Finding lncRNA-Protein Interactions Based on Deep Learning With Dual-Net Neural Architecture

Lihong Peng, Chang Wang, Xiongfei Tian, Liqian Zhou, and Keqin Li

Abstract—The identification of lncRNA-protein interactions (LPIs) is important to understand the biological functions and molecular mechanisms of lncRNAs. However, most computational models are evaluated on a unique dataset, thereby resulting in prediction bias. Furthermore, previous models have not uncovered potential proteins (or lncRNAs) interacting with a new lncRNA (or protein). Finally, the performance of these models can be improved. In this study, we develop a Deep Learning framework with Dual-net Neural architecture to find potential LPIs (LPI-DLDN). First, five LPI datasets are collected. Second, the features of lncRNAs and proteins are extracted by Pyfeat and BioTriangle, respectively. Third, these features are concatenated as a vector after dimension reduction. Finally, a deep learning model with dual-net neural architecture is designed to classify lncRNA-protein pairs. LPI-DLDN is compared with six state-of-the-art LPI prediction methods (LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI) under four cross validations. The results demonstrate the powerful LPI classification performance of LPI-DLDN. Case study analyses show that there may be interactions between RP11-439E19.10 and Q15717, and between RP11-196G18.22 and Q9NUL5. The novelty of LPI-DLDN remains, integrating various biological features, designing a novel deep learning-based LPI identification framework, and selecting the optimal LPI feature subset based on feature importance ranking.

Index Terms—Deep learning, dual-net neural architecture, feature importance ranking, lncRNA-protein interaction

1 Introduction

1.1 Motivation

Over the past decades, the explosion of multiple high-throughput genomic analyses has suggested that most noncoding regulatory elements control the developmental processes regulating organism complexity [1], [2]. Noncoding elements are generally transcribed into noncoding RNAs (ncRNAs), thereby implicating the significant regulatory roles of ncRNAs in complex organisms. In fact, studies demonstrate that ncRNAs can regulate many major biological activities impacting development, differentiation, and metabolism [2], [3]. In contrast to small ncRNAs (for example, miRNAs [4]), which have high conservation and affect transcriptional and posttranscriptional gene silencing, long noncoding RNAs (lncRNAs) have poor conservation and control gene expression based on various unknown mechanisms [5].

Although only a few lncRNAs have been well studied, they have been demonstrated to affect every stage of the gene expression program [6]. lncRNAs regulate posttranscriptional genes by controlling biological activities such as protein synthesis and RNA maturation and affect transcriptional gene silencing by controlling chromatin structures [7]. Given the vast number of lncRNAs whose biological functions are still unknown, there is clear significance for finding widespread regulation of gene expression and chromatin modification.

Recent studies show that lncRNAs generally regulate cellular processes to exert their functions through associations with RNA-binding proteins [8], [9]. Therefore, identifying possible lncRNA-protein interactions (LPIs) is vital to demonstrate the functions and mechanisms of lncRNAs. Experimental methods have uncovered some LPIs; however, the methods require a large amount of time and resources. Thus, computational methods were designed to discover LPI candidates [7], [10].

1.2 Study Contributions

In this manuscript, a deep learning-based framework (LPI-DLDN) is developed to find new LPIs. This framework utilizes various biological data, feature selection, dimensional reduction, dual-net neural architecture, Feature Importance Ranking (FIR), and Multiple-Layer Perceptron (MLP). The study has the following three main contributions:

1) Multiple biological features of lncRNAs and proteins are reasonably integrated to more effectively depict lncRNA-protein pairs.
2) A deep learning model with dual-net neural architecture, composed of the FIR and MLP nets, is developed to classify unknown lncRNA-protein pairs.
3) The exploration-exploitation strategy is used to select the most representative features and boost the generalization ability of LPI-DLDN.

The remainder of this manuscript is organized as follows. Section 2 introduces related work. Section 3 describes the LPI-

- *Lihong Peng, Chang Wang, Xiongfei Tian, and Liqian Zhou are with the School of Computer, Hunan University of Technology, Zhuzhou, Hunan 412007, China. E-mail: {plhhnu, wsrgyw, txfhut, zhoulq11}@163.com.*
- *Keqin Li is with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA. E-mail: lik@newpaltz.edu.*

Manuscript received 4 Apr. 2021; revised 4 Aug. 2021; accepted 27 Aug. 2021. Date of publication 29 Sept. 2021; date of current version 8 Dec. 2022.
This work was supported in part by the National Natural Science Foundation of China under Grants 62072172, 61803151, 62172158, the Natural Science Foundation of Hunan province under Grant 2021JJ30219, Scientific Research Project of Hunan Provincial Department of Education under Grant 20C0636, Scientific Research and Innovation Foundation of Hunan University of Technology under Grant CX2031.
(Corresponding authors: Liqian Zhou and Keqin Li.)
Digital Object Identifier no. 10.1109/TCBB.2021.3116232

DLDN framework. Section 4 gives the results from a series of comparative experiments. Section 5 discusses the LPI-DLDN method and provides directions for future research.

2 Related Work

2.1 LPI Prediction

Computational methods for LPI prediction roughly contain network-based methods and machine learning-based methods [11]. Network-based methods, for example, random walk [12], linear neighborhood propagation [13], and bipartite network projection [14], [15], [16], integrated related biological information and network propagation algorithms to predict LPI candidates. Machine learning-based methods, for example, matrix factorization techniques [17], [18] and ensemble learning-based methods [9], [19], used matrix factorization and ensemble learning to discover potential interactions between lncRNAs and proteins.

In these LPI identification methods, LPI-HeteSim [20], LPI-NRLMF [21], LPI-XGBoost [22], PLIPCOM [23], LPI-CNNCP [24], and Capsule-LPI [25] are six state-of-the-art approaches. LPI-HeteSim utilized the HeteSim method to evaluate the relevance between lncRNAs and proteins in the heterogeneous lncRNA-protein network. LPI-NRLMF employed a neighborhood regularized logistic matrix factorization method to score unknown lncRNA-protein pairs. LPI-HeteSim and LPI-NRLMF are two network-based LPI prediction approaches. PLIPCOM extracted diffusion and HeteSim features from the heterogeneous lncRNA-protein networks and then developed a gradient tree boosting approach to find LPI candidates. LPI-XGBoost used an innovative algorithm to process categorical LPI features and an ordered boosting technique to predict new LPIs. PLIPCOM and LPI-XGBoost are two ensemble learning-based LPI identification methods. LPI-CNNCP exploited a convolutional neural network model with the copy-padding trick to investigate potential interactions between lncRNAs and proteins. Capsule-LPI fused multiple protein and lncRNA features and designed a capsule network for LPI identification. LPI-CNNCP and Capsule-LPI are two deep learning-based LPI prediction frameworks.

Although these computational models were effectively applied to LPI identification, they have a few limitations. First, the majority were measured on a unique dataset, thereby possibly resulting in prediction bias. Second, most of them did not find potential proteins (or lncRNAs) interacting with a new lncRNA (or protein). Finally, the prediction performance has room for improvement.

2.2 Deep Learning in Bioinformatics

To obtain knowledge from biomedical data, machine learning algorithms (i.e., random forests, support vector machines, and Bayesian networks) have been widely used [26]. Machine learning uses training data to make predictions by building a best fit model. However, the performance of these traditional machine learning methods relies heavily on data feature representation [27]. However, features are generally exploited by engineers with extensive expertise knowledge, and it is difficult to uncover features appropriate for a given task [28]. Therefore, deep learning, as a branch of machine learning, is widely used in the areas of bioinformatics.

Deep learning has overcome the above limitations and boosted major advances in various fields of bioinformatics [28], [29]. For example, Shaw *et al.* [30] adopted a hybrid framework (DeepLPI) based on a multimodal deep neural network combined with a conditional random field. DeepLPI obtained better prediction performance for LPI discovery and is a representative LPI identification method. However, deep learning is very difficult to apply to areas demanding explainability/interpretability because of its purported "black box" nature [28], [31].

2.3 Feature Selection

lncRNAs and proteins have diverse biological features, which possibly results in a dimensional curse due to the effects of irrelevant features on supervised learning. Generally, in machine learning methods, an optimal feature subset is selected to maximize the learning ability of a model based on prespecified criteria. Feature selection methods provide clearer ways to remove redundant and irrelevant information and obtain the best feature subset [32]. The methods help construct a better classifier by extracting significant features and reducing computational overload [33].

Traditional feature selection techniques contain filter methods, embedded methods, and wrapper methods. Filter methods retain the top features based on the alternating conditional expectations algorithm. Embedded methods incorporate their own feature selection process. Wrapper methods provide a better feature list. However, these conventional methods actually produce interpretability and stability problems [34]. Stability represents the reproducibility of feature selection approaches. The strong correlation among features frequently generates multiple equally optimal signatures, thereby making traditional feature selection techniques unstable and reducing the confidence level of the selected features [32], [34], [35].

The FIR methods facilitate the understanding of classification tasks and discovery of key features and thus have been validated as powerful tools in solving explainable/interpretable problems [36]. The methods construct a representative feature subset by evaluating the role of individual input feature in a classification model. It helps reduce space and time complexity and further boosts the purity of a classifier [32].

3 Materials and Methods

3.1 Data Preparation

In this manuscript, we collect five different LPI datasets. Datasets 1, 2, and 3 are from humans and the remaining datasets are from plants. Dataset 1 was constructed by Li *et al.* [12]. ncRNA-protein interactions are downloaded from the NPInter 2.0 database [37] and filtered by restricting the type and organism to NONCODE and HOMO sapiens. A total of 3,487 human LPIs between 938 lncRNAs and 59 proteins are then selected based on the lncRNAs in the NONCODE 4.0 database [38]. Finally, we remove lncRNAs and proteins without known sequences in the NPInter [37], NONCODE [38], and UniProt [39] databases and achieve 3,479 LPIs between 935 lncRNAs and 59 proteins.

Dataset 2 was collected by Zheng *et al.* [40]. Human ncRNA-protein interactions and lncRNAs are first downloaded from the NPInter 2.0 [37] and NONCODE 4.0

TABLE 1
The Statistics of LPI Data

Dataset	lncRNAs	Proteins	LPIs
Dataset 1	935	59	3,479
Dataset 2	885	84	3,265
Dataset 3	990	27	4,158
Dataset 4	109	35	948
Dataset 5	1,704	42	22,133

databases [38], respectively. A total of 4,467 LPIs from 1,050 lncRNAs and 84 proteins are then obtained after preprocessing. Finally, we obtain 3,265 LPIs between 885 lncRNAs and 84 proteins by removing lncRNAs and proteins without any sequence information.

Dataset 3 was compiled by Zhang *et al.* [13]. Experimentally confirmed LPIs from 1,114 lncRNAs and 96 proteins provided by Ge *et al.* [14] are first downloaded. The sequence and expression information of lncRNAs and the sequence information of proteins are extracted from the NONCODE 4.0 [38] and SUPERFAMILY databases [41], respectively. Finally, a total of 4,158 LPIs between 990 lncRNAs and 27 proteins are selected by manually removing the lncRNAs and proteins without any sequence or expression information or only interacting with one protein (or lncRNA).

