LPI-deepGBDT: a multiple-layer deep framework based on gradient boosting decision trees for IncRNA–protein interaction identification

Liqian Zhou¹, Zhao Wang¹, Xiongfei Tian¹ and Lihong Peng¹,2*

Abstract

Background: Long noncoding RNAs (lncRNAs) play important roles in various biological and pathological processes. Discovery of IncRNA–protein interactions (LPIs) contributes to understand the biological functions and mechanisms of lncRNAs. Although wet experiments find a few interactions between lncRNAs and proteins, experimental techniques are costly and time-consuming. Therefore, computational methods are increasingly exploited to uncover the possible associations. However, existing computational methods have several limitations. First, majority of them were measured based on one simple dataset, which may result in the prediction bias. Second, few of them are applied to identify relevant data for new lncRNAs (or proteins). Finally, they failed to utilize diverse biological information of lncRNAs and proteins.

Results: Under the feed-forward deep architecture based on gradient boosting decision trees (LPI-deepGBDT), this work focuses on classify unobserved LPIs. First, three human LPI datasets and two plant LPI datasets are arranged. Second, the biological features of lncRNAs and proteins are extracted by Pyfeat and BioProt, respectively. Thirdly, the features are dimensionally reduced and concatenated as a vector to represent an IncRNA–protein pair. Finally, a deep architecture composed of forward mappings and inverse mappings is developed to predict underlying linkages between lncRNAs and proteins. LPI-deepGBDT is compared with five classical LPI prediction models (LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF, and LPI-HNM) under three cross validations on lncRNAs, proteins, lncRNA–protein pairs, respectively. It obtains the best average AUC and AUPR values under the majority of situations, significantly outperforming other five LPI identification methods. That is, AUCs computed by LPI-deepGBDT are 0.8321, 0.6815, and 0.9073, respectively and AUPRs are 0.8095, 0.6771, and 0.8849, respectively. The results demonstrate the powerful classification ability of LPI-deepGBDT. Case study analyses show that there may be interactions between GAS5 and Q15717, RAB30-AS1 and O00425, and LINC-01572 and P35637.

Conclusions: Integrating ensemble learning and hierarchical distributed representations and building a multiple-layered deep architecture, this work improves LPI identification.

*Correspondence: plhhnu@163.com
¹ School of Computer Science, Hunan University of Technology, No. 88, Taishan West Road, Tanyuan District, Zhuzhou, China
Full list of author information is available at the end of the article
prediction performance as well as effectively probes interaction data for new IncRNAs/proteins.

Keywords: IncRNA–protein interaction, Multiple-layer deep architecture, Gradient boosting decision tree

Introduction
Long noncoding RNAs (lncRNAs) are a class of important noncoding RNAs with the length more than 200 nucleotides. The class of RNAs have been reported to have dense associations with multiple biological processes including RNA splicing, transcriptional regulation, and cell cycle [1–3]. More importantly, the mutations and dysregulations of lncRNAs have important affects on multiple cancers [4, 5], for instance, lung cancer [6], colon cancer [7], and prostate cancer [8]. For example, lncRNAs UCA1, PCA3, and HOTAIR have been used as possible biomarkers of bladder cancer detection, prostate cancer aggressiveness, and hepatocellular carcinoma recurrence, respectively [9–11]. Although lncRNAs have been intensively investigated, functions and molecular mechanisms of lncRNAs still largely remain elusive [2, 12]. Recent researches have revealed that lncRNAs densely link to the corresponding binding-proteins. Therefore, the identification of the binding proteins for lncRNAs is urgent for better understanding the biological functions and molecular mechanisms of lncRNAs [1].

Although wet experiments for IncRNA–protein Interaction (LPI) discovery have been designed, computational methods are appealing to infer the relevances between IncRNAs and proteins [13]. The computational methods can be roughly divided into two categories: network-based methods and machine learning-based methods. Network-based LPI inference methods integrated various biological data and designed network propagation methods to find potential LPIs in the heterogeneous IncRNA–protein network. For example, Li et al. [14] proposed a random walk with restart-based LPI prediction model. Zhou et al. [15] took miRNAs as mediators to predict LPIs in a heterogeneous network (LPI-HNM). Yang et al. [16] used the HeteSim algorithm to compute the associated scores between lncRNAs and proteins. Zhao et al. [17], Ge et al. [18], and Xie et al. [19] explored a few bipartite network projection-based recommendation techniques to compute the interaction probabilities between lncRNAs and proteins. Zhang et al. [20] explored a novel LPI prediction framework combining a linear neighborhood propagation algorithm. Zhou et al. [21] combined similarity kernel fusion and Laplacian regularized least squares to find unobserved LPIs (LPI-SKF).

Machine learning-based LPI inference methods characterized the biological features of lncRNAs and proteins and exploited machine learning algorithms to probe LPI candidates [22]. Machine learning-based LPI prediction methods contain matrix factorization techniques and ensemble learning techniques [23]. Matrix factorization-based LPI prediction approaches used various matrix factorization techniques. Liu et al. [24] identified new LPIs comapping neighborhood regularized logistic matrix factorization. Zhao et al. [25] inferred LPI candidates combining the neighborhood regularized logistic matrix factorization model and random walk. Zhang et al. [26] proposed a graph regularized nonnegative matrix factorization method to uncover unobserved LPIs.

Ensemble learning-based LPI inference methods utilized diverse ensemble techniques. Zhang et al. [27] exploited an ensemble learning model to discover the interactions
between lncRNAs and proteins. Liu et al. [24] designed three ensemble strategies to predict LPIs based on support vector machine, random forest and extreme gradient boosting, respectively. Deng et al. [1] extracted HeteSim features and diffusion features of lncRNAs and proteins and constructed a gradient tree boosting-based LPI prediction algorithm (PLIPCOM). Fan and Zhang [28] explored a stacked ensemble-based LPI classification model via logistical regression (LPI-BLS). Deng et al. [29] proposed a gradient boosted regression tree for finding possible LPIs. Wekesa et al. [30] designed a categorical boosting-based LPI discovery framework (LPI-CatBoost). In addition, deep learning (such as deep graph neural network [31]) is increasingly developed to identify LPI candidates.

Computational methods effectively identified potential LPIs. However, there are a few problems to solve. First, the majority of computational models were evaluated on one dataset, which may result in predictive bias. Second, they were not used to infer potential proteins (or lncRNAs) associated with a new lncRNA (or protein). Finally, their prediction performance need to further improve.

To solve the above problems, in this study, inspired by Gradient Boosting Decision Trees (GBDT) provided by Feng et al. [32], we exploit a multiple-layer Deep structure with GBDT to predict unobserved LPIs (LPI-deepGBDT). First, five LPI datasets are constructed. Second, lncRNA and protein features are extracted by Pyfeat and BioProt, respectively. Third, a feature vector is built to represent an lncRNA–protein pair. Finally, a multiple-layer deep architecture integrating tree ensembles and hierarchical distributed representations is developed to classify lncRNA–protein pairs.

The remaining of this manuscript is organized as follows. “Materials and methods” section describes data resources and the LPI-deepGBDT framework. “Results” section illustrates the results from a series of experiments. “Discussion and further research” section discusses the LPI-deepGBDT method and provides directions for further research.

**Materials and methods**

**Data preparation**

In this manuscript, we collect three human LPI datasets and two plant LPI datasets. Dataset 1 provided by Li et al [14] contains 3,487 LPIs from 938 lncRNAs and 59 proteins. 3,479 LPIs between 935 lncRNAs and 59 proteins are finally obtained by removing the lncRNAs without sequence information in the NONCODE [33], NPInter [34] and UniProt [35] databases.

Dataset 2 build by Zheng et al. [36] contains human 4,467 LPIs between 1,050 lncRNAs and 84 proteins. 3,265 LPIs from 885 lncRNAs and 84 proteins are extracted after removing the lncRNAs without any sequence information. Dataset 3 constructed by Zhang et al. [20] contains 4,158 LPIs between 990 lncRNAs and 27 proteins.

