Data and text mining

DMIL-IsoFun: predicting isoform function using deep multi-instance learning

Guoxian Yu¹,²,³⁺, Guangjie Zhou¹,², Xiangliang Zhang³, Carlotta Domeniconi⁴ and Maozu Guo⁵⁺

¹School of Software, Shandong University, Jinan 250101, China.
²College of Computer and Information Sciences, Southwest University, Chongqing 400715, China.
³Computer, Electrical, and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology, SA.
⁴Department of Computer Science, George Mason University, Fairfax 22030, USA.
⁵School of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture, Beijing, China.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Alternative splicing creates the considerable proteomic diversity and complexity on relatively limited genome. Proteoforms translated from alternatively spliced isoforms of a gene actually execute the biological functions of this gene, which reflect the functional knowledge of genes at a finer granular level. Recently, some computational approaches have been proposed to differentiate isoform functions using sequence and expression data. However, their performance is far from being desirable, mainly due to the imbalance and lack of annotations at isoform-level, and the difficulty of modeling gene-isoform relations.

Result: We propose a deep multi-instance learning based framework (DMIL-IsoFun) to differentiate the functions of isoforms. DMIL-IsoFun firstly introduces a multi-instance learning convolution neural network trained with isoform sequences and gene-level annotations to extract the feature vectors and initialize the annotations of isoforms, and then uses a class-imbalance Graph Convolution Network to refine the annotations of individual isoforms based on the isoform co-expression network and extracted features. Extensive experimental results show that DMIL-IsoFun improves the $S_{\text{min}}$ and $F_{\text{max}}$ of state-of-the-art solutions by at least 29.6% and 40.8%. The effectiveness of DMIL-IsoFun is further confirmed on a testbed of human multiple-isoform genes, and Maize isoforms related with photosynthesis.


Contact: guomaozu@bucea.edu.cn (M. Guo) and guoxian85@gmail.com (G. Yu)

Supplementary information: Supplementary data are available at Bioinformatics online.
have a distinct impact on protein structure and function. For example, two isoforms (SERCA2a and SERCA2b) are differentially expressed during muscle differentiation via a splice process at the 3’ end of the primary SERCA2 transcript. Replacement of SERCA2 with SERCA2a in a mouse model results in mild hypertrophy and impaired contraction-relaxation in the heart (Ver Heyen et al., 2001). In contrast, over-expressed SERCA2b shows an increased cardiac SERCA activity and contractility (Greene et al., 2000). Therefore, the individual functional annotations of isoforms are crucial to decipher the functional complexity of genomes.

Gene Ontology (GO) is a major bioinformatics initiative that unifies the representation of gene and gene products functional attributes of multiple species (Dessimoz and Škunca, 2017). GO has been extensively used as a golden standard to encode the function of gene products; it includes plenty of GO terms and each GO term describes a distinct biological concept. GO includes three subontologies: Molecular Function (MF), Biological Process (BP), and Cellular Component (CC). Each subontology structurally organizes GO terms as a direct acyclic graph (DAG). In the DAG, each node corresponds to a GO term and the edge describes the relationship between terms. The prediction of GO annotations of genes has been studied for many years and diverse computational methods have been developed (Zhou et al., 2019; Zhao et al., 2020). Existing solutions mainly focus on the gene-level and collectively assign all functions of gene products to the same gene; thus, typical gene function prediction methods cannot be directly adopted to differentiate functions of isoforms.

Some pioneers model the isoform function prediction as a multi-instance learning (MIL) problem (Zhou et al., 2012), where each bag (gene) is composed of a set of instances (isoforms), and the labels of a bag are induced from the label of its instances. This setting aligns with the convention of GO annotations of genes, in which a gene is positive for a GO term if at least one of its isoforms is positively annotated with that GO term; on the other hand, a gene is negative for a GO term if all its isoforms are not annotated with that GO term. Several MIL-based solutions have been developed to predict isoform functions using GO and RNA-seq data. Eksi et al. (2013) applied an isoform-level maximum margin classifier (miSVM) using GO annotations of genes, gene-isoform relation, and isoform expression data to predict isoform functions. However, miSVM is seriously affected by the initial annotations of isoforms from positive genes and the preset threshold. JMILP (Li et al., 2013), a multiple instance-based label propagation method, predicts annotations of isoforms by constructing the isoform function association network and by uniformly initializing the annotations of isoforms using GO annotations of genes. Then, it iteratively normalizes and propagates annotations in the association network to associate isoforms with GO terms. WLMR (Luo et al., 2017) differentiates the functions of isoforms by integrating sparse simplex projection into a nonconvex sparsity-induced regularizer within the MIL framework, and efficiently solves the highly nontrivial non-convex and non-smooth optimization problem in miSVM.

