A deep learning framework for gene ontology annotations with sequence- and network-based information

Fuhao Zhang[1], Hong Song[1], Min Zeng[1], Fang-Xiang Wu[2], Yaohang Li[3], Yi Pan[4], and Min Li[1*]

Abstract—Knowledge of protein functions plays an important role in biology and medicine. With the rapid development of high-throughput technologies, a huge number of proteins have been discovered. However, there are a great number of proteins without functional annotations. A protein usually has multiple functions and some functions or biological processes require interactions of a plurality of proteins. Additionally, Gene Ontology provides a useful classification for protein functions and contains more than 40,000 terms. We propose a deep learning framework called DeepGOA to predict protein functions with protein sequences and protein-protein interaction (PPI) networks. For protein sequences, we extract two types of information: sequence semantic information and subsequence-based features. We use the word2vec technique to numerically represent protein sequences, and utilize a Bi-directional Long and Short Time Memory (Bi-LSTM) and multi-scale convolutional neural network (multi-scale CNN) to obtain the global and local semantic features of protein sequences, respectively. Additionally, we use the InterPro tool to scan protein sequences for extracting subsequence-based information, such as domains and motifs. Then, the information is plugged into a neural network to generate high-quality features. For the PPI network, the Deepwalk algorithm is applied to generate its embedding information of PPI. Then the two types of features are concatenated together to predict protein functions. To evaluate the performance of DeepGOA, several different evaluation methods and metrics are utilized. The experimental results show that DeepGOA outperforms DeepGO and BLAST.

Index Terms—deep learning, protein function, protein-protein interaction, protein sequence, protein domain.

1 Introduction

Proteins perform specific functions in organisms and are virtually involved in various biological activities, such as body movement, metabolism, and structural support [1]. With the rapid development of high-throughput technologies, many protein databases have been available. However, there are a great number of proteins without functional annotations. For instance, only about 1% of proteins have been confirmed with experiments and manually annotated in the UniProt database [2]. Protein functions are usually discovered via in vitro or in vivo experiments [3]. However, biological experimental methods are expensive and time-consuming. Thus, it is a difficult task to determine the functions of a huge number of unannotated proteins with experimental methods.

In the past decades, a number of computational methods have been proposed to predict protein functions and could be classified into the following three categories. The first category of methods is sequence-based and the most famous and wildly used method is BLAST [4]. BLAST [4] assigns the functions of annotated proteins to unannotated proteins based on the homologous similarity of sequences. It has a disadvantage that this method can only be used to predict the function of proteins with high homologous similarity. In order to overcome this disadvantage, other sequence-based methods with additional biological information have been proposed to predict protein functions. FFPred3 [5] predicts protein functions with the biological information of the secondary structures, transmembrane helices, intrinsically disordered regions, signal peptides, and other motifs. GOLabeler [6] is a recently developed method that improves the prediction of protein functions with a combination of diverse sequence-based features, such as 3-mer, protein domains, families, motifs, and biophysical properties. The second category of methods is focusing on phylogenomic and genomic information. Proteins are translated from genes and the changes of protein functions are related to the changes of the physiologies in different species. Thus, SVD-phy [7] uses the singular value decomposition of phylogenetic profiles for the protein function prediction. SIFTER [8] improves the function annotation by a statistical model with the phylogenetic tree. TreeGrafter [9] annotates protein functions with phylogenetic tree data. In addition, with the development of high-throughput microarray technology, there are methods that

- *F. Zhang, H. Song, M. Zeng, and M. Li are with the School of Computer Science and Engineering, Central South University, Changsha, P.R. China. Email: fhzhang@csu.edu.cn, songhong@csu.edu.cn, zengmin@csu.edu.cn, limin@mail.csu.edu.cn*
- *F.X.Wu is with the Division of Biomedical Engineering and Department of Mechanical Engineering, University of Saskatchewan, Saskatoon, SKS7N5A9, Canada. E-mail: faw341@mail.usask.ca.*
- *Y. Li is with the Department of Computer Science, Old Dominion University, Norfolk, USA. Email: yaohang@cs.odu.edu*
- *Y.Pan is with the Department of Computer Science, Georgia State University, Atlanta, GA30302, USA E-mail: yipan@gsu.edu*
- ** Corresponding author*

xxxx-xxxx/0x/$xx.00 © 200x IEEE Published by the IEEE Computer Society

use the gene expression data for the accurate protein function prediction [3, 10]. The third category of methods is predicting protein functions with other biological information (do not contain protein sequences). Many proteins that have similar protein functions do not mean they have a similar sequence. As we know, many different types of biological information (e.g. PPI network, genetic interaction, genomic context, and protein structure) have complex relationships with protein functions [11, 12]. Thus, these different types of biological information are applied to the prediction of protein functions. For example, a protein is not isolated, yet interact with other proteins to perform its functions in many situations. Therefore, several researchers [13-17] use a PPI network to predict protein functions. Moreover, there are methods that predict protein functions with multiple network information. NetGO [18] improves the protein function prediction by incorporating massive protein-protein network information. DeepNF [19] uses multimodal Deep Autoencoders to extract high-level features of proteins from multiple heterogeneous interaction networks.

In common, a huge number of researchers annotate protein functions with Gene Ontology (GO) [20]. GO has three major branches, biological processes (BP), molecular functions (MF) and cellular components (CC). GO contains more than 40,000 terms and a lot of proteins have more than one function. It is infeasible to train a model for each GO term with traditional machine learning methods (e.g. Support Vector Machine (SVM) and Logistic regression). Recently, deep learning techniques have been extensively applied to several fields, such as computer vision, natural language processing, and speech recognition. Inspired by their success, some researchers use deep learning methods to identify protein functions to address this issue. DeepGO [21] presents a deep ontology-aware model for protein function prediction and achieves good results. It learns sequence features of proteins with convolutional neural networks and obtains topological features of the PPI network by using a network representation learning technique.

Deep learning techniques have improved the performance of some biological problems [22-26]. In this study, we present a novel deep learning framework called DeepGOA that predicts protein functions based on protein sequences and PPI networks. Firstly, we generate the dense vectors of each amino acid code of each protein sequence by using the word2vec technique which is a recently developed distribution representation technique. Secondly, in order to extract more effectively features of protein sequences, a single convolutional layer is replaced by a multi-scale convolutional layer which has a stronger ability to capture features than a single convolutional layer. Additionally, we extract global features of sequences with Bi-LSTM [27] before the multi-scale convolutional layer. Bi-LSTM [27] can capture the global features of protein sequences that can provide a preliminary processing result to the multi-scale convolutional layer. Protein subsequence-based information including protein domains and motifs plays an important role in predicting protein functions. Thus, we obtain the subsequence-based information by InterPro [28] and represent the subsequence-based information with one-hot coding. Then the combination of diverse sequence-based features is fed into a fully connected layer to generate sequence-based features. Thirdly, we use the Deepwalk algorithm to extract topological features without hand-crafted feature vectors. Finally, the sequence-based features and topological features of the PPI network are combined to perform the task of the protein functions prediction.