In addition, a protein is regarded as a redundant protein if it exists in any two human datasets. There are 100 different proteins in datasets 1, 2, and 3. That is, there are 100 proteins which only exist in any one human dataset. There are 991 different lncRNAs in datasets 1 and 3. Redundant lncRNAs between dataset 2 and the other two human LPI datasets are not analyzed due to the existence of different versions. More importantly, LPIs in datasets 1, 2, and 3 are from different publications. Therefore, a known LPI in one dataset may be unknown lncRNA-protein pair in another dataset. For example, some unlabeled lncRNA-protein pairs in dataset 3 are validated in datasets 1 or 2.

Datasets 4 and 5 contain LPI-related information about Arabidopsis thaliana and Zea mays, respectively. The sequence data of lncRNAs and proteins can be extracted from the plant lncRNA database (PlncRNADB [42]) and LPIs can be obtained at http://bis.zju.edu.cn/PlncRNADB/. Dataset 4 contains 948 LPIs between 109 lncRNAs and 35 proteins and dataset 5 contains 22,133 LPIs between 1,704 lncRNAs and 42 proteins. These two datasets have no redundant data. The details are shown in Table 1.

We represent an LPI network as a matrix Y with the element

$$y_{ij} = \begin{cases} 1, \text{ if lncRNA } l_i \text{ interacts with protein } p_j \\ 0, \qquad\qquad\qquad \text{otherwise} \end{cases}. \tag{1}$$

3.2 Overview of LPI-DLDN

In this study, motivated by the FIR method proposed by Wojtas and Chen [36], we develop a deep learning model with a dual-net neural architecture to predict potential LPIs based on feature extraction, dimension reduction, FIR, and MLP. Fig. 1 describes the pipeline of LPI-DLDN.

As shown in Fig. 1, the LPI-DLDN framework consists of three main steps after five LPI datasets are collected. (1) LPI feature extraction. Pyfeat [43] and BioTriangle [44] are applied to obtain the original features of lncRNAs and proteins. (2) LPI feature selection. The two types of features are transformed into two d-dimensional vectors based on Principal Component Analysis (PCA). The two vectors are concatenated as a $2d$-dimensional vector to represent a lncRNA-protein pair. (3) LPI classification. A deep learning model with dual-net neural architecture is developed to discover possible LPIs. The architecture consists of two nets: the FIR net and the MLP net. The FIR net selects the optimal LPI feature subset based on the classification ability obtained from the MLP net in the last iteration. The MLP net classifies lncRNA-protein pairs based on the extracted optimal LPI feature subset in the FIR net. The two nets are alternately trained on five LPI datasets. Finally, FIR is utilized to identify an optimal LPI feature subset, while MLP classifies unknown lncRNA-protein pairs based on the extracted optimal LPI feature subset.

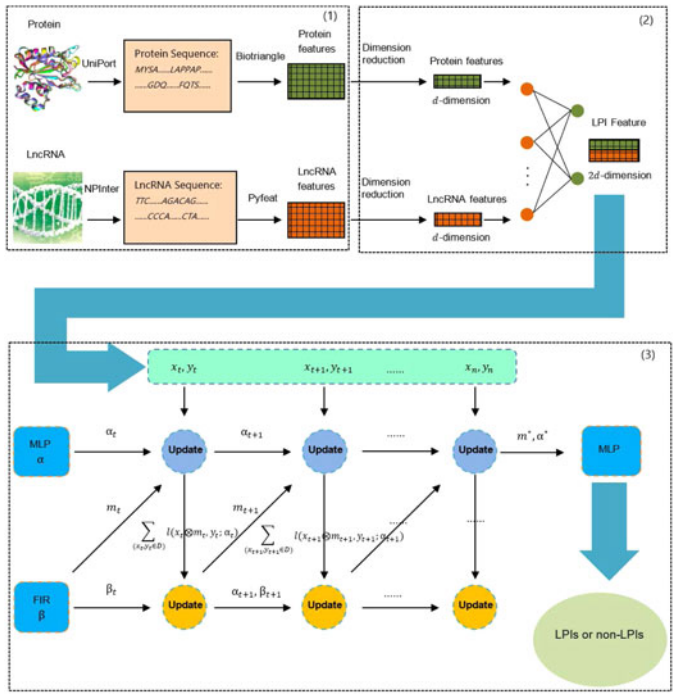


Fig. 1. The Flowchart of the LPI-DLDN framework. (1) LPI feature selection. (2) LPI dimension reduction. (3) LPI classification.

3.3 Feature Selection

3.3.1 Feature Selection of lncRNAs

To describe DNAs and RNAs, Pyfeat [43] ensembles thirteen types of features, including zCurve, gcContent, atgcRatio, cumulativeSkew, pseudoKNC, monoMonoKGap, monoDiKGap, monoTriKGap, diMo-noKGap, diDiKGap, diTriKGap, triMonoKGap, and tri-DiKGap. In this study, we use Pyfeat to extract lncRNA features and obtain a 14,892-dimensional vector.

3.3.2 Feature Selection of Proteins

BioTriangle [44] uses fourteen types of features to represent proteins: amino acid composition, dipeptide composition, tripeptide composition, CTD composition, CTD transition, CTD distribution, M-B autocorrelation, Moran autocorrelation,

Geary autocorrelation, conjoint triad features, quasi-sequence order descriptors, sequence order coupling number, pseudo amino acid composition 1, and pseudo amino acid composition 2. Features generated by BioTriangle can effectively capture the discriminatory information of amino acids. In this study, we utilize BioTriangle to extract protein features and obtain a 10,029-dimensional vector.

3.4 Dimension Reduction

We separately carry out dimensional reduction for lncRNA and protein features based on PCA and obtain two d-dimensional vectors. The two vectors are then concatenated, and thus, a lncRNA-protein pair is represented as a $2d$-dimensional vector $\boldsymbol{x}$.

3.5 LPI Prediction Framework

3.5.1 Problem Formulation

Assume that $D = (X, Y)$ denotes an LPI dataset, where $(\boldsymbol{x}, \boldsymbol{y})$ represents a training example (a lncRNA-protein pair), $\boldsymbol{x} \in X$ denotes a $2d$-dimensional LPI feature vector and $\boldsymbol{y} \in Y$ denotes the corresponding label of the lncRNA-protein pair. We aim to find labels for unknown lncRNA-protein pairs.

Let $\boldsymbol{m} \in M$ represent a $2d$-dimensional binary mask vector composed of the elements with values of 0 or 1, where $||\boldsymbol{m}||_0 = s$, $s < 2d$, and $|M| = \binom{2d}{s}$. A mask vector $\{\boldsymbol{x} \otimes \boldsymbol{m}\}_{\boldsymbol{x} \in X}$ is used to denote an LPI feature subset of s features for any lncRNA-protein pair $\boldsymbol{x}$, where $\otimes$ indicates the Hadamard product. Suppose that $Q(\boldsymbol{x}, \boldsymbol{m})$ denotes the prediction performance obtained from MLP trained on D via the masked feature subset, we rank the features based on their importance

$$(\boldsymbol{m}^*, Score(\boldsymbol{m}^*)) = \arg\max_{\boldsymbol{m} \in M} \sum_{\boldsymbol{x} \in X} Q(\boldsymbol{x}, \boldsymbol{m}), \tag{2}$$

where $\boldsymbol{m}^*$ denotes the indicators of the learned optimal LPI feature subset identified by Eq. (2) and $Score(\boldsymbol{m}^*)$ denotes the importance scores of all the extracted features in the optimal subset. The labels for each lncRNA-protein pair can be computed based on the selected optimal LPI feature subset $\boldsymbol{m}^*$ and an MLP.

3.5.2 Model Description

The model in Eq. (2) describes a combinatorial optimization problem. No algorithm can outperform a random strategy in the combinatorial optimization problem based on the theory of no free lunch. Noise is thus injected into candidate LPI feature subsets $M^{'} \subset M$ to enhance a stochastic local search procedure [45], where $M^{'}$ may change during learning. Each training sample $(\boldsymbol{x}, \boldsymbol{y}) \in D$ is converted into $|M^{'}|$ samples: $\{(\boldsymbol{x} \otimes \boldsymbol{m}, \boldsymbol{y})\}_{\boldsymbol{m} \in M^{'}}$.

The MLP net is trained on D based on different LPI feature subsets to learn $f_{\mathrm{MLP}} : X \times M \rightarrow Y$. The loss functions on $|M^{'}|$ for the MLP net are defined as Eq. (3)

$$L_{\mathrm{MLP}}(D, M^{'}; \alpha) = \frac{1}{|M^{'}||D|} \sum_{\boldsymbol{m} \in M^{'}} l(\boldsymbol{x} \otimes \boldsymbol{m}, \boldsymbol{y}; \alpha), \tag{3}$$

where $l(\boldsymbol{x} \otimes \boldsymbol{m}, \boldsymbol{y}; \alpha)$ with the parameter α denotes a binary cross-entropy loss during LPI classification. The loss is utilized to characterize its learning performance, $Q(\boldsymbol{x}, \boldsymbol{m})$. In the MLP net, sigmoid and softmax are used as the activation functions in all intermediate layers and the final output layer, respectively. After learning, the trained MLP, $f_{\mathrm{MLP}}(\alpha^*; \boldsymbol{x}, \boldsymbol{m}^*)$ with an optimal parameter α^*, is used to uncover possible LPIs on the test dataset.

The FIR net selects the optimal LPI feature subset based on the prediction ability from the MLP net. For each lncRNA-protein pair $\boldsymbol{x} \in X$, the extracted optimal LPI feature subset should maximize the performance of MLP quantified by $Q(\boldsymbol{x}, \boldsymbol{m})$. An exploration-exploitation strategy is exploited to rank features and produce an optimal LPI feature subset with the index of $\boldsymbol{m}^*$ via $Score(\boldsymbol{m}^*)$. The loss function on $|M^{'}|$ for the FIR net is defined as Eq. (4)

$$\begin{aligned} L_{\mathrm{FIR}}(M^{'}, \beta) = & \frac{1}{2|M^{'}|} \sum_{\boldsymbol{m} \in M^{'}} (f_{\mathrm{FIR}}(\beta; \boldsymbol{m}) \\ & - \frac{1}{|D|} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in D} l(\boldsymbol{x} \otimes \boldsymbol{m}, \boldsymbol{y}; \alpha). \end{aligned} \tag{4}$$

In the FIR net, sigmoid and linear functions are used as the activation functions in all intermediate layers and the final output layer, respectively. After learning, the trained FIR, $f_{\mathrm{FIR}}(\beta^*; \boldsymbol{x}, \boldsymbol{m}^*)$ with an optimal parameter β^*, is used to extract the optimal LPI features on the test dataset.

In the alternative learning process, the FIR net assists in MLP to provide an optimal LPI feature subset $|M^{'}|$, while the MLP net feeds back the performance $l(\boldsymbol{x} \otimes \boldsymbol{m}, \boldsymbol{y}; \alpha)$ to the FIR net for all $\boldsymbol{m} \in |M^{'}|$. The detailed procedures are described as Fig. 1.