The aforementioned methods implicitly assume GO annotations of genes are complete. Yu et al. (2020) introduced an approach (IsoFun) that performs bi-random walks on a heterogeneous network composed with genes, isoforms and GO terms to replenish the missing GO annotations of genes and dispatch them to individual isoforms. Wang et al. (2020) assumed that the annotations of genes are aggregated from key isoforms and proposed DissoFun, which collaboratively factorizes the isoform expression data matrix and gene-term association matrix into low-rank matrices to discover the latent key isoforms and identify their functions. All the above methods can be regarded as shallow solutions, as they cannot mine nonlinear relationships between isoforms and GO terms. DeepIsoFun (Shaw et al., 2018) uses auto-encoders to extract features from expression data, and combines MIL with domain adaption to provide extra annotated data, and thus to more credibly transfer the annotations to isoforms. DIFFUSE (Chen et al., 2019) uses a convolutional neural network to predict the functions of isoforms from the genomic sequence, and then applies a conditional random field (CRF) to refine the prediction based on the isoform co-expression network. However, these two deep methods do not take into account the important relationship between a gene (bag) and its isoforms (instances) during training, and thus suffer from inconsistent predictions for genes and corresponding isoforms.

In this paper, we propose a deep multi-instance learning based solution called DMIL-IsoFun to predict isoform functions by leveraging RNA-seq datasets, isoform sequences and GO annotations. The main idea of DMIL-IsoFun and its workflow are illustrated in Fig. 1. DMIL-IsoFun introduces a Multi-Instance Learning Convolution Neural Network (MILCNN) to transfer the gene-level annotations to isoforms by max-pooling, and then proposes a class-imbalance Graph Convolution Network (GCN) (Kipf and Welling, 2016) to refine the transferred annotations of individual isoforms. Particularly, MILCNN takes the gene bag composed with isoform sequences as input and uses the 1D convolution kernel to scan isoforms of a gene bag, along with two maximum pooling layers, one to get the isoform-level final annotations, and the other to aggregate the initial annotations to the hosting gene bag, and thus learns the representational features of isoforms and initializes the annotations of isoforms by gene-level annotations in a concordant way. The class-imbalance GCN takes the focal loss (Lan et al. (2020)), instead of cross-entropy loss, to handle imbalance annotations, and fuses the representational features and the isoform co-expression network induced from multiple RNA-Seq datasets to further differentiate the annotations of individual isoforms. We conducted experiments on Maize and Human datasets from public repositories, and found that MIL-IsoFun can more credibly differentiate GO annotations of isoforms than other related and competitive approaches (Eksi et al., 2013; Li et al., 2013; Shaw et al., 2018; Chen et al., 2019; Yu et al., 2020; Wang et al., 2020). Further experiments confirm that DMIL-IsoFun can accurately identify the functions of isoforms spliced from multiple isoform genes (MIGs) of Human, and Maize isoforms related with photosynthesis.

2 System and methods

2.1 Overview and Formulation

DMIL-IsoFun predicts isoform functions by deep multi-instance learning and graph convolution network to integrate isoform sequence and expression data, and gene-level annotations. Fig. 1 shows the basic workflow of our model. First, the spliced isoform sequences of a gene form the feature vectors of a gene bag. In the MILCNN stage, the 1D convolution kernel is used to extract the features for each isoform instance in the bag, and then two maximum pooling layers are used to aggregate the predicted functions of isoforms to the hosting gene and to reversely back propagate known annotations of this gene to its isoforms, respectively. The aggregation and back propagation strategy is motivated by the observation that the collected GO annotations of a gene are the union of annotations of individual isoforms spliced from this gene (Dessimoz and Škunca, 2017). These predicted annotations are taken as the initial annotations of isoforms for the follow-up fine-tuning. In the GCN stage, GCN fuses the isoform sequences and co-expression network to further refine the GO annotations of individual isoforms.

