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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

A deep learning framework for identifying essential proteins by integrating multiple types of biological information

Min Zeng, Min Li*, Zhihui Fei, Fang-Xiang Wu, Yaohang Li, Yi Pan and Jianxin Wang

Abstract—Computational methods including centrality and machine learning-based methods have been proposed to identify essential proteins for understanding the minimum requirements of the survival and evolution of a cell. In centrality methods, researchers are required to design a score function which is based on prior knowledge, yet is usually not sufficient to capture the complexity of biological information. In machine learning-based methods, some selected biological features cannot represent the complete properties of biological information as they lack a computational framework to automatically select features. To tackle these problems, we propose a deep learning framework to automatically learn biological features without prior knowledge. We use node2vec technique to automatically learn a richer representation of protein-protein interaction (PPI) network topologies than a score function. Bidirectional long short term memory cells are applied to capture non-local relationships in gene expression data. For subcellular localization information, we exploit a high dimensional indicator vector to characterize their feature. To evaluate the performance of our method, we tested it on PPI network of S. cerevisiae. Our experimental results demonstrate that the performance of our method is better than traditional centrality methods and is superior to existing machine learning-based methods. To explore which of the three types of biological information is the most vital element, we conduct an ablation study by removing each component in turn. Our results show that the PPI network embedding contributes most to the improvement. In addition, gene expression profiles and subcellular localization information are also helpful to improve the performance in identification of essential proteins.

Index Terms—Deep learning, essential proteins, protein-protein interaction network, gene expression, subcellular localization.

1 Introduction

Essential proteins which play important roles in various biological activities are indispensable in cellular life [1, 2]. If one of essential proteins has been removed from an organism, then the organism cannot survive or develop [3]. Thus identification of essential proteins is of great significance in biology. Determination of essential proteins can help us to understand the minimum requirements of the survival and evolution of a cell. Additionally, essential proteins are potential targets of new antibacterial drugs and can help find human disease genes. During the past several decades, traditional biological experimental methods, such as single gene knockout [4], conditional knockout [5], and RNA interference [6, 7] have been employed to identify essential proteins. However, these biological experimental methods are expensive, time-consuming and laborious. On the other hand, traditional biological experimental methods have some limitations -- they are not suitable for humans and other complex organisms. Thus developing accurate computational approaches to identify essential proteins would be of great value to biologists.

For more than two decades, a lot of computational approaches for identifying essential proteins have been developed. Jeong et al. [8] pointed out that there is a positive correlation between the topological properties of proteins in protein-protein interaction (PPI) networks and protein essentiality. Thus a lot of centrality methods which are based on topological features of PPI networks, such as degree centrality (DC) [9], betweenness centrality (BC) [10], closeness centrality (CC) [11], subgragh centrality (SC) [12], eigenvector centrality (EC) [13], information centrality (IC) [14] and edge clustering coefficient centrality (NC) [15], have been proposed to identify essential proteins. These centrality methods have been integrated in the cytoscape plugins CytoNCA[16] and DyNetViewer [17]. Additionally, previous published studies have pointed out that some biological information associates with gene essentiality [18]. For example, gene expression profiles and subcellular localization information have some relationship with gene essentiality. Some researchers believed that gene expression profiles are useful to identify essential proteins because proteins are products of gene expressions. Localization of proteins in cells are usually related to protein functions because most essential biological processes take place in certain subcellular

- *M. Zeng, M. Li, Z. Fei, and J. Wang are with the School of Information Science and Engineering, Central South University, Changsha, P.R. China. Email: zengmin@csu.edu.cn, limin@mail.csu.edu.cn, zhihuifei@foxmail.com, jxwang@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*

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

localization [19, 20]. On the basis of these ideas, methods which combine network topological features with different biological information have been proposed [21-27].

With the rapid development of high-throughput sequencing techniques, a lot of protein sequences and their properties have been obtained, which make it possible for us to use machine learning methods. Recent years, some machine learning-based methods have been used for identifying essential proteins and the most common used machine learning algorithms are support vector machine (SVM) [28, 29] and ensemble learning-based model [30, 31]. Additionally, Naive Bayes, decision tree, neural network, and genetic algorithms are also commonly used algorithms [32-37].

Although centrality and mahine learning-based methods have obtained good results, they still have some limitations and there is room for the improvement. Regardless of a centrality method or a machine learning-based method, the biggest limitation is the feature representation of biological information including network topology of PPI network, gene expression, and subcellular localization. In centrality methods, researchers usually design a score function to represent the importance of each piece of biological information and combine these functions into an equation to determine the essentiality of a protein. Although centrality method is simple and convenient, but the main drawback is that we have to master a lot of prior knowledge to design a good score function. Moreover, the designed score function cannot always characterize the comprehensive biological information. For example, a single value of a score function cannot represent the comprehensive topological information of PPI networks. A PPI network usually has thousands of vertices and tens of thousands edges and the output of designed score function is just a real number. Yet it is difficult to represent the complete topological features by a real number. In machine learning-based methods, we usually collect some biological properties as features and then apply them into a machine learning classifier. The main problem with this method is that some selected biological features cannot represent the complete properties of biological information. For example, network topological features were the widely used features in previous studies. DC, BC, and CC are the most frequently used network topological features. Furthermore, there is lack of a computational framework to automatically select features from various pieces of biological information. The commonly used method for the feature selection is based on the results of statistical methods and the prior knowledge of researchers. As a result, it is difficult to explain why these features are chosen and what roles they play.

To tackle the above limitations, we propose a deep learning framework to automatically learn biological features without prior knowledge. We have used three different types of biological information including PPI network, gene expression and subcellular localization information in our study. Specifically, for PPI network, we employ a network representation learning technique called node2vec [38] to learn its topological features automatically, which have two advantages. First, it can learn a richer representation of PPI network than a score function in centrality methods. Second, it can automatically learn dense vectors without manually selecting some topological features such as DC, BC, and CC. For gene expression data, we use bidirectional long short term memory (LSTM) cells [39] to extract features. Gene expression data are sequential data and bidirectional LSTM cells can capture non-local relationships in sequential data more efficiently than a score function. For subcellular localization information, we exploit an indicator vector to characterize their features. Compared with using a score function, the indicator vector has higher dimension and richer representation.