Before we present our method, we would like to discuss the difference between our method DeepGOA and the previously proposed approach DeepGO [21]. Firstly, DeepGO extracts k-mer features from protein sequences with one convolutional layer and ignores the global information of the whole sequence. In order to extract more effectively features of protein sequences, DeepGOA extracts global and multi-size local features of sequences with Bi-LSTM [27] and a multi-scale convolutional layer. In addition, DeepGOA obtains domains, families, and motifs from sequences by InterPro. Secondly, the PPI network used in DeepGO and DeepGOA is different. The PPI network used in DeepGO has 8,789,935 vertices and 11,586,695,610 edges. Such a big network has a lot of noise, which will bring negative effects for the prediction, increase the cost of computation. Here, we filter the proteins and their interactions if they are not included in Uniport. The filtered PPI network contains 354,687 vertices and 54,253,077 edges, which is much smaller than that of DeepGO. Deepwalk is applied on the PPI network to extract the topological features without hand-crafted feature vectors. In DeepGO, Neuro-symbolic method is used.

2 METHODS

Our deep learning framework of DeepGOA proposed to predict protein functions is shown in Figure 1. This framework includes the feature extraction and classification sections.

2.1 Network Architecture

In the feature extraction, DeepGOA first numerically represents protein sequences with the word2vec technique and uses one-hot coding to represent information of protein domains, families, motifs from InterPro. Second, DeepGOA extracts the global features and local features of sequences, with a Bi-LSTM [27] and a multi-scale convolutional layer, respectively. Moreover, DeepGOA obtains high-quality features of protein domains, families, and motifs with a neural network. Then DeepGOA combines these subsequence-based features and generates sequence-based features. Additionally, DeepGOA uses the Deepwalk algorithm to obtain topological features of the PPI network. Finally, a combination of the complex sequence-based features and topological features of the PPI network is fed into the classification section of DeepGOA.

2.2 Extracting sequence-based features

This subsection discusses the various steps involved in feature extraction from protein sequences.

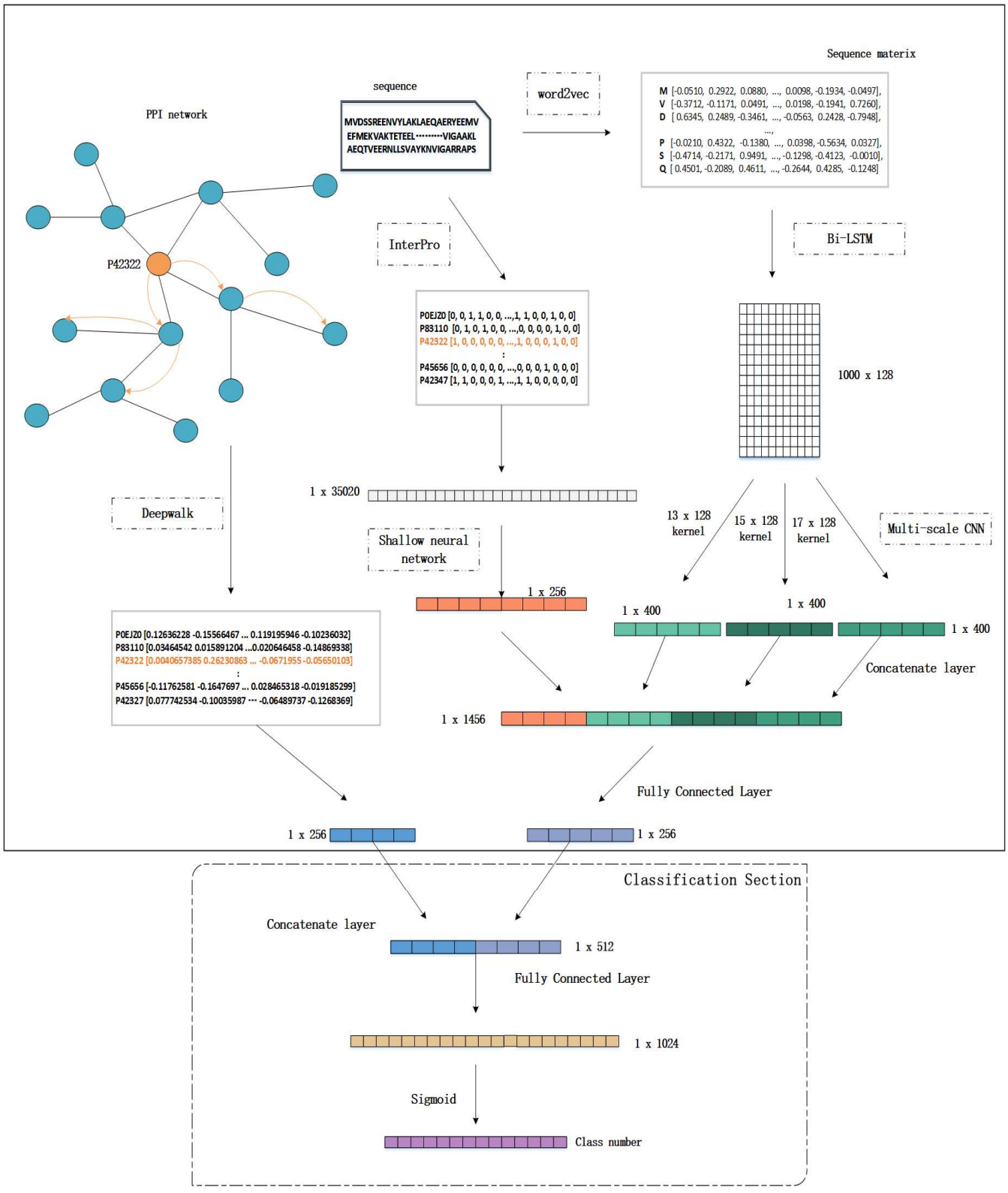

Fig. 1. An overview of our proposed deep learning framework for identifying protein functions.

2.2.1 Numerical representation of the sequence via the word2vec technique

Proteins consist of different kinds of amino acids and many amino acids have important biological functions. Traditional computational methods use one-hot coding to numerically represent the inputs of amino acids. However,

one-hot coding is sparse and cannot reflect the relationship between different kinds of amino acids. In recent years, the distribution representation technique has been rapidly developed in the field of natural language processing (NLP). The distribution representation technique uses a dense vector to represent a word, which can describe the semantic distance between words to a certain extent. The Word2vec algorithm is one of the most classic models and widely used in various fields [29]. Inspired by the word2vec algorithm, we regard a protein sequence as a sentence and amino acid in the protein sequence as words and then use the word2vec algorithm to numerically represent amino acid codes.