We formulate the isoform function prediction as a multi-instance learning task. Let \( n \) represent the number of isoforms, \( m \) denote the number of genes, and \( T \) denote the set of studied GO terms. A gene and its annotations are given as pairs \( \{ x_i, y_i \} \), where \( x_i = \{ x_{i,1}, x_{i,2}, \ldots, x_{i,T} \} \), \( T \) is the maximum number of isoforms in a gene, which is set to 20 in our experiments based on statistics of isoforms spliced from the same gene, as reported in Section 5 of the Supplementary file.
A gene with more than $\tau$ isoforms keeps the first $\tau$ isoforms, and a gene with fewer than $\tau$ isoforms is zero-padded to generate the gene bag feature matrix $X_i \in \mathbb{R}^{n \times d}$ is the representation feature vector of the $j$-th isoform of the $i$-th gene. $Y_i \subset T$ stores the GO annotations of this gene, $Z_{ij} \subset \bigcup_i Y_i$ (where $Z_{ij}$ stores the GO annotations of the $j$-th isoform spliced from the $i$-th gene, but is typically unknown).

2.2 Datasets

Amino acid sequences have been used extensively for predicting protein functions (Zhou et al., 2019), and have recently been used for isoform function prediction as well (Chen et al., 2019). Unlike the canonically studied mammal genomes (i.e., Human and Mouse), which have very scarce direct GO annotations of isoforms, the Maize B73 v5 genome assembly data provides a number of isoform-level GO annotations, and can serve as a natural testbed for isoform function prediction. As such, we perform our experiments using primarily the Maize data. We downloaded the B73 v5 genome assembly data generated by the NAM project from MaizeGDB. Coding DNA Sequence (CDS) is extracted for each isoform through the RefSeq CDS annotation file. The combination of different CDSs of a gene produces a variety of isoforms. The codon table is used to convert the nucleotide sequence into a protein sequence. Isoform expression data of Maize genome were downloaded from NCBI Sequence Read Archive (SRA), which includes 10 RNA-seq datasets from 6 tissues. The details of these RNA-seq datasets are given in the supplementary Table S1. As in previous studies (Li et al., 2013; Chen et al., 2019; Wang et al., 2020), we also collected human data to comprehensively evaluate our model.

We apply the True Path Rule (Dessimoi and Škunca, 2017; Zhao et al., 2020) and direct annotations in the corresponding GO file to expand GO annotations of genes, namely if a GO term is annotated to a gene, then its ancestor GO terms are annotated to the gene too. We then select terms annotated with at least 50 genes for the experiments. As a result, 387 GO terms are chosen. We considered and created a binary label vector for each gene bag. If a gene bag is annotated with a GO term from the selected terms, we assign 1 to the term position in the binary vector and use it as a positive sample for this GO term. Otherwise, we assign 0 and use it as a negative sample. Similarly, isoforms annotated with the selected terms are kept for the experiments. In the end, we obtained 31,640 isoforms spliced from 14,605 Maize genes, among which 7,587 are multiple isoform genes (MIGs) and 7,018 are single isoform genes (SIGs).

2.3 Extracting isoform features and initializing annotations

Gene-isoform relation is the bridge that transfers gene-level annotations to isoforms. Most isoform function prediction approaches use this relation to initialize the annotations of isoforms, but neglect it during the training process (Li et al., 2013; Shaw et al., 2018; Chen et al., 2019), and thus result in inconsistent predictions at the gene- and isoform-levels. Some other approaches make use of such relations during the training process (Eksi et al., 2013; Yu et al., 2020; Wang et al., 2020), but they mainly focus on the RNA-seq data, or fuse multi-type data at a coarse level, which prohibit them from obtaining a prominent performance. Here, we introduce a multi-instance learning convolution neural network (MILCNN) trained on gene-level annotations to extract isoform features in a gene bag, and to initialize the individual annotations of isoforms in a coherent manner.