Datasets 4 provides 948 Arabidopsis thaliana LPIs from 109 lncRNAs and 35 proteins. Dataset 5 provides 22,133 Zea mays LPIs from 1,704 lncRNAs and 42 proteins. The sequence information of two entities is downloaded from the PlncRNADB database [37] and LPIs are extracted at http://bis.zju.edu.cn/PlncRNADB/. The details are described in Table 1.

We denote an LPI network via a matrix $Y$:
Overview of LPI-deepGBDT

In this study, we develop a feed-forward deep framework to infer new LPIs. Figure 1 describes the flowchart of LPI-deepGBDT. As shown in Fig. 1, the LPI-deepGBDT framework consists of three main processes after LPI datasets are built. (1) Feature extraction. Pyfeat [38] and BioProt [39] are used to extract the original features for lncRNAs and proteins. (2) Feature selection. The lncRNA and protein features are reduced into two $d$-dimensional vector based on dimensional reduction analysis with Principle Component Analysis (PCA). The two vectors are then concatenated to depict lncRNA–protein pairs. (3) Classification. A multiple-layer deep

$$y_{ij} = \begin{cases} 
1, & \text{if } \text{lncRNAs } l_i \text{ interacts with protein } p_j \\
0, & \text{otherwise} 
\end{cases} \quad (1)$$

Table 1 The statistics of LPI information

| Dataset   | lncRNAs | Proteins | LPIs  |
|-----------|---------|----------|-------|
| Dataset 1 | 935     | 59       | 3479  |
| Dataset 2 | 885     | 84       | 3265  |
| Dataset 3 | 990     | 27       | 4158  |
| Dataset 4 | 109     | 35       | 948   |
| Dataset 5 | 1704    | 42       | 22,133|

Fig. 1 The flowchart of the LPI-deepGBDT framework. (1) Feature selection. (2) Dimension reduction. (3) Classification.
structure, composed of forward mapping and inverse mapping, is developed to classify lncRNA–protein pairs.

**Feature extraction**

*Feature extraction of lncRNAs*

Pyfeat [38] is widely applied to generate numerical features via sequence information. In this study, we use Pyfeat to obtain lncRNA features and represent an lncRNA as a 3,051-dimensional vector. The details are shown in Table 2.

*Feature extraction of proteins*

BioProt [39] utilizes various information to represent a protein. In this study, we use BioProt to obtain protein features and represent each protein as a 9,890-dimensional vector. The details are shown in Table 3.

**Dimension reduction**

The feature dimensions of lncRNAs and protein are reduced based on PCA, respectively. Two $d$-dimensional feature vectors are obtained and concatenated as a $2d$-dimensional vector $x$ applied to represent an lncRNA–protein pair.

**LPI prediction framework**

*Problem description*

For a given LPI dataset $D = (X, Y)$, where $(x, y)$ represents an lncRNA–protein pair (a training example), $x \in X$ denotes a $2d$-dimensional LPI feature vector and $y \in Y$ denotes its label, we aim to classify unknown lncRNA–protein pairs.

For a feed-forward deep architecture with one original input layer, one output layer and $(m-1)$ intermediate layers, suppose that $o_i$ ($i \in \{0, 1, 2, \ldots, m\}$) denotes the output in the $i$-th layer. For an lncRNA–protein pair $(x, y)$, we want to learn a mapping $F_i$ based on GBDT to minimize the empirical loss $L$ between the desired output $y$ and the final real output $o_m$ on the training data.

| Feature name                  | Number of features |
|-------------------------------|--------------------|
| zCurve                       | 3                  |
| gcContent                    | 1                  |
| ATGC ratio                   | 1                  |
| Cumulative Skew              | 2                  |
| Chou’s Pseudocomposition     | 84                 |
| monoMonoKGap                 | 16                 |
| monoDiKGap                   | 256                |
| monoTriKGap                  | 64                 |
| diMonoKGap                   | 64                 |
| diDiKGap                     | 1024               |
| diTriKGap                    | 256                |
| triMonoKGap                  | 256                |
| triDiKGap                    | 1024               |
GBDT can generate highly robust, interpretable and competitive classification procedures, especially for exploiting less than clean data [29, 40, 41]. For an lncRNA–protein pair \((x, y)\), an estimator \(f(x)\) denotes an approximate function response to the label \(y\), the GBDT model iteratively builds \(K\) different individual decision tree \(\{g(x; \alpha_1), \ldots, g(x; \alpha_K)\}\) using the training data \(D = (X, Y)\). And \(f(x)\) can be denoted as an expansion of individual decision tree \(g(x; \alpha_k)\) by Eq. (2).

\[
\begin{align*}
  f(x) &= \sum_{k=1}^{K} f_k(x) = \sum_{k=1}^{K} \beta_k g(x; \alpha_k) \\
  g(x; \alpha_k) &= \sum_{j=1}^{J} \gamma_{jk} I(x \in R_{jk})
\end{align*}
\]  

(2)

where each tree splits the input space into \(N\) disjoint regions \(\{R_{1k}, \ldots, R_{jk}\}\) and calculates a constant value \(\gamma_{jk}\) for the region \(R_{jk}\) where \(I = 1\) if \(x \in R_{jk}\); \(I = 0\), otherwise. \(f_k(x)\) denotes an addition function combined from the first decision tree to the \(k\)-th decision tree. The parameters \(\alpha_k\) denotes the mean values of partition locations and the terminal leaf nodes for each partitioning variables in the \(k\)-th decision tree. The parameters \(\beta_k\) denotes the weights used to determine how to effectively integrate the prediction results from individual decision trees when the leaf nodes of each collection are known. The two parameters \(\alpha_k\) and \(\beta_k\) can be estimated by minimizing a loss function \(L(y, f(x))\) by Eq. (3).

| Feature group              | Features                      | Number |
|---------------------------|-------------------------------|--------|
| Amino acid composition    | Amino acid composition        | 20     |
|                           | Dipeptide composition         | 400    |
|                           | Tripeptide composition        | 8000   |
| Autocorrelation           | Normalized Moreau–Broto       | 240    |
|                           | Moran autocorrelation         | 240    |
|                           | Geary autocorrelation         | 240    |
| CTD                       | Composition                  | 21     |
|                           | Transition                   | 21     |
|                           | Distribution                 | 105    |
| Conjoint triad            | Conjoint triad features      | 343    |
| Quasi-sequence order      | Sequence order coupling       | 60     |
|                           | Quasi-sequence order descriptors | 100   |
| Pseudo amino acid composition | Pseudo amino acid composition | 50     |
|                           | Amphiphilic pseudo amino acid composition | 50  |

Table 3 The protein features by BioProt
to solve the model (3), Friedman [42] proposed a gradient boosting approach. First, the parameters \( \alpha_m \) can be estimated based on least square error:

\[
(\alpha_k, \beta_k) = \arg \min_{\alpha, \beta} \sum_{i=1}^{N} L(y_i, f_{k-1}(x_i) + \beta g(x_i; \alpha))
\]

\[
= \arg \min_{\alpha, \beta} \sum_{i=1}^{N} L(y_i, f_{k-1}(x_i) + \beta \sum_{j=1}^{J} \gamma_j I(x_i \in R_j))
\]

and

\[
f_k(x) = f_{k-1}(x) + \beta_k g(x; \alpha_k) = f_{k-1}(x) + \beta_k \sum_{j=1}^{J} \gamma_j I(x \in R_j)
\]

To solve the model (3), Friedman [42] proposed a gradient boosting approach. First, the parameters \( \alpha_m \) can be estimated based on least square error:

\[
\alpha_k = \arg \min_{\alpha, \beta} \sum_{i=1}^{N} [\tilde{y}_{ik} - \beta g(x_i; \alpha)]^2 = \arg \min_{\alpha, \beta} \sum_{i=1}^{N} [\tilde{y}_{ik} - \beta \sum_{j=1}^{J} \gamma_j I(x_i \in R_j)]^2
\]

where \( \tilde{y}_{im} \) denotes the gradient and is defined by Eq. (6).

\[
\tilde{y}_{ik} = -\left[ \frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f(x)=f_{k-1}(x)}
\]

The parameters \( \beta_k \) can be determined by Eq. (7).

\[
\beta_k = \arg \min_{\beta} \sum_{i=1}^{N} L(y_i, f_{k-1}(x_i) + \beta g(x_i; \alpha_k))
\]

\[
= \arg \min_{\beta} \sum_{i=1}^{N} L(y_i, f_{k-1}(x_i) + \beta \sum_{j=1}^{J} \gamma_j I(x_i \in R_j))
\]

The estimator \( f_k(x) \) for the \( k \)-th regression tree can be updated by Eq. (8)

\[
f_k(x) = f_{k-1}(x) + \beta_k g(x, \alpha_k)
\]

The final estimator \( f(x) \) can be obtained by Eq. (9)

\[
f(x) = \sum_{k=1}^{K} f_k(x)
\]

The gradient boosting approach calculates the optimal values of the parameters \( \alpha_m \) via minimizing the least square function defined by Eq. (5). The parameters \( \beta_m \) can be solved by Eqs. (5) and (7). And the GBDT algorithm is described as Algorithm 1.