Firstly, the word2vec algorithm calculates the word frequency of each word in the input text and selects the N words with the highest word frequency to form a vocabulary. Then it generates a one-hot vector for each word in the input text and uses the one-hot vector as the input of the Skip-gram model [30] which predicts the probability of a word around the input word by maximizing the possibility of co-occurrence between words. After the training steps, we obtain an embedding vector for each word in the vocabulary.

In our experiments, we regard a protein sequence as a sentence and each amino acid code as a word. We generate a dense vector of each amino acid code with the word2vec technique. The dense vectors of all amino acid codes of the sequence form a feature matrix of the sequence, which is treated as an image. As a result, we can use deep learning techniques to capture sequential features.

2.2.2 Acquiring the global information of sequences with Bi-LSTM

The global information about the whole sequence plays an important role in the classification of protein functions. The multi-scale convolutional layer has little capacity to obtain long-range features of sequences with some small kernel sizes. If we increase the convolutional kernel size to get a larger receptive field, it generates noises when addressing short sequences and amino acid patterns. In addition, due to the length of the sequence is 1000, it is difficult to choose an appropriate convolution kernel size in the large range of the region. In order to overcome these limitations, we first use Bi-LSTM [27], which is a variant of the recurrent neural network (RNN), to extract global features from protein sequences.

RNN analyzes a text word by word and stores the semantics of all the previous texts in a fixed-sized hidden layer [31]. An important advantage of the RNN is to utilize context information in the mapping process between input and output sequences. Unfortunately, the range of context information captured by standard RNNs is limited and there exists a vanishing gradient problem in the back-propagation process. In order to solve this problem, some researchers proposed the Long and Short Time Memory (LSTM) structure, an excellent variant of RNN, which inherits the characters of most RNN models and alleviates the vanishing gradient problem. LSTM only access to past contextual information and not to future information that is very beneficial for many sequence annotation tasks. Based on this idea, the Bi-LSTM [27] has been proposed, which provides complete past and future information for each point of the input sequence in the output layer.

In order to capture past and future context information of protein sequences, the feature matrix of a sequence is fed to the Bi-LSTM [27] part. The hidden layer size is 64 and the number of hidden layers is 2. We set the dropout rate as 0.2 to avoid overfitting.

2.2.3 Obtain more local features of sequences with Multi-scale convolutional layer

The local features of protein sequences are important for the prediction. Convolutional filters can be used to obtain local features and acquire more features with multi-layer stacking [32]. Previous studies use a single convolution kernel to extract features of protein sequences and work well. However, a single convolutional kernel cannot capture satisfactory features for classification. For instance, the length of the sequence and amino acid patterns with biological information are different. Thus, using a single fixed convolution kernel does not work well with protein sequences that have different lengths. In addition, due to the fixed input scale, the sequences whose length is less than the fixed length must be filled with zero which may introduce noises for the prediction. To address these problems, we use multi-scale convolutional kernels to extract more effective local features. Besides, 1D max-pooling is applied to filter zero paddings. A multi-scale convolutional layer not only is suitable for sequences and the amino patterns with different lengths but also increases the diversity of local features. It turns out that the multi-scale convolutional layer is more powerful than a single convolutional layer. For instance, TextCNN [33] uses a multi-scale convolutional layer for the sentence classification, which outperforms a single convolutional layer and other deep learning structure. The convolutional kernels of our multi-scale convolutional layer are 13, 15, and 17, respectively, and the number of channels is 400. The convolutional layer is followed by the 1D-max-pooling with the size of 1000.

2.2.4 Obtain subsequence-based features

Protein subsequence-based features are very important for the prediction of protein functions. In this study, we use the InterPro tool to obtain protein subsequence-based features including protein domains and motifs. InterPro analyses protein sequences according to diverse databases, including CCD [34], Pfam [35], CATH-Gene3D [36], and SUPERFAMILY [37]. InterPro provides a useful tool called InterProScan which is a software package can be downloaded from the InterPro database. InterProScan can create a binary vector with 33,520-dimensional features that code the information of protein domains, families and motifs. In order to obtain non-linear features, the binary vector is fed into fully connected layers. Then we combine diverse sequence features and generate comprehensive and high-quality protein features.

2.3 Extracting PPI network topological features

Networks have been widely used to model various biological problems and network topological features are very

important in the study of biological prediction problems[38, 39]. ThrRW [11] predicts protein functions by using multiple random walks to extract network topological features. When a PPI network contains thousands or millions of nodes and edges, it is computationally expensive or even infeasible to extract topological features by using a random walk on the network. Thus some representation learning techniques have been proposed, including Deepwalk [40], LINE [41], node2vec [42], HOPE [43], SiNE [44], SNE [45]. These methods are neural network-based approaches and their performances are better than traditional approaches, such as PCA [46] and MDS [47].

The Deepwalk algorithm is the first network embedding method based on deep learning and the most popular method. The Deepwalk algorithm treats nodes as words and combines a random walk with the. Skip-gram model [30]. The first step of the Deepwalk algorithm is representing the input network with a matrix, such as an adjacency matrix or a Laplacian matrix. The second step is generating sequences of nodes with random walk. Finally, the Deepwalk algorithm uses the Skip-gram model [30] to learn to embed nodes from sequence nodes. In our study, we utilize the Deepwalk algorithm as the method for learning node embedding of the PPI network. In order to cover the adjacent vertices of each vertex as many as possible, we use a sampling method. The formula is as follows:

$$(1-p)^k \le \alpha \tag{1}$$

where p is the ratio of vertices to edges. The left part of the formula represents the probability that one adjacent vertex of the vertex is not picked at least once after k iterations of random walks. When this probability is smaller than α, it is reasonable to believe that all adjacent vertices of the vertex are covered. In this study, we set α as 0.1 and the approximate value of the walk number is 300. The walk-length, the window-size, and the output vector size is 20, 10, and 256, respectively.