3.5.3 Initial LPI Classification in the MLP Net

First, we train the MLP net based on several random LPI feature subsets for a few epochs until it can stably produce different performances on different LPI feature subsets. In each epoch, an LPI feature subset with different masks, $M_1^{'}$, is randomly extracted from M, $M_1^{'} = \{\boldsymbol{m}_i | \boldsymbol{m}_i = Random(M, s)\}_{i=1}^{|M^{'}|}$ where $Random(M, s)$ denotes a function applied to randomly extract a $2d$-dimensional mask with s one-elements and $(2d\text{-}s)$ zero-elements from M. α is trained by the Nesterov-accelerated adaptive moment estimation algorithm and updated by Eq. (5)

$$\alpha^{''} = \alpha^{'} - \eta \nabla_{\alpha} L_{\mathrm{MLP}}(D, M_1^{'}; \alpha)|_{\alpha = \alpha^{'}}, \tag{5}$$

where η denotes a learning rate. After E epochs,

$$\alpha_1 = \alpha^{''}(E) \tag{6}$$

$$\boldsymbol{m}_{1,opt}^{'} = \arg\min_{\boldsymbol{m} \in M_1^{'}} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in D} l(\boldsymbol{x} \otimes \boldsymbol{m}, \boldsymbol{y}; \alpha_1). \tag{7}$$

The above parameters is used as the input of the FIR net.

3.5.4 Optimal LPI Feature Subset Construction via MLP Feedback

As shown in Fig. 1, the training samples in the FIR net were provided by the MLP net at the tth step: $\{(\boldsymbol{m}, \frac{1}{|D|} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in D} l(\boldsymbol{x} \otimes \boldsymbol{m},\ \boldsymbol{y}; \alpha_t))\}_{\boldsymbol{m} \in M_t^{'}}$. The parameters β are first updated based on the adaptive moment estimation algorithm with random initialization β_1

$$\beta_{t+1} \triangleq \beta_t - \eta \nabla_\beta L_{\mathrm{FIR}}(M_t^{'}; \beta)|_{\beta=\beta_t}. \quad (8)$$

An exploration-exploitation strategy is then adopted to produce a new masked LPI feature subset $M_{t+1}^{'}$ applied to the FIR net at the (t+1)th step. The feature subset $M_{t+1}^{'}$ is separated into two mutually exclusive subsets: $M_{t+1}^{'} = M_{t+1,1}^{'} \cup M_{t+1,2}^{'}$. In terms of the role of noise data in the stochastic local search [45], a random function $M_{t+1,1}^{'} = \{\boldsymbol{m}_i | \boldsymbol{m}_i = Random(M, s)\}_{i=1}^{|M_{t+1,1}^{'}|}$ is applied to generate $M_{t+1,1}^{'}$ and reduce overfitting. Inspired by the input gradient technique proposed by Hechtlinger *et al*. [46], $M_{t+2,1}^{'}$ is produced based on Algorithm 1.

Algorithm 1. Generation of the Optimal LPI Feature Subset $M_{t+1,2}^{'}$

Input: m_0, s, and s_p;
Output: The integrated optimal LPI feature subset $M_{t+1,2}^{'}$;

Phase I: Generate initial optimal LPI feature subset $\boldsymbol{m}_{t+1,opt}$
1: Compute $\delta_{\boldsymbol{m}_0} = \frac{\partial f_{\mathrm{FIR}}(\beta_{t+1}; \boldsymbol{m})}{\partial \boldsymbol{m}}|_{\boldsymbol{m}=\boldsymbol{m}_0}$
2: Select the top s LPI features $\boldsymbol{m}_{t+1,opt}$ via $(\boldsymbol{m}_{opt}, \overline{\boldsymbol{m}}_{opt}) = \arg sort(\delta_{\boldsymbol{m}_{opt}}, s)$ based on the following four-step procedure:
(1) Re-measure the contributions of the selected top LPI features by $(\boldsymbol{m}_{opt}, \overline{\boldsymbol{m}}_{opt}) = \arg sort(\delta_{\boldsymbol{m}_{opt}}, s)$ where $\delta_{\boldsymbol{m}_{opt}} = \frac{\partial f_{\mathrm{FIR}}(\beta_{t+1}; \boldsymbol{m})}{\partial \boldsymbol{m}}|_{\boldsymbol{m}=\boldsymbol{m}_{opt}}$;
(2) Re-generate the optimal LPI feature subset by replacing an LPI feature with negative gradient in $\boldsymbol{m}_{opt}$ with a feature with the largest gradient in $\overline{\boldsymbol{m}}_{opt}$ if there exists;
(3) Generate the optimal LPI feature subset $\boldsymbol{m}_{opt}^{'}$ by $(\boldsymbol{m}_{opt}^{'}, \overline{\boldsymbol{m}}_{opt}^{'}) = swap(\boldsymbol{m}_{opt}, \overline{\boldsymbol{m}}_{opt})$.
(4) Repeat Steps (1)-(3) until $f_{\mathrm{FIR}}(\beta_{t+1}; \boldsymbol{m}_{opt}) \leq f_{\mathrm{FIR}}(\beta_{t+1}; \boldsymbol{m}_{opt}^{'})$ and obtain the optimal LPI feature subset $\boldsymbol{m}_{t+1,opt}$.

Phase II: Generate multiple optimal LPI feature subsets via perturbation
1: Randomly convert $s_p (s_p < s)$ different elements in $\boldsymbol{m}_{opt}/\overline{\boldsymbol{m}}_{opt}$ from 1/0 to 0/1 based on the perturbation function $Perturb(\boldsymbol{m}_{opt}, s_p)$ and swap the corresponding elements in $\boldsymbol{m}_{opt}$ and $\overline{\boldsymbol{m}}_{opt}$;
2: Repeat perturbation and obtain multiple optimal LPI feature subset candidates $\{\boldsymbol{m}_i | \boldsymbol{m}_i = Perturb(\boldsymbol{m}_{t+1,opt}, s_p)\}$.

Phase III: Integrate optimal LPI feature subset candidates
1: Let $\boldsymbol{m}_{t,best}$ represent the LPI feature subset candidate, which contributes to the best predictive performance in the MLP net at the tth step;
2: Obtain the feature subset $M_{t+1,2}^{'} = \{\boldsymbol{m}_{t,best}\} \cup \{\boldsymbol{m}_{t+1,opt}\} \cup \{\boldsymbol{m}_i | \boldsymbol{m}_i = Perturb(\boldsymbol{m}_{t+1,opt}, s_p)\}_{i=1}^{|M_{t+1,2}^{'}|-2}$ based on the above Phases I and II.

In phase I, as illustrated in Algorithm 1, an initial $2d$-dimensional LPI feature vector $\boldsymbol{m}_0 = (\frac{1}{2}, \ldots, \frac{1}{2})$ is used to demonstrate that every LPI feature can be selected with equal opportunity. The input features with larger gradients can better boost the learning ability in the MLP net; therefore, we select the top s LPI features based on their gradients via $(\boldsymbol{m}_{opt}, \overline{\boldsymbol{m}}_{opt}) = \arg sort(\delta_{\boldsymbol{m}_{opt}}, s)$, where $\boldsymbol{m}_{opt}$ denotes the mask of the top s features and $\overline{\boldsymbol{m}}_{opt}$ represents the mask of the remaining $(2d\text{-}s)$ features. $\boldsymbol{m}_{opt}$ can be selected based on the four-step validation procedure in phase I. The function $(\boldsymbol{m}_{opt}^{'}, \overline{\boldsymbol{m}}_{opt}^{'}) = swap(\boldsymbol{m}_{opt}, \overline{\boldsymbol{m}}_{opt})$ denotes that an LPI feature with the least gradient in m_{opt} is swapped with the feature with the largest gradient in $\overline{\boldsymbol{m}}_{opt}$.

In phase II, to avoid the local optimum produced by $\boldsymbol{m}_{t+1,opt}$ and obtain multiple better LPI feature subsets, noise is injected based on a perturbation function $Perturb(\boldsymbol{m}_{opt}, s_p)$.

In phase III, the optimal LPI feature subset candidates $M_{t+1,2}^{'}$ are integrated based on the obtained optimal subset candidates $\boldsymbol{m}_{t,best}$ in the tth step, the optimal subsets $m_{t+1,opt}$ in the (t+1)th step, and the subsets $\{\boldsymbol{m}_i | \boldsymbol{m}_i = Perturb(\boldsymbol{m}_{t+1,opt}, s_p)\}$ via perturbation.

3.5.5 MLP Training via Optimal LPI Feature Subset Candidates

The FIR net provides the optimal LPI feature subset $M_{t+1}^{'} = M_{t+1,1}^{'} \cup M_{t+1,2}^{'}$ for the MLP net based on the training process in the above section. The MLP net was then trained on $M_{t+1}^{'}$ via the stochastic local search method: $\alpha_{t+1} \triangleq \alpha_t - \eta \nabla_\alpha L_{\mathrm{MLP}}(D, M_{t+1}^{'}; \alpha)|_{\alpha=\alpha_t}$. The two nets are alternately trained until a predefined iterative stops.

3.5.6 Classification of lncRNA-Protein Pairs

The optimal parameters α^* and β^* in the MLP and FIR nets can be obtained based on the above three sections. In addition, an optimal feature subset $\boldsymbol{m}^*$ is extracted by Algorithm 1. Thus, all lncRNA-protein pairs can be classified based on Algorithm 2.

Algorithm 2. Classification of lncRNA-Protein Pairs

Input: LPI feature vector, LPIs, lncRNA-protein pairs, α^*, and β^*;
Output: The labels of lncRNA-protein pairs;
1: Calculate the gradient $\delta_{\boldsymbol{m}_0} = \frac{\partial f_{\mathrm{FIR}}(\beta^*; \boldsymbol{m})}{\partial \boldsymbol{m}}|_{\boldsymbol{m}=\boldsymbol{m}_0}$ with $\boldsymbol{m}_0 = (\frac{1}{2}, \ldots, \frac{1}{2})$
2: Find the top s LPI features and obtain the optimal feature subset $\boldsymbol{m}^*$ composed of the s features by $(\boldsymbol{m}^*, \overline{\boldsymbol{m}}^*) = \arg sort(\delta_{\boldsymbol{m}_0}, s)$
3: Ensure the optimality of $\boldsymbol{m}^*$ based on Algorithm 1
4: Obtain the optimal LPI feature subset based on $Score(\boldsymbol{m}^*) = \frac{\partial f_{\mathrm{FIR}}(\beta^*; \boldsymbol{m})}{\partial \boldsymbol{m}}|_{\boldsymbol{m}=\boldsymbol{m}^*}$
5: Predict the label for each lncRNA-protein pair $\hat{\boldsymbol{x}}$ in the test set with the trained MLP, $f_{\mathrm{MLP}}(\alpha^*; \hat{\boldsymbol{x}}, \boldsymbol{m}^*)$ via $\hat{\boldsymbol{x}} \otimes \boldsymbol{m}^*$, in the MLP net

4 Results

4.1 Evaluation Metrics

Six measurements are utilized to evaluate the performance of our proposed LPI-DLDN framework: precision, recall, accuracy, F1-score, Area Under the ROC curve (AUC) and Area Under the Precision-Recall curve (AUPR). The six metrics are statistically consistent criteria. Higher precision, recall, accuracy, F1-score, AUC and AUPR denote better performance. The comparative experiments are repeated 20 times and the final performance is obtained by averaging the results from the 20 iterations.