\textbf{Algorithm 1: The gradient boosting decision tree algorithm}

\textbf{Require:} The LPI feature vector \( x \); the label \( y \) for each IncRNA-protein pair;
\textbf{Ensure:} The final estimator \( f(x) \)

1: Initialize \( f_0(x) = f_0(x) = \arg \min_{\beta} \sum_{i=1}^{N} L(y_i, \beta) \)

2: Compute the negative gradient \( \tilde{y}_{ik} \) by Eq. (6)

3: Compute the parameters \( \alpha_k \) by Eq. (5)

4: Fit the decision tree \( g(x; \alpha_k) \) to the gradient \( \tilde{y}_{ik} \)

5: Compute the parameters \( \beta_k \) by Eq. (7)

6: Update \( f_k(x) \) by Eq. (8)

7: Repeat steps 2-6 for each decision tree

8: Obtain the final estimator \( f(x) \) by Eq. (9)
The multi-layered deep architecture with GBDT

We exploited a multi-layered deep architecture with GBDT to classify unknown lncRNA–protein pairs. Firstly, \( m \) gradient boosting decision trees are initialized. Initial forward mapping, inverse mapping, and output are then computed. Second, pseudo-label in the \( m \)-th layer is obtained based on the initialized output and real label. Third, the forward mapping for each regression tree is iteratively updated based on the computed pseudo-label at the last iteration. Fourth, the inverse mapping is iteratively learned based on the achieved forward mapping at the last iteration. Finally, the final label is output after \( m \) iterations.

Phase I: Initialize GBDT

It is very difficult to design a random tree structure based on the distribution from all potential tree configurations. Therefore, multiple Gaussian noise data are injected to the output in all intermediate layers. Given a deep structure with \( m \) layers, the initial forward mapping \( F_0^m (i \in \{1, 2, \ldots, m\}) \) and the inverse mapping \( G_0^m (i \in \{2, 3, \ldots, m\}) \) are obtained by a few very tiny trees, where index 0 represents the tree structures achieved in the initialization procedure. In addition, the initial output \( o_0 \) is set as \( X \) and \( o_i = F_0^m (o_{i-1}) (i \in \{1, 2, \ldots, m\}) \).

The iterations are updated based on the learned forward mappings and inverse mappings. At each iteration \( t \), we conduct Phases II-IV.

Phase II: Compute the pseudo-label in the \( m \)-th layer

The pseudo-label in the \( m \)-th layer can be computed based on the final output \( o_m \) and the real label \( y \), \( \alpha \) is the learning rate by Eq. (10)

\[
p_m^t = o_m - \alpha \frac{\partial L(o_m, y)}{\partial o_m}
\]  

(10)

Phase III: Forward mapping

At the \( t \)-th iteration, during the forward mapping, \( F_t^i \) for each regression tree in a GBDT is first initialized by \( F_t^i = F_{t-1}^i \) and updated based on a pseudo-labels \( p_{t-1}^i \) with \( p_{t-1}^i = G_i(p_t^i) \). The details are described as follows.

For each regression tree in a GBDT, we define a reconstruction loss function as Eq. (11).

\[
L_{\text{forw}}^i = ||F_t^i(o_{i-1}) - p_t^i||
\]  

(11)

The pseudo-residuals for each tree can be computed by Eq. (12).

\[
r_{\text{forw}}^i = -\frac{\partial L_{\text{forw}}^i}{\partial F_t^i(o_{i-1})}
\]  

(12)

When the pseudo-label in each layer is calculated, each \( F_{t-1}^i \) can implement a gradient ascent towards its pseudo-residual by Eq. (12).

Each regression tree \( g_k \) is fitted to \( r_{\text{forw}}^i \) based on the training set \((o_{i-1}, r_{\text{forw}}^i)\) and the forward mapping \( F_t^i \) for each tree can be updated by Eq. (13).

\[
F_t^i = F_t^i + \gamma g_k
\]  

(13)

Finally, we obtain the output for each layer by the forward mapping by Eq. (14).
The forward mapping procedures are described as Algorithm 2.

**Algorithm 2: The forward mapping**

**Require:** Training data $(X, Y)$, loss function $L$, the number of layers $m$, the number of regression trees in a GBDT $K_1$, and \( m \)

**Ensure:** The output in the \( i \)-th layer \( o_i \) (\( i \in \{1, 2, ..., m\} \))

1. Initialize $F_i^{(0)}$ by $F_i^{(0)} = F_i^{(t-1)}$
2. For each decision tree in a GBDT (\( k = 1 \) to \( K_1 \)), conduct the following steps 1-4:
3. Step 1: Reconstruct the loss function by Eq. (11)
4. Step 2: Compute the pseudo-residuals by Eq. (12)
5. Step 3: Fit regression tree \( g_k \) to \( r_k^{inv} \) using the training data \((o_{i-1}, r_k^{forw})\)
6. Step 4: Update $F_i^{(t)}$ by Eq. (13)
7. Compute the output \( o_i \) in each layer by Eq. (14)

In this phase, we use a bottom up update technique, that is, \( F_i \) will be updated before \( F_j \) when \( i < j \). In addition, each \( F_i \) can run multiple rounds of additive boosting operations towards its current pseudo-label.

**Phase IV: Inverse mapping**

At the \( t \)-th iteration, for each decision tree, given the forward mapping \( F_i^{(t-1)} \) learned from the \((t-2)\)-th iteration, we intend to achieve an “pseudo-inverse” mapping \( G_i^{(t)} \) paired with each \( F_i^{(t-1)} \) satisfying \( G_i^{(t)}(F_i^{(t-1)}(o_{i-1})) \approx o_{i-1} \) based on the following expected value of the reconstructed loss function by Eq. (15):

$$
\hat{G}_i^{(t)} = \arg \min_{G_i^{(t)}} \mathbb{E}_x[L_i^{inv}(o_{i-1}, G_i^{(t)}(F_i^{(t-1)}(o_{i-1})))]
$$

where \( L_i^{inv} \) denotes the reconstructed loss in the \( i \)-th layer.

To build a more robust and generative model, random noises \( \sigma \) are injected into the output in all intermediate layers:

$$
o_{i-1}^{noise} = o_{i-1}^{noise} + \epsilon, \epsilon \sim \mathcal{N}(0, \text{diag}(\sigma^2))
$$

For each regression tree \( g_k \) in a GBDT, the reconstructed error can be computed by Eq. (17):

$$
L_i^{inv} = ||G_i^{(t)}(F_i^{(t-1)}(o_{i-1}^{noise})) - (o_{i-1}^{noise})||
$$

Based on the noise injection, each \( G_i^{(t-1)} \) follows a gradient ascent towards the pseudo-residuals by Eq. (18):

$$
r_k^{inv} = -\frac{\partial L_i^{inv}}{\partial G_i^{(t)}(F_i^{(t-1)}(o_{i-1}^{noise}))}
$$

where \( r_k^{inv} \) denotes the pseudo-residuals of the \( k \)-th regression tree during the inverse mapping. For each regression tree \( g_k \) in GBDT, we fit it to \( r_k \) via the training set \((F_i^{(t-1)}(o_{i-1}^{noise}), r_k^{inv})\) and then update \( G_i^{(t)} \) by Eq. (19).
Finally, the pseudo-label in each intermediate layer can be propagated from the final layer to the first layer by Eq. (20):

$$p_{i-1} = G_i^t(p_i^t)$$  \hspace{1cm} (20)$$

For all intermediate layers and the final output layer ($i \in \{m, m-1, \ldots, 2\}$), the inverse mapping procedures are described as Algorithm 3.