Since different isoforms have different lengths of amino acids, we need to numerically encode each amino acid sequence while retaining the relationship between the isoform and the gene bag. Specifically, we first translate the CDS of each isoform into a sequence of amino acids. Then each sequence is represented as a series of overlapping trigrams. The feature vector $x_{ij}$ for an isoform is a combination of trigrams. We splice the feature vectors of all isoforms of a gene to generate the bag representation matrix $X_i$, of isoform trigrams with one-hot encoding. Formally, given the gene bag representation matrix $X_i$, the initial annotations are computed as follows:

$$Z_i = \text{mPool}(pPool(iConv(wEmbed(X_i))))$$  

where $wEmbed$ is the word-embedding of isoform sequences and it can retain the local features of isoform sequences to some extent. Particularly,
each trigram composed with three amino acids is treated as a word, then the combination of trigrams is used as a sentence to train \textit{wEmbded}. We adopt the default word-embedding model already integrated with PyTorch to obtain the embedding for each isoform sequence. \textit{iConv}, \textit{mPool}, and \textit{mPool} are operations of MILCNN, and are described below.

Dense embedding (Bengio et al., 2003) can address the sparsity of one-hot encoding, which has a limited generalization capability. To discover latent features of isoform sequences, we choose four different sizes (8, 16, 24 and 32) of convolution kernels, and set different sliding steps. The convolution portion takes the gene bag \(X_i\) as input and extracts isoform sequence features as follows:

\[
P_{i1}^{(1)} x_{i1}^{(1)}, \ldots, x_{iv}^{(1)} = \text{iConv} \ast [x_{i1}^{(0)}, x_{i2}^{(0)}, \ldots, x_{iv}^{(0)}] \tag{2}
\]

where \(\text{iConv}\) is the 1D convolution, and \(x_{i}^{(0)}\) is the word embedding of the \(j\)-th isoolform of the \(i\)-th gene. We observe that the 1D kernels convolve each isoform by retaining the relationship between the gene and its isoforms, and the convolutional sliding window does not contain two isoforms at the same time when sliding, and thus it ensures the features are independently extracted among the isoforms.

The convoluted isoform vectors vary in length, thus making it difficult to set the connections between the pooling layer and the dense layer. To handle this, an improved pyramid pooling (He et al., 2015) called Instances pyramid pooling is used to encode isoform sequences into fixed length has no clear positive effect on the prediction results. Given that, we set \(d = 256\). After the pyramid pooling, we use the output of the penultimate layer as the representational feature vectors \(H \in \mathbb{R}^{n \times l}\) of \(n\) isoforms.

Finally, two maximum pooling layers are sequentially used to get the latent features of isoform sequences, we choose four different sizes \((8, 16, 24, 32)\) of convolution kernels, and set different sliding steps. The convolution portion takes the gene bag \(X_i\) as input and extracts isoform sequence features as follows:

\[
x_{i1}, \ldots, x_{iv} = \text{mPool}(x_{i1}^{(1)}, \ldots, x_{iv}^{(1)}) \tag{3}
\]

where \(x_{i} \in \mathbb{R}^d\) is the representational feature vector of an isoform. Zhou et al. (2020) found that if the extracted feature dimension exceeds 128, the protein sequence information can be well preserved, and further increasing the dimension has no clear positive effect on the prediction results. Given that, we set \(d = 256\). After the pyramid pooling, we use the output of the penultimate layer as the representational feature vectors \(H \in \mathbb{R}^{n \times l}\) of \(n\) isoforms.

Finally, two maximum pooling layers are sequentially used to get the isoform initial annotations \((\hat{x}_{ij} \in \mathbb{R}^{l \times T})\) and aggregate these annotations to their hosting gene as follows:

\[
\hat{x}_{11}, \ldots, \hat{x}_{iv} = \text{mPool}(x_{i1}, \ldots, x_{iv}) \tag{4}
\]

\[
\hat{Y}_i = \text{mPool}(\hat{x}_{i1}, \ldots, \hat{x}_{iv}) \tag{5}
\]

The first maximum pooling maps the \(d\)-dimensional isoform feature vectors with respect to \(T\) GO terms, and the second one aims to aggregate individual annotations of \(\tau\) isoforms to the hosting gene. Note, the zero-padded isoform feature vectors are filtered out in this process. To ensure the consistent annotations between gene and isoform levels, we should reversely distribute known annotations of a gene to its isoforms. To do this, we define the loss function of MILCNN as the cross-entropy between the isoform-level aggregated annotations and the corresponding gene-level known annotations:

\[
\text{loss}_\text{MILCNN} = \sum_{l=1}^{m} \sum_{i \in Y_l} \log(\sigma(\hat{Y}_i)) + (1 - Y_l) \log(1 - \sigma(\hat{Y}_i)) \tag{6}
\]

Here \(Y_i\) is the one-hot label vector for the \(i\)-th gene, \(Y_i = 1\) indicate \(i\) is annotated to the \(l\)-th gene, \(Y_i = 0\) otherwise. \(\sigma(\cdot)\) is the Sigmoid activation function. Minimizing the above loss can induce consistent annotations of a gene and its isoforms. In addition, it unites the learning of isoform representational features and initial annotations, the last two are given in input to the GCN sub-branch for further differentiating the annotations of individual isoforms.

