# A Graph Transformer-Based Method for Predicting LncRNA-Disease Associations Using Matrix Factorization and Automatic Meta-Path Generation

Dengju Yao[1], Yuehu Wu[1], and Xiaojuan Zhan[2]

[1] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China ydkvictory@hrbust.edu.cn

[2] College of Computer Science and Technology, Heilongjiang Institute of Technology, Harbin 150050, China

**Abstract.** LncRNAs are crucial regulators of gene expression that exert their influence on diverse cellular processes. Exploring the potential connections between lncRNAs and various pathological conditions holds significant promise for unraveling the intricate mechanisms that underlie disease onset and progression. Due to traditional biological experimentation for probing lncRNA-disease associations is often hampered by substantial financial constraints and prolonged timelines. Consequently, computational modeling and bioinformatics methodologies have emerged as efficient and cost-effective alternatives. This paper proposes a lncRNA-disease association prediction model that integrates graph transformers, matrix factorization, and automatic meta-path generation. First, a transformer encoder network encodes the nonlinear node features, capturing their deep semantic representations. Simultaneously, matrix factorization learns linear embedding vectors for diseases and lncRNAs, representing their latent features. Moreover, an automatic meta-path generation algorithm dynamically updates the graph structure and reconstructs node representations to capture their topological characteristics. Finally, these diverse node representations are merged and fed into a multilayer perceptron for comprehensive learning, yielding the final prediction scores. Under 5-fold cross-validation experiments, our approach outperforms other existing methods in several performance metrics on the dataset. Additionally, case studies further provide further evidence of the effectiveness of our approach.

**Keywords:** transformer · matrix factorization· meta-path· nonlinear features· linear embedding vectors · topological features.

## 1 Introduction

Long non-coding RNAs (lncRNAs) constitute a distinct class of transcribed RNA molecules that do not encode proteins, characterized by their substantial length, exceeding 200 nucleotides. Recent years have witnessed a burgeoning interest in the study of lncRNAs [1]. Delving into the intricate biological roles and molecular pathways of lncRNAs holds a profound influence on advancing disease diagnosis and therapeutics [2]. Due to the shortcomings of traditional biological experiments, such as long cycle times and high costs. Therefore, it is essential to develop more efficient calculation methods.

In order to reduce data dimensionality and capture the main features. Zeng et al. proposed a computational framework integrating matrix decomposition and deep learning. Singular value decomposition (SVD) extracted linear features of lncRNAs and diseases, while a fully connected layer captured their nonlinear features. The linear and nonlinear features were then combined to predict potential LDAs [3]. Xi et al. introduced LD-CMFC, a collaborative matrix factorization approach that maximizes covariance entropy to enhance robustness and prediction accuracy [4]. The technique of matrix decomposition described above converts raw data into a low-dimensional representation. However, it has a limited effect on complex nonlinear relationships.

Machine learning and deep learning learn abstract representations and correlations of data by training models and using deep neural networks, which can handle more complex non-linear relations. Zeng et al. proposed DMFLDA, a deep learning framework with nonlinear hidden layers, to capture complex LDAs [5]. While numerous approaches extract both linear and non-linear features from lncRNAs and diseases to predict LDAs, there is a lack of models that integrate both feature types.

In this paper, to address some of the aforementioned problems, we introduce GTMALDA, a novel prediction model for LDAs, leveraging graph transformer, matrix factorization, and automatic meta-path generation techniques. The GTMALDA first integrates various databases of lncRNA and disease to obtain their respective similarity matrices, including lncRNA Jaccard similarity, GIP kernel similarity, cosine similarity, and disease Jaccard similarity, GIP kernel similarity, cosine similarity, and semantic similarity. Then, The similarity matrices of lncRNA and disease are fused into a comprehensive similarity network using the non-linear fusion (SNF) method [6]. Based on this, the two comprehensive networks and the LDA matrix are combined to form a multi-source heterogeneous network. Thirdly, a graph transformer is trained to obtain non-linear, multi-level, and granular features for each node. Subsequently, a non-negative matrix decomposition of the LDA matrix is performed to obtain a linear feature representation of lncRNA and disease. Additionally, to uncover the biological links and potential mechanisms of action between diseases and lncRNAs, we learn topological features based on the network structure using an automatic meta-path generation algorithm. Finally, linear, non-linear, and topological features are combined to obtain the final LDA scores by MLP. The GTMALDA outperforms several leading algorithms under multiple datasets using 5-CV for several metrics. Additionally, the case study confirms the presence of associations with related diseases among the top 15 expected lncRNAs. The workflow of the GTMALDA model is shown in Figure 1.