We carry out computational experiments on the PPI network of S. cerevisiae. Accuracy, precision, recall, F-measure and AUC (Area Under receiver operating characteristic Curve) obtained by our method are 0.850, 0.680, 0.505, 0.579 and 0.832, respectively, which is better than traditional centrality methods including DC, BC, CC, EC, NC, local average connectivity (LAC) [40], PeC [23], and WDC [41]. It also outperforms machine learning methods including SVM, decision tree, random forest and Adaboost. In order to investigate what role each piece of biological information plays in the success of our proposed deep learning framework, we compared the results obtained by removing each individual component in our network. Detailed analyses show that the PPI network embedding is the major contribution to the improvement. Gene expression data and subcellular localization information are also helpful for improving the performance of essential proteins identification as auxiliary biological information.

2 METHODS

In this section, we first present an overview of our proposed deep learning framework in section 2.1 and then give the details of network representation learning, recurrent neural networks, subcellular localization information embedding, and assessment metrics in sections 2.2-2.5, respectively.

2.1 Network Architecture

As shown in Figure 1, our deep learning framework for identifying essential protein is an end-to-end model. The framework consists of two parts, biological information feature extraction and classification part. In this study, we make use of three types of biological data, PPI network, gene expression, and subcellular localization information. The biological information feature extraction part is responsible for learning useful features and patterns from these different types of biological data. For PPI network, a network representation learning technique called node2vec is utilized to learn a dense vector for each vertex to represent the topological information of the network. Then a fully connected layer with rectified linear unit (ReLU) activation function [42] is used for continue processing. Considering gene expression data are sequential data, we employ a recurrent neural network module called bidirectional long short term memory (LSTM) to

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

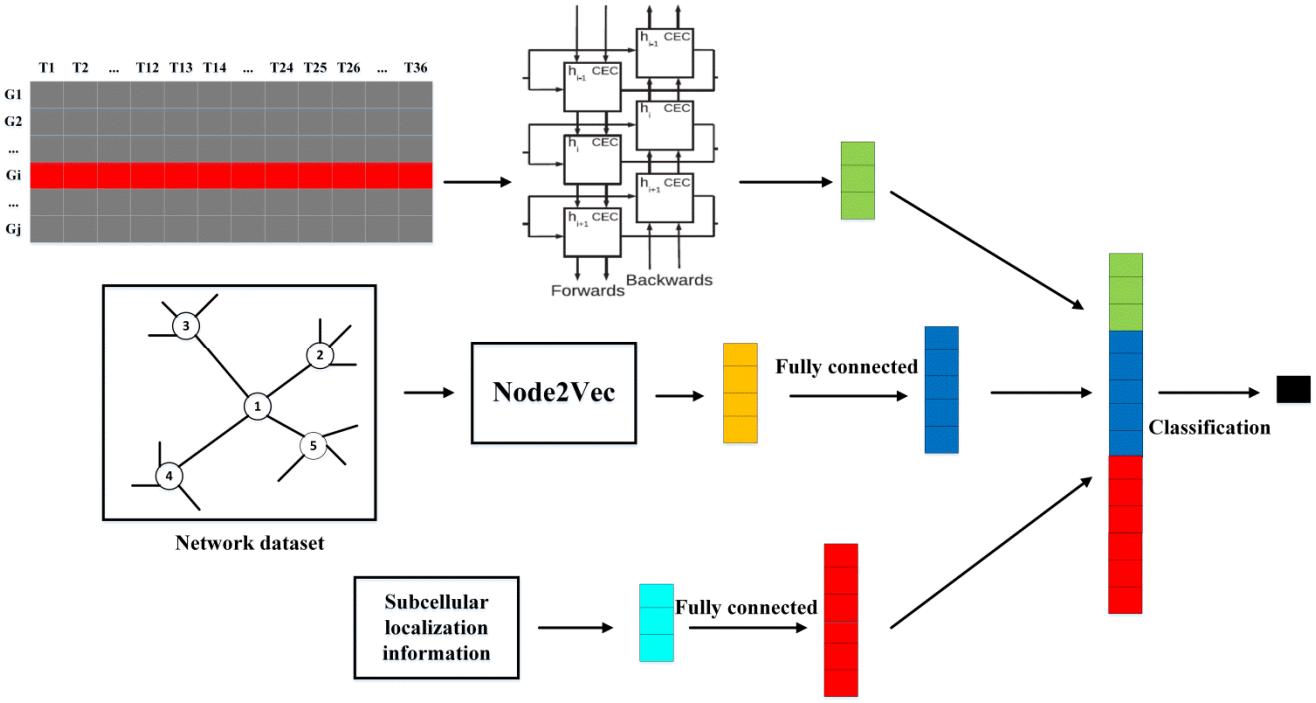

Fig. 1. An overview of proposed deep learning framework for identifying essential proteins. The input consists of three parts: PPI network, gene expression profiles, and subcellular localization information. For different types of biological data, different preprocessing methods are used. For PPI network, we use node2vec technique to obtain a 64-dimensional vector. The 64-dimensional vector is fed into a fully connected layer with 312 hidden units. We use an 11-dimensional vector to encode the information of subcellular localization, and the 11-dimensional vector is fed into a fully connected layer with 512 hidden units. For gene expression data, a Bi-LSTM with 8 hidden units is used to capture patterns. The output is a 16-dimensional vector. Then we concatenate the three vectors to perform classification task.

extract long-range dependencies. For subcellular localization information, we design an indicator vector to represent the subcellular localizations of each protein. Then we utilize a fully connected layer to continuously process it. After the part of biological information feature extraction, three output vectors are concatenated together as the input of classification part which is made of a fully connected layer with sigmoid function.

2.2 Network Representation Learning

As previously mentioned, the topological properties of proteins in PPI network have a direct relationship with gene essentiality. Thus network topological feature extraction plays a vital role in the study of identifying essential proteins. In recent years researchers have proposed a lot of centrality methods to predict essential proteins. Meanwhile a lot of machine learning-based methods select some of these centralities as the features to train their classifiers. However, the output of these centrality methods is a real number and the topological features of a complex biological network usually cannot be captured by a real number. We believe it can offer a number of benefits by using a vector to replace a real number to capture the topological features of a biological network. Motivated by this idea, we employ the network representation learning to learn network topological features without any prior knowledge.

Network representation learning aims at learning dense vector representation automatically for each vertex in a network. These learned dense vectors contain rich semantic and topological information and they can be applied to improve the performance of network analysis tasks. In this study, we use the node2vec technique. It is a deep learning method which learns vector representations of vertexes based on local network information. Node2vec technique utilizes a biased random walk algorithm to obtain each vertex sequence. For a source node u, node2vec uses the following formula to generate a sequence which has a fixed length of l.

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{\pi_{vx}}{Z} & \text{if } (v,x) \epsilon E \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where c_i stands for the ith node in the walk, π_{vx} is the unnormalized transition probability between nodes v and x, and Z is the normalization constant.