In the deep learning-based prediction model, an LPI with an association probability greater than 0.5 is classified to the positive class, while those less than 0.5 are classified to the

TABLE 2
Parameter Settings

Method	Parameter Setting
LPI-XGBoost	num_boost_round=10, evals=(), obj=None, callbacks=None, feval=None, maximize=False, learning_rates=None, evals_result=None, verbose_eval=True, xgb_model=None, early_stopping_rounds=None
LPI-NRLMF	cfix=5, K1=5, K2=5, num_factors=10, alpha=0.1, beta=0.1, theta=1.0, lambda_d=0.625, lambda_t=0.625, max_iter=100
PLIPCOM	learning_rate=1, n_estimators=100, random_state= 10 max_depth=3, min_samples_leaf=10, min_samples_split= 2, max_features=30
LPI-DLDN	max_batches = 2500, N_FEATURES = 200,s = 150, FEATURE_SHAPE = 200, dataset_label = RNA_PRO, data_batch_size = 32, mask_batch_size = 32, s_p = 2, phase_2_start = 200, early_stopping_patience = 200
LPI-CNNCP	filters1 = 24, kernel_size1 = (49, 10), strides1 = (1, 1), filters2 = 24, kernel_size2 = (64, 10), strides2 = (1, 3)
Capsule-LPI	EPOCH = 30, BATCH_SIZE = 100, lr = 0.001

negative class. In this study, we adopt the function sklearn.metrics.roc_curve in the sklearn package to generate a threshold array. The thresholds in the array are used to calculate the six measurements, and finally, the average performance is computed.

4.2 Experimental Settings

Pyfeat is used to extract lncRNA features and the parameters are set as follows: kGap = 5, kTuple = 3, opti-mumDataset = 0, pseudoKNC = 1, zCurve = 1, gcContent = 1, cumulativeSkew = 1, atgcRatio = 1, monoMono = 1, monoDi = 1, monoTri = 1, diMono = 1, diDi = 1, diTri = 1, triMono = 1, and triDi = 1. All features in BioTriangle are applied to represent proteins. The parameters in LPI-HeteSim are set as the default values provided by Zhou *et al*. [47]. The parameters in the other six LPI prediction models are set to the corresponding values where the models obtain optimal performance through grid search. The details are shown in Table 2.

We adopt grid search and found that when $d = 100$, LPI-DLDN obtains better performance. Therefore, we extract two 100-dimensional vectors to represent lncRNA and protein features. Four different 5-fold cross-validations (CVs) are performed to measure the performance of LPI-DLDN.

1) 5-fold CV on lncRNAs (CV1): random rows in Y are masked for testing, that is, 80% of lncRNAs are selected as train set and the remaining 20% are selected as test set in each round.
2) 5-fold CV on proteins (CV2): random columns in Y are masked for testing, that is, 80% of proteins are selected for the train set and the remaining 20% are selected as test set in each round.
3) 5-fold CV on lncRNA-protein pairs (CV3): random lncRNA-protein pairs in Y are masked for testing, that is, 80% of lncRNA-protein pairs are selected as the train set and the remaining 20% are selected as the test set in each round.
4) 5-fold CV on independent lncRNAs and independent proteins (CV4) [48]: First, 20% of lncRNAs and 20% of proteins are randomly selected to form the "node test set". Second, the remaining nodes (lncRNAs or proteins) are regarded as the "node train set". Third, all edges connecting a node from the node train set to a node from the node test set are discarded and excluded from the analysis. Finally, one classifier is trained only on edges within the node train set to predict edges within the node test set.

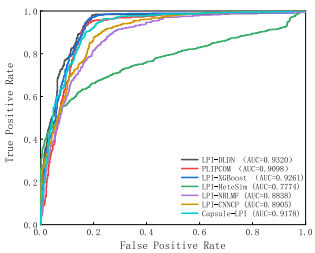
The above four CVs refer to LPI prediction for (1) new (unknown) lncRNAs (that is, lncRNAs that do not interact with any protein), (2) new proteins (that is, proteins that do not interact with any lncRNA), (3) new lncRNA-protein pairs, and (4) new independent lncRNAs and independent proteins.

In addition, negative LPIs are randomly selected from unlabeled lncRNA-protein pairs. The ratio of the selected negative samples to positive samples is 1, that is, the number of negative LPIs is the same as that of known LPIs.

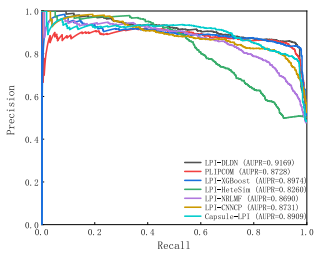
4.3 Comparison With Six State-of-the-Art LPI Prediction Methods

We compare our proposed LPI-DLDN method with six state-of-the-art LPI prediction methods, that is, LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI, to evaluate the prediction ability and robustness of LPI-DLDN. The ROC curves and the PR curves of the seven LPI prediction methods on 5 datasets under four different cross validations are shown in Figs. 2, 3, 4, and 5, respectively.

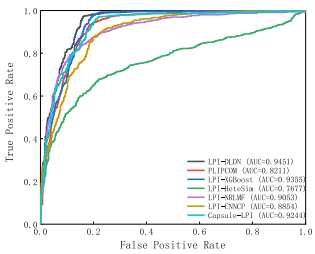
Table 1 in the Supplementary Materials, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TCBB.2021.3116232, shows the performance of seven LPI prediction models in terms of precision, recall, accuracy, F1-score, AUC and AUPR under CV1. LPI-DLDN obtains the highest average precision, F1-score, and AUC over the five datasets, significantly outperforms LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI. Although the average recall, accuracy and AUPR computed by LPI-DLDN are slightly lower than those computed by Capsule-LPI, LPI-XGBoost and LPI-HeteSim, respectively, the differences are small enough to be negligible. For example, the average recall calculated by Capsule-LPI is 0.7722, while the value obtained by LPI-DLDN is 0.7687, which is less than the 0.46% of Capsule-LPI. The average accuracy computed by LPI-XGBoost is 0.8199, while the value from LPI-DLDN is 0.8165, which is only smaller than 0.40% for LPI-XGBoost. The average AUPR obtained by LPI-HeteSim is 0.8185 while the value obtained by LPI-DLDN is 0.8150, which is only smaller than 0.43%. Fig. 2 demonstrates the ROC curves and the precision-recall (PR) curves of seven LPI prediction models on five datasets under CV1. LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI are state-of-the-art LPI prediction methods and obtain superior performance for new LPI identification. LPI-DLDN either significantly outperforms the six competing models or has very few differences. Therefore, LPI-DLDN is powerful for finding proteins that interact with a new lncRNA.



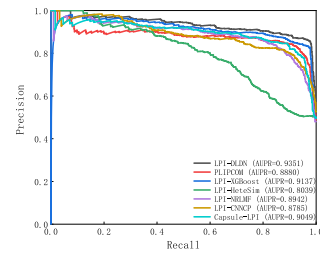
(a) The ROC curves of different methods on dataset 1



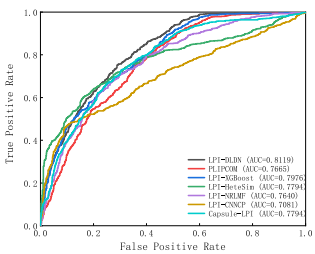
(b) The PR curves of different methods on dataset 1



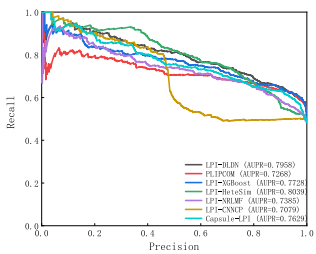
(c) The ROC curves of different methods on dataset 2



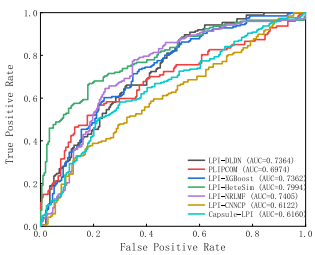
(d) The PR curves of different methods on dataset 2



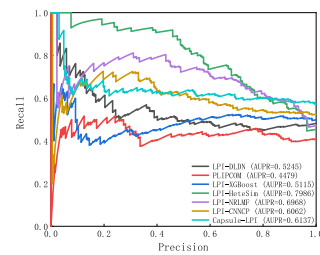
(e) The ROC curves of different methods on dataset 3



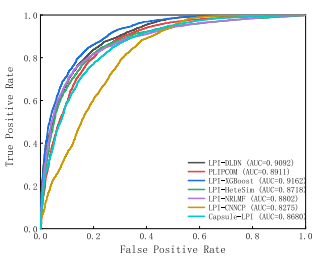
(f) The PR curves of different methods on dataset 3



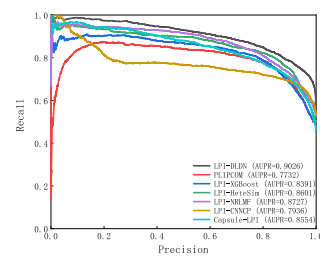
(g) The ROC curves of different methods on dataset 4



(h) The PR curves of different methods on dataset 4

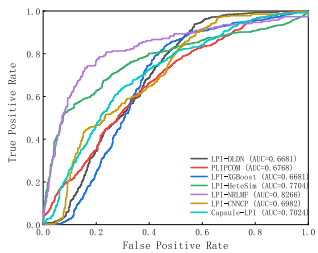


(i) The ROC curves of different methods on dataset 5

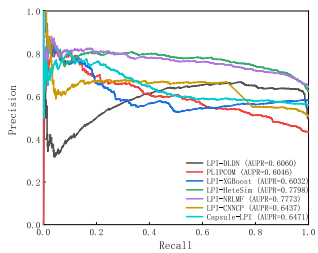


(j) The PR curves of different methods on dataset 5

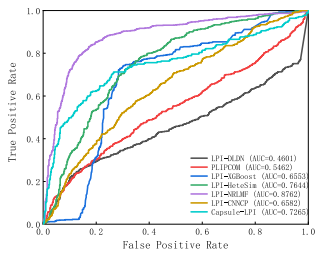
Fig. 2. The ROC curves and the PR curves of different methods under CV1.