## 2 Materials and methods

### 2.1 Baseline datasets

The dataset is obtained from the LncRNADisease2.0 public dataset [7], and after removing duplicate non-human LDAs, we obtain 1690 experimentally validated LDAs, including 447 lncRNAs and 218 diseases. We analyze the dataset and transform it into adjacency matrices $A$, where rows represent lncRNAs, and columns represent diseases. If the disease in column $j$ is associated with the lncRNA in row $i$, then $A_{ij} = 1$; otherwise $A_{ij} = 0$.

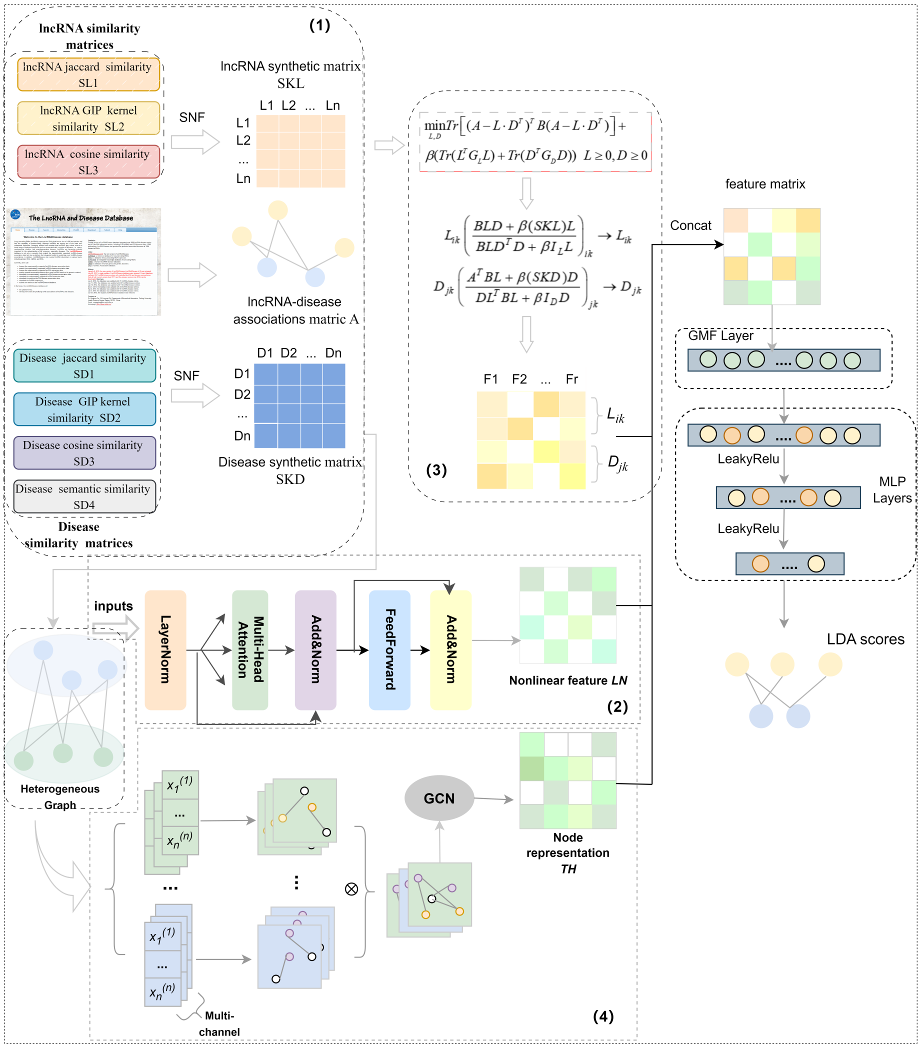
(1)
lncRNA similarity matrices
lncRNA jaccard similarity SL1
lncRNA GIP kernel similarity SL2
lncRNA cosine similarity SL3
SNF
lncRNA synthetic matrix SKL
L1 L2 ... Ln
L1
L2
...
Ln
The LncRNA and Disease Database
lncRNA-disease associations matric A
Disease jaccard similarity SD1
Disease GIP kernel similarity SD2
Disease cosine similarity SD3
Disease semantic similarity SD4
Disease similarity matrices
SNF
D1 D2 ... Dn
D1
D2
...
Dn
Disease synthetic matrix SKD
$\min_{L,D} Tr\left[(A-L\cdot D^T)^T B(A-L\cdot D^T)\right]+$
$\beta(Tr(L^T G_L L)+Tr(D^T G_D D))\ \ L\geq 0, D\geq 0$
$L_{ik}\left(\frac{BLD+\beta(SKL)L}{BLD^T D+\beta I_L L}\right)_{ik}\rightarrow L_{ik}$
$D_{jk}\left(\frac{A^T BL+\beta(SKD)D}{DL^T BL+\beta I_D D}\right)_{jk}\rightarrow D_{jk}$
F1 F2 ... Fr
$L_{ik}$
$D_{jk}$
(3)
feature matrix
Concat
GMF Layer
....
....
LeakyRelu
....
LeakyRelu
....
MLP Layers
LDA scores
inputs
LayerNorm
Multi-Head Attention
Add&Norm
FeedForward
Add&Norm
Nonlinear feature LN
(2)
Heterogeneous Graph
$x_1^{(1)}$
...
$x_n^{(n)}$
...
$x_1^{(1)}$
...
$x_n^{(n)}$
Multi-channel
⊗
GCN
Node representation TH
(4)