The 2nd order random walk with two parameters p and q is applied to guide this walk. Researchers set the unnormalized transition probability to $\pi_{vx} = \alpha_{pq}(t,x) \cdot w_{vx}$, $\alpha_{pq}(t,x)$ is given by the formula below.

$$\alpha_{pq}(t,x) = \begin{cases} \frac{1}{p} & \text{if } d_{tx} = 0 \\ 1 & \text{if } d_{tx} = 1 \\ \frac{1}{q} & \text{if } d_{tx} = 2 \end{cases} \quad (2)$$

where d_{tx} denotes the shortest path distance between nodes t and x.

Node2vec technique introduces the Skip-Gram model which is a powerful and effective word representation method in the field of network representation learning to obtain represative vectors. The Skip-Gram model [43] is employed to predict surrounding context words given a center word. Following the idea of the Skip-Gram model,

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

we aim at maximizing the co-occurrence likelihood between a target vertex and its context vertices. Then the learned dense vectors are successively updated by using the Skip-Gram model. These dense vectors are considered to be rich topological representation of a network.

2.3 Recurrent Neural Networks

The essentiality of a protein not only depends on topological features of PPI networks, but also gene expression data. Some researchers have pointed that gene expression data can improve the performance for identifying essential proteins [22, 23, 37, 44, 45]. The gene expression data we used are sequential data and we thus use recurrent neural networks (RNNs) to handle them.

RNNs are a family of deep neural networks that have a powerful capacity to deal with sequential data. RNNs can produce an output at each time step and have recurrent connections between hidden units. Thus they can read complete sequence and then produce outputs. However, the conventional RNNs only use previous context of the input sequence and they cannot exploit future context [46]. Inspired by the idea, bidirectional RNNs have been proposed to process sequential data in both directions with two hidden layers. LSTM uses three gates (i.e. input gate, forget gate, output gate) to store and process information. The mechanism of a LSTM can be presented as follows.

$$\begin{aligned} i_t &= \sigma(W_i \mathrm{h}_{t-1} + U_i x_t + b_i) \\ f_t &= \sigma\left(W_f \mathrm{h}_{t-1} + U_f x_t + b_f\right) \\ \tilde{c}_t &= \tanh(W_c \mathrm{h}_{t-1} + U_c x_t + b_c) \\ c_t &= \mathrm{f_t} \odot \mathrm{c_{t-1}} + \mathrm{i_t} \odot \tilde{\mathrm{c}}_\mathrm{t} \\ o_t &= \sigma(W_o \mathrm{h}_{t-1} + U_o x_t + b_o) \\ \mathrm{h}_t &= o_t \odot \tanh(c_t) \end{aligned} \tag{3}$$

where $\mathrm{W_i}$, $\mathrm{W_f}$, $\mathrm{W_c}$, $\mathrm{W_o}$,$\mathrm{U_i}$, $\mathrm{U_f}$, $\mathrm{U_c}$, $\mathrm{U_o}$ are weight matrices and $\mathrm{b_i}$, $\mathrm{b_f}$, $\mathrm{b_c}$, $\mathrm{b_o}$ are bias terms; σ, tanh and ⊙ denote element-wise sigmoid, hyperbolic and product functions, respectively.

Bidirectional LSTM, which combines bidirectional RNNs with LSTM, has more powerful ability than conventional RNNs and LSTM [47]. Considering the success of RNNs in sequential data, we apply bidirectional LSTM cells to extract features of gene expression data. Bidirectional LSTM cells combine two LSTM cells, one moves forward from the start of the sequence and another moves forward from the end of the sequence. Thus the output of each time step depends on both the past and the future data, which can access long-range context in complete sequence.

2.4 Subcellular Localization Information Embedding

In addition to network topological features of PPI networks and gene expression data, subcellular localization information also associates with protein essentiality. Most essential biological processes take place in certain subcellular localization. Thus it is reasonable to believe that localization of proteins in a cell usually determines the protein functions. To better identify essential proteins, subcellular localization information is used in our experiments. According to previous studies, the subcellular localizations are usually classified into eleven categories: 1) Cytoskeleton, 2) Cytosol, 3) Endoplasmic, 4) Endosome, 5) Extracellular, 6) Golgi, 7) Mitochondrion, 8) Nucleus, 9) Peroxisome, 10) Plasma and 11) Vacuole. We count the number of essential and non-essential proteins in each subcellular location and present these statistical results in Table 1.

TABLE 1.
THE NUMBER OF ESSENTIAL AND NON-ESSENTIAL PROTEINS IN EACH SUBCELLULAR LOCATION.

Subcellular localization	Number of essential proteins	Number of Non-essential proteins
Cytoskeleton	95	133
Cytosol	138	289
Endoplasmic	137	292
Endosome	22	109
Extracellular	1	70
Golgi	61	184
Mitochondrion	173	753
Nucleus	809	1407
Peroxisome	4	61
Plasma	53	354
Vacuole	19	238

Centrality methods usually design a score function to measure the contribution of subcellular localization information. These methods are both simple and convenient. However, there are two limitations. First, researchers are required to master a lot of prior knowledge in the field of proteomics to manually design a score function. Nevertheless, researchers come from different background, such as computer and life science, which cause some difficulty to explain which score function contains enough biological information. Second, these score functions only use a real number to measure the contribution of subcellular localization information. As previously mentioned in section 2.2, biological process is very complicated and thus is difficult to measure with a real number. To better embed subcellular localization information, we exploit an indicator vector to characterize their features. The indicator vector is encoded in the following way. The subcellular localizations are generally classified into eleven categories, and thus we use an 11-dimensional vector to encode this information. In this 11-dimensional vector, each dimension represents a specific subcellular localization of a protein. If a protein performs its function at a certain subcellular localization, we assign the value of one to that localization; otherwise we assign the value of zero. For instance, in Figure 2, a protein performs its function at cytoskeleton and nucleus. Thus we assign one to these two dimensions and zero to others dimensions. Using such an indicator vector has several advantages. First, the indicator vector is a high dimensional vector that will have richer expression ability than a real number. Second, there is no need to have prior knowledge for researchers. We just input the raw information of subcellular localiza-

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

tion and the deep learning framework can learn parameters automatically.