2.4 Assessment metrics

In this study, we use F_{max}, AvgPr, AvgRc, MCC (Mathews Correlation Coefficient), AUC (Area Under The Curve) to evaluate the performance of models [48, 49]. F_{max} is a protein-centric maximum F-measure. AvgPr and AvgRc are the average precision and average recall for all proteins that have at least one GO term, respectively. They are calculated as follows:

$$pr_i(t) = \frac{\sum_f I(f \in P_i(t) \wedge f \in T_i)}{\sum_f I(f \in P_i(t))} \tag{2}$$

$$rc_i(t) = \frac{\sum_f I(f \in P_i(t) \wedge f \in T_i)}{\sum_f I(f \in T_i)} \tag{3}$$

$$Avg\,\Pr(t) = \frac{1}{m(t)} \cdot \sum_{i=1}^{m(t)} pr_i(t) \tag{4}$$

$$AvgRc(t) = \frac{1}{n} \cdot \sum_{i=1}^{n} rc_i(t) \tag{5}$$

$$F_{\max} = \max\left\{\frac{2 \cdot Avg\,\Pr(t) \cdot AvgRc(t)}{Avg\,\Pr(t) + AvgRc(t)}\right\} \tag{6}$$

$$AUC = \int_{-\infty}^{\infty} TPR(t)(-FPR'(t))dt \tag{7}$$

$$TPR(t) = \frac{TP(t)}{TP(t) + FN(t)} \tag{8}$$

$$FPR(t) = \frac{FP(t)}{FP(t) + TN(t)} \tag{9}$$

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{10}$$

where f represents a GO term. When the threshold is t, $P_i(t)$ and T_i are the set of predicted GO terms and the set of annotation GO terms for protein i, respectively. n is the number of all proteins and m(t) is the number of predicted proteins with at least one GO term. TP and TN represent the numbers of the positive and negative terms of predicted proteins which are classified correctly, respectively. FP and FN represent the numbers of positive and negative terms of proteins which are misclassified, respectively.

3 Datasets

In this study, we use the same datasets as the previous study [21]. Specifically, we use three datasets including training dataset, testing dataset and benchmark evaluation dataset. The training dataset contains 48,568 proteins and the testing dataset contains 12,142 proteins. These proteins have experimental evidence codes (EXP, IDA, IPI, IMP, IGI, IEP, TAS, and IC) and we ignore some proteins which contain ambiguous amino acid codes (B, O, J, U, X, Z). The benchmark evaluation dataset is released as part of the CAFA3 competition. We train one model for each subontology in GO. We select the top 589 terms of MF, 439 terms of CC, and 932 terms of BP with the sorted order of GO classes, respectively.

We construct a PPI network of multiple species from the STRING database [50]. In addition, we connect these proteins with orthology relations from the EggNOG database [51]. We acquire the mapping relationship files provided by the STRING and SwissProt [2] databases, respectively.

4 Experimental results

4.1 Implemental Details

In order to choose the best parameters, we also randomly select 20% proteins from the training dataset as the validation dataset. If a protein has a GO term in our selected terms, we assign 1 to the term's position in the label vector and use it as a positive sample of the term. Otherwise, we assign 0. During training and testing processes, we use proteins that have at least one GO term in our selected terms. The InterProScan tool and the InterPro entry list are down-loaded from the InterPro database. The InterPro entry list contains 2,865 homologous super-families, 21,695

families, 9,268 domains, 280 repeats, and 912 sites. We select proteins that have the mapping relationship between the STRING and SwissProt and pick up interactions of selected proteins whose confidence score is at least 300. Then we add the orthologous relations from the EggNOG database for selected proteins. We combine selected proteins, PPIs and orthologous relations to construct the PPI network, which contains 354,687 vertices, and 54,552,077 edges. During training and testing processes, we assign a zero vector to those proteins without the embedding representations.

Our deep learning framework is implemented by Pytorch [52], a public deep learning framework developed by Facebook. The detail of the network structure is described below. We use a grid search method to choose the parameters and structure of our method. Reasonable parameters and structures are clearly described as follows. Firstly, we generate a dense vector for each amino acid code with the word2vec technique. Then we represent each protein sequence with a feature matrix which is 1000x128. In addition, InterPro generates a 33,520-dimensional vector to represent features of protein domains, families and motifs. Secondly, we use the Deepwalk algorithm to generate a 256-dimensional vector to represent PPI features. The walk number, the walk-length and the window-size of Deepwalk are 20, 10, and 256, respectively. Thirdly, after trying different kernel sizes, we determine the kernel sizes of the multi-scale convolutional layer are 13x128, 15x128, and 17x128, respectively. The performance of different kernel sizes of our model is provided in supplementary Table S1. The output size of the multi-scale CNN layer is 1000x1 by using the zero-padding and the stride is 1. By using a multi-scale CNN layer, we obtain three feature maps that have 400 channels. The results of different hidden layer sizes of our model are provided in supplementary Table S2. We use two fully connected layers to extract high-quality features of protein domains, families and motifs. The number of neural units in the two fully connected layers is 1024 and 512, respectively. The activation function of the two fully connected layers is the sigmoid function. Two kinds of sequence-based features are concatenated together as input to a fully connected hidden layer for extracting high-quality sequence-based features. On top of the fully connected hidden layer, there is another fully connected hidden layer taking concatenated high-quality sequence-based features and embedding vectors from the PPI network as input. The detailed results of DeepGOA with different walk numbers are found in supplementary Table S3. We used a dropout rate of 0.2 on the fully connected layer in the network to avoid over-fitting. The output from the fully connected layer is fed into the prediction layer which performs the classification task. Finally, Adam optimizer is used to train our deep learning framework. The batch size is set to 128 and the initial learning rate is set to 0.002.

4.2 Comparison with other methods

To examine the performance of DeepGOA, we first compare DeepGOA with BLAST [4] and DeepGO on the testing dataset. To our best knowledge, DeepGO is the first method to use deep learning techniques with protein sequences and PPI networks. In this study, for a protein from the testing dataset, we use BLAST [4] to find the most similar protein from the training dataset. Then we assign all GO terms of the most similar protein to it. Table 1 shows that DeepGOA achieves the best values in all assessment metrics on all branches. For example, in terms of F_{max}, DeepGOA improves about 34.3% (BP), 85.9% (CC), 50.0% (MF) than BLAST, and about 6.8% (BP), 6.3% (CC), 18.7% (MF) than DeepGO. DeepGOA achieves the AUC of 0.906 (BP), 0.976 (CC), 0.947 (MF), respectively, which is better than DeepGO (0.896, 0.967, 0.928). In terms of MCC measure, DeepGOA is also better than DeepGO in the BP, CC, and MF branches. The prediction of DeepGOA is available on this website (http://bioinformatics.csu.edu.cn/DeepGOA/).

We also compare DeepGOA with DeepGO and FFPred3 [5] in previous CAFA challenges [53] on the benchmark evaluation dataset. All methods (DeepGOA, DeepGO, FFPred3) did not use protein annotations in the benchmark evaluation dataset during the training process. Table 2 shows the performance of DeepGOA comparing with DeepGO, FFPred3, and Phylo-PFP on the benchmark evaluation dataset of CAFA3. In the CC branch, the performance values of DeepGOA are 0.538 (F_{max}), 0.582 (AvgPr), 0.496 (AvgRc), 0.502 (MCC), 0.953 (AUC), respectively, which is about 21.4% (F_{max}), 27.3% (AvgPr), 15.3% (AvgRc), 28.1% (MCC), 6.7% (AUC) better than FFPred3. We find that DeepGOA and Phylo-PFP are better than FFPred3 and Deep GO in the MF branch. In terms of F_{max}, the performance value of Although Phylo-PFP is 0.539, which is better than FFPred3 (0.376) and DeepGO (0.472). The results show that our method achieves state-of-the-art performance.