**Fig. 1.** The workflow of the GTMALDA.

## 2.2 Similarity networks

(1) Disease semantic similarity: To calculate the semantic similarity, we consider the contributions of the common nodes and edges of the two DAGs using the method of a previous study [8]. Finally, we obtain the semantic similarity matrix, which is represented as $SD_4 \in R^{218*218}$. (2) Jaccard similarity: The similarity between sets is obtained by measuring the size of the intersection of two sets relative to the size of their concatenation. By the previous study, we get Jaccard similarity $SL_1 \in R^{447*447}$ and $SD_1 \in R^{218*218}$ for lncRNA and disease. (3) Gaussian interaction profile (GIP) kernel similarity: ector interactions are modeled with a Gaussian function, and vector similar-

ity is computed from the interaction patterns. As per a former study [9], the GIP kernel similarity of lncRNA and disease is obtained as $SL_2 \in R^{447*447}$ and $SD_2 \in R^{218*218}$. (4) Cosine similarity: The cosine similarity, ranging from -1 (dissimilar vectors) to 1 (similar vectors), measures the angle between two vectors. According to a previous study, we get the cosine similarity of lncRNA and disease as $SL_3 \in R^{447*447}$ and $SD_3 \in R^{218*218}$.

### 2.3 Fusion of similarity feature matrices

Inspired by previous research [6], we use a similarity network fusion (SNF)-based method for integrating different types of similarity network data, to obtain comprehensive views of lncRNAs and diseases. The following is an example of feature matrix fusion for lncRNA.

To prevent the scale of certain features from adversely affecting the model, we normalize each lncRNA similarity matrix to ensure that the range of values for each feature is consistent. The normalization process is shown below:

$$n_a(l_i, l_j) = \frac{SL(l_i, l_j)}{\sum\limits_{l_p \in L} SL(l_p, l_j)} \tag{1}$$

Here, $L$ is a set that represents all lncRNAs. $n_a$ is a normalized matrix and satisfies $\sum\limits_{l_p \in L} n_a(l_p, l_j) = 1$.

After that, we introduce neighborhood constraints to the lncRNA similarity matrix by constructing a neighborhood constraint kernel to capture their neighborhood relationships in the similarity matrix. It is shown below:

$$S_b(l_i, l_j) = \begin{cases} \frac{SL(l_i,l_j)}{\sum\limits_{l_p \in N_i} SL(l_i,l_p)} & \text{if } l_j \in N_i \\ 0 & \text{if } l_j \notin N_i \end{cases} \tag{2}$$

where $S_b(l_i, l_j)$ is a neighborhood-constrained kernel satisfying $\sum\limits_{l_p \in N_i} SL(l_i, l_p) = 1$, $N_i$ denotes the set of $p$ lncRNAs that are most similar to lncRNA $l_i$.

Then, we fuse the previous normalization kernel and the neighborhood constraint kernel as follows:

$$p_d^{t+1} = \alpha(S_b \times \frac{\sum\limits_{r \neq a} n_r^t}{2} \times S_b^T) + (1 - \alpha)\frac{\sum\limits_{r \neq a} n_r^0}{2} \tag{3}$$

The initial value $n_a$ is denoted by $n_r^0$. The weight parameter is represented by $\alpha$ and $p_d^{t+1}$ is the $dth$ similarity kernel after $t$ iterations. The final similarity kernel after $t$ iterations can be expressed as follows:

$$SL = \frac{1}{3}(\sum_{d=1}^{3} P_d^{t+1}) \tag{4}$$

To further eliminate noise and improve the quality of the similarity kernel, we build a weight matrix as follows:

$$w(l_i, l_j) = \begin{cases} 0 & \text{if } l_i \notin N_j \cap l_j \notin N_i \\ 0.5 & otherwise \\ 1 & \text{if } l_i \in N_j \cap l_j \in N_i \end{cases} \tag{5}$$

The final similarity matrix for fused lncRNA is obtained as follows:

$$SKL = w(l_i, l_j) \times SL \tag{6}$$

Similarly, we obtain the fused disease similarity matrix $SKD \in R^{nd \times nd}$.