### 2.4 Predicting isoform functions using a Graph Convolution Network

The functions of an isoform depend on its sequence and interaction partners (Yang et al., 2016; Yu et al., 2021). Like genes, isoforms with similar functions are usually co-expressed (Park et al., 2013; Li et al., 2014). The advent of RNA-seq techniques provides an unprecedented amount of isoform-level expression data, and enables the detection of alternative splicing events at a deeper level. The initial annotations \(Z\) are still learned at a coarse level using the isoform sequence and gene-isoform relations. In this work, a Graph Convolution Network (GCN) (Kipf and Welling, 2016) is adopted to further refine annotations of individual isoforms based on the functional associations between isoforms. Diverse isoform expression measurements have been developed (Teng et al., 2016), such as Reads Per Kilobase of exon model per Million mapped reads (RPKM), Fragments Per Kilobase of exon model per Million mapped reads (FPKM), Transcripts Per Kilobase of exon model per Million mapped reads (TPM), and so on. Here, we use TPM to quantify the expression value of isoforms from collected RNA-seq datasets. TPM can reduce the impact of sequencing depth and gene length by first standardizing the gene length, and then standardizing the sequencing depth. Based on the quantified expression profiles of isoforms across multiple RNA-seq datasets, we construct a co-expression network \(W^c\) using weighted correlation network analysis (WGCNA) (Langfelder and Horvath, 2008). Particularly, the edge weights between two isoforms are determined by the Pearson correlation coefficient with the soft threshold method of WGCNA.

For each isoform, we retain the \(k\) \((k = 10\) for our experiments) neighbors with the highest co-expression values to avoid a too dense co-expression network and to omit trivial co-expressions. Due to the sparseness of RNA-seq data, the isoform co-expression networks are still not very effective for function prediction. Chen et al. (2019) reported that the integration of RNA-seq and sequence data can boost the performance of isoform function prediction. Motivated by this study, we build a composite isoform functional association network by integrating isoform co-expression networks \(W^c\) and isoform sequence similarity networks \(W^s\) (using BLAST with the cutoff E-value of 0.001) as follows:

\[
A_{ij} = \max(W^c_{ij}, W^s_{ij}) \tag{7}
\]

Here \(A_{ij}\) encodes the functional association strength between the \(i\)-th and \(j\)-th isoforms. Since the two networks are relatively sparse, the summation in the fusion process causes similar isoforms to have a disproportionately high association in the composite network. To avoid this, we adopt the maximum function to integrate the two networks into a composite one \(A \in \mathbb{R}^{n \times n}\).

The composite network \(A\) and the isoform representation feature vectors \(H\) naturally give an attributed isoform functional association network. To mine the underlying non-linear and interdependent relationships among isoforms and GO terms, we leverage a GCN and the initialized GO annotations of isoforms on this attributed network to further differentiate isoform functions. GCN (Kipf and Welling, 2016) has been widely used in network representation learning and semi-supervised learning. GCN updates node representations by propagating information among connected nodes. In our case, the GCN takes \(H\) and \(A \in \mathbb{R}^{n \times n}\) as the input, and updates the representation of \(i\) isoforms layer-by-layer. The updated representation of \(i\) isoforms in the \(l + 1\) layer is defined as follows:

\[
H^{l+1} = \phi(AH^lW^c) \tag{8}
\]
where $A$ is the normalized version of $A$, and $\phi(\cdot)$ denotes a non-linear activation function (LeakyReLU is used in this work). $W^l \in \mathbb{R}^{512 \times 1024}$ is a transformation matrix to be learned, $d_i = d$ and $H^{l+1} = H$.