To discover the vital elements in the success of DeepGOA, we compare our model with component methods. The results are shown in Table 3. DeepGOA_Bi-LSTM [27] and DeepGOA_MultiCNN adopt Bi-LSTM [27] model and MultiCNN model with only protein sequences, respectively. DeepGOA_Seq only uses protein sequences to predict protein functions with a combination of Bi-LSTM [27] and MultiCNN models. The input of the DeepGOA_PPI method is only from the PPI network and DeepGOA_InterPro only uses features from protein domains, families, motifs to predict protein functions. DeepGOA_Seq_InterPro, DeepGOA_Seq_PPI and DeepGOA_InterPro_PPI are three independent methods that combine DeepGOA_Seq and DeepGOA_InterPro, DeepGOA_Seq and DeepGOA_PPI, and DeepGOA_InterPro and DeepGOA_PPI, respectively.

First, we compare models only using sequence-based features as input. Table 3 shows that DeepGOA_Seq outperforms DeepGOA_MultiCNN and DeepGOA_Bi-LSTM, which indicates a combination of Bi-LSTM and Mul-tiCNN makes a model to extract more effective features than only using a single model. In terms of subsequence-based feat-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2020.2968882, IEEE/ACM Transactions on Computational Biology and Bioinformatics

Tabel 1. The performance of DeepGOA and comparison to DeepGO and BLAST.

Method	BP					CC					MF				
	F_{max}	AvgPr	AvgRc	MCC	AUC	F_{max}	AvgPr	AvgRc	MCC	AUC	F_{max}	AvgPr	AvgRc	MCC	AUC
BLAST	0.314	0.302	0.327	-	-	0.362	0.321	0.417	-	-	0.372	0.367	0.377	-	-
DeepGO	0.395	0.412	0.379	0.397	0.896	0.633	0.643	0.624	0.592	0.967	0.470	0.577	0.397	0.438	0.928
DeepGOA	**0.422**	**0.443**	**0.403**	**0.420**	**0.906**	**0.673**	**0.684**	**0.661**	**0.621**	**0.976**	**0.558**	**0.667**	**0.480**	**0.528**	**0.947**

Tabel 2. Evaluation of DeepGOA, DeepGO, Phylo-PFP, and FFPred3 on the benchmark evaluation dataset of CAFA3.

Method	BP					CC					MF				
	F_{max}	AvgPr	AvgRc	MCC	AUC	F_{max}	AvgPr	AvgRc	MCC	AUC	F_{max}	AvgPr	AvgRc	MCC	AUC
FFPred3	0.262	0.304	0.228	0.231	0.828	0.443	0.457	0.430	0.392	0.893	0.376	0.352	0.401	0.293	0.858
Phylo-PFP	0.256	0.388	0.191	0.186	0.599	0.417	0.409	0.426	0.357	0.724	0.539	0.570	0.512	0.332	0.719
DeepGO	0.344	0.309	0.365	0.319	0.884	0.521	0.549	0.493	0.497	**0.953**	0.472	0.614	0.387	0.371	0.902
DeepGOA	**0.369**	**0.376**	**0.366**	**0.373**	**0.904**	**0.538**	**0.582**	**0.496**	**0.502**	**0.953**	**0.570**	**0.637**	**0.521**	**0.465**	**0.954**

ures from protein domains, families and motifs, DeepGOA_InterPro achieves better results than DeepGOA_Bi-LSTM, DeepGOA_MultiCNN, DeepGOA_Seq in all assessment metrics in BP and MF branches. Second, compared only with using sequence-based features, DeepGOA_PPI only with topological features of the PPI network achieves better performance in both BP and CC branches.

Third, we compare the performance of a combination of diverse single component methods. We observe that models with a combination of diverse single component methods are better than models with single component methods in most assessment metrics. For example, DeepGOA_Seq_PPI obtains the highest F_{max} of 0.673 and AUC of 0.977 in the CC branch. However, the results of DeepGOA_Seq_InterPro and DeepGOA_InterPro_PPI are better than DeepGOA_Seq_PPI in the MF branch except for AUC. For example, In terms of F_{max}, AvgPr, MCC in MF branch, DeepGOA_InterPro_PPI improves about 13.2%, 14.7%, 12.1% than DeepGOA_Seq_PPI. Then we examine the performance of component methods in Table 3 and DeepGO in Table 1. The results show that DeepGOA_PPI, DeepGOA_Seq_PPI and DeepGOA_InterPro_PPI outperform DeepGO in terms of all assessment metrics. While DeepGOA_Seq_InterPro obviously outperforms than DeepGO in MF branch, we observe that DeepGO achieves higher results than DeepGOA_Seq_InterPro in BP and CC branches.

DeepGOA, the combination of DeepGOA_Seq, DeepGOA_PPI and DeepGOA_InterPro, achieves the highest F_{max} in BP, CC and MF branches. Furthermore, in terms of AvgPr, AvgRc, MCC, and AUC, DeepGOA performs comparably to methods that achieve the highest values. Table 3 shows other interesting results. For example, methods using features from the PPI network outperform other methods without considering the PPI network in BP and CC branches. The performance of methods with protein families, domains and motifs perform better than other methods. The results indicate that topological features provide a better understanding of cellular components and the biological process of protein functions. Additionally, protein families, domains and motifs are useful for predicting molecular functions of proteins.

4.3 Case studies

Firstly, we choose one protein (Name: RENT3_ARATH) from the benchmark evaluation dataset of CAFA3 to illustrate the real effect of the performance of DeepGOA and other competing methods in the MF branch. Table 4 shows the results. Although Phylo-PFP and DeepGOA predict the same number of real functions, Phylo-PFP annotates three negative functions. The predictions of FFPred3 contain 3 real functions that are more than DeepGO. However, FFPred3 predicts many negative functions. In addition, we find GO:0003676 (nucleic acid binding) and GO:0005488 (binding) are annotated by most methods in Table 4. In summary, the results show that DeepGOA performs better than other compared methods. Secondly, we choose some examples of the results predicted by DeepGOA_PPI, DeepGOA_seq, and DeepGOA_InterPro to see which protein functions that can be easily predicted by sequence or PPI features alone. The results are shown in supplementary Table S4. We find that the sequence features prefer

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2020.2968882, IEEE/ACM Transactions on Computational Biology and Bioinformatics

Table 3. The performance of DeepGOA and component methods.