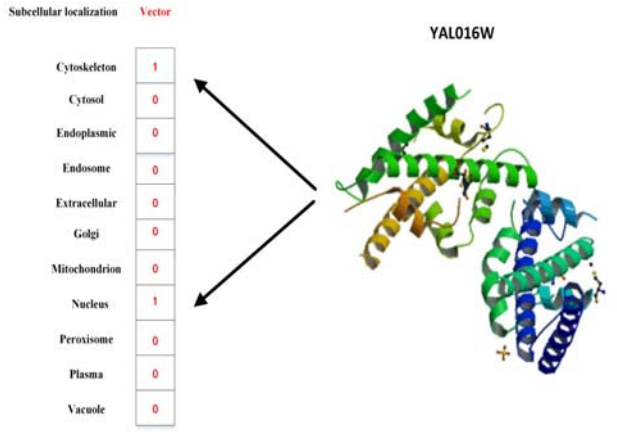

Fig. 2. An example of subcellular localization information embedding.

2.5 Assessment Metrics

To assess the performance of our deep learning framework and other methods in identifying essential proteins, six measures: accuracy, precision, recall, F-measure, AUC, and average precision (AP) score are used. Accuracy is defined as:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad (4)$$

Where TP and TN represent the number of samples of the essential and non-essential proteins which are classified correctly, respectively, and FN and FP represent the number of samples of the essential and non-essential proteins which are classified wrongly, respectively.

Precision and recall are defined as:

$$precision = TP/(TP + FP) \quad (5)$$

$$recall = TP/(TP + FN) \quad (6)$$

Precision and AUC are the most important effective assessment metrics. F-measure is a tradeoff of precision and recall, and it is defined as:

$$F - measure = \frac{(1+\beta^2)\cdot recall\cdot precision}{\beta^2\cdot precision + recall} \quad (7)$$

where β is a parameter to adjust the weight between precision and recall. In this study, we set β=1.

AUC is defined as the area under the Receiver Operating Characteristic (ROC) curve. In general, a classifier which provides a larger AUC shows it has better performance. AUC is 1 represents perfect performance.

AP score is the area under the precision-recall (PR) curve and it summarizes a PR curve as the weighted mean of precisions achieved at each threshold. PR curve is a statistical method used for visualizing and evaluating classifiers. It has been widely used for performance evaluation in identifying essential proteins.

It is worth noting that in an imbalanced learning problem, we pay more attention to F-measure, AUC, and AP score, because these metrics can provide more insight into the performance of a classifier than accuracy, precision, and recall.

3 Data Sources

This study uses multiple biological datasets, including PPI network dataset, essential protein dataset, gene expression dataset and subcellular localization dataset.

PPI network of S. cerevisiae is the most complete and widely used in the study of identifying essential proteins. This dataset is downloaded from BioGRID database [48]. After removing repeated interaction and self-interactions, the processed dataset consists of 5616 proteins and 52833 interactions.

The essential proteins dataset of S. cerevisiae is obtained from the following databases: MIPS [49], SGD [50], DEG [2] and SGDP. After integrating these databases, we obtained 1199 essential proteins.

Gene expression dataset is retrieved from GEO (Gene Expression Omnibus) [51] with accession number GSE3431, which contain 6777 proteins and 36 time points in total. This dataset has three successive metabolic cycles with 12 time points in a cycle.

Subcellular localization dataset of yeast is downloaded from knowledge channel of COMPARTMENTS database [52] on August 30, 2014. It integrates several source databases including UniProtKB [53], MGD [54], SGD [50], FlyBase [55] and WormBase [56] database. After preprocessing, there are 3923 proteins which have subcellular localization information.

4 Experiments and Discussion

4.1 Implementation Details

There are three different sources of biological data in our experiment and we deal with them separately. After preprocessing three datasets (raw PPI network, gene expression profiles, subcellular localization information), we used 5297 proteins with known gene expression profiles and subcellular localization information in raw PPI network. Among these proteins, 1185 are essential and 4112 are non-essential. The ratio of essential proteins to non-essential proteins is 1: 3.470. For PPI network, the node2vec technique was applied to generate representation vectors. The source codes of node2vec can be obtained from https://github.com/aditya-grover/node2vec. We use the node2vec technique to generate a 64-dimensional vector for each node in the PPI network. The detailed parameters are listed below. The window context size for optimization is 10, the length of walk per source is 20, and the number of walk per source is 10. Then a fully connected layer of 312 units with ReLU activation function is used for preliminary processing. For gene expression data, bidirectional RNNs were utilized to extract long-range dependencies patterns. The bidirectional RNN layer has 8 hidden units and the output from the bidirectional RNN layer is regularized with dropout (= 0.05) to avoid overfitting. For subcellular localization information, we use the embedding technique to obtain an 11-dimensional vector for each protein, each dimension of which represents a specific subcellular localization of a protein. Then we use a fully connected layer of 512 units with ReLU activation function to continuously process it.

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

After using different methods to process these biological data, three different dimensional vectors are obtained. Then the three vectors are concatenated together to be fed to the fully connected layer with sigmoid activate function for classification. Dropout rate of 0.05 was used on fully connected layer in the network to avoid overfitting. The deep learning framework is implemented in Tensorflow [57]. To test the robustness of the model, the 5 fold cross-validation is applied to model evaluation. The Adam optimizer is used in the deep learning framework for training and the initial learning rate is set to 0.0005. The batch size is set to 32.

In node2vec, researchers use the 2nd order random walk with two parameters p and q to sample a node sequence. Parameter q is the in-out parameter and parameter p is the return parameter. Parameter q allows the search to differentiate between "inward" and "outward" nodes. If $q > 1$, the random walk is biased towards nodes close to starting node. If $q < 1$, the walk is more inclined to visit nodes which are further away from the starting node. Parameter p controls the likelihood for immediately revisiting a node in the walk. If $p > \max(q, 1)$, it is less likely to sample an already visited node. If $p < \max(q, 1)$, it would lead the walk to backtrack a step. According to the original paper of the node2vec technique, the best in-out and return hyper-parameters were learned using 5-fold cross-validation with a grid search over $p, q \in \{0.5, 1, 2\}$. The results are shown in Table 2. We find that the best performance is achieved when p=2 and q=1.

TABLE 2.
EFFECTS OF CHANGING DIFFERENT P AND Q ON PREDICTION PERFORMANCE.