### 2.4 LncRNA-disease heterogeneous network

A heterogeneous network is built using similarity matrices for both lncRNA and diseases, comprised of the lncRNA fusion similarity matrix ($SKL$), the disease fusion similarity matrix ($SKD$), and the lncRNA-disease adjacency matrix ($A$), represented as follows:

$$HeN = \begin{bmatrix} SKL & A \\ A^T & SKD \end{bmatrix} \in R^{(nl+nd) \times (nl+nd)} \tag{7}$$

### 2.5 Generate non-linear features

The multi-head attention mechanism is used to model the dependency relationship between words at any distance, fully capturing the sentence's semantic information.The first component is the Multi-Head Attention, which consists of $S_a$ the Self-Attention layer. Specifically, the self-attention mechanism efficiently computes three vectors: $Q$, $K$, and $V$ through matrix operations. The similarity of $Q$ and $K$ is then computed through the inner product as the Attention score, which is used to weighted average the $V$ vectors as output.

The attention scores for each node are computed by feeding the node representations from lncRNA and disease heterogeneous networks into a Multi-Head Attention mechanism. The formula for this process is as follows:

$$R_n = soft\max\left(\frac{Q_n K_n^T}{\sqrt{d_K}}\right) V_n \tag{8}$$

$$R = concat(R_1 \cdot \cdot R_n \cdot \cdot R_{head}) \tag{9}$$

Here, the output of each attention head in the Multi-Head Attention mechanism is represented by $R_n$. $d_K$ represents the dimension of the vector $K$, and the $head$ represents the number of attention heads. Finally, $R_n$ is combined to form the final attention vector for each node.

To obtain higher-level semantic features, the attention output is continuously fed into a feed-forward fully connected network. This network further extracts and expresses the features of the attention output. The feedforward neural network is expressed as follows:

$$FFN(R) = Leaky\,\mathrm{Re}\,Lu(XW + b) \tag{10}$$

Finally, residual linking is used to sum up the features learned by the attention mechanism with the input features. Layer normalization is also used to improve numerical stability. The final output of the nonlinear feature LN fusing the attentional expression and the original expression is shown below:

$$X = FFN(R) + x \tag{11}$$

$$LN(x_i) = \alpha \times \frac{x_i - \mu_L}{\sqrt{\sigma_L^2 + \varepsilon}} + \beta \tag{12}$$

Where $x$ represents the node input feature, $\varepsilon$ is a regularisation term to avoid a denominator of zero. $\alpha, \beta$ is the learnable parameter, and $\mu$, $\sigma$ represents the mean and variance, respectively.

## 2.6 Generate linear features

Non-negative matrix factorization (NMF) decomposes a non-negative matrix into the product of two or more non-negative matrices. The LDA matrix $A$ is decomposed into two low-rank non-negative matrices $L \in R^{nl \times r}$ and $D \in R^{nd \times r}$ , where $r < \min(nl, nd)$ is the dimension of the two submatrices. The product of $L$ and $D$ is infinitely close to $A$, meaning $A \approx L \cdot D^T$.

Introducing a constraint paradigm in the loss function helps feature selection and improves the generalization of the model.The loss function for a non-negative matrix decomposition problem can be defined accordingly as follows:

$$\min_{L,D} ||A - L \cdot D^T||_{2,1} = \min_{L,D} Tr\left[(A - L \cdot D^T)^T B (A - L \cdot D^T)\right] \tag{13}$$

Here $B$ is a diagonal matrix with the values of the elements on the diagonal $B_{ii} = \frac{1}{||(A - L \cdot D^T)_i||_2}$, and $||(A - L \cdot D^T)_i||$ represents the $i - th$ row of $||(A - L \cdot D^T)||$.