Method	BP					CC					MF				
	F_{max}	AvgPr	AvgRc	MCC	AUC	F_{max}	AvgPr	AvgRc	MCC	AUC	F_{max}	AvgPr	AvgRc	MCC	AUC
DeepGOA_Bi-LSTM	0.307	0.316	0.299	0.283	0.824	0.568	0.576	0.561	0.517	0.929	0.355	0.424	0.305	0.320	0.876
DeepGOA_MultiCNN	0.317	0.335	0.301	0.282	0.816	0.587	0.601	0.573	0.528	0.941	0.385	0.502	0.313	0.348	0.875
DeepGOA_Seq	0.322	0.343	0.303	0.286	0.824	0.603	0.610	0.596	0.542	0.947	0.415	0.521	0.345	0.383	0.904
DeepGOA_InterPro	0.354	0.384	0.328	0.338	0.847	0.566	0.596	0.539	0.514	0.926	0.530	0.672	0.437	0.497	0.929
DeepGOA_PPI	0.417	0.447	0.392	0.427	0.912	0.640	0.650	0.631	0.604	0.973	0.485	0.578	0.419	0.458	0.928
DeepGOA_Seq_InterPro	0.361	0.394	0.332	0.339	0.854	0.616	0.632	0.600	0.554	0.950	0.512	0.643	0.425	0.481	0.935
DeepGOA_Seq_PPI	0.420	0.438	**0.404**	0.418	**0.911**	**0.673**	**0.686**	0.659	**0.625**	**0.977**	0.492	0.584	0.426	0.470	0.943
DeepGOA_InterPro_PPI	0.416	**0.446**	0.389	0.413	0.899	0.635	0.659	0.612	0.592	0.968	0.557	**0.670**	0.476	0.527	**0.948**
DeepGOA	**0.422**	0.443	0.403	**0.420**	0.906	**0.673**	0.684	**0.661**	0.621	0.976	**0.558**	0.667	**0.480**	**0.528**	0.947

Table 4. The prediction of the protein (RENT3_ARATH) with different methods.

Real label	DeepGOA	DeepGO	FFPred3	Phylo-PFP
GO:0003674	**GO:0003676**	**GO:0003674**	GO:0000166	GO:0000166
GO:0003676	**GO:0003723**	**GO:0005488**	**GO:0003676**	**GO:0003676**
GO:0003723	**GO:0005488**		**GO:0003723**	**GO:0003723**
GO:0003729	**GO:0097159**		GO:0003779	**GO:0005488**
GO:0005488	**GO:1901363**		GO:0008092	GO:0036094
GO:0044822			GO:0008134	**GO:0097159**
GO:0097159			GO:0015631	GO:1901265
GO:1901363			GO:0036094	**GO:1901363**
GO:1901576			**GO:0097159**	
GO:1901661				
GO:1901663				

to common functions while the PPI features are useful for both common and uncommon functions.

5 Discussions and Conclusions

Due to the development of high throughput measures, there are diverse heterogeneous data that are created, such as protein sequences, PPI networks and so on. Many techniques and computational tools have been proposed to predict protein functions with various categories of data. There are still some challenges for predicting protein functions. Firstly, there are proteins that need to interact with neighbor proteins to achieve the functions in many situations. Secondly, it is not obvious which kind of features is efficiently useful for large amounts of proteins. In this study, we propose a deep learning model called DeepGOA that combines protein sequences and PPI networks. First, we represent protein sequences with the word2vec technique and use one-hot coding to represent information of protein domains, families, motifs from InterPro. Second, DeepGOA extracts global features and local features of sequences, with Bi-LSTM and Multi-scale convolutional year, respectively. Moreover, a few fully connected layers are used to generate high-quality features of protein domains, families, and motifs. Then, DeepGOA combines these features to create a comprehensive sequence. Finally, a combination of comprehensive sequence features and topological features of the PPI network is fed into the classification section of DeepGOA. The source code of DeepGOA is available at https://github.com/CSUBioGroup/DeepGOA.

The results show that DeepGOA outperforms BLAST, DeepGO, and FFPred3 in terms of all assessment metrics. We observe that our models achieve higher performance with topological features from the PPI network in both BP and CC branches. Protein domains, families and motifs are substantially useful for the prediction of molecular functions. The possible future work is integrating additional heterogeneous data, such as gene co-expression [54], protein structure [55], text mining [56]. It is also possible to further improve the performance by effectively using GO term information.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants (No. 61832019,

No. 61622213 and No. 61728211), the Project (G20190018001), the Hunan Provincial Science and Technology Program (2018WK4001), and the Hunan Graduate Research and Innovation Project (CX20190082).

REFERENCES

[1] R. R. Rani, and D. Ramyachitra, "A SURVEY ON PROTEIN FUNCTION PREDICTION: COMPUTATIONAL METHODS AND TOOLS," 2018.

[2] E. Boutet, D. Lieberherr, M. Tognolli, M. Schneider, P. Bansal, A. J. Bridge, S. Poux, L. Bougueleret, and I. Xenarios, "UniProtKB/Swiss-Prot, the manually annotated section of the UniProt KnowledgeBase: how to use the entry view," *Plant Bioinformatics*, pp. 23-54: Springer, 2016.

[3] M. Costanzo, B. VanderSluis, E. N. Koch, A. Baryshnikova, C. Pons, G. Tan, W. Wang, M. Usaj, J. Hanchard, and S. D. Lee, "A global genetic interaction network maps a wiring diagram of cellular function," *Science,* vol. 353, no. 6306, pp. aaf1420, 2016.

[4] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *Journal of molecular biology,* vol. 215, no. 3, pp. 403-410, 1990.

[5] D. Cozzetto, F. Minneci, H. Currant, and D. T. Jones, "Ffpred 3: feature-based function prediction for all gene ontology domains," *Scientific reports,* vol. 6, pp. 31865, 2016.

[6] R. You, Z. Zhang, Y. Xiong, F. Sun, H. Mamitsuka, and S. Zhu, "GOLabeler: improving sequence-based large-scale protein function prediction by learning to rank," *Bioinformatics,* vol. 1, pp. 9, 2018.

[7] A. Franceschini, J. Lin, C. von Mering, and L. J. Jensen, "SVD-phy: improved prediction of protein functional associations through singular value decomposition of phylogenetic profiles," *Bioinformatics,* vol. 32, no. 7, pp. 1085-1087, 2015.

[8] S. M. Sahraeian, K. R. Luo, and S. E. Brenner, "SIFTER search: a web server for accurate phylogeny-based protein function prediction," *Nucleic acids research,* vol. 43, no. W1, pp. W141-W147, 2015.

[9] H. Tang, R. D. Finn, and P. D. Thomas, "TreeGrafter: phylogenetic tree-based annotation of proteins with Gene Ontology terms and other annotations," *Bioinformatics,* vol. 35, no. 3, pp. 518-520, 2018.

[10] L. Tran, "Hypergraph and protein function prediction with gene expression data," *arXiv preprint arXiv:1212.0388*, 2012.