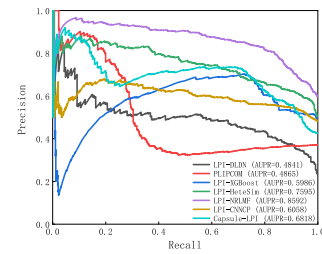
(a) The ROC curves of different methods on dataset 1



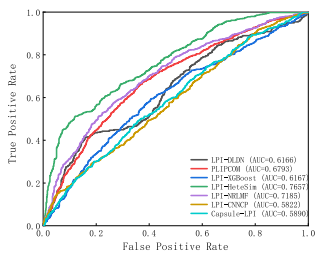
(b) The PR curves of different methods on dataset 1



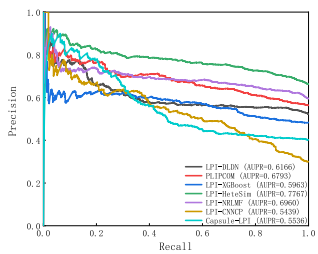
(c) The ROC curves of different methods on dataset 2



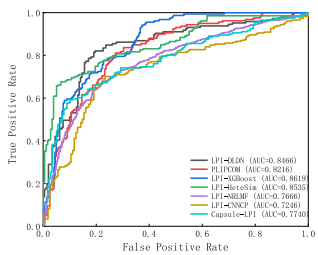
(d) The PR curves of different methods on dataset 2



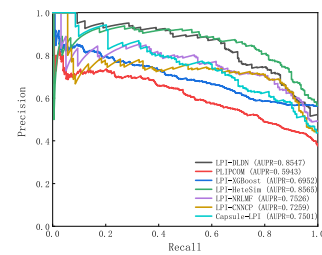
(e) The ROC curves of different methods on dataset 3



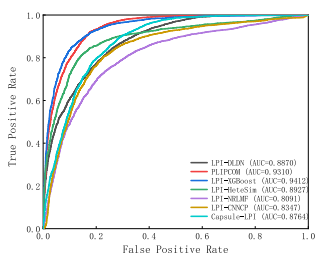
(f) The PR curves of different methods on dataset 3



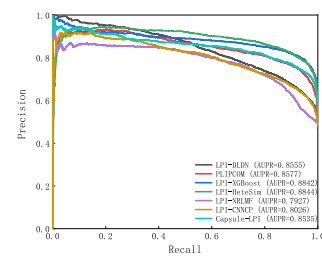
(g) The ROC curves of different methods on dataset 4



(h) The PR curves of different methods on dataset 4



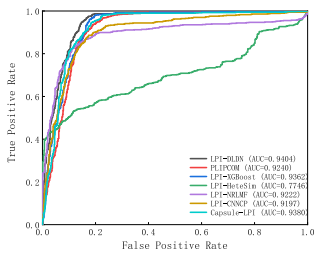
(i) The ROC curves of different methods on dataset 5



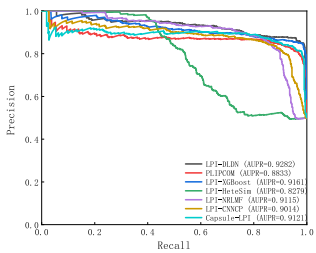
(j) The PR curves of different methods on dataset 5

Fig. 3. The ROC curves and the PR curves of different methods under CV2.

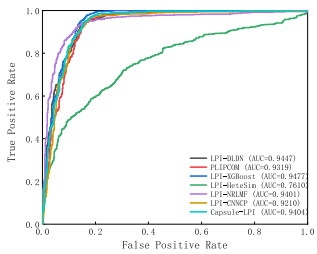
Table 2 in the Supplementary Materials, available online, demonstrates the comparison results under CV2. As shown in Table 2 in the Supplementary Materials, available online, although the average performances obtained from LPI-HeteSim and LPI-NRLMF are slightly better than those of LPI-DLDN, they are two network-based LPI prediction models, which have one severe shortage: network-based models cannot find possible interaction information for an orphan lncRNA (or protein). More importantly, under the majority of conditions, LPI-DLDN outperforms LPI-XGBoost and PLIPCOM, which are two better ensemble learning-based LPI inference approaches. In particular, AUPR is a more important measurement than the other five metrics. The average AUPR of LPI-DLDN outperforms LPI-XGBoost and PLIPCOM. The results suggest that LPI-DLDN may be an effective supervised learning method applied to identify potential lncRNAs associated with a new protein. In addition, LPI-DLDN outperforms LPI-CNNCP, which is a deep



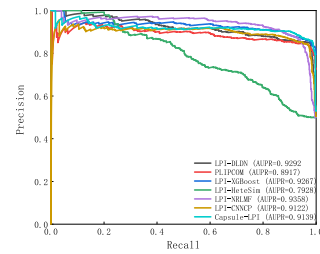
(a) The ROC curves of different methods on dataset 1



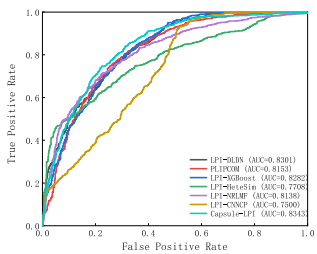
(b) The PR curves of different methods on dataset 1



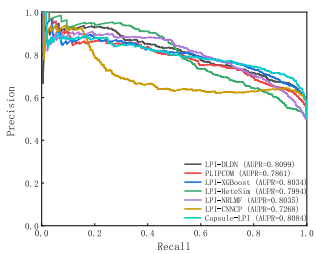
(c) The ROC curves of different methods on dataset 2



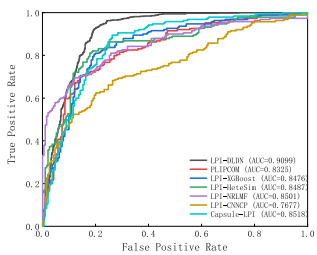
(d) PRs of different methods on dataset 2



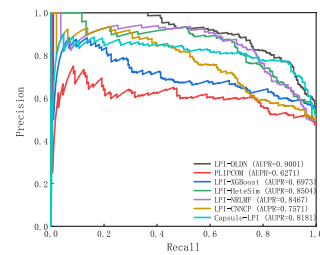
(e) The ROC curves of different methods on dataset 3



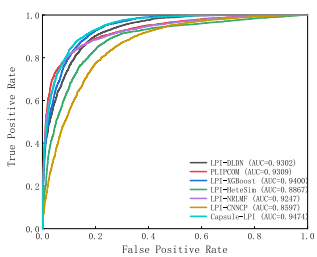
(f) PRs of different methods on dataset 3



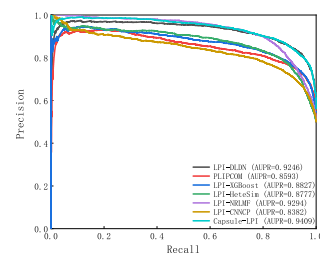
(g) The ROC curves of different methods on dataset 4



(h) The PR curves of different methods on dataset 4

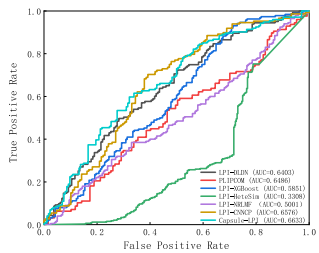


(i) The ROC curves of different methods on dataset 5

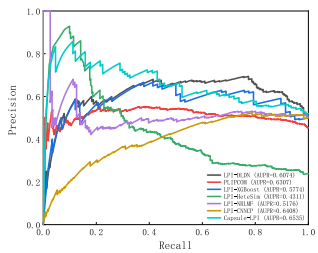


(j) The PR curves of different methods on dataset 5

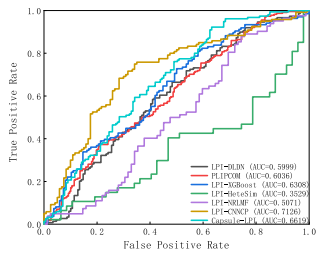
Fig. 4. The ROC curves and the PR curves of different methods under CV3.



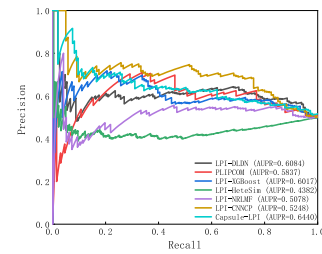
(a) The ROC curves of different methods on dataset 1



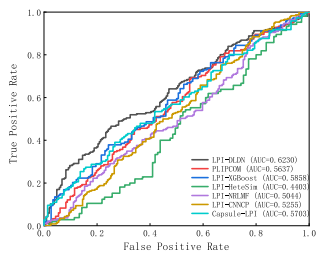
(b) The PR curves of different methods on dataset 1



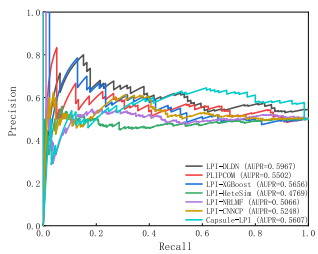
(c) The ROC curves of different methods on dataset 2



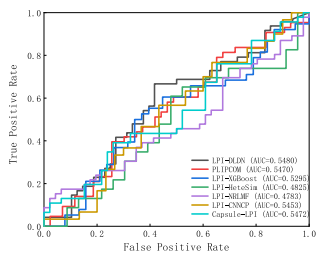
(d) The PR curves of different methods on dataset 2



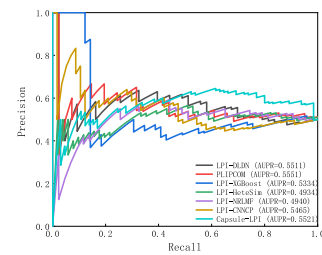
(e) The ROC curves of different methods on dataset 3



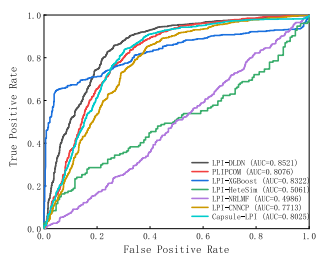
(f) The PR curves of different methods on dataset 3



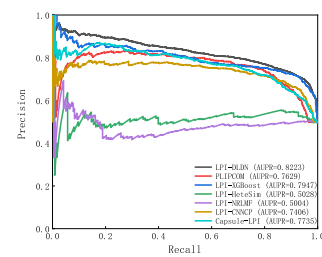
(g) The ROC curves of different methods on dataset 4



(h) The PR curves of different methods on dataset 4



(i) The ROC curves of different methods on dataset 5



(j) The PR curves of different methods on dataset 5

Fig. 5. The ROC curves and the PR curves of different methods under CV4.

learning-based LPI identification technique, again demonstrating the superiority of LPI-DLDN. Fig. 3 describes the ROC curves and the textcolorredPR curves of seven LPI prediction methods on five datasets under CV2.

The comparative results under CV3 are described in Table 3 in the Supplementary Materials, available online. The results demonstrate that LPI-DLDN significantly outperforms the other six LPI prediction models over all datasets in terms of precision, recall, F1-score, AUC, and AUPR. For example, LPI-DLDN computes the best average AUC value of 0.9110, which is 1.22%, 11.27%, 2.29%, 2.65%, 7.40%, and 0.95% better than LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI, respectively. More importantly, for the AUPR metric, LPI-DLDN achieves the best average AUPR of 0.8984, which is 1.46% superior to the second-best method and 2.0% superior to the third-best method. Fig. 4 illustrates the ROC curves and the PR curves of seven LPI prediction models on

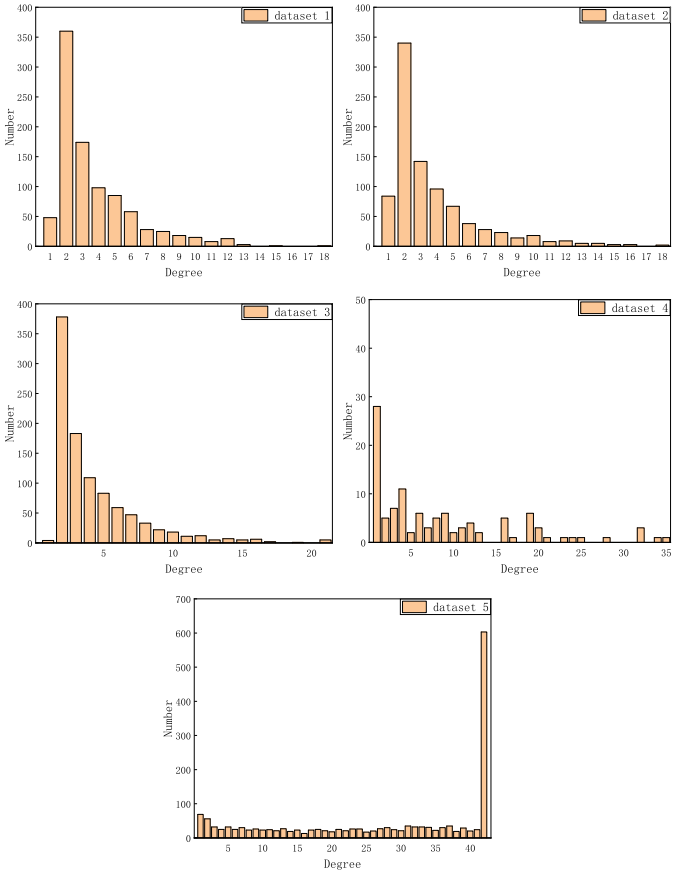


Fig. 6. The degrees of LPI networks in five LPI datasets.

five datasets under CV3. The results demonstrate the powerful classification ability of LPI-DLDN. Therefore, LPI-DLDN can be used to find new interactions between lncRNAs and proteins based on known LPIs.