---

**Algorithm 3: The inverse mapping**

Require: Training data ($X, Y$), loss function $L$, number of layers $m$, the number of regression trees in a GBDT $K$, noise injection $\sigma$, and $\gamma$

Ensure: $G_i^t$ and $p_{i-1}^t$ ($i \in \{m, m-1, \ldots, 2\}$)

1: Initialize $G_0^t$: $G_0^t = G_{i-1}^t$
2: Inject noise data to all intermediate layers by Eq. (16)
3: For each decision tree in a GBDT ($k=1$ to $K$), conduct the following Steps 4-7:
   4: Step 1 Reconstruct the loss function by Eq. (17)
   5: Step 2 Compute the pseudo-residuals by Eq. (18)
   6: Step 3 Fit regression tree $g_k$ to $r_{i-1}^{uv}$ using the training data ($F_{i-1}^{uv}(\alpha^{u+1}), r_{i-1}^{uv}$)
   7: Step 4 Update $G_i^t$ by Eq. (19)
8: Compute the pseudo-label $p_{i-1}^t$ by Eq. (20)

We can obtain the inverse mapping $G_i^t$ for the final output layer and all intermediate layers and the pseudo-labels $p_i^t$ for the first layer and all the intermediate layers. After finishing the $t$-th iteration, we continue the $(t+1)$-th iteration to update $F_i$ and $G_i$.

During LPI prediction, a linear classifier $Y = XW^T + b$ is applied to the forward mapping in the $m$-th layer. There are two main advantages. First, the $m-1$ layers can re-represent the LPI features as linearly separable as possible. Second, the corresponding inverse mapping in the $m$-th layer does not have to be computed because the pseudo-label in the $(m-1)$-th layer can be obtained based on the gradient of global loss related to the output in the $(m-2)$-th layer.

**Results**

The experiments is mainly explored to empirically examine if the proposed LPI-deepGBDT method can effectively predict new LPIs.

**Evaluation metrics**

The six measurements are utilized to evaluate the performance of LPI-deepGBDT: precision, recall, accuracy, F1-score, AUC and AUPR. For the six evaluation criteria, higher values depict better performance [43]. The experiments are repeatedly implemented for 20 times. The average performance for the 20 rounds is taken as the final performance. The six measurements are defined by Eqs. (21)–(24).

$$\text{Precision} = \frac{TP}{TP + FP}$$  \hspace{1cm} (21)
where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively. Precision denotes the ratio of correctly predicted positive samples among all predicted positive samples. Recall represents the ratio of correctly predicted positive samples among all real positive samples. Accuracy denotes the ratio of correctly predicted positive and negative samples among all samples. F1-Score is harmonic mean between precision and recall. Area Under receiver operating Characteristic Curve (AUC) is used to measure the trade-off between TP ratio and FP ratio. Area Under Precision-Recall curve (AUPR) is applied to evaluate the trade-off between precision and recall.

**Experimental settings**

The parameters in Pyfeat are set as: kgap=5, ktuple=3, optimum=1, pseudo=1, zcurve=1, gc=1, skew=1, atgc=1, monoMono=1, monoDi=1, monoTri=1, diMono=1, diDi=1, diTri=1, triMono=1, and triDi=1. All parameters in BioProt and LPI-SKF are the corresponding values provided by refs. [39] and [21], respectively. The deep GBDT architecture we used is (input-16-16-output). The parameters in the remaining methods are set the values when the corresponding methods obtain the best performance. The details are described in Table 4.

Therefore, we select two 100-dimensional vectors to represent lncRNA and protein, respectively. Three 5-fold Cross Validations (CVs) are carried out to evaluate the performance of LPI-deepGBDT.

5-fold CV on lncRNAs (CV1): 80% of lncRNAs are extracted as train set and the remaining is test set in each round.

5-fold CV on proteins (CV2): 80% of proteins are extracted as train set and the remaining is test set in each round.

5-fold CV on lncRNA–protein pairs (CV3): 80% of lncRNA–protein pairs are extracted as train set and the remaining is test set in each round.

The three CVs refer to potential LPI identification for (1) a new (unknown) lncRNA without interaction information, (2) a new protein without interaction information, and (3) lncRNA–protein pairs, respectively.

**Comparison with five state-of-the-art LPI prediction methods**

We compare the proposed LPI-deepGBDT framework with five classical LPI identification models to measure the classification performance and robustness of LPI-deepGBDT, that is, LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF and LPI-HNM. The number
of negative samples is set as the same as positive samples. The best performance is illustrated in boldface in each row in Tables 5, 6 and 7.

Table 5 gives the comparative results of the five LPI identification models in terms of the six measurements under CV1. It can be observed that LPI-deepGBDT achieves better average recall, accuracy, F1-score, AUC and AUPR than LPI-BLS, LPI-CatBoost, PLIPCOM, and LPI-HNM on five LPI datasets. For example, LPI-deepGBDT obtains the best average F1-score of 0.7586, 8.99%, 9.83%, 1.61%, 22.70% and 8.37% superior than LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF, and LPI-HNM, respectively. More importantly, it calculates the best AUC value of 0.8321, 1.63%, 8.32%, 2.37%, 0.02% and 6.26% better than the above five models, respectively. It also achieves the best average AUPR of 0.8095, 1.85%, 5.53%, 0.77%, 0.02% and 0.24% higher than the five methods, respectively.

LPI-BLS, LPI-CatBoost, PLIPCOM and LPI-HNM are four state-of-the-art supervised learning-based LPI prediction methods and LPI-deepGBDT computes better performance than them. The results suggest the powerful classification ability of LPI-deepGBDT under CV1. More importantly, although LPI-deepGBDT computes slightly lower precision than LPI-SKF, other five measurements are better than LPI-SKF. LPI-SKF is one network-based LPI inference algorithm. The type of methods have one limitation, that is, they can not be applied to predict possible interaction information for an orphan lncRNA. Therefore, LPI-deepGBDT is appropriate to prioritize underlying proteins associated with a new lncRNA.

Table 6 depicts the performance of LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF, LPI-HNM, and LPI-deepGBDT under CV2. The results show that the performance of LPI-deepGBDT is slightly lower than LPI-HNM. Under CV2, 80% proteins are extracted as training set and the remaining is test set in each round. That is, there will be relatively higher proteins for which association information is masked, thereby resulting in the reduction of samples and affecting the performance of LPI-deepGBDT. Compared to other five methods, LPI-HNM may be relatively robust to data abundant level when predicting possible lncRNAs for a new protein.

More importantly, LPI-deepGBDT computes the best average AUC and AUPR in comparing to LPI-BLS, LPI-CatBoost, and PLIPCOM. For example, LPI-deepGBDT obtains the best average AUC of 0.6815, 21.97%, 9.24%, and 4.01% superior than LPI-BLS, LPI-CatBoost, and PLIPCOM, respectively. LPI-deepGBDT achieves the best average AUPR of 0.6771, 15.74%, 10.37%, and 6.78% better than the above three methods, respectively.

| Method           | Parameter setting                                      |
|------------------|--------------------------------------------------------|
| LPI-BLS          | \( s = 1, c = 10^{-10}, N1 = 3, N2 = 60, N3 = 900 \) |
| LPI-CatBoost     | learning_rate = 0.5, loss_function = 'Logloss', logging_level = 'Verbose' |
| PLIPCOM          | learning_rate = 0.01, n_estimators = 100 |
| min_samples_split = 2, max_depth = 3 |
| LPI-deepGBDT     | target_lr = 1.0, epsilon = 0.3, n_rounds = 3, d = 100 |
| max_depth = 5, num_boost_round = 5, n_epochs = 15 |

Table 4 Parameter settings
AUC and AUPR are two more important evaluation criteria compared to other four measurements. LPI-deepGBDT outperforms LPI-BLS, LPI-CatBoost, and PLIPCOM in terms of AUC and AUPR. The results suggest that LPI-deepGBDT is one appropriate LPI prediction algorithm.