To consider higher-order relations between isoforms, the convolution on the attributed network $\hat{A}$ can depend on the nodes that are $S$ steps away from the target node. The output signals of the GCN are defined by an $S$-order approximation of localized spectral filters on the network. Then, the convolution operation can be further formulated as:

$$H^{l+1} = \phi(\sum_{s=1}^{S} \hat{A}^s H^l W^l)$$

We can learn a deep representation of isoforms by stacking the GCN layers and the non-linear mapping from isoform representations to GO terms. We take the final output as the predicted association likelihood between isoforms and GO terms. Unlike the convolution part, we train a GCN classifier for each GO term:

$$\hat{Z} = \text{GCN}(A, H, Z)$$

Class imbalance is one of the main obstacles that impede deep neural networks from achieving state-of-the-art accuracy. Unfortunately, the GO annotations of genes and their hosted isoforms are rather imbalanced and sparse. Given this, we use the focal loss (Lin et al., 2020) to deal with imbalanced annotations. Different from the traditional cross entropy loss, the focal loss is a dynamically scaled cross entropy loss, which can automatically down-weight ‘easy’ isoforms that can be correctly predicted without much effort, and rapidly leads the model to focus on ‘hard’ ones. The focal loss on a particular isoform is defined as:

$$\text{loss}_f(X, t) = \sum_{i \in I} \alpha(1 - \hat{y}_i)^\gamma \log \hat{y}_i + (1 - \alpha)\hat{y}_i^{\gamma} \log(1 - \hat{y}_i)$$

where $\hat{y}_i$ denotes the initial association probability between the $k$-th isoform and GO term $t$, and $\hat{y}_i$ is the predicted likelihood for this association. Focal loss assigns a weight $\alpha$ to the negative isoforms, thus focusing more on positive isoforms. When a positive isoform is judged to be negative, a large weight is imposed on the loss according to the $\alpha$. $\gamma$ adjusts the rate of sample weight reduction. For example, for a positive isoform with respect to $t$, the prediction value of 0.95 is definitely an easy isoform, so the power $(1-0.95)^\gamma$ of $z_i$ where $z_i = \sum_{s=1}^{S} \hat{A}^s \hat{H} \hat{W}^l$ is the predicted likelihood for this isoform with respect to $t$.

Algorithm 1 DMIL-IsoFun: Deep Multi-Instance Learning for Isoform Function prediction.

Input: Isoform sequences $X$, isoform co-expression network $\tilde{W}$, gene annotations $Y$. Max epochs $M_{ep}$, $\alpha = 0.25$, $\gamma = 2$.

Output: Network parameters $f_m$ for MILCNN and $f_g$ for GCN.

1: for $e = 1 \rightarrow M_{ep}$ do
2: for each gene $X_i$ do
3: $[x_1, \cdots, x_r] = pPool(wConv(wEmbed(X)));
4: $H_i = [x_1, \cdots, x_r]$; $\triangleright$ Representation of isoforms
5: $Z_i = mPool(H_i)$; $\triangleright$ Initial annotations of isoforms
6: $Y_i = mLPool(Z_i)$; $\triangleright$ Gene annotations from isoforms
7: $\text{loss}_{gcn} = \text{CELoss}(Y, Y_i)$; $\triangleright$ Compute the cross entropy loss
8: end for
9: Update $f_m$ based on loss$_{gcn}$ by gradient descent.
10: $A = \max(W_{ij}, W_{ij}^\gamma)$; $\triangleright$ Construct the composite network
11: for each term $t$ do
12: $\hat{Z} = \text{GCN}(A, H, Z)$; $\triangleright$ Predict isoform annotations
13: $\text{loss}_f = \text{FocalLoss}(\hat{Z}, Z)$; $\triangleright$ Compute focal loss
14: end for
15: Update $f_g$ based on loss$_{f}$ by gradient descent.
16: end for

### 3.1 Performance comparison with the existing methods

The 2017 B73 v5 genome assembly project provides annotations, which enable direct performance evaluation and comparison at the isoform level, instead of the typical approximate gene-level evaluation done by aggregating isoform-level predictions. We randomly partition the isoforms into a training set (80%) and a validation set (20%) for ten independent rounds, and ensure that the isoforms of the same gene are partitioned into the same set in each round. We compare DMIL-IsoFun against miSVM (Eksi et al., 2013), iMILP (Li et al., 2013), DeepIsoFun (Shaw et al., 2018), DIFFUSE (Chen et al., 2019), IsoFun (Yu et al., 2020), and Disofun (Wang et al., 2020). All the input parameters are set as suggested by the authors, or optimized in the suggested ranges. The values of the parameters for DMIL-IsoFun are given in the supplementary Table S4.