Parameters setting	Accuracy	Precision	Recall	F-measure	AUC
p=1, q=0.5	0.846	0.695	0.450	0.546	0.841
p=1, q=1	0.844	0.691	0.440	0.538	0.839
p=1, q=2	0.845	0.708	0.422	0.529	0.835
p=2, q=0.5	0.844	0.691	0.440	0.538	0.836
p=2, q=1	0.850	0.680	0.505	0.579	0.832
p=2, q=2	0.846	0.695	0.450	0.546	0.833
p=0.5, q=0.5	0.853	0.777	0.399	0.527	0.835
p=0.5, q=1	0.847	0.726	0.413	0.526	0.831
p=0.5, q=2	0.855	0.739	0.454	0.563	0.837

4.2 Comparisons with Other Topology-based Methods

To evaluate the performance of our deep learning framework, we compare our model with 6 other common used topology-based methods, DC, BC, CC, EC, NC and LAC. In addition, PeC and WDC, two methods which are based on the integration of PPI and gene expression data are compared. For these topology-based methods, we use the following way for evaluation. First, we calculate the scores of proteins by each topology-based method and ranked them in descending order according the scores. Then we select the top 1185 proteins (our processed PPI network has 1185 essential proteins) ranked by these methods as their predicted essential proteins. The rest of proteins are considered to be non-essential proteins. Finally, according to the comparison with the true labels of essential proteins and non-essential proteins, we obtain a confusion matrix to calculate each metric. In Table 3 we present experimental results including accuracy, precision, recall, and F-measure of DC, BC, CC, EC, NC, LAC, PeC, WDC and our method. From Table 3, we can see that almost all assessment metrics obtained by our deep learning framework are higher than those of other topology-based methods except for recall obtained by NC. Accuracy, precision, recall, and F-measure obtained by our method are 0.850, 0.680, 0.505, and 0.579, respectively, which are better than DC (0.740, 0.436, 0.430 and 0.433), BC (0.722, 0.398, 0.393 and 0.395), CC (0.665, 0.262, 0.260, and 0.261), EC (0.727, 0.408, 0.401, and 0.404), NC (0.752, 0.468, 0.464 and 0.466), LAC (0.745, 0.467, 0.409 and 0.436), PeC (0.747, 0.438, 0.430 and 0.434), and WDC (0.742, 0.455, 0.459, and 0.457). These experimental results show that our method performs better than all those topology-based methods for identifying essential proteins.

TABLE 3.
COMPARISON OF PERFORMANCES OBTAINED BY OUR METHOD AND OTHER TOPOLOGY-BASED METHODS.

Method	Accuracy	Precision	Recall	F-measure
DC	0.740	0.436	0.430	0.433
BC	0.722	0.398	0.393	0.395
CC	0.665	0.262	0.260	0.261
EC	0.727	0.408	0.401	0.404
NC	0.752	0.468	0.464	0.466
LAC	0.745	0.467	0.409	0.436
PeC	0.747	0.438	0.430	0.434
WDC	0.742	0.455	0.459	0.457
Our method	**0.850**	**0.680**	**0.505**	**0.579**

4.3 Comparisons with Other Machine Learning Algorithms

We have also compared our methods with commonly used algorithms including SVM, decision tree, random forest, and Adaboost. Specifically, we concatenate each node vectors which are generated by node2vec, gene expression data, and subcellular localization embedding vector into a long vector as the input of the four machine learning methods. All machine learning methods are implemented by using scikit-learn [58]. To test the robustness

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

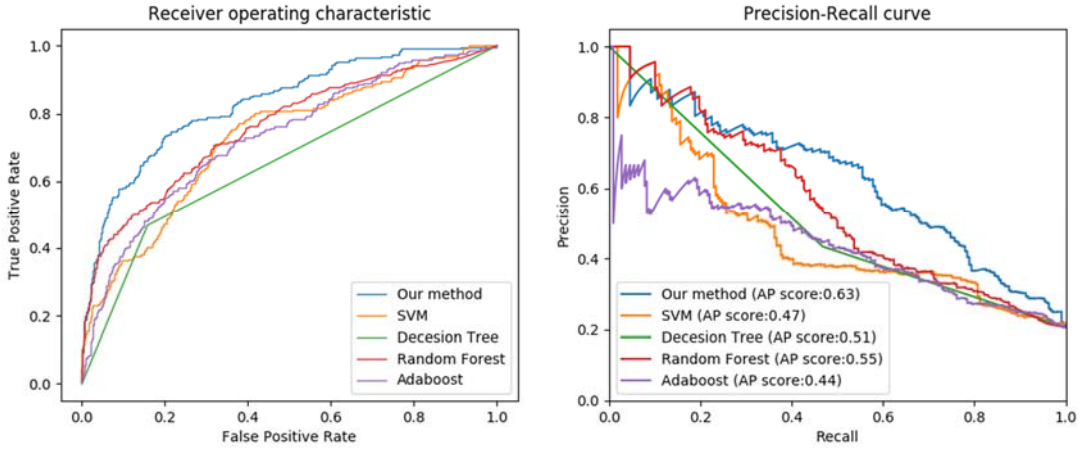

Fig. 3. ROC and PR curves of our model and other machine learning methods.

of these models, a 5 fold cross-validation is applied to model evaluation and the performance results in Table 4. From Table 4, we also find that almost all assessment metrics obtained by our method outperform that of other topology-based methods except for precision obtained by SVM and random forest. Our method obtained accuracy, precision, recall, F-measure, and AUC with values of 0.85, 0.68, 0.50, 0.58, and 0.83, respectively, which is better than SVM (0.82, 0.85, 0.13, 0.23, and 0.73), decision tree (0.76, 0.43, 0.47, 0.45, and 0.65), random forest (0.83, 0.76, 0.23, 0.36, and 0.76), and Adaboost (0.81, 0.55, 0.34, 0.42, and 0.73). Figure 3 shows the ROC and PR curves of our method and other machine learning methods. The ROC curve of our method is significantly higher than other machine learning methods. Furthermore, we obtained the AP score of 0.63 which is better than SVM (0.47), decision tree (0.51), random forest (0.55), and Adaboost (0.44). From these results, we can conclude that our method significantly outperforms other machine learning methods.

TABLE 4.
PERFORMANCES (ACCURACY, PRECISION, RECALL, F-MEASURE, AND AUC) OF OUR MODEL AND OTHER MACHINE LEARNING METHODS.