In particular, LPI-BLS is an ensemble learning-based model. LPI-deepGBDT significantly outperforms LPI-BLS based on AUC and AUPR. The results illustrate that LPI-deepGBDT may obtain better ensemble performance. In addition, LPI-CatBoost and PLIPCOM are two categorical boosting techniques. LPI-deepGBDT, integrating the idea of deep architecture, obtains better performance than the two methods. It shows that deep learning may more effectively learn the relevances between lncRNAs

| Metric | Dataset 1 | Dataset 2 | Dataset 3 | Dataset 4 | Dataset 5 | Ave. |
|--------|-----------|-----------|-----------|-----------|-----------|------|
| Precision | 0.8458 ± 0.0014 | 0.8377 ± 0.0032 | 0.8428 ± 0.0060 | 0.8757 ± 0.0086 | 0.7006 ± 0.0171 | 0.8457 ± 0.0046 |
| Recall | 0.6550 ± 0.0009 | 0.8331 ± 0.0140 | 0.9632 ± 0.0028 | 0.9592 ± 0.0156 | 0.7134 ± 0.0152 | 0.9456 ± 0.0070 |
| Accuracy | 0.7512 ± 0.0005 | 0.8310 ± 0.0071 | 0.8917 ± 0.0039 | 0.7254 ± 0.0032 | 0.6751 ± 0.0112 | 0.8964 ± 0.0032 |
| F1-score | 0.7381 ± 0.0012 | 0.8314 ± 0.0067 | 0.8899 ± 0.0033 | 0.6298 ± 0.0070 | 0.7059 ± 0.0148 | 0.8897 ± 0.0033 |
| AUC | 0.7353 ± 0.0020 | 0.6827 ± 0.0092 | 0.9628 ± 0.0043 | 0.7921 ± 0.0120 | 0.6889 ± 0.0043 | 0.9592 ± 0.0024 |
| AUPR | 0.7120 ± 0.0020 | 0.8468 ± 0.0041 | 0.7377 ± 0.0034 | 0.6394 ± 0.0052 | 0.6226 ± 0.0058 | 0.6693 ± 0.0113 |
Table 6: The performance of five LPI prediction methods on CV2

| Metric     | Dataset 1 | LPI-BlS       | LPI-CatBoost   | PLIPCOM       | LPI-SKF       | LPI-HNM       | LPI-deepGBDT  |
|------------|-----------|---------------|---------------|---------------|---------------|---------------|---------------|
| Precision  |           | 0.5370 ± 0.0347 | 0.3405 ± 0.1562 | 0.5341 ± 0.1209 | 0.7009 ± 0.1208 | 0.6836 ± 0.1148 | 0.4413 ± 0.1452 |
| Recall     |           | 0.5264 ± 0.0130 | 0.2567 ± 0.1423 | 0.2165 ± 0.0725 | 0.5415 ± 0.0702 | 0.7060 ± 0.0876 | 0.2298 ± 0.1220 |
| F1-score   |           | 0.5285 ± 0.0228 | 0.2567 ± 0.1423 | 0.2494 ± 0.0853 | 0.5399 ± 0.0745 | 0.6818 ± 0.0428 | 0.2697 ± 0.1242 |
| AUC        | Dataset 1 | 0.5701 ± 0.0255 | 0.5659 ± 0.0734 | 0.5397 ± 0.0855 | 0.6293 ± 0.1142 | 0.8013 ± 0.0902 | 0.5419 ± 0.0863 |
|            | Dataset 2 | 0.6227 ± 0.0328 | 0.5173 ± 0.0987 | 0.5895 ± 0.0743 | 0.6233 ± 0.0899 | 0.7578 ± 0.1278 | 0.6347 ± 0.0798 |
|            | Dataset 3 | 0.4443 ± 0.0269 | 0.5373 ± 0.0421 | 0.5084 ± 0.0512 | 0.5848 ± 0.1577 | 0.7595 ± 0.0402 | 0.5625 ± 0.0508 |
|            | Dataset 4 | 0.5206 ± 0.0088 | 0.6004 ± 0.1148 | 0.7791 ± 0.0124 | 0.7020 ± 0.0571 | 0.7134 ± 0.0528 | 0.7883 ± 0.0115 |
|            | Dataset 5 | 0.5013 ± 0.0025 | 0.8717 ± 0.1138 | 0.8544 ± 0.0063 | 0.8000 ± 0.1136 | 0.8959 ± 0.0212 | 0.8802 ± 0.0172 |
| Ave.       |           | 0.5318           | 0.6185           | 0.6542           | 0.6516           | 0.7856           | 0.6815           |

and proteins. Although LPI-SKF computes better AUPR than LPI-deepGBDT, LPI-SKF is a network-based model. Network-based methods can not reveal association information for an orphan protein. In summary, LPI-deepGBDT may be applied to infer possible interacting IncRNAs for a new protein.

The experimental results under CV3 are shown in Table 7. The comparative results demonstrate that LPI-deepGBDT computed the best average precision, recall, accuracy, F1-score, AUC, and AUPR over all datasets. For example, LPI-deepGBDT obtains the best average F1-score value of 0.8429, 14.83%, 10.77%, 3.10%, 16.73% and 18.43% superior than LPI-BlS, LPI-CatBoost, PLIPCOM, LPI-SKF and LPI-HNM, respectively. More importantly, it calculates the best AUC value of 0.9073, 4.93%, 11.21%, 3.32%, 0.12% and 14.49%, better than the above five models, respectively. It
also achieves the best average AUPR of 0.8849, 5.82%, 8.84%, 2.59%, 2.62% and 9.13% higher than the five methods, respectively. The results characterize the superior classification performance of LPI-deepGBDT. Therefore, LPI-deepGBDT can precisely discover the potential relationships between lncRNAs and proteins based on known association information.

In addition, we investigate the performance computed by all six LPI prediction methods under the three different cross validations. The results from Tables 5, 6 and 7 show that LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF, and LPI-deepGBDT achieve much better performance under CV3 than CV1, followed by CV2, regardless of precision, recall, accuracy, F1-score, AUC or AUPR. Under CV3, cross validations are conducted on all lncRNA–protein pairs and 80% lncRNA–protein pairs are used to train the model.
and the remaining 20% lncRNA–protein pairs are applied to test the model. However, under CV1 or CV2, cross validations are implemented on lncRNAs or proteins, that is, 80% lncRNAs or proteins are applied to train the model and the remaining 20% lncRNAs or proteins are used to test the model. CV3 may provide more LPI information relative to CV1 and CV2. The result suggest that abundant data contribute to improve the prediction performance of LPI identification models.

Case study
In this section, we aim to mine possible association data for a new lncRNA/protein or based on known LPIs.

Identifying potential proteins for a new lncRNA
RN7SL1 is an endogenous RNA. The lncRNA is usually protected by RNA-binding protein SRP9/14. Its increase can alter the stoichiometry with SRP9/14 and thus produce unshielded RN7SL1 in stromal exosomes. After exosome transfer to breast cancer cells, unshielded RN7SL1 can activate breast cancer RIG-I and promote tumor growth, metastasis, and therapy resistance [44]. Hepatocellular carcinoma patients with higher RN7SL1 concentrations also show lower survival rates. RN7SL1 may enhance hepatocellular carcinoma cell proliferation and clonogenic growth [45].