To preserve the inherent structure of the data, we employ graph regularization [10]. Specifically, we introduce the previously constructed lncRNA similarity matrix $SKL$ and disease similarity matrix $SKD$ into the objective function as regularisation terms to maintain the structural information in the graph. as shown below:

$$\frac{1}{2} \sum_{i,j=1}^{L_m} ||L(l_i) - L(l_j)||^2 SKL(l_i, l_j) = Tr(L^T G_L L) \tag{14}$$

$$\frac{1}{2} \sum_{i,j=1}^{D_n} ||D(d_i) - D(d_j)||^2 SKD(d_i, d_j) = Tr(D^T G_D D) \tag{15}$$

where $\mathrm{G}_L$ denotes the Laplacian matrix in lncRNA feature space.The formula for $\mathrm{G}_L$ is $G_L = I_L - SKL$. where $I_L$ is the diagonal matrix whose diagonal elements are the row sums of the lncRNA feature matrix $SKL$. Similarly, calculating the $G_D$.By integrating Eq. 13, Eq. 14, and Eq. 15, we can rewrite the objective function as:

$$\begin{aligned} &\min_{L,D} Tr\left[(A - L \cdot D^T)^T B (A - L \cdot D^T)\right] + \\ &\beta(Tr(L^T G_L L) + Tr(D^T G_D D)) \; L \geq 0, D \geq 0 \end{aligned} \tag{16}$$

where $\beta$ is a variable parameter. After further optimization of the above equation, we can obtain the corresponding Lagrangian function $L_f$ as follows:

$$L_f = \mathrm{Tr}(L^T BL) - 2\mathrm{Tr}(L^T BL \cdot D^T) + \mathrm{Tr}(DL^T BLD^T) + \\ \beta(\mathrm{Tr}(L^T G_L L) + \mathrm{Tr}(D^T G_D D)) + \mathrm{Tr}(\varphi L^T) + \mathrm{Tr}(\phi D^T) \tag{17}$$

Here $\varphi$ and $\phi$ are Lagrange multipliers and the partial derivatives of $L$ and $D$ are computed via the Karush-Kuhn-Tucker (KKT) condition [11], and $L$ and $D$ are finally updated as follows:

$$L_{ik}\left(\frac{BLD + \beta(SKL)L}{BLD^T D + \beta I_L L}\right)_{ik} \rightarrow L_{ik} \tag{18}$$

$$D_{jk}\left(\frac{A^T BL + \beta(SKD)D}{DL^T BL + \beta I_D D}\right)_{jk} \rightarrow D_{jk} \tag{19}$$

By iteratively updating Eqs. 18 and 19, the final linear low-rank representations $L$ and $D$ are obtained when the final convergence criterion is reached.

## 2.7 Generating topological features

Graph Transformer Networks (GTNs) is a framework for learning novel graph structure. GTNs consist of two components. The GT layer first selects two candidate graphs, $P_1$ and $P_2$, from the $HeN$. Then, it learns the combined new relationships through matrix multiplication of $P_1$ and $P_2$. The candidate graph $P$ is selected as follows:

$$P = conv(HeN; soft\max(W_{conv})) \tag{20}$$

where $conv$ is the convolutional layer and $W_{conv}$ is the weight parameter in the convolutional layer.

In GTNs, each candidate graph $P_i$ can be represented as:

$$\sum\nolimits_{e_t \in E_t} W_{e_n}^{(n)} A_{e_n} \tag{21}$$

where $E_t$ is the set of edge types, $W_{e_n}^{(n)}$ is the weight of the $nth$ edge type $e_t$ in the $nth$ layer, and $A_{e_t}$ is the corresponding matrix in each layer. So the adjacency matrix of a source path of arbitrary length can be represented as:

$$A_n = (\sum\nolimits_{e_t \in E_t} W_{e_1}^{(1)} A_{e_1})(\sum\nolimits_{e_t \in E_t} W_{e_2}^{(2)} A_{e_2}) \cdots (\sum\nolimits_{e_t \in E_t} W_{e_n}^{(n)} A_{e_n}) \tag{22}$$

After stacking $n$ GT layers, the GCN is applied to each channel of the meta-path $A^{(n)} = R^{N \times N \times T}$, connecting the representations of multiple nodes.

Long meta paths contain more nodes, spanning multiple local subgraphs, representing complex relationships. Short meta paths contain fewer nodes, confined to local regions, representing direct relationships. For example, A->B->C denotes a two-element path, while A->B->C->D denotes a three-element path. Learning short and long meta paths allows the automatic discovery of topological features from the network structure. Moreover, To learn short and long meta-paths containing primitive edges, a unit matrix

$I$ ($A_0 = I$) is added to $A$. This enables the GTN to learn metapaths of any length. The meta-path representation of a node is then determined as follows:

$$TH = ||_{i=1}^{T}\sigma(\widetilde{D_i}\widetilde{A_i}^{(n)}XW) \tag{23}$$

where || is the join operator, $T$ is the number of channels, $\widetilde{A_i}^{(n)} = A_i^{(n)} + I$ denotes the adjacency matrix of the $nth$ channel, $\widetilde{D_i}$ is the degree matrix of $\widetilde{A_i}^{(n)}$, $W$ is the trainable weight matrix shared across channels, and $X$ is the feature matrix.