The experimental results under CV4 are shown in Table 4 in the Supplementary Materials, available online. CV4 can ensure that no data leakage occurs in the analysis and that the training model has not seen any of the nodes in the test set by means of connection to another node in the train set. LPI-DLDN significantly outperforms LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, and LPI-CNNCP under the majority of conditions. More importantly, LPI-DLDN obtains the optimal average AUC and AUPR on five datasets although Capsule-LPI computes better precision, recall, accuracy, and F1-score. AUC and AUPR are two more representative measurements compared to the other four evaluation metrics. LPI-DLDN calculates the best average AUC of 0.6527, 3.7% higher than Capsule-LPI, and the best AUPR of 0.6372, 0.78% higher than Capsule-LPI. Fig. 5 shows the ROC curves and the PR curves of seven LPI prediction methods on five datasets under CV4.

4.4 The Degree of LPI Networks

Inspired by the description of the data distribution provided by Lan *et al.* [49], the degree in each LPI network is investigated. In this section, lncRNAs are taken as nodes and used to analyze five LPI networks. The results are shown as Fig. 6. In datasets 1-3, the distribution of the degrees of the nodes is very uneven. For example, the degrees of the majority of nodes are less than 12 in datasets 1-3. The number of nodes with a degree of 2 is 360, 340, and 378, respectively, accounting for a large proportion of the three human datasets. In dataset 4, the distribution of the degrees of the nodes is relatively even. The degrees of most of nodes are less than 20 and there are 22 nodes with a degree of 1. In dataset 5, the distribution of degrees of nodes is even, while the number of nodes with a degree of 42 is 603. The nonuniformity and imbalanced features of data result in prediction bias when data are not screened. That is, the prediction result will favor a certain category. Therefore, we select the same number of positive and negative samples in the train set and the test set to reduce the prediction bias.

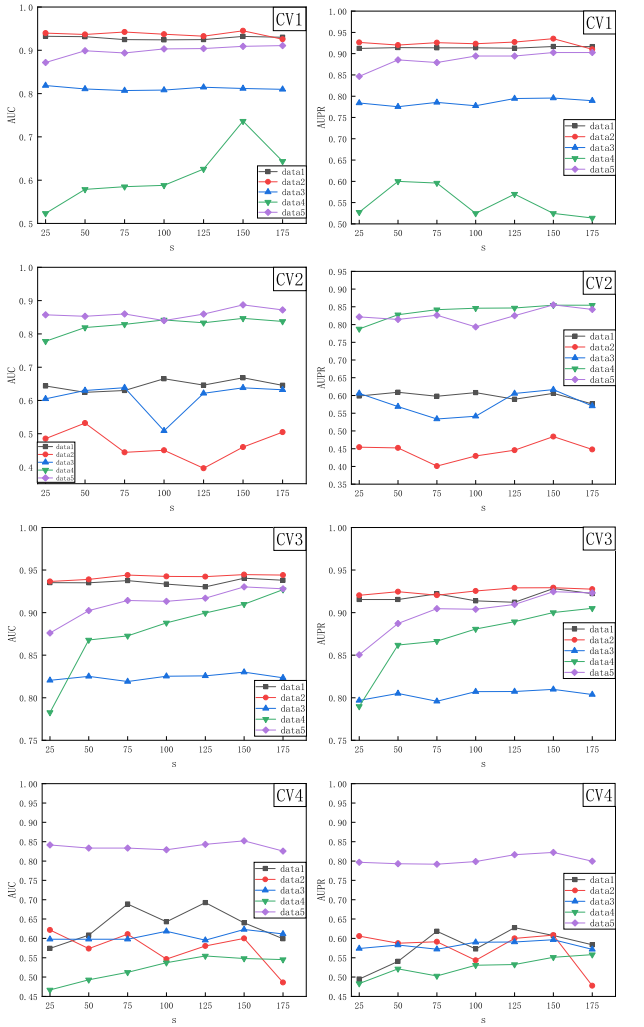


Fig. 7. AUCs and AUPRs based on different feature number s under four cross validations.

4.5 Evaluation of the Effect of Hyperparameter

In this section, we measure the effect of hyperparameter s on the prediction performance. s denotes the highest s features selected from $2d$ LPI features and $s < 2d$. To evaluate the effect of s on the classification performance, we set it in the range of (1, 200) with an interval of 25 and investigate the performance of LPI-DLDN on the five LPI datasets under the four cross validations. The results are shown in Fig. 7. From Fig. 7, we can find that LPI-DLDN computes the optimal AUCs and AUPRs when the feature number s is set as 150. Therefore, we choose s as 150.

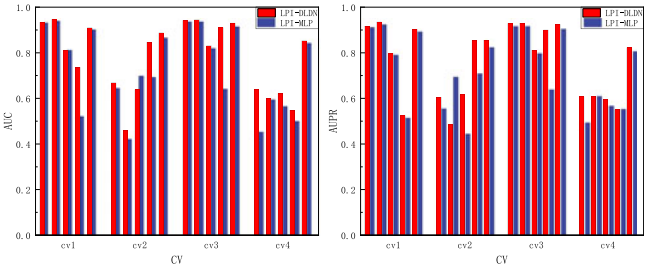


Fig. 8. AUCs and AUPRs of LPI-DLDN and LPI-MLP.

4.6 Comparison of the Dual-Net Architecture With the MLP Network

In the LPI-DLDN framework, we use a dual-net architecture composed of the MLP network and the FIR network, to select the best features and predict labels for each lncRNA-protein pair. To compare the performance of the dual-net architecture (LPI-DLDN) with the MLP network (LPI-MLP), we investigate the AUC and AUPR values of LPI-DLDN and LPI-MLP. The results are shown in Fig. 8. From Fig. 8, we can observe that LPI-DLDN obtains better AUCs and AUPRs than LPI-MLP on five LPI datasets under the four cross validations. The results suggest that the dual-net architecture significantly outperforms the MLP net and demonstrate the optimal classification capability of deep learning. Therefore, it is necessary to use the dual-net structure during LPI classification.

4.7 Case Study

In the last section, the performance of LPI-DLDN is validated. We further discover potential LPIs, especially possible proteins (or lncRNAs) for a new lncRNA (or protein).

4.7.1 Finding Associated Proteins for New lncRNAs

lncRNA FGD5 antisense RNA 1 (FGD5-AS1) has an important effect on multiple human cancers. For example, FGD5-AS1 can be used as a possible therapeutic target for colorectal cancer by suppressing cell migration, invasion, and proliferation, and accelerating cell apoptosis in colorectal cancer [50]. It may serve as a possible diagnostic biomarker for oral squamous cell carcinoma through binding to miR-520b against USP21 [51]. It could regulate human gastric cancer via the downstream epigenetic axis of hsa-miR-153-3p/CITED2 [52] and promote the cell proliferation of non-small cell lung cancer by sponging hsa-miR-107 to upregulate FGFRL1 [53].

In datasets 1-3, FGD5-AS1 (NONHSAT088370, n384228, and NONHSAT088370) interacts with 6, 6, and 8 proteins, respectively. To identify new proteins interacting with FGD5-AS1, all its association information is masked and it is taken as a new lncRNA. The seven LPI identification methods are then used to identify potential proteins associated with FGD5-AS1. The experiments are repeated 10 times, and the top 5 predicted proteins interacting with FGD5-AS1 are selected. The results are shown in Table 5 in the Supplementary Materials, available online. We observe that O00425, Q9Y6M1, and Q9NZI8 are predicted to interact with FGD5-AS1 in dataset 3. Although the associations between the above three proteins and FGD5-AS1 are unknown in dataset 3, O00425 has been validated to interact with FGD5-AS1 in dataset 1 and Q9Y6M1 and Q9NZI8 have been reported to interact with FGD5-AS1 in datasets 1 and 2. More importantly, all the top 5 predicted proteins interacting with FGD-AS1 have higher ranking in LPI-XGBoost, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI. The results show the powerful interaction prediction ability of LPI-DLDN for a new lncRNA.

4.7.2 Finding Potential lncRNAs Interacting With a New Protein

We intend to identify possible lncRNAs that interact with a new protein. Q9H9G7 is required for RNA-mediated gene silencing. The protein binds to short RNAs and represses the translation of mRNAs complementary to them. This process involves the stabilization of small RNA derivates in stem cells and the siRNA-dependent degradation of RNA polymerase II-transcribed coding mRNAs. Meanwhile, it still possesses RNA slicer activity [54].

Q9H9G7 interacts with 126, 126, and 137 lncRNAs in datasets 1, 2 and 3, respectively. We mask all association information for Q9H9G7 and use the proposed LPI-DLDN method to identify potential lncRNAs interacting with the protein. We repeatedly perform the experiment 10 times and obtain the average association scores for all lncRNA-protein pairs. The top 5 associated lncRNAs for Q9H9G7 on three human datasets are listed in Table 6 in the Supplementary Materials, available online. We predict that lncRNA n343060 may link to Q9H9G7 with a ranking of 3 on dataset 2. In addition, among 885 lncRNAs possibly associated with Q9H9G7, the interaction between n343060 and Q9H9G7 is ranked as 18, 207, 322, 2, 820, and 738 on the other six LPI prediction methods. The results suggest that n343060 may interact with Q9H9G7 but this remains to be further experimentally validated.

4.7.3 Finding New LPIs Based on Known LPIs

We further predict new LPIs based on LPI-DLDN. We repeat the experiment 10 times and compute the average interaction probabilities for all lncRNA-protein pairs on datasets 1-5. The predicted top 50 LPIs on five datasets, which contain known LPIs, are illustrated in Fig. 9. In the figure, sky blue solid lines and black dotted lines represent known and unknown LPIs obtained from LPI-DLDN, respectively. Deep sky blue circles and dark orange hexagons represent lncRNAs whose association information is known and unknown, respectively. Green diamonds denote proteins.

The interactions between NON-HSAT011709 (RPI001_236932) and Q15717, n338615 (RP11-439E19.10) and Q15717, NONHSAT006254 (RP11-196G18.22) and Q9NUL5, AthlncRNA309 (TCONS_00051077) and F4JLJ3, and ZmalncRNA1625 and B8A305 have the highest probability among unknown lncRNA-protein pairs on five datasets. There are 55,165, 74,340, 26,730, 3,815, and 71,568 lncRNA-protein pairs in the five datasets. Among all lncRNA-protein pairs, the five predicted interactions are ranked as 3, 13, 7, 583, and 853.