[11] W. Peng, M. Li, L. Chen, and L. Wang, "Predicting protein functions by using unbalanced random walk algorithm on three biological networks," *IEEE/ACM transactions on computational biology and bioinformatics,* vol. 14, no. 2, pp. 360-369, 2017.

[12] A. Sokolov, and A. Ben-Hur, "Hierarchical classification of gene ontology terms using the GOstruct method," *Journal of bioinformatics and computational biology,* vol. 8, no. 02, pp. 357-376, 2010.

[13] J. Q. Jiang, and L. J. McQuay, "Predicting protein function by multi-label correlated semi-supervised learning," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB),* vol. 9, no. 4, pp. 1059-1069, 2012.

[14] M. Kirac, and G. Ozsoyoglu, "Protein function prediction based on patterns in biological networks." pp. 197-213.

[15] C. D. Nguyen, K. J. Gardiner, and K. J. Cios, "Protein annotation from protein interaction networks and Gene Ontology," *Journal of biomedical informatics,* vol. 44, no. 5, pp. 824-829, 2011.

[16] J. Hou, *New Approaches of Protein Function Prediction from Protein Interaction Networks*: Academic Press, 2017.

[17] R. Sharan, I. Ulitsky, and R. Shamir, "Network - based prediction of protein function," *Molecular systems biology,* vol. 3, no. 1, pp. 88, 2007.

[18] R. You, S. Yao, Y. Xiong, X. Huang, F. Sun, H. Mamitsuka, and S. Zhu, "NetGO: improving large-scale protein function prediction with massive network information," *Nucleic acids research,* vol. 47, no. W1, pp. W379-W387, 2019.

[19] V. Gligorijević, M. Barot, and R. Bonneau, "deepNF: Deep network fusion for protein function prediction," *Bioinformatics*, 2018.

[20] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, and J. T. Eppig, "Gene Ontology: tool for the unification of biology," *Nature genetics,* vol. 25, no. 1, pp. 25, 2000.

[21] M. Kulmanov, M. A. Khan, and R. Hoehndorf, "DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier," *Bioinformatics,* vol. 34, no. 4, pp. 660-668, 2017.

[22] M. Li, Z. Fei, M. Zeng, F. Wu, Y. Li, Y. Pan, and J. Wang, "Automated ICD-9 Coding via A Deep Learning Approach," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, no. 1, pp. 1-1, 2018.

[23] M. Zeng, M. Li, Z. Fei, Y. Yu, Y. Pan, and J. Wang, "Automatic ICD-9 coding via deep transfer learning," *Neurocomputing,* vol. 324, pp. 43-50, 2019.

[24] F. Zhang, H. Song, M. Zeng, Y. Li, L. Kurgan, and M. J. P. Li, "DeepFunc: A Deep Learning Framework for Accurate Prediction of Protein Functions from Protein Sequences and Interactions," pp. 1900019, 2019.

[25] M. Li, Y. Wang, R. Zheng, X. Shi, F. Wu, and J. Wang, "DeepDSC: A Deep Learning Method to Predict Drug Sensitivity of Cancer Cell Lines," *IEEE/ACM transactions on computational biology and bioinformatics*, 2019.

[26] M. Zeng, F. Zhang, F.-X. Wu, Y. Li, J. Wang, and M. Li, "Protein–protein interaction site prediction through combining local and global features with deep neural networks," *Bioinformatics*, 2019.

[27] A. Graves, and J. J. N. n. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," vol. 18, no. 5-6, pp. 602-610, 2005.

[28] A. Mitchell, H.-Y. Chang, L. Daugherty, M. Fraser, S. Hunter, R. Lopez, C. McAnulla, C. McMenamin, G. Nuka, and S. Pesseat, "The InterPro protein families database: the classification resource after 15 years," *Nucleic acids research,* vol. 43, no. D1, pp. D213-D221, 2014.

[29] M. Zeng, M. Li, Z. Fei, F. Wu, Y. Li, Y. Pan, and J. Wang, "A deep learning framework for identifying essential proteins by integrating multiple types of biological information," *IEEE/ACM Trans Comput Biol Bioinform*, Feb 5, 2019.

[30] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality." pp. 3111-3119.

[31] J. L. Elman, "Finding structure in time," *Cognitive science,* vol.

14, no. 2, pp. 179-211, 1990.

[32] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning." p. 12.

[33] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[34] A. Marchler-Bauer, M. K. Derbyshire, N. R. Gonzales, S. Lu, F. Chitsaz, L. Y. Geer, R. C. Geer, J. He, M. Gwadz, and D. I. Hurwitz, "CDD: NCBI's conserved domain database," *Nucleic acids research,* vol. 43, no. D1, pp. D222-D226, 2014.

[35] E. L. Sonnhammer, S. R. Eddy, and R. Durbin, "Pfam: a comprehensive database of protein domain families based on seed alignments," *Proteins: Structure, Function, and Bioinformatics,* vol. 28, no. 3, pp. 405-420, 1997.

[36] I. Sillitoe, T. E. Lewis, A. Cuff, S. Das, P. Ashford, N. L. Dawson, N. Furnham, R. A. Laskowski, D. Lee, and J. G. Lees, "CATH: comprehensive structural and functional annotations for genome sequences," *Nucleic acids research,* vol. 43, no. D1, pp. D376-D381, 2014.

[37] D. A. de Lima Morais, H. Fang, O. J. Rackham, D. Wilson, R. Pethica, C. Chothia, and J. Gough, "SUPERFAMILY 1.75 including a domain-centric gene ontology method," *Nucleic acids research,* vol. 39, no. suppl_1, pp. D427-D434, 2010.

[38] M. Li, X. Meng, R. Zheng, F.-X. Wu, Y. Li, Y. Pan, and J. Wang, "Identification of protein complexes by using a spatial and temporal active protein interaction network," *IEEE/ACM transactions on computational biology and bioinformatics*, 2017.

[39] M. Li, P. Ni, X. Chen, J. Wang, F. Wu, and Y. Pan, "Construction of refined protein interaction network for predicting essential proteins," *IEEE/ACM transactions on computational biology and bioinformatics*, 2017.

[40] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations." pp. 701-710.

[41] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding." pp. 1067-1077.

[42] A. Grover, and J. Leskovec, "node2vec: Scalable feature learning for networks." pp. 855-864.

[43] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, "Asymmetric transitivity preserving graph embedding." pp. 1105-1114.

[44] S. Wang, J. Tang, C. Aggarwal, Y. Chang, and H. Liu, "Signed network embedding in social media." pp. 327-335.

[45] S. Yuan, X. Wu, and Y. Xiang, "SNE: signed network embedding." pp. 183-195.

[46] I. Jolliffe, "Principal component analysis," *International encyclopedia of statistical science*, pp. 1094-1096: Springer, 2011.