For a comprehensive evaluation, we use four widely-used evaluation metrics: AUROC, AUPRC, DIFFUSE (Chen et al., 2019), IsoFun (Yu et al., 2020), and Disofun (Wang et al., 2020). All the input parameters are set as suggested by the authors, or optimized in the suggested ranges. The values of the parameters for DMIL-IsoFun are given in the supplementary Table S4.

For a comprehensive evaluation, we use four widely-used evaluation metrics: AUROC, AUPRC, DIFFUSE (Chen et al., 2019), IsoFun (Yu et al., 2020), and Disofun (Wang et al., 2020). All the input parameters are set as suggested by the authors, or optimized in the suggested ranges. The values of the parameters for DMIL-IsoFun are given in the supplementary Table S4.

For a comprehensive evaluation, we use four widely-used evaluation metrics: AUROC, AUPRC, DIFFUSE (Chen et al., 2019), IsoFun (Yu et al., 2020), and Disofun (Wang et al., 2020). All the input parameters are set as suggested by the authors, or optimized in the suggested ranges. The values of the parameters for DMIL-IsoFun are given in the supplementary Table S4.

For a comprehensive evaluation, we use four widely-used evaluation metrics: AUROC, AUPRC, DIFFUSE (Chen et al., 2019), IsoFun (Yu et al., 2020), and Disofun (Wang et al., 2020). All the input parameters are set as suggested by the authors, or optimized in the suggested ranges. The values of the parameters for DMIL-IsoFun are given in the supplementary Table S4.
the effectiveness of DMIL-IsoFun on leveraging isoform sequences, RNA-seq datasets, and gene-isofrom relations to differentiate the functions of individual isoforms. Like DMIL-IsoFun and DIFFUSE, DeepIsoFun also builds on deep neural networks, but it is outperformed by the former two. This is because DeepIsoFun solely uses isoform expression data and equally initializes all annotations of a gene to its isoforms, without accounting for the important gene-isofrom relation. For similar reasons, iMILP also loses to DIFFUSE and DMIL-IsoFun. miSV takes into account the gene-isofrom relations, but it’s still outperformed by the other methods due to the sole utilization of isoform expression data. DIFFUSE also leverages the sequence and co-expression information, but it doesn’t model well the gene-isofrom relation. As a result, it loses to DMIL-IsoFun by a large margin. Another possible cause of this is that DMIL-IsoFun combines both effectively sequence and co-expression network data by using a GCN than DIFFUSE. The performance margin between DMIL-IsoFun and DIFFUSE suggests that our choice of GCN for refining isoform-level annotations is more effective than the Conditional Random Field approach adopted by DIFFUSE. Both Disofun and IsoFun integrate the gene-level interactions and the co-expression network; they neglect the important isoform sequence data, which encode important functional sites and domains that help differentiate the functions of individual isoforms. As such, they often lose to DIFFUSE and DMIL-IsoFun. We observe that the AUPRC value of MF and BP is significantly lower than that of CC. That is because among 387 GO terms and 31,040 isoforms retained for experiment, 32 CC terms have an average of 1,054 annotations for each term, and 227 BP terms have an average of 919 annotations for each term. Each CC term has more annotated isoforms than others, 14.7% larger than BP and 3% higher than MF. The number of CC terms used for experiments is also the smallest. So the prediction of CC terms is relatively easier than BP and MF. We further used a signed rank test to check the difference between DMIL-IsoFun and each compared method; all the p-values are smaller than 0.01. In summary, these results and comparisons demonstrate the effectiveness of leveraging deep multi-instance learning and GCNs for isoform function prediction.

Besides the isoform-level evaluation, we report the gene-level evaluation results of DMIL-IsoFun and other compared methods in the supplementary Table S3. Moreover, we applied DMIL-IsoFun and the other methods on the Human dataset and report the results (approximate evaluation at the gene-level) in the supplementary Table S4. We observe that DMIL-IsoFun again achieves significantly better results than the competitive methods on different genomes and evaluation measures.