Model	Accuracy	Precision	Recall	F-measure	AUC
SVM	0.82	**0.85**	0.13	0.23	0.73
Decision tree	0.76	0.43	0.47	0.45	0.65
Random forest	0.83	0.76	0.23	0.36	0.76
Adaboost	0.81	0.55	0.34	0.42	0.73
Our method	**0.85**	0.68	**0.50**	**0.58**	**0.83**

4.4 Ablation Study

In this study, three different sources of biological information were used in experiments. In order to discover what role each piece of biological information plays in the success of our proposed deep learning framework, we conduct an ablation study by removing individual input component in our network. Specifically, we test the performances of models without PPI network embedding, gene expression data, or subcellular localization information in turn. We use the remaining two components to form a large vector to retrain the model. From the results (accuracy, precision, recall, F-measure, and AUC) presented in Table 5, we find that PPI network embedding is the most important component in our deep learning framework. Without the PPI network embedding layer, accuracy, precision, recall, F-measure, and AUC drops from 0.850, 0.680, 0.505, 0.579, and 0.832 to 0.805, 0.593, 0.160, 0.252, and 0.755, respectively. This finding is also consistent with previous study [59]. Subcellular localization information is also important as accuracy, precision, recall, F-measure, and AUC drop to 0.830, 0.661, 0.358, 0.464, and 0.813 without them. In addition, compared with other biological information, gene expression data seems not as important, but they are also beneficial for enhancing the performance. Without gene expression data, accuracy, precision, recall, F-measure, and AUC drop to 0.832, 0.641, 0.417, 0.506, and 0.826, respectively. In Figure 4 we show that the ROC and PR curves of our method and other models removing different components. The ROC curve of our method is clearly higher than models without PPI network embedding and subcellular localization information and a little bit higher than model without gene expression data. Moreover, the AP score of 0.63 is obtained by our method, which is better than models without PPI network embedding (0.41), gene expression data (0.60), or subcellular localization information (0.59), respectively.

Our results indicate that three different types of biological information play their roles in identifying essential proteins and the importance of these different types of information are not the same. The most vital element is PPI network embedding. Gene expression profiles and subcellular localization information are used as auxiliary data to improve the performance for identifying essential proteins. We hypothesize that there are two main reasons. The first reason is that topological features of PPI network indeed have a strong relationship with gene essentiality. Since previous study have showed that there is a strong positive correlation between topological features of PPI network and gene essentiality, and thus almost all methods use the PPI network topology as their input features for prediction. The second resson is that the dimension of dense vector generated by the

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

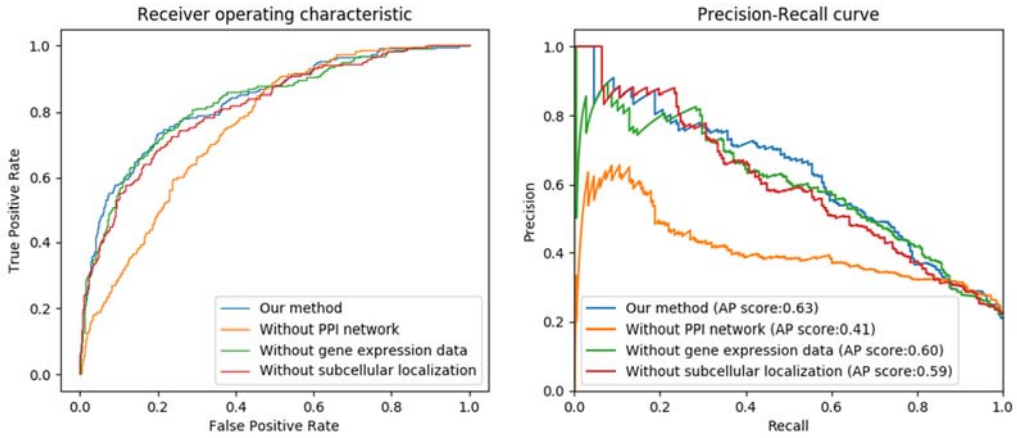

Fig. 4. ROC and PR curves of our model and other models removing different component.

node2vec technique is higher than that of the other two input vectors in our study. The dense vector generated by the node2vec technique is a 64-dimensional dense vector, the vectors of gene expression data processed by RNNs are two 8-dimensional vectors, and the subcellular localization information embedding vector is an 11-dimensional sparse vector. Among the three input vectors, dense vector generated by node2vec occupies the largest proportion and thus has the largest affect to the results.

TABLE 5.
PERFORMANCES (ACCURACY, PRECISION, RECALL, F-MEASURE, AND AUC) OF OUR MODEL AND OTHER MODELS REMOVING DIFFERENT COMPONENT.

Model	Accuracy	Precision	Recall	F-measure	AUC
Without PPI network	0.805	0.593	0.160	0.253	0.755
Without gene expression data	0.832	0.641	0.417	0.506	0.826
Without subcellular localization	0.830	0.661	0.358	0.464	0.813
Our method	**0.850**	**0.680**	**0.505**	**0.579**	**0.832**

5 CONCLUSION

Many computational methods have been proposed to identify essential proteins, but most of them require a lot of prior knowledge. Centrality methods need prior knowledge to design good score functions and machine learning-based methods need prior knowledge to select representative biological properties as features. But some designed score functions cannot capture the complexity of biological information and it is very difficult to select well-expressed features in machine learning-based methods. To tackle these problems, we propose a deep learning framework for identifying essential proteins without any prior knowledge. In this framework, we have used three different types of biological information including PPI network, gene expression data, and subcellular localization information. For PPI network, we employ network representation learning technique called node2vec to automatically learn semantic and topological features. This technique can maps vertices to a vector space and thus have a richer representation of PPI network than traditional score function. For gene expression data, we utilize bidirectional LSTM cells to better extract non-local relationships considering they are sequential data. For subcellular localization information, we design a high dimensional indicator vector to encode the location of each protein perform its function. This indicator vector is more expressive than a real number. After preprocessing the different types of biological information, three output vectors are concatenate together to be fed to a fully connected layer for classification. In order to evaluate the efficiency of our method, we have carried out experiments on PPI network of S. cerevisiae. Comparison with widely used centrality methods of DC, BC, CC, EC, NC, LAC, PeC, and WDC demonstrate that our method performs better than these existing centrality methods. Additionally, comparisons with widely used machine learning methods of SVM, decision tree, random forest, and Adaboost show that our method outperforms machine learning-based methods. To explore which piece of biological information is the most vital element in the success of the proposed method, we have conducted experiments by removing PPI network, gene expression data, or subcellular localization information. The results show that the major contribution to the improvement originates from the PPI network features learned by the node2vec technique while subcellular localization information and gene expression profiles also play roles in the improvement.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grants (No. 61832019, No. 61622213 and No. 61728211) and the 111 Project (No.B18059).

REFERENCES

[1] J. I. Glass, C. A. Hutchison, H. O. Smith, and J. C. Venter, "A systems biology tour de force for a near‐minimal bacterium," *Molecular systems biology*, vol. 5, no. 1, pp. 330, 2009.

[2] R. Zhang, and Y. Lin, "DEG 5.0, a database of essential genes in both prokaryotes and eukaryotes," *Nucleic acids research*, vol. 37, no. suppl_1, pp. D455-D458, 2008.