### 2.8 Predicting potential LDAs

To establish the LDAs and enhance prediction accuracy, we utilized the generalized matrix factorization (GMF) model [12]. The GMF model splices the learned vector representations of lncRNAs and diseases to obtain a high-dimensional joint representation vector that contains information about both. Next, the high-dimensional vector is input into a fully connected layer for nonlinear transformation. The layer utilizes element-wise multiplications, enabling the model to better capture intricate relationships between lncRNAs and diseases. Ultimately, the MLP network learns to model the intricate LDAs, producing the final association scores. The resulting high-dimensional association representation vector is as follows:

$$F = \left\{ LN_{N_l+N_d} \cdots \begin{pmatrix} L_{N_l} \\ D_{N_d} \end{pmatrix} \cdots TH_{N_l+N_d} \right\} \tag{24}$$

where $N_l$ and $N_d$ denote the number of lncRNAs and disease nodes, respectively. The element-by-element multiplication operation of GMF is expressed as follows:

$$EW(l_i, d_i) = \left\{ \begin{matrix} F_{l_i} \odot F_{d_j} \\ F_{l_i} \in F\,[1 : N_l] \\ F_{d_j} \in F\,[N_{l+1} : N_{l+d}] \end{matrix} \right\} \tag{25}$$

where $\odot$ denotes element-by-element multiplication, and $F_{l_i}$, $F_{d_j}$ denote the $i - th$ and $j - th$ rows in the matrix $F$ respectively. The final MLP network is defined as follows:

$$\widehat{Y}\left(l_i, d_j\right) = sigmoid(W_L \cdots (Leaky\,\mathrm{Re}\,LU(W_1 EW\left(l_i, d_j\right) + b_1) + \cdots)) \tag{26}$$

where $W$ is the weight matrix and $b$ is the bias vector. Our model is using the binary cross entropy loss function (BCELoss) as the loss function of the model, defined as follows:

$$Loss(A, \widehat{Y}) = -\frac{1}{N}\sum\nolimits_{i=1}^{N}\left(A_i \log(\widehat{Y_i}) + (1 - A_i)\log(1 - \widehat{Y_i})\right) \tag{27}$$

where $\widehat{Y}$ is the association prediction score for lncRNA and disease and $N$ is the number of lncRNAs.

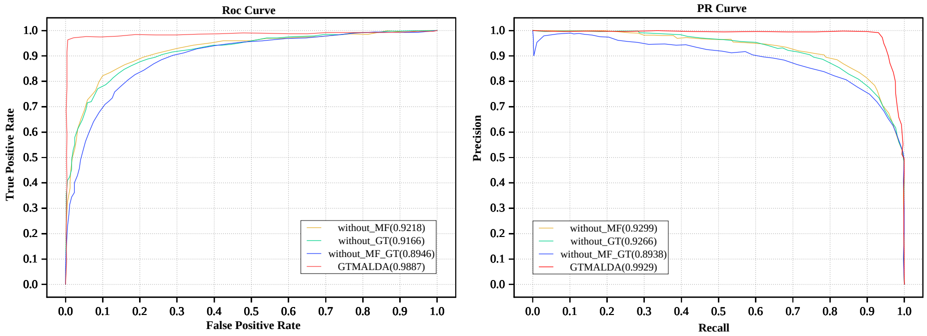

**Fig. 2.** Ablation experiment.

# 3 Experiments and results

## 3.1 Evaluation metrics

To evaluate the impact of the GTMALDA model, we utilize a 5-cv method. Our experiments randomly divide the LDAs matrix into 5 equal subsets, followed by conducting 5-cv experiments. During each experiment, we use 4 subsets as the training set and the 5th subset as the test set. After training the GTMALDA model on the training set, we remove all known LDAs from the test set, then use the trained model to predict potential associations in the test set and calculate the model performance metrics on the test set. The final evaluation metric of the GTMALDA model is the average of the results from the 5-cv experiments. When evaluating model performance, we focus on metrics such as AUC, AUPR, F1-score, recall, and precision. The experimental results indicate that our model performs better on the LDA prediction task.