3.2 Further analysis

3.2.1 Analyzing the effects of model components

To investigate which component of DMIL-IsoFun contributes to the improved performance of DMIL-IsoFun, we perform an ablation study by removing the components from our model and measuring how the performance of the model is affected. We introduced two variants: (i) DMIL-IsoFun-CNN only uses one-hot encoding to encode the isoform sequence and the GCN to fuse sequence and expression data, along with the known isoform-level annotations; (ii) DMIL-IsoFun-CNN directly uses the initialized annotations of isoforms from MILCNN, without using the RNA-seq data and GCN. Following the same experimental configuration, we list the prediction results of these variants in Table 2. We observe that the results of DMIL-IsoFun-CNN are the lowest. This proves the necessity of utilizing GCN and RAN-seq datasets to further differentiate the annotations of individual isoforms initialized by MILCNN subnet. DMIL-IsoFun-GCN ranks second, which again confirms the power of the GCN on merging sequence and expression data to explore the nonlinear relationship between isoforms and GO terms. In practice, the AUROC of DMIL-IsoFun-GCN reduces by at least 12.2%, and the AUPRC drops by at least 25% when not using the sequence data, which is consistent with previous study that sequence data contain important functional sites and domains that help differentiate the function of individual isoforms (Chen et al., 2019). From the ablation study, we can conclude that both the MILCNN and the GCN subnets contribute to the improved performance of isoform function prediction.

3.2.2 Validation of predicted isoform functions

We further take ‘DNA binding’ (GO:0003677), ‘zinc ion binding’ (GO:0008270), and ‘phosphatidylinositol phosphate kinase activity’ (GO:0016307) as the testbed, which contains annotated isoforms spliced from the same gene (Breuza et al., 2016). Table 3 lists the known annotations of 10 isoforms of 4 genes. DMIL-IsoFun correctly differentiates 9 out of 10, which results in higher accuracy than the other methods. We observe that the two deep neural network based models (DeepIsoFun and DIFFUSE) are inclined to assign the same GO term to all isoforms of a multi-isoform gene, since they initialize all annotations of important isoform sequence data, which encode important functional sites and domains that help differentiate the functions of individual isoforms. In contrast, our DMIL-IsoFun takes into account this important relation and differentiates the initial annotations by MILCNN. We also see that the other four methods (Disofun, IsoFun, iMILP, and miSVM) are biased towards negative predictions. This is because the
positive annotations of isoforms are much fewer than the negative ones, and these methods do not take into account the intrinsic class imbalance issue of isoform function prediction. For a similar reason, DIFFUSE and DeepIsoFun also make more negative predictions than DMIL-IsoFun, which instead accounts for the class imbalance issue. We also observe that DMIL-IsoFun has a higher recall than the other methods, owing to the consideration of class-imbalance. Overall, these case studies confirm that DMIL-IsoFun can differentiate the GO annotations of individual isoforms spliced from the same gene.

Table 4. Prediction of the compared methods on Maize isoform (Zm00001e042100-T005) with respect to nine positive annotations.

<table>
<thead>
<tr>
<th>GO Term</th>
<th>Known Annotations</th>
<th>DNA binding (GO:0000677)</th>
<th>Zinc ion binding (GO:0000102666)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GO:0005979</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GO:0016021</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO:0001021</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO:0044425</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO:0009987</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO:0044237</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO:0008152</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GO:0006120</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Accuracy</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0/6</td>
<td>6/10</td>
<td>9/10</td>
</tr>
</tbody>
</table>

4 Discussion

The differentiation of functions of alternatively spliced isoforms helps explaining the proteome complexity and various complex diseases at a higher resolution than the canonical gene-level analysis. In this paper, we introduced DMIL-IsoFun, a method that merges genomics and transcriptomics data to identify the functions of individual isoforms spliced from the same gene. DMIL-IsoFun builds on the principle that the functions of a gene are aggregated from its isoforms, and isoforms with similar sequences and co-expression share similar functions. DMIL-IsoFun firstly introduces a Multi-Instance Learning Convolutional Neural Network to extract the feature vectors of isoform sequences and to initialize the annotations of individual isoforms using gene-isoform relations; then it alters the GCN to account for the class-imbalance data to further differentiate annotations of individual isoforms. DMIL-IsoFun significantly outperforms state-of-the-art methods for predictions at both the gene and isoform levels. In the future, we will study how to reliably combine multiple gene-level, transcript-level, and phenotype heterogeneous data sources to further improve the performance of DMIL-IsoFun, and to explore isoform-disease associations.

Funding: This work is supported by NSFC (61872300 and 62031003).

Conflict of Interest: none declared.

References