In this section, we mask all interaction information for RN7SL1 and want to infer possible proteins interacting with the lncRNA. The experiments are repeated for 10 times and the interaction probabilities between RN7SL1 and other proteins are averaged over the 10 time results. The predicted top 5 proteins interacting with RN7SL1 on human LPI datasets are described in Table 8. In Dataset 1, we can observe that RN7SL1 is predicted to interact with Q15465. Q15465 displays a cholesterol transferase and autoproteolysis activity in the reticulum endoplasmic. Its N-product is a morphogen required for diverse patterning events during development. It induces ventral cell fate in somites and the neural tubes. It is required for axon guidance and densely related to the anterior-posterior axis patterning in the developing limb bud [35]. In the dataset, RN7SL1 may associate with 59 proteins. In other two datasets, there does not exist any associated lncRNAs for Q15465. Although the interaction between RN7SL1 and Q15465 hasn’t been validated, among all possible associated 59 proteins, the protein is ranked as 4, 6, 8, 9, and 14 by LPI-CatBoost, PLIPCOM, LPI-SKF, LPI-HNM, and LPI-BLS, respectively. Therefore, the association between RN7SL1 and Q15465 need further validation.

In Dataset 2, we predict that Q13148, P07910, and Q9NZI8 may interact with RN7SL1. The interaction between Q9NZI8 and RN7SL1 is known in Dataset 3. Q13148 is a RNA-binding protein involved in various procedures in RNA biogenesis and processing. The protein controls the splicing in numerous non-coding and protein-coding RNAs, for example, proteins involved in neuronal survival and mRNAs encoding proteins related to neurodegenerative diseases. It plays important roles in maintaining mitochondrial homeostasis, mRNA stability and circadian clock periodicity, the normal skeletal muscle formation and regeneration. In Dataset 2, RN7SL1 may associate with 84 proteins. Among the 84 underlying proteins for RN7SL1, the rankings of Q13148 predicted by LPI-deepGBDT LPI-CatBoost, PLIPCOM, LPI-SKF, LPI-BLS, and LPI-HNM are 2, 3, 1, 3, 2, and 6, respectively. That is, all the six
LPI identification models predict that there may be interaction between Q13148 and RN7SL1. Therefore, we infer that Q13148 may possibly interact with RN7SL1.

More importantly, in Dataset 2, P07910 binds to pre-mRNA and regulates the stability and translation level of bound mRNA molecules. The protein is involved in the early procedures of spliceosome assembly and pre-mRNA splicing. In other two human LPI datasets, there are no any known associated lncRNAs for P07910. Among 84 potential associated proteins for RN7SL1, P07910 is ranked as 3, 7, 8, 9, 11, and 9 by LPI-deepGBDT, LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF, and LPI-HNM, respectively. The ranking are relatively higher. Therefore, we predict that P07910 may associate with RN7SL1.

In Dataset 3, we observe that Q9UKV8 and Q9Y6M1 may interact with RN7SL1. The interactions between RN7SL1 and the two proteins can be retrieved in Dataset 1. That is, the predicted top 5 interaction data by LPI-deepGBDT can be validated by publications. In summary, the results from case analyses based on association prediction for a new lncRNA suggest that LPI-deepGBDT can be utilized to identify new proteins associated with a new lncRNA.

### Finding potential lncRNAs interacting with a new protein

Q9UL18 is a protein required by RNA-mediated gene silencing. The protein can repress the translation of mRNAs complementary to them by binding to short RNAs or short interfering RNAs. It lacks endonuclease activity and thus can cleave target mRNAs. It is still required by transcriptional gene silencing of promoter regions complementary to bound short antigen RNAs [35]. In this section, we mask the interaction information for Q9UL18 and intend to find associated lncRNAs for the protein. The predicted top 5 lncRNAs on three human LPI dataset are shown in Table 9.

In Datasets 1-3, Q9UL18 may interact with 935, 885, and 990 lncRNAs. It can be seen that all the predicted top 5 interactions on each dataset are validated as known LPIs.

#### Table 8 The predicted top 5 proteins interacting with RN7SL1

| Dataset | Proteins | Confirmed | LPI-deepGBDT | LPI-BLS | LPI-CatBoost | PLIPCOM | LPI-SKF | LPI-HNM |
|---------|----------|-----------|---------------|----------|--------------|----------|---------|---------|
| Dataset 1 | O00425 | YES | 1 | 2 | 2 | 4 | 7 | 3 |
| Q9Y6M1 | YES | 2 | 8 | 3 | 8 | 6 | 2 |
| Q15465 | NO | 3 | 14 | 4 | 6 | 8 | 9 |
| Q15717 | YES | 4 | 1 | 1 | 2 | 21 | 4 |
| Q9UKV8 | YES | 5 | 4 | 7 | 14 | 1 | 8 |
| Dataset 2 | Q8IUX4 | YES | 1 | 6 | 2 | 2 | 8 | 5 |
| Q13148 | NO | 2 | 2 | 3 | 1 | 3 | 6 |
| P07910 | NO | 3 | 7 | 8 | 9 | 11 | 9 |
| Q9NZI8 | NO | 4 | 5 | 6 | 3 | 5 | 7 |
| Q9HCE1 | YES | 5 | 9 | 4 | 4 | 10 | 4 |
| Dataset 3 | Q9UKV8 | NO | 1 | 7 | 5 | 9 | 10 | 7 |
| Q9NU5 | YES | 2 | 1 | 1 | 1 | 1 | 5 |
| Q9Y6M1 | NO | 3 | 4 | 4 | 5 | 6 | 6 |
| O00425 | YES | 4 | 3 | 3 | 2 | 3 | 1 |
| Q9NZI8 | YES | 5 | 6 | 2 | 3 | 2 | 2 |
results suggest that LPI-deepGBDT can be applied to prioritize possible lncRNAs for a new protein.

Finding new LPIs based on known LPIs

We further infer new LPIs based on LPI-deepGBDT. We rank all lncRNA–protein pairs based on the computed average interaction probabilities. Figures 2, 3, 4, 5 and 6 give the predicted 50 LPIs with the highest interaction scores. In the five figures, black dotted lines and solid lines represent unknown and known LPIs obtained from LPI-deepGBDT, respectively. Gold ovals denote proteins, deep sky blue rounded rectangles denote RNA.

There are 55,165, 74,340, 26,730, 3,815, and 71,568 known and unknown lncRNA–protein pairs on given five datasets, respectively. We observe that unknown lncRNA–protein pairs between NONHSAT023366 (RAB30-AS1) and O00425, n378107 (NONHSAT007673, GAS5) and Q15717, NONHSAT143568 (LINC-01572) and P35637, AthlncRNA376 (TCONS_00057930) and O22823, and ZhangRNA530 (TCONS_00007931) and COPLI2, which are predicted to have the highest association scores on the five datasets, are ranked as 1, 3, 1, 6, and 113, respectively.

lncRNA GAS5 has close linkages with multiple complex diseases. The lncRNA is a repressor of the glucocorticoid receptors associated with growth arrest and starvation [46]. It is downregulated in breast cancer [47]. It can also promote microglial inflammatory response in Parkinson's disease [48], control apoptosis in non-small-cell lung cancer [49] and prostate cancer cell [50]. Its decreased expression indicates a poor prognosis in cervical cancer [51] and gastric cancer [52].

Q15717 increases the stability of mRNA and mediates the CDKN2A anti-proliferative activity and regulates p53/TP53 expression. It increases the leptin mRNA's stability and is involved in embryonic stem cells differentiation. In dataset 2, GAS5 have been validated to interact with P35637, and Q13148. P35637 plays an important role in diverse cellular processes including transcription regulation, DNA repair and damage response, RNA transport, and RNA splicing. It helps RNA transport, mRNA

| Table 9 The predicted top 5 lncRNAs interacting with Q9UL18 |
|---------------------------------|----------------|----------------|----------------|----------------|----------------|
| Dataset | IncRNAs | Confirmed | LPI-deepGBDT | LPI-BLS | LPI-CatBoost | PLIPCOM | LPL-SKF | LPI-HNM |
|---------|---------|-----------|--------------|---------|--------------|---------|---------|---------|
| Dataset 1 | RP0001_1006774 | YES | 1 | 439 | 614 | 566 | 29 | 593 |
|         | RP11-401 | YES | 2 | 169 | 177 | 204 | 558 | 10 |
|         | LUCAT1 | YES | 3 | 110 | 315 | 48 | 930 | 48 |
|         | RP0001_685651 | YES | 4 | 696 | 310 | 94 | 925 | 14 |
|         | RP0001_25361 | YES | 5 | 411 | 819 | 83 | 234 | 94 |
| Dataset 2 | RP5-1085F17 | YES | 1 | 396 | 116 | 11 | 104 | 116 |
|         | RP0001_79181 | YES | 2 | 521 | 276 | 302 | 78 | 15 |
|         | RP0001_114047 | YES | 3 | 687 | 567 | 315 | 45 | 63 |
|         | RP0001_81047 | YES | 4 | 789 | 330 | 125 | 88 | 17 |
|         | RP0001_139850 | YES | 5 | 204 | 360 | 167 | 8 | 65 |
| Dataset 3 | RP0001_1036776 | YES | 1 | 5 | 469 | 3 | 810 | 107 |
|         | RP11-357C3 | YES | 2 | 141 | 344 | 16 | 933 | 133 |
|         | RP0001_878565 | YES | 3 | 148 | 561 | 50 | 221 | 74 |
|         | HCG17 | YES | 4 | 22 | 118 | 4 | 707 | 129 |
|         | AL139819 | YES | 5 | 251 | 533 | 34 | 131 | 111 |
Fig. 2 The predicted top 50 LPIs on Dataset 1

