization subcategories and predictive capabilities for all prokaryotes. *Bioinformatics*, 26(13):1608–1615, 2010.

[165] Jie Zeng, Guoxian Yu, Jun Wang, Maozu Guo, and Xiangliang Zhang. Dmil-iii: Isoform-isoform interaction prediction using deep multi-instance learning method. In *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 171–176. IEEE, 2019.

[166] Chiyuan Zhang, Samy Bengio, Moritz Hardt, et al. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

[167] Sai Zhang, Hailin Hu, Tao Jiang, Lei Zhang, and Jianyang Zeng. TITER: predicting translation initiation sites by deep learning. *Bioinformatics*, 33(14):i234–i242, 2017.

[168] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1529–1537, 2015.

[169] Yi Zheng, Chen Jiao, Honghe Sun, Hernán Rosli, Marina Alejandra Pombo, Peifen Zhang, Michael Banf, Xinbin Dai, Gregory B Martin, James J Giovannoni, et al. itak: a program for genome-wide prediction and classification of plant transcription factors, transcriptional regulators, and protein kinases. *Molecular plant*, 9, 2016.

[170] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning with hypergraphs: Clustering, classification, and embedding. *Advances in neural information processing systems*, 19:1601–1608, 2006.

[171] Yingyao Zhou, Bin Zhou, Lars Pache, Max Chang, Alireza Hadj Khodabakhshi, Olga Tanaseichuk, Christopher Benner, and Sumit K Chanda. Metascape provides a biologist-oriented resource for the analysis of systems-level datasets. *Nature communications*, 10(1):1–10, 2019.

[172] Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560, 1997.

[173] Marinka Zitnik and Jure Leskovec. Predicting multicellular function through multi-layer tissue networks. *Bioinformatics*, 33(14):i190–i198, 2017.