## 3.2 Parameter Selection

The GTMALDA's performance depends on various parameters. During training, the learning rate is 0.001, and the number of epochs is 130. For feature matrix fusion, the weight is 0.1, the number of iterations is 20, and the number of neighbors is 40. For nonlinear features, the self-attention stacks are set to 8 from {5,6,7,8,9}. For linear features, the rank is 128 from {32,64,128,256}, and the convergence criterion is 0.01 from {1,0.1,0.01,0.001}. For topological features with GTNs, the GT layer stacks are 7 from {2,3,4,5,6,7,8,9}, and the convolution output channel is 2.

## 3.3 Ablation experiments

Ablation experiments, depicted in Figure 2, validate the effectiveness of the model's individual components. When NMFs are removed only, the AUC and AUPR of the model are 0.9218 and 0.9299, which are 6.67% and 6.3% lower respectively. Removing only GTNs results in a decrease in the model's AUC to 0.9166 and AUPR to 0.9266. The model's performance is the worst when it only has a transformer, with an AUC and

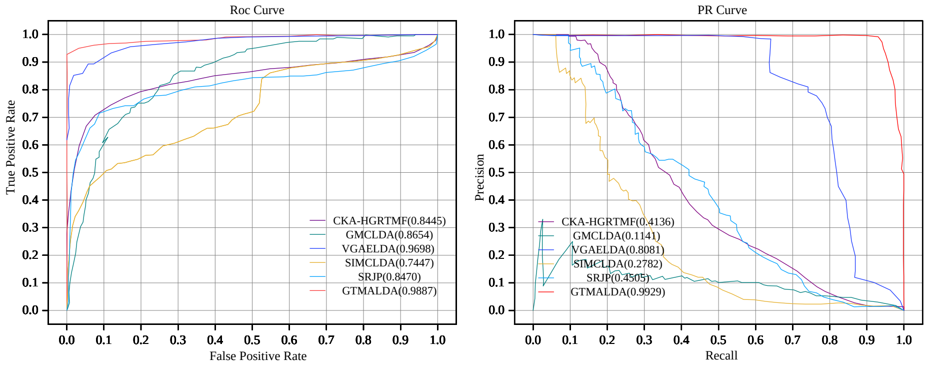

**Fig. 3.** Comparison of ROC curves and PR curves for all methods on Dataset

AUPR of 0.8946 and 0.8938, respectively. Adding NMF and GTNs modules improves the model's prediction performance. The results show that incorporating both NMF and GTNs improves the model's predictive performance.

**Table 1.** The performance comparison of different methods.

| Methods | AUC | AUPR | F1-score | Recall | Precision |
|---|---|---|---|---|---|
| CKA-HGRTMF | 0.8445 | 0.4136 | 0.0429 | 0.9035 | 0.0226 |
| GMCLDA | 0.8655 | 0.1141 | 0.7269 | 0.8879 | 0.7241 |
| SRJP | 0.8372 | 0.4325 | 0.7025 | 0.7001 | 0.8634 |
| SIMCLDA | 0.7448 | 0.2782 | 0.6675 | 0.6507 | 0.7265 |
| VGAELDA | 0.9698 | 0.8081 | 0.7557 | 0.9649 | 0.0758 |
| GTMALDA | **0.9887** | **0.9929** | **0.9853** | **0.9727** | **0.9981** |

### 3.4 Comparison with other methods

We evaluated GTMALDA's performance against five classical and state-of-the-art algorithms to validate its effectiveness. These five methods are CKA-HGRTMF [13], GMCLDA [14], SRJP [15], SIMCLDA [16] and VGAELDA [17].The parameter settings of all compared methods are taken from the original paper. Table 1 and figure 3 present the comparison results for each performance metric. Our proposed new method demonstrated excellent performance in all evaluation metrics. In terms of AUC values, our results are 1.8% higher than the sub-optimal method, VGAELDA, which indicates that our predictive model has higher classification accuracy. In addition, our method is also 18% ahead on AUPR, indicating that we can detect potential LDAs with higher accuracy.

**Table 2.** The top 15 HCC-related lncRNA candidates.

| Rank | LncRNA | PMID | Rank | LncRNA | PMID |
|---|---|---|---|---|---|
| 1 | MALAT1 | 26887056 | 9 | SNHG1 | 30266084 |
| 2 | H19 | 27027436 | 10 | DANCR | 26617879 |
| 3 | CDKN2B-AS1 | 27314206 | 11 | CRNDE | 22393467 |
| 4 | MEG3 | 25636452 | 12 | SNHG7 | 33685194 |
| 5 | TUG1 | 26856330 | 13 | KCNQ1OT1 | 16965397 |
| 6 | GAS5 | 24926850 | 14 | TP73-AS1 | 30010111 |
| 7 | PVT1 | 24196785 | 15 | SNHG6 | 30157475 |
| 8 | NEAT1 | 26314847 | | | |