Fig. 3 The predicted top 50 LPIs on Dataset 2
Fig. 4 The predicted top 50 LPIs on Dataset 3

Fig. 5 The predicted top 50 LPIs on Dataset 4
stability and synaptic homeostasis in neuronal cells. Q13148 plays a crucial role in maintaining mitochondrial homeostasis. It participates in the formation and regeneration of normal skeletal muscle, negatively regulates the expression of CDK6. The three proteins are RNA-binding proteins and have in part similar biological functions. Therefore, we infer that Q15717 may be the corresponding protein of GAS5.

Discussion and further research

lncRNAs regulate many important biological processes. They have close relationships with multiple human complex diseases. However, most of them are not annotated because of the poor evolutionary conservation. Recent researches suggest that lncRNAs implement their functions by binding to the corresponding proteins. Therefore, it is a significant work to infer potential interactions between lncRNAs and proteins. Various computational methods were designed to identify new LPIs. These models improved LPI prediction and found many potential linkages between the two entities.
The predicted LPIs with higher rankings are worthy of further biomedical experimental validation.

In this manuscript, we explore an LPI identification framework (LPI-deepGBDT) based on a feed-forward deep architecture with GBDTs. First, three LPI datasets and two plant datasets are retrieved. Second, the biological features of IncRNAs and proteins are selected via Pyfeat and BioProt, respectively. Third, the features are reduced based on dimensional reduction technique and concatenated to depict an IncRNA–protein pair. Finally, a multi-layered deep framework is developed to find the potential relationships between the two entities. We compare LPI-deepGBDT with five classical LPI discovery methods, LPI-BLS, LPI-CatBoost, PLIPCOM, LPI-SKF and LPI-HNM, on the five datasets under three cross validations. The results demonstrate the superior classification ability of LPI-deepGBDT. Case studies are further implemented to conduct interaction prediction for new IncRNAs (or proteins) or based on known LPIs.

LPI-deepGBDT computes the best performance on the collected five LPI datasets. It may be in large part due to the following features. First, LPI-deepGBDT fuses multiple biological features. Second, the constructed multi-layered deep framework with non-differentiable components helps to distributedly represent the outputs in intermediate layers. Thirdly, the update procedure for each intermediate layer can reduce the global loss by updating its pseudo-label and reducing the loss in the previous layer. Finally, the random noises added in the loss function can better map the neighbor training samples to right manifold.

In the future, we will collect multiple LPI datasets from different species to better mine the relevances between IncRNAs and proteins for different species. More importantly, we will develop more effective ensemble learning model to improve the performance of LPI prediction.

Abbreviations
LPI-deepGBDT: Feed-forward deep architecture based on gradient boosting decision trees used to discover unobserved LPIs; LPI: Long noncoding RNA–protein interaction; GBDT: Gradient boosting decision trees; IncRNAs: Long noncoding RNAs; CVs: Cross validations.

Acknowledgements
We would like to thank all authors of the cited references.

Authors’ contributions
Conceptualization: L-HP, ZW and L-QZ; Funding acquisition: L-HP and L-QZ; Investigation: L-HP and ZW; Methodology: L-HP and ZW; Project administration: L-HP and L-QZ; Software: ZW; Validation: ZW, X-FT; Writing – original draft: L-HP; Writing – review and editing: L-HP and ZW. All authors read and approved the final manuscript.

Funding
This research was funded by the National Natural Science Foundation of China (Grant 61803151, 62072172).

Availability of data and materials
Source codes and datasets are freely available for download at https://github.com/plhhnu/LPI-deepGBDT.

Declarations
Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.
References

1. Deng L, Wang J, Xiao Y, Wang Z, Liu H. Accurate prediction of protein-IncRNA interactions by diffusion and hetesim features across heterogeneous network. BMC Bioinform. 2018;19(1):1–11.

2. Liu Z-P Predicting IncRNA-protein interactions by machine learning methods: a review. Curr Bioinform. 2020;15(8):831–40.

3. Chen X, Sun Y-Z, Guan N-N, Qu J, Huang Z-A, Zhu Z-X, Li J-Q. Computational models for IncRNA function prediction and functional similarity calculation. Brief Funct Genom. 2019;18(1):58–82.

4. Chen X, Yan CC, Zhang X, You Z-H. Long non-coding rnas and complex diseases: from experimental results to computational models. Brief Bioinform. 2017;18(4):558–76.

5. Wang W, Dai Q, Li F, Xiong Y, Wei D-Q. Mlcpp: multi-label classification with deep forest in disease prediction for long non-coding rnas. Brief Bioinform. (2020)

6. Zhang X, Zhou Y, Mehta KR, Danila DC, Scolavino S, Johnson SR, Klubanski A. A pituitary-derived meg3 isoform functions as a growth suppressor in tumor cells. J Clin Endocrinol Metab. 2003;88(11):519–26.

7. Pibouin L, Villaudy J, Feraud D, Mulers M, Prosperi M-T, Remvikos Y, Goubin G. Cloning of the mna of overexpressed in colon carcinoma-1: a sequence overexpressed in a subset of colon carcinomas. Cancer Genet Cytofgenet. 2002;133(1):55–60.

8. Cui, Z., Ren, S., Lu, J., Wang, F., Xu, W., Sun, Y., Wei, M., Chen, J., Gao, X., Xu, C., et al.: The prostate cancer-up-regulated long noncoding rna plncrna-1 modulates apoptosis and proliferation through reciprocal regulation of androgen receptor. In: Urologic Oncology: Seminars and Original Investigations, vol. 31. pp. 1117–1123. Elsevier (2013)

9. Chen X, Yan G-Y. Novel human IncRNA-disease association inference based on IncRNA expression profiles. Bioinformatics. 2013;29(20):2617–24.

10. van Poppel H, Haese A, Graefen M, de la Taille A, Irani J, de Reijke T, Remzi M, Marberger M. The relationship between prostate cancer gene 3 (pc antigen) and prostate cancer significance. BJU Int. 2012;109(3):360–6.

11. Yang Z, Zhou L, Wu L-M, Lai M-C, Xie H-Y, Zhang F, Zheng S-S. Overexpression of long non-coding rna hotair predicts tumor recurrence in hepatocellular carcinoma patients following liver transplantation. Ann Surg Oncol. 2011;18(12):4343–50.

12. Wang, W., Guan, A., Khan, M.T., Xiong, Y., Wei, D-Q.: Lmi-dforest: a deep forest model towards the prediction of IncRNA-mirna interactions. Comput. Biol. Chem. 107406 (2020)

13. Li Y, Sun H, Feng S, Zhang Q, Han S, Du W. Capsule-lpi: a IncRNA-protein interaction predicting tool based on a capsule network. BMC Bioinform. 2021;22(1):1–19.

14. Li, A., Ge, M., Zhang, Y., Peng, C., Wang, M. Predicting long noncoding ma and protein interactions using heterogeneous network model. Biomed. Res. Int. 2015 (2015)

15. Zhou Y-K, Shen Z-A, Yu H, Luo T, Gao Y, Du P-F. Predicting IncRNA-protein interactions with mirnas as mediators in a heterogeneous network model. Front Genet. 2020;11:1341.