RP11-439E19.10 has been revealed to be upregulated. The lncRNA may promote ovarian tumor initiation and progression by interacting with proinflammatory cytokines [55]. More

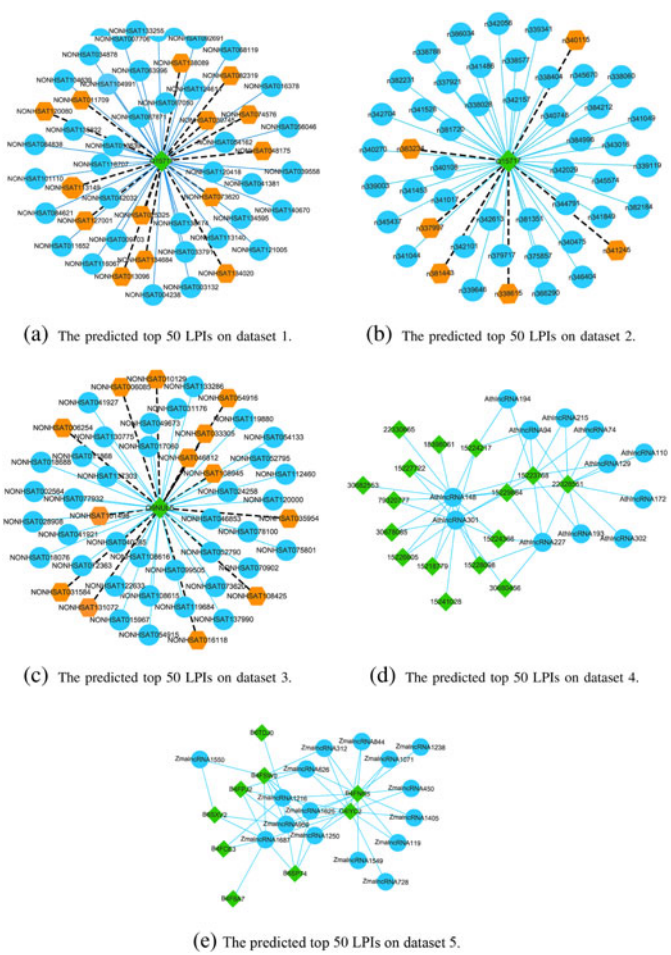


Fig. 9. The predicted top 50 LPIs on five LPI datasets.

importantly, it may be associated with the radiosensitivity of esophageal cancer stem cells and could possibly be used as a new target of esophageal squamous carcinoma. Q15717 is a RNA-binding protein [56], that assists in embryonic stem cell differentiation, regulates p53/TP53 expression, mediates the CDKN2A antiproliferative activity, and increases the stability of leptin mRNA [39].

In dataset 2, RP11-439E19.10 is verified to interact with Q13148, P35637, and Q01844. Q13148 regulates the splicing of proteins related to neuronal survival and mRNAs encoding proteins in neurodegenerative diseases. It can control mRNA stability and plays an important role in maintaining circadian clock periodicity and mitochondrial homeostasis. It also participates in the formation and regeneration of normal skeletal muscle [39]. P35637 is densely associated with various cellular processes. The protein can bind its own pre-mRNA and autoregulate its expression. It plays a key role in dendritic spine formation and stability, mRNA stability and synaptic homeostasis in neuronal cells [39]. Q01844 plays an important role in the tumorigenic process. The protein may disturb gene expression and assist in aberrant activation of fusion protein target genes [39]. Q15717 has similar functions to Q13148, P35637, and Q01844. Based on the "guilt-by-association" principle, similar lncRNAs may interact with similar proteins. More importantly, among all 55,165 lncRNA-protein pairs in dataset 1, the interaction between RP11-439E19.10 and Q15717 is ranked as 3 by LPI-DLDN. Therefore, we infer that RP11-439E19.10 may have dense linkage with Q15717.

In addition, we predict that RP11-196G18.22 may be closely associated with lung adenocarcinoma and adjacent normal samples [57]. Q9NUL5 can inhibit programmed -1 ribosomal frameshifting (-1PRF) of multiple mRNAs from viruses and cellular genes. The protein may cause premature translation termination. It may prevent the translation of DENV RNA, interrupt Zika virus replication, and limit hepatitis C virus replication [39]. We infer that RP11-196G18.22 may interact with Q9NUL5 with a ranking of 7 among all 26,730 lncRNA-protein pairs; however, further validation is needed.

5 Discussion and Further Research

lncRNAs have been validated to play significant roles in many biological processes. Furthermore, lncRNAs have close linkages with the origin and development of multiple human complex diseases. However, the majority of lncRNAs have not obvious functional annotations because of their poor evolutionary conservation. Therefore, it is instrumental to find the associations between lncRNAs and other biological entities (for example, proteins) and further interpret their biological functions and molecular mechanisms.

Recently, researchers have focused on constructing various computational models to identify new LPIs. Based on computational methods, the interaction probabilities between lncRNAs and proteins can be quantified and lncRNA-protein pairs with top rankings can be applied to further biomedical experimental validation, thereby reducing the time and cost of experiments. Therefore, computational methods provide effective guidance and support for new LPI identification.

In this manuscript, we exploit an LPI prediction method (LPI-DLDN) based on deep learning with a dual-net neural architecture. First, five LPI datasets are integrated based on existing data resources. Second, the features of lncRNAs and proteins are extracted via Pyfeat and BioTriangle, respectively. Third, the features are subjected to dimensional reduction based on PCA and concatenated as a vector to represent a lncRNA-protein pair. Finally, a deep learning model composed of the FIR and MLP nets is explored to predict new LPIs. We compare LPI-DLDN with six state-of-the-art LPI prediction models, LPI-XGBoost, LPI-HeteSim, LPI-NRLMF, PLIPCOM, LPI-CNNCP, and Capsule-LPI, on five LPI datasets under three cross validations. The experimental results show its powerful classification ability for unknown lncRNA-protein pairs. We further apply case studies to discover potential proteins (or lncRNAs) for a new lncRNA (or protein) and new LPIs based on known LPIs.

We investigate the classification ability of different models under four different cross validations. In particular, CV4 can ensure that no data leakage occurs in the analysis. LPI-DLDN obtains the best average performance on five datasets under four cross validations, especially for CV4. It may be attributed to the following features. First, it reasonably integrates multiple biological features of lncRNAs and proteins. Second, in three human LPI datasets, known LPIs are obtained from different resources. Some unlabeled lncRNA-protein pairs in one dataset have been reported in another dataset, thereby increasing the antiinterference capability of

models. Third, a deep learning model with dual-net neural architecture, composed of the FIR and MLP nets, demonstrates extreme classification power and interpretability. Fourth, the FIR method is designed to select the optimal LPI features and further boost the generation ability of the proposed LPI-DLDN model. Finally, the exploration-exploitation strategy used in LPI prediction can simultaneously utilize different feature subsets, which generates more training samples with fewer random LPI features.

In addition, LPI-DLDN obtains slightly lower performance under CV2. Compared to the other six LPI identification methods, LPI-DLDN performs better on datasets 4 and 5, while it achieves relatively lower performance on datasets 1, 2 and 3. This may be caused by different data structures. Under 5-fold cross validation, CV2 selects 20% of proteins as the test set. However, the number of proteins on five LPI datasets is 59, 84, 27, 35, and 42, respectively. That is, proteins are very small on the datasets, so the lack of samples may lead to unstable performance of LPI-DLDN under CV2. In addition, the degree of each protein (i.e., the number of lncRNAs interacting with each protein) on five datasets is unevenly distributed, thereby resulting in uneven data distribution during 5-fold cross validation.

In the future, we will integrate LPI data from multiple different species to more effectively probe the biological functions and mechanisms of lncRNAs. More importantly, we will design a better LPI feature selection model combining available data resources and the FIR algorithm. In addition, LPI-DLDN produces a high computational burden because of the computational complexity of the dual-net neural architecture. We will address this issue via the newest deep learning technique in subsequent research.

6 Data Availability

Source codes and datasets are freely available for download at https://github.com/plhhnu/LPI-DLDN.

Acknowledgments

The authors would like to thanks two anonymous reviewers for their valuable comments.