[3] A. E. Clatworthy, E. Pierson, and D. T. Hung, "Targeting virulence: a new paradigm for antimicrobial therapy," *Nature chemical biology*, vol. 3, no. 9, pp. 541, 2007.

[4] G. Giaever, A. M. Chu, L. Ni, C. Connelly, L. Riles, S. Veronneau, S. Dow, A. Lucau-Danila, K. Anderson, and B. Andre, "Functional profiling of the Saccharomyces cerevisiae genome," *nature*, vol. 418, no. 6896, pp. 387, 2002.

[5] L. M. Cullen, and G. M. Arndt, "Genome‐wide screening for gene function using RNAi in mammalian cells," *Immunology & Cell Biology*, vol. 83, no. 3, pp. 217-223, 2005.

[6] T. Roemer, B. Jiang, J. Davison, T. Ketela, K. Veillette, A. Breton, F. Tandia, A. Linteau, S. Sillaots, and C. Marta, "Large‐scale essential gene identification in Candida albicans and applications to antifungal drug discovery," *Molecular microbiology*, vol. 50, no. 1, pp. 167-181, 2003.

[7] J. Harborth, S. M. Elbashir, K. Bechert, T. Tuschl, and K. Weber, "Identification of essential genes in cultured mammalian cells using small interfering RNAs," *Journal of cell science*, vol. 114, no. 24, pp. 4557-4565, 2001.

[8] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai, "Lethality and centrality in protein networks," *Nature*, vol. 411, no. 6833, pp. 41, 2001.

[9] M. W. Hahn, and A. D. Kern, "Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks," *Molecular biology and evolution*, vol. 22, no. 4, pp. 803-806, 2004.

[10] M. P. Joy, A. Brock, D. E. Ingber, and S. Huang, "High-betweenness proteins in the yeast protein interaction network," *BioMed Research International*, vol. 2005, no. 2, pp. 96-103, 2005.

[11] S. Wuchty, and P. F. Stadler, "Centers of complex networks," *Journal of Theoretical Biology*, vol. 223, no. 1, pp. 45-53, 2003.

[12] E. Estrada, and J. A. Rodriguez-Velazquez, "Subgraph centrality in complex networks," *Physical Review E*, vol. 71, no. 5, pp. 056103, 2005.

[13] P. Bonacich, "Power and centrality: A family of measures," *American journal of sociology*, vol. 92, no. 5, pp. 1170-1182, 1987.

[14] K. Stephenson, and M. Zelen, "Rethinking centrality: Methods and examples," *Social networks*, vol. 11, no. 1, pp. 1-37, 1989.

[15] J. Wang, M. Li, H. Wang, and Y. Pan, "Identification of essential proteins based on edge clustering coefficient," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 4, pp. 1070-1080, 2012.

[16] Y. Tang, M. Li, J. Wang, Y. Pan, and F.-X. Wu, "CytoNCA: a cytoscape plugin for centrality analysis and evaluation of protein interaction networks," *Biosystems*, vol. 127, pp. 67-72, 2015.

[17] M. Li, J. Yang, F.-X. Wu, Y. Pan, and J. Wang, "DyNetViewer: a Cytoscape app for dynamic network construction, analysis and visualization," *Bioinformatics*, vol. 34, no. 9, pp. 1597-1599, 2017.

[18] S. Saha, and S. Heber, "In silico prediction of yeast deletion phenotypes," *Genet Mol Res*, vol. 5, no. 1, pp. 224-232, 2006.

[19] G. Li, M. Li, J. Wang, J. Wu, F.-X. Wu, and Y. Pan, "Predicting essential proteins based on subcellular localization, orthology and PPI networks," *BMC bioinformatics*, vol. 17, no. 8, pp. 279, 2016.

[20] M. Li, W. Li, F.-X. Wu, Y. Pan, and J. Wang, "Identifying essential proteins based on sub-network partition and prioritization by integrating subcellular localization information," *Journal of theoretical biology*, vol. 447, pp. 65-73, 2018.

[21] M. Li, R. Zheng, H. Zhang, J. Wang, and Y. Pan, "Effective identification of essential proteins based on priori knowledge, network topology and gene expressions," *Methods*, vol. 67, no. 3, pp. 325-333, 2014.

[22] X. Lei, J. Zhao, H. Fujita, and A. Zhang, "Predicting essential proteins based on RNA-Seq, subcellular localization and GO annotation datasets," *Knowledge-Based Systems*, vol. 151, pp. 136-148, 2018.

[23] M. Li, H. Zhang, J. Wang, and Y. Pan, "A new essential protein discovery method based on the integration of protein-protein interaction and gene expression data," *BMC systems biology*, vol. 6, no. 1, pp. 15, 2012.

[24] W. Peng, J. Wang, Y. Cheng, Y. Lu, F. Wu, and Y. Pan, "UDoNC: an algorithm for identifying essential proteins based on protein domains and protein-protein interaction networks," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 12, no. 2, pp. 276-288, 2015.

[25] X. Peng, J. Wang, J. Wang, F.-X. Wu, and Y. Pan, "Rechecking the centrality-lethality rule in the scope of protein subcellular localization interaction networks," *PloS one*, vol. 10, no. 6, pp. e0130743, 2015.

[26] 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.

[27] M. Li, P. Ni, X. Chen, J. Wang, F.-X.Wu, and Y. Pan, "Construction of refined protein interaction network for predicting essential proteins," *IEEE/ACM transactions on computational biology and bioinformatics*, no. 1, pp. 1-1, 2017.

[28] Y.-C. Hwang, C.-C. Lin, J.-Y. Chang, H. Mori, H.-F. Juan, and H.-C. Huang, "Predicting essential genes based on network and sequence analysis," *Molecular BioSystems*, vol. 5, no. 12, pp. 1672-1678, 2009.

[29] K. Plaimas, R. Eils, and R. König, "Identifying essential genes in bacterial metabolic networks with machine learning methods," *BMC systems biology*, vol. 4, no. 1, pp. 56, 2010.

[30] J. Deng, L. Deng, S. Su, M. Zhang, X. Lin, L. Wei, A. A. Minai, D. J. Hassett, and L. J. Lu, "Investigating the predictability of essential genes across distantly related organisms using an integrative approach," *Nucleic acids research*, vol. 39, no. 3, pp. 795-807, 2010.

[31] Y. Lu, J. Deng, J. C. Rhodes, H. Lu, and L. J. Lu, "Predicting essential genes for identifying potential drug targets in Aspergillus fumigatus," *Computational biology and chemistry*, vol. 50, pp. 29-40, 2014.