16. Yang L, Li A, Ge M, Wang M. Relevance search for predicting IncRNA-protein interactions based on heterogeneous network. Neurocomputing. 2016;206(19):81–8.

17. Zhao Q, Yu H, Ming Z, Hu H, Ren G, Liu H. The bipartite network projection-recommeded algorithm for predicting long non-coding ma-protein interactions. Mol Therapy-Nucleic Acids. 2018;13:464–71.

18. Ge M, Li A, Wang M. A bipartite network-based method for prediction of long non-coding ma-protein interactions. Genom Proteom Bioinform. 2016;14(1):62–71.

19. Xie G, Wu W, Sun Y, Fan Z, Liu J. Lpi-ibnra: long non-coding ma-protein interaction prediction based on improved bipartite network recommender algorithm. Front Genet. 2019;10:343.

20. Zhang W, Qu Q, Zhang Y, Wang W. The linear neighborhood propagation method for predicting long non-coding ma-protein interactions. Neurocomputing. 2018;273:526–34.

21. Zhou Y-K, Hu J, Shen Z-A, Zhang W-Y, Du P-F. Lpi-skl: predicting IncRNA-protein interactions using similarity kernel fusions. Front Genet. 2020;11:1554.

22. Chen Y, Fu X, Li Z, Peng L, Zhuo L. Prediction of IncRNA-protein interactions via the multiple information integration. Front Bioeng Biotechnol. 2021;9:60.

23. Peng L, Liu F, Yang J, Liu X, Meng Y, Deng X, Peng C, Tian G, Zhou L. Probing IncRNA-protein interactions: data repositories, models, and algorithms. Front Genet. 2020;10:1346.

24. Liu H, Ren G, Hu H, Zhang L, Ai H, Zhang W, Zhao M. Lpi-nrmf: IncRNA-protein interaction prediction by neighborhood regularized logistic matrix factorization. Oncotarget. 2017;8(61):103975.

25. Zhao Q, Zhang Y, Hu H, Ren G, Zhang W, Liu H. Irwrnlpi: integrating random walk and neighborhood regularized logistic matrix factorization for IncRNA-protein interaction prediction. Front Genet. 2018;9:239.

26. Zhang T, Wang M, Xi J, Li A. Lpgnmf: predicting long non-coding rna and protein interaction using graph regularized nonnegative matrix factorization. IEEE/ACM Trans Comput Biol Bioinf. 2018;17(1):89–97.

27. Zhang W, Yue X, Tang G, Wu W, Huang F, Zhang X. Sfpel-lpi: sequence-based feature projection ensemble learning for predicting IncRNA-protein interactions. PLoS Comput Biol. 2018;14(12):1006616.

28. Fan X-N, Zhang S-W. Lpi-bl: predicting IncRNA-protein interactions with a broad learning system-based stacked ensemble classifier. Neurocomputing. 2019;37088–93.

29. Deng L, Yang W, Liu H. Predicting prediction of protein-ma binding affinity using gradient boosted regression trees. Front Genet. 2019;10:637.
30. Wekesa JS, Meng J, Luan Y. Multi-feature fusion for deep learning to predict plant IncRNA-protein interaction. Genomics. 2020;112(5):2928–36.
31. Shen, Z.-A., Luo, T., Zhou, Y.-K., Yu, H., Du, P.-F.: Npi-gnn: predicting ncRNA-protein interactions with deep graph neural networks. Brief. Bioinform. (2021)
32. Feng, J., Yang, Y., Zhou, Z.H.: Multi-layered gradient boosting decision trees (2018)
33. Xie C, Yuan J, Li H, Li M, Zhao G, Bu D, Zhu W, Wu W, Chen R, Zhao Y. Noncodev4: exploring the world of long non-coding rna genes. Nucleic Acids Res. 2014;42(D1):98–103.
34. Yuan J, Wu W, Xie C, Zhao G, Zhao Y, Chen R. Npinter v2. O: an updated database of ncRNA interactions. Nucleic Acids Res. 2014;42(D1):104–8.
35. Consortium, U.: Uniprot: a worldwide hub of protein knowledge. Nucleic Acids Res. 2019;47(D1):506–15.
36. Zheng X, Wang Y, Tian K, Zhou J, Guan J, Luo L, Zhou S. Fusing multiple protein–protein similarity networks to effectively predict IncRNA-protein interactions. BMC Bioinform. 2017;18(12):11–8
37. Bai Y, Dai X, Ye T, Zhang P, Yan X, Gong X, Liang S, Chen M. Pincrnda: a repository of plant Incrnas and Incrna-rbp protein interactions. Curr Bioinform. 2019;14(7):621–7.
38. Muhammad R, Ahmed S, Md Farid D, Shatabda S, Sharma A, Dehzangi A. PyFeat: a python-based effective feature generation tool for DNA, RNA and protein sequences. Bioinformatics. 2019;35(19):3831–3.
39. Márquez, B., Amaya, J.C.: Bioprot contenedor autónomo de residuos biológicos. Revista colombiana de tecnologías de avanzada 1(53) (2019)
40. Ding C, Wang D, Ma X, Li H. Predicting short-term subway ridership and prioritizing its influential factors using gradient boosting decision trees. Sustainability. 2016;8(11):1100.
41. Shi Z, Chu Y, Zhang Y, Wang Y, Wei D-Q. Prediction of blood–brain barrier permeability of compounds by fusing resampling strategies and extreme gradient boosting. IEEE Access. 2020;9:9557–66.
42. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. 1189–1232 (2001)
43. Jiao Y, Du P. Performance measures in evaluating machine learning based bioinformatics predictors for classifications. Quant Biol. 2016;4(4):320–30.
44. Nabet BY, Qiu Y, Shabason JE, Wu TJ, Yoon T, Kim BC, Benci JL, DeMichele AM, Tchou J, Marcoratiano J, et al. Exosome marker(s) shielding couples stromal activation to pattern recognition receptor signaling in cancer. Cell. 2017;170(2):352–66.
45. Tan C, Cao J, Chen L, Xi X, Wang S, Zhu Y, Yang L, Ma L, Wang D, Yin J, et al. Noncoding rnas serve as diagnosis and prognosis biomarkers for hepatocellular carcinoma. Clin Chem. 2019;65(7):1905–15.
46. Kino T, Hurt DE, Ichiho T, Nader N, Chrousos GP. Noncoding RNA gas5 is a growth arrest-and starvation-associated repressor of the glucocorticoid receptor. Sci Signal. 2010;3(107):6–8.
47. Mourtada-Maarabouni M, Pickard M, Hedge V, Farzanefar F, Williams G. Gas5, a non-protein-coding rna, controls apoptosis and is downregulated in breast cancer. Oncogene. 2009;28(2):195–208.
48. Xu W, Zhang L, Geng Y, Liu Y, Zhang N. Long noncoding rna gas5 regulates microglial inflammatory response in Parkinson's disease by regulating ndlrp3 pathway through sponging mir-223-3p. Int Immunopharmacol. 2020;85:106614.
49. Shi X, Sun M, Liu H, Yao Y, Kong N, Chen Z, Song Y. A critical role for the long non-coding rna gas5 in proliferation and apoptosis in non-small-cell lung cancer. Mol Carcinog. 2015;54(5):11–12.
50. Pickard M, Mourtada-Maarabouni M, Williams G. Long non-coding rna gas5 regulates apoptosis in prostate cancer cell lines. Biochimica et Biophysica Acta. 2013;1832(10):1613–23.
51. Cao S, Liu W, Li F, Zhao W, Qin C. Decreased expression of IncRNA gas5 predicts a poor prognosis in cervical cancer. Int J Clin Exp Pathol. 2014;7(9):6776.
52. Sun M, Jin F-Y, Xia R, Kong R, Li J-H, Xu T-P, Liu Y-W, Zhang F-B, Liu X-H, De W. Decreased expression of long noncoding rna gas5 indicates a poor prognosis and promotes cell proliferation in gastric cancer. BMC Cancer. 2014;14(1):1–12.

Publisher’s Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.