References

[1] J. S. Mattick, "RNA regulation: A new genetics?," *Nat. Rev. Genet.*, vol. 5, no. 4, pp. 316–323, 2004.
[2] K. C. Wang and H. Y. Chang, "Molecular mechanisms of long noncoding RNAs," *Mol. Cell*, vol. 43, no. 6, pp. 904–914, 2011.
[3] T. R. Mercer, M. E. Dinger, and J. S. Mattick, "Long non-coding RNAs: Insights into functions," *Nat. Rev. Genet.*, vol. 10, no. 3, pp. 155–159, 2009.
[4] X. Chen, C.-C. Zhu, and J. Yin, "Ensemble of decision tree reveals potential miRNA-disease associations," *PLoS Comput. Biol.*, vol. 15, no. 7, 2019, Art. no. e1007209.
[5] J. Whitehead, G. K. Pandey, and C. Kanduri, "Regulation of the mammalian epigenome by long noncoding RNAs," *Biochimica et Biophysica Acta (BBA)-General Subjects*, vol. 1790, no. 9, pp. 936–947, 2009.
[6] O. Wapinski and H. Y. Chang, "Long noncoding RNAs and human disease," *Trends Cell Biol.*, vol. 21, no. 6, pp. 354–361, 2011.
[7] X. Chen *et al.*, "Computational models for lncRNA function prediction and functional similarity calculation," *Brief. Functional Genomics*, vol. 18, no. 1, pp. 58–82, 2019.
[8] X. Chen and G.-Y. Yan, "Novel human lncRNA–disease association inference based on LncRNA expression profiles," *Bioinformatics*, vol. 29, no. 20, pp. 2617–2624, 2013.
[9] W. Zhang, X. Yue, G. Tang, W. Wu, F. Huang, and X. Zhang, "SFPEL-LPI: Sequence-based feature projection ensemble learning for predicting lncRNA-protein interactions," *PLoS Comput. Biol.*, vol. 14, no. 12, 2018, Art. no. e1006616.
[10] A. C. Bester *et al.*, "An integrated genome-wide CRISPRa approach to functionalize LncRNAs in drug resistance," *Cell*, vol. 173, no. 3, pp. 649–664, 2018.
[11] L. Peng *et al.*, "Probing LncRNA–protein interactions: Data repositories, models, and algorithms," *Front. Genet.*, vol. 10, 2020, Art. no. 1346.
[12] A. Li, M. Ge, Y. Zhang, C. Peng, and M. Wang, "Predicting long noncoding RNA and protein interactions using heterogeneous network model," *BioMed Res. Int.*, vol. 2015, 2015, Art. no. 671950.
[13] W. Zhang, Q. Qu, Y. Zhang, and W. Wang, "The linear neighborhood propagation method for predicting long non-coding RNA–protein interactions," *Neurocomputing*, vol. 273, pp. 526–534, 2018.
[14] M. Ge, A. Li, and M. Wang, "A bipartite network-based method for prediction of long non-coding rna–protein interactions," *Genomics, Proteomics Bioinformat.*, vol. 14, no. 1, pp. 62–71, 2016.
[15] Q. Zhao, H. Yu, Z. Ming, H. Hu, G. Ren, and H. Liu, "The bipartite network projection-recommended algorithm for predicting long non-coding RNA-protein interactions," *Mol. Ther.-Nucleic Acids*, vol. 13, pp. 464–471, 2018.
[16] G. Xie, C. Wu, Y. Sun, Z. Fan, and J. Liu, "LPI-IBNRA: Long noncoding RNA-protein interaction prediction based on improved bipartite network recommender algorithm," *Front. Genet.*, vol. 10, 2019, Art. no. 343.
[17] T. Zhang, M. Wang, J. Xi, and A. Li, "LPGNMF: Predicting long non-coding RNA and protein interaction using graph regularized nonnegative matrix factorization," *IEEE/ACM Trans. Comput. Biol. Bioinformat.*, vol. 17, no. 1, pp. 189–197, Jan./Feb. 2018.
[18] Q. Zhao, Y. Zhang, H. Hu, G. Ren, W. Zhang, and H. Liu, "IRWNRLPI: Integrating random walk and neighborhood regularized logistic matrix factorization for LncRNA-protein interaction prediction," *Front. Genet.*, vol. 9, 2018, Art. no. 239.
[19] H. Hu *et al.*, "HLPI-Ensemble: Prediction of human LncRNA-protein interactions based on ensemble strategy," *RNA Biol.*, vol. 15, no. 6, pp. 797–806, 2018.
[20] J. Yang, A. Li, M. Ge, and M. Wang, "Relevance search for predicting LncRNAprotein interactions based on heterogeneous network," *Neurocomputing*, vol. 206, no. 19, pp. 81–88, 2016.
[21] H. Liu *et al.*, "LPI-NRLMF: LncRNA-protein interaction prediction by neighborhood regularized logistic matrix factorization," *Oncotarget*, vol. 8, no. 61, pp. 103975–103984, 2017.
[22] J. S. Wekesa, J. Meng, and Y. Luan, "Multi-feature fusion for deep learning to predict plant LncRNA-protein interaction," *Genomics*, vol. 112, no. 5, pp. 2928–2936, 2020.
[23] L. Deng, J. Wang, Y. Xiao, Z. Wang, and H. Liu, "Accurate prediction of protein-LncRNA interactions by diffusion and hetesim features across heterogeneous network," *BMC Bioinformat.*, vol. 19, no. 1, pp. 1–11, 2018.
[24] S.-W. Zhang, X.-X. Zhang, X.-N. Fan, and W.-N. Li, "LPI-CNNCP: Prediction of lncRNA-protein interactions by using convolutional neural network with the copy-padding trick," *Anal. Biochem.*, vol. 601, 2020, Art. no. 113767.
[25] Y. Li, H. Sun, S. Feng, Q. Zhang, S. Han, and W. Du, "CAPSULE-LPI: A lncRNA–protein interaction predicting tool based on a capsule network," *BMC Bioinformat.*, vol. 22, no. 1, pp. 1–19, 2021.
[26] P. Larranaga *et al.*, "Machine learning in bioinformatics," *Brief. Bioinformat.*, vol. 7, no. 1, pp. 86–112, 2006.
[27] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*. Cambridge, MA, USA: MIT press, 2016.
[28] S. Min, B. Lee, and S. Yoon, "Deep learning in bioinformatics," *Brief. Bioinformat.*, vol. 18, no. 5, pp. 851–869, 2017.
[29] D. Ravì, C. Wong, F. Deligianni, M. Berthelot, J. Andreu-Perez, B. Lo, and G.-Z. Yang, "Deep learning for health informatics," *IEEE J. Biomed. Health Informat.*, vol. 21, no. 1, pp. 4–21, Jan. 2017.
[30] D. Shaw, H. Chen, M. Xie, and T. Jiang, "DEEPLPI: A multimodal deep learning method for predicting the interactions between lncRNAs and protein isoforms," *BMC Bioinformat.*, vol. 22, no. 1, pp. 1–22, 2021.
[31] C. Angermueller, T. Pärnamaa, L. Parts, and O. Stegle, "Deep learning for computational biology," *Mol. Syst. Biol.*, vol. 12, no. 7, 2016, Art. no. 878.

[32] A. M. A. and P. A. Thomas, "Comparative review of feature selection and classification modeling," in *Proc. Int. Conf. Adv. Comput., Commun. Control*, 2019, pp. 1–9.
[33] Y. M. Masoudi-Sobhanzadeh, H. Motieghader, and A. Masoudi-Nejad, "FeatureSelect: A software for feature selection based on machine learning approaches," *BMC Bioinformat.*, vol. 20, pp. 1–17, 2019, Art. no. 170.
[34] U. M. Khaire and R. Dhanalakshmi, "Stability of feature selection algorithm: A review," *J. King Saud*, Univ. Comput. Inf. Sci., 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157819304379
[35] S. *et al.*, "A matlab toolbox for feature importance ranking," in *Proc. Int. Conf. Med. Imaging Phys. Eng.*, 2019, pp. 1–6.
[36] M. Wojtas and K. Chen, "Feature importance ranking for deep learning," 2020, *arXiv:2010.08973*.
[37] J. Yuan, W. Wu, C. Xie, G. Zhao, Y. Zhao, and R. Chen, "Npinter v2. 0: An updated database of ncRNA interactions," *Nucleic Acids Res.*, vol. 42, no. D1, pp. D104–D108, 2014.
[38] C. Xie *et al.*, "NONCODEv4: Exploring the world of long non-coding RNA genes," vol. 42, no. D1, pp. D98–D103, 2014.
[39] U. Consortium, "Uniprot: A worldwide hub of protein knowledge," *Nucleic Acids Res.*, vol. 47, no. D1, pp. D506–D515, 2019.
[40] X. Zheng *et al.*, "Fusing multiple protein-protein similarity networks to effectively predict lncRNA-protein interactions," *BMC Bioinformat.*, vol. 18, no. 12, pp. 11–18, 2017.
[41] A. P. Pandurangan, J. Stahlhacke, M. E. Oates, B. Smithers, and J. Gough, "The superfamily 2.0 database: A significant proteome update and a new webserver," *Nucleic Acids Res.*, vol. 47, no. D1, pp. D490–D494, 2019.
[42] Y. Bai *et al.*, "PlncRNADB: A repository of plant lncRNAs and lncRNA-RBP protein interactions," *Curr. Bioinformat.*, vol. 14, no. 7, pp. 621–627, 2019.
[43] R. Muhammod, S. Ahmed, D. Md Farid, S. Shatabda, A. Sharma, and A. Dehzangi, "PyFeat: A python-based effective feature generation tool for DNA, RNA and protein sequences," *Bioinformatics*, vol. 35, no. 19, pp. 3831–3833, 2019.
[44] J. Dong *et al.*, "Biotriangle: A web-accessible platform for generating various molecular representations for chemicals, proteins, DNAs/RNAs and their interactions," *J. Cheminformat.*, vol. 8, no. 1, pp. 1–13, 2016.
[45] O. J. Mengshoel, "Understanding the role of noise in stochastic local search: Analysis and experiments," *Artif. Intell.*, vol. 172, no. 8–9, pp. 955–990, 2008.
[46] Y. Hechtlinger, "Interpretation of prediction models using the input gradient," 2016, *arXiv:1611.07634*.
[47] Y.-K. Zhou, Z.-A. Shen, H. Yu, T. Luo, Y. Gao, and P.-F. Du, "Predicting LncRNA-protein interactions with miRNAs as mediators in a heterogeneous network model," *Front. Genet.*, vol. 10, 2020, Art. no. 1341.
[48] Y. Park and E. M. Marcotte, "Flaws in evaluation schemes for pair-input computational predictions," *Nat. Methods*, vol. 9, no. 12, pp. 1134–1136, 2012.
[49] W. Lan *et al.*, "Predicting drug–target interaction using positive-unlabeled learning," *Neurocomputing*, vol. 206, pp. 50–57, 2016.
[50] D. Li, X. Jiang, X. Zhang, G. Cao, D. Wang, and Z. Chen, "Long noncoding RNA FGD5-as1 promotes colorectal cancer cell proliferation, migration, and invasion through upregulating CDCA7 via sponging miR-302e," *Vitro Cellular Develop. Biol.-Animal*, vol. 55, no. 8, pp. 577–585, 2019.
[51] L. Liu, Y. Zhan, Y. Huang, and L. Huang, "Lncrna FGD5-as1 can be predicted as therapeutic target in oral cancer," *J. Oral Pathol. Med.*, vol. 49, no. 3, pp. 243–252, 2020.
[52] Y. Gao, M. Xie, Y. Guo, Q. Yang, S. Hu, and Z. Li, "Long non-coding RNA FGD5-as1 regulates cancer cell proliferation and chemoresistance in gastric cancer through miR-153–3p/cited2 axis," *Front. Genet.*, vol. 11, 2020, Art. no. 715.
[53] Y. Fan, H. Li, Z. Yu, W. Dong, X. Cui, J. Ma, and S. Li, "Long non-coding RNA FGD5-as1 promotes non-small cell lung cancer cell proliferation through sponging hsa-miR-107 to up-regulate FGFRL1," *Biosci. Rep.*, vol. 40, no. 1, 2020, Art. no. BSR20193309.
[54] M. S. Park *et al.*, "Human Argonaute3 has slicer activity," *Nucleic Acids Res.*, vol. 45, no. 20, pp. 11 867–11 877, 2017.
[55] J. Song, W. Zhang, S. Wang, K. Liu, F. Song, and L. Ran, "A panel of 7 prognosis-related long non-coding RNAs to improve platinum-based chemoresistance prediction in ovarian cancer," *Int. J. Oncol.*, vol. 53, no. 2, pp. 866–876, 2018.
[56] J. Li and W. Sun, "Exploration of radiosensitivity-related LncRNAs in esophageal cancer stem cell," *Int. J. Radiation Oncol. Biol. Phys.*, vol. 102, no. 3, 2018, Art. no. e33.
[57] L. Zhang *et al.*, "Systematic identification of cancer-related long noncoding RNAs and aberrant alternative splicing of quintuple-negative lung adenocarcinoma through RNA-seq," *Lung Cancer*, vol. 109, pp. 21–27, 2017.

Lihong Peng received the PhD degree from the College of Information Science and Engineering, Hunan University, China. She is currently an associate professor with the Hunan University of Technology. Her research interests include machine learning, data mining, and bioinformatics.

Chang Wang is currently working toward the postgraduation degree with the School of Computer, Hunan University of Technology, China. His research interests include machine learning and bioinformatics.

Xiongfei Tian is currently working toward the postgraduation degree with the School of Computer, Hunan University of Technology, China. His research interests include machine learning and bioinformatics.

Liqian Zhou received the PhD degree from Xiangtan University, China. He is currently a professor with Hunan University of Technology . His research interests include machine learning, data mining, and bioinformatics.

Keqin Li (Fellow, IEEE) is currently a SUNY distinguished professor of computer science with the State University of New York and a distinguished professor with Hunan University, China. He has authored or coauthored more than 760 journal articles, book chapters, and refereed conference papers. His current research interests include cloud computing, fog computing and mobile edge computing, energy-efficient computing and communication, embedded systems and cyber-physical systems, heterogeneous computing systems, big data computing, high-performance computing, CPU-GPU hybrid and cooperative computing, computer architectures and systems, computer networking, machine learning, and intelligent and soft computing. He has chaired many international conferences. He is currently an associate editor for the *ACM Computing Surveys* and the *CCF Transactions on High Performance Computing*. He was on the editorial boards of the *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Computers*, *IEEE Transactions on Cloud Computing*, *IEEE Transactions on Services Computing*, and the *IEEE Transactions on Sustainable Computing*. He was the recipient of the several best paper awards.