[47] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika,* vol. 29, no. 1, pp. 1-27, 1964.

[48] M. Zeng, B. Zou, F. Wei, X. Liu, and L. Wang, "Effective prediction of three common diseases by combining SMOTE with Tomek links technique for imbalanced medical data." pp. 225-228.

[49] M. Zeng, M. Li, Z. Fei, F.-X. Wu, Y. Li, and Y. Pan, "A Deep Learning Framework for Identifying Essential Proteins Based on Protein-Protein Interaction Network and Gene Expression Data."

[50] D. Szklarczyk, A. Franceschini, S. Wyder, K. Forslund, D. Heller, J. Huerta-Cepas, M. Simonovic, A. Roth, A. Santos, and K. P. Tsafou, "STRING v10: protein–protein interaction networks, integrated over the tree of life," *Nucleic acids research,* vol. 43, no. D1, pp. D447-D452, 2014.

[51] J. Huerta-Cepas, D. Szklarczyk, K. Forslund, H. Cook, D. Heller, M. C. Walter, T. Rattei, D. R. Mende, S. Sunagawa, and M. Kuhn, "eggNOG 4.5: a hierarchical orthology framework with improved functional annotations for eukaryotic, prokaryotic and viral sequences," *Nucleic acids research,* vol. 44, no. D1, pp. D286-D293, 2015.

[52] A. Paszke, S. Gross, S. Chintala, and G. Chanan, "PyTorch," 2017.

[53] P. Radivojac, W. T. Clark, T. R. Oron, A. M. Schnoes, T. Wittkop, A. Sokolov, K. Graim, C. Funk, K. Verspoor, and A. Ben-Hur, "A large-scale evaluation of computational protein function prediction," *Nature methods,* vol. 10, no. 3, pp. 221, 2013.

[54] M. N. Wass, G. Barton, and M. J. Sternberg, "CombFunc: predicting protein function using heterogeneous data sources," *Nucleic acids research,* vol. 40, no. W1, pp. W466-W470, 2012.

[55] C. Zhang, P. L. Freddolino, and Y. Zhang, "COFACTOR: improved protein function prediction by combining structure, sequence and protein–protein interaction information," *Nucleic acids research,* vol. 45, no. W1, pp. W291-W299, 2017.

[56] R. Cao, and J. Cheng, "Integrated protein function prediction by mining function associations, sequences, and protein–protein and gene–gene interaction networks," *Methods,* vol. 93, pp. 84-91, 2016.

Fuhao Zhang received his BSc degrees in Chongqing University of Posts and Telecommunications, China in 2014. He is currently a postgraduate student in Bioinformatics at Central South University. His current research interests include bioinformatics, network representation learning, and deep learning.

Hong Song received Ph.D. degrees in Computer Engineering from Central South University, China, in 2010. She is an associate professor in the School of Computer Science and Engineering, Central South University, Changsha, Hunan, P.R. China. Her current research interests include information security, transparent computing, and operating system.

Min Zeng received a B.S. degree from Lanzhou University in 2013, and the M.S. degree from Central South University in 2016. He is currently working toward the Ph.D. degree in the School of Computer Science and Engineering, Central South University, China. His research interests include bioinformatics, machine learning and deep learning.

Fang-Xiang Wu (M'06-SM'11) received the B.Sc. degree and the M.Sc. degree in applied mathematics, both from Dalian University of Technology, Dalian, China, in 1990 and 1993, respectively, the first Ph.D. degree in control theory and its applications from Northwestern Polytechnical University, Xi'an, China, in

1998, and the second Ph.D. degree in biomedical engineering from University of Saskatchewan (U of S), Saskatoon, Canada, in 2004. During 2004-2005, he worked as a Postdoctoral Fellow at the Laval University Medical Research Center (CHUL), Quebec City, Canada. He is currently a Professor of the Division of Biomedical Engineering and the Department of Mechanical Engineering at the U of S. His current research interests include computational and systems biology, genomic and proteomic data analysis, biological system identification and parameter estimation, applications of control theory to biological systems. Dr. Wu is serving as the editorial board member of five international journals, the guest editor of several international journals, and as the program committee chair or member of several international conferences. He has also reviewed papers for many international journals.

Yaohang Li received the M.S. and Ph.D. degrees in computer science from Florida State University, Tallahassee, FL, USA, in 2000 and 2003, respectively. He is an Associate Professor in the Department of Computer Science at Old Dominion University, Norfolk, VA, USA. His research interests are in computational biology, Monte Carlo methods, and scientific computing. After graduation, he worked at Oak Ridge National Laboratory as a Research Associate for a short period. Before joining ODU, he was an Associate Professor in the Computer Science Department at North Carolina A&T State University.

Yi Pan is a Regents' Professor of Computer Science and an Interim Associate Dean and Chair of Biology at Georgia State University, USA. Dr. Pan joined Georgia State University in 2000 and was promoted to full professor in 2004, named a Distinguished University Professor in 2013 and designated a Regents' Professor (the highest recognition given to a faculty member by the University System of Georgia) in 2015. He served as the Chair of the Computer Science Department from 2005-2013. He is also a visiting Changjiang Chair Professor at Central South University, China. Dr. Pan received his B.Eng. and M.Eng. degrees in computer engineering from Tsinghua University, China, in 1982 and 1984, respectively, and his Ph.D. degree in computer science from the University of Pittsburgh, USA, in 1991. His profile has been featured as a distinguished alumnus in both Tsinghua Alumni Newsletter and the University of Pittsburgh CS Alumni Newsletter. Dr. Pan's research interests include parallel and cloud computing, wireless networks, and bioinformatics. Dr. Pan has published more than 330 papers including over 180 SCI journal papers and 60 IEEE/ACM Transactions papers. In addition, he has edited/authored 40 books. His work has been cited more than 6500 times. Dr. Pan has served as an editor-in-chief or editorial board member for 15 journals including 7 IEEE Transactions. He is the recipient of many awards including IEEE Transactions Best Paper Award, 4 other international conferences or journal Best Paper Awards, 4 IBM Faculty Awards, 2 JSPS Senior Invitation Fellowships, IEEE BIBE Outstanding Achievement Award, NSF Research Opportunity Award, and AFOSR Summer Faculty Research Fellowship. He has organized many international conferences and delivered keynote speeches at over 50 international conferences around the world.

Min Li received the PhD degree in Computer Science from Central South University, China, in 2008. She is currently the vice dean and a Professor at the School of Computer Science and Engineering, Central South University, Changsha, Hunan, P.R. China. Her research interests include computational biology, systems biology and bioinformatics. She has published more than 80 technical papers in refereed journals such as Bioinformatics, IEEE/ACM Transactions on Computational Biology and Bioinformatics, Proteomics, and conference proceedings such as BIBM, GIW and ISBRA.