[32] M. L. Acencio, and N. Lemke, "Towards the prediction of

essential genes by integration of network topology, cellular localization and biological process information," *BMC bioinformatics*, vol. 10, no. 1, pp. 290, 2009.

[33] Y. Chen, and D. Xu, "Understanding protein dispensability through machine-learning analysis of high-throughput data," *Bioinformatics*, vol. 21, no. 5, pp. 575-581, 2004.

[34] J. Cheng, Z. Xu, W. Wu, L. Zhao, X. Li, Y. Liu, and S. Tao, "Training set selection for the prediction of essential genes," *PloS one*, vol. 9, no. 1, pp. e86805, 2014.

[35] A. M. Gustafson, E. S. Snitkin, S. C. Parker, C. DeLisi, and S. Kasif, "Towards the identification of essential genes using targeted genome sequencing and comparative analysis," *Bmc Genomics*, vol. 7, no. 1, pp. 265, 2006.

[36] M. Seringhaus, A. Paccanaro, A. Borneman, M. Snyder, and M. Gerstein, "Predicting essential genes in fungal genomes," *Genome research*, vol. 16, no. 9, pp. 1126-1135, 2006.

[37] J. Zhong, J. Wang, W. Peng, Z. Zhang, and Y. Pan, "Prediction of essential proteins based on gene expression programming," *BMC genomics*, vol. 14, no. 4, pp. S7, 2013.

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

[39] S. Hochreiter, and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[40] M. Li, J. Wang, X. Chen, H. Wang, and Y. Pan, " A local average connectivity-based method for identifying essential proteins from the network level," *Computational biology and chemistry*, vol. 35, no. 3, pp. 143-150, 2011.

[41] X. Tang, J. Wang, J. Zhong, and Y. Pan, "Predicting essential proteins based on weighted degree centrality," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 11, no. 2, pp. 407-418, 2014.

[42] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks." pp. 315-323.

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

[44] M. Li, Z. Niu, X. Chen, P. Zhong, F. Wu, and Y. Pan, "A reliable neighbor-based method for identifying essential proteins by integrating gene expressions, orthology, and subcellular localization information," *Tsinghua Science and Technology*, vol. 21, no. 6, pp. 668-677, 2016.

[45] Y. Fan, X. Tang, X. Hu, W. Wu, and Q. Ping, "Prediction of essential proteins based on subcellular localization and gene expression correlation," *BMC bioinformatics*, vol. 18, no. 13, pp. 470, 2017.

[46] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM." pp. 273-278.

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

[48] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers, "BioGRID: a general repository for interaction datasets," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D535-D539, 2006.

[49] H.-W. Mewes, C. Amid, R. Arnold, D. Frishman, U. Güldener, G. Mannhaupt, M. Münsterkötter, P. Pagel, N. Strack, and V. Stümpflen, "MIPS: analysis and annotation of proteins from whole genomes," *Nucleic acids research*, vol. 32, no. suppl_1, pp. D41-D44, 2004.

[50] J. M. Cherry, C. Adler, C. Ball, S. A. Chervitz, S. S. Dwight, E. T. Hester, Y. Jia, G. Juvik, T. Roe, and M. Schroeder, "SGD: Saccharomyces genome database," *Nucleic acids research*, vol. 26, no. 1, pp. 73-79, 1998.

[51] B. P. Tu, A. Kudlicki, M. Rowicka, and S. L. McKnight, "Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes," *Science*, vol. 310, no. 5751, pp. 1152-1158, 2005.

[52] J. X. Binder, S. Pletscher-Frankild, K. Tsafou, C. Stolte, S. I. O'Donoghue, R. Schneider, and L. J. Jensen, "COMPARTMENTS: unification and visualization of protein subcellular localization evidence," *Database*, vol. 2014, pp. bau012, 2014.

[53] M. Magrane, "UniProt Knowledgebase: a hub of integrated protein data," *Database*, vol. 2011, 2011.

[54] J. T. Eppig, J. A. Blake, C. J. Bult, J. A. Kadin, J. E. Richardson, and M. G. D. Group, "The Mouse Genome Database (MGD): comprehensive resource for genetics and genomics of the laboratory mouse," *Nucleic acids research*, vol. 40, no. D1, pp. D881-D886, 2011.

[55] P. McQuilton, S. E. St. Pierre, J. Thurmond, and F. Consortium, "FlyBase 101–the basics of navigating FlyBase," *Nucleic acids research*, vol. 40, no. D1, pp. D706-D714, 2011.

[56] T. W. Harris, I. Antoshechkin, T. Bieri, D. Blasiar, J. Chan, W. J. Chen, N. De La Cruz, P. Davis, M. Duesbury, and R. Fang, "WormBase: a comprehensive resource for nematode research," *Nucleic acids research*, vol. 38, no. suppl_1, pp. D463-D467, 2009.

[57] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, and M. Devin, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[58] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, and V. Dubourg, "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825-2830, 2011.

[59] X. Zhang, M. L. Acencio, and N. Lemke, "Predicting essential genes and proteins based on machine learning and network topological features: a comprehensive review," *Frontiers in physiology*, vol. 7, pp. 75, 2016.

Min Zeng received the 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 PhD degree in the School of Information Science and Engineering, Central South University, China. His research interests include bioinformatics, machine learning and deep learning.

Min Li Min Li received the PhD degree in Computer Science from Central South University, China, in 2008. She is currently a Professor at the School of Information Science and Engineering, Central South University, Changsha, Hunan, P.R. China. Her research interests include computa-

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.2019.2897679, IEEE/ACM Transactions on Computational Biology and Bioinformatics

tional 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. According to Google scholar, her paper citations is more than 2500 and H-index is 28.

Zhihui Fei received his BSc degrees in Hubei Normal University, China in 2015. He is currently a postgraduate student in Bioinformatics at Central South University. His currently research interests include bioinformatics, medical data mining 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 in 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. Hisresearch 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 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 Computer Science Department from 2005-2013. 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 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 300 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 9000 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 conference 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.

Jianxin Wang received the BEng and MEng degrees in computer engineering from Central South University, China, in 1992 and 1996, respectively, and the PhD degree in computer science from Central South University, China, in 2001. He is the vice dean and a professor in School of Information Science and Engineering, Central South University, Changsha, Hunan, P.R. China. His current research interests include algorithm analysis and optimization, parameterized algorithm, bioinformatics and computer network. He has published more than 150 papers in various International journals and refereed conferences. He is a senior member of the IEEE.