### 3.5 Case studies

Numerous experiments have confirmed the close relationship between lncRNAs and various types of cancers. For instance, Liu et al. discovered that the lncRNA HOTTIP is upregulated in colorectal cancer tissues, and its knockdown suppresses cancer cell proliferation, migration, and invasion [18]. We select the Hepatocellular carcinoma (HCC) as a case study to evaluate the reliability of GTMALDA. HCC is a liver cancer frequently arising from underlying cirrhosis. After sorting the candidate RNAs associated with HCC by descending order, we find that in 15 candidate RNAs are all confirmed in the lncRNADisease database, see Table 2. This further proves the reliability of our model.

## 4 Conclusion

A growing body of research suggests that lncRNAs play a key role in the onset and progression of many types of various diseases. The paper introduces the GTMALDA model, which combines linear and non-linear features of lncRNA and disease using non-negative matrix decomposition and the encoder part of the transformer, respectively. The GTNs model is then used to obtain meta-path topological feature information of nodes from the network structure. Finally, an MLP is employed to calculate the final association score. The comparative experiments and case studies in the paper also confirm the effectiveness and robustness of GTMALDA.

Although the model has achieved some success, there are still some shortcomings. Firstly, the model contains several modules with a large number of parameters, and selecting the appropriate parameter combinations requires numerous experiments. Secondly, the coordination between modules needs to be strengthened due to the different parameter choices, making it difficult to ensure a perfect fit. Therefore, our next focus is to explore the efficient adjustment of parameters in anticipation of a breakthrough in parameter selection and module coordination.

**Acknowledgments**

This work is supported by the National Natural Science Foundation of China (Grant No. 62172128).

## References

1. Awn N S, Li Y, Zhao B, et al. LDAGSO: Predicting IncRNA-Disease Associations from Graph Sequences and Disease Ontology via Deep Learning techniques[C]//2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). 2022: 398-403Las Vegas, NV, USA: IEEE, 2022: 398-403.
2. Zeng M, Lu C, Zhang F, et al. SDLDA: lncRNA-disease association prediction based on singular value decomposition and deep learning[J]. Methods, 2020, 179: 73-80.
3. Zeng M, Lu C, Zhang F, et al. LncRNA–disease association prediction through combining linear and non-linear features with matrix factorization and deep learning techniques[C]//2019 BIBM. 2019: 577-582San Diego, CA, USA: IEEE, 2019: 577-582.
4. Xi W-Y, Zhou F, Gao Y-L, et al. LDCMFC: Predicting Long Non-Coding RNA and Disease Association Using Collaborative Matrix Factorization Based on Correntropy[J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2023, 20(3): 1774-1782.
5. Zeng M, Lu C, Fei Z, et al. DMFLDA: A Deep Learning Framework for Predicting lncRNA–Disease Associations[J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2021, 18(6): 2353-2363.
6. Wang B, Mezlini A M, Demir F, et al. Similarity network fusion for aggregating data types on a genomic scale[J]. Nature Methods, 2014, 11(3): 333-337.
7. Chen G, Wang Z, Wang D, et al. LncRNADisease: a database for long-non-coding RNA-associated diseases[J]. Nucleic Acids Research, 2012, 41(D1): D983-D986.
8. Wang J Z, Du Z, Payattakool R, et al. A new method to measure the semantic similarity of GO terms[J]. Bioinformatics, 2007, 23(10): 1274-1281.
9. Van Laarhoven T, Nabuurs S B, Marchiori E. Gaussian interaction profile kernels for predicting drug–target interaction[J]. Bioinformatics, 2011, 27(21): 3036-3043.
10. Wang B, Mezlini A M, Demir F, et al. Similarity network fusion for aggregating data types on a genomic scale[J]. Nature Methods, 2014, 11(3): 333-337.
11. Tang X, Cai L, Meng Y, et al. Indicator Regularized Non-Negative Matrix Factorization Method-Based Drug Repurposing for COVID-19[J]. Frontiers in Immunology, 2021, 11: 603615.
12. Zhang Z-C, Zhang X-F, Wu M, et al. A graph regularized generalized matrix factorization model for predicting links in biomedical bipartite networks[J]. Bioinformatics, 2020, 36(11): 3474-3481.
13. Wang H, Tang J, Ding Y, et al. Exploring associations of non-coding RNAs in human diseases via three-matrix factorization with hypergraph-regular terms on center kernel alignment[J]. Briefings in Bioinformatics, 2021, 22(5): bbaa409.
14. Lu C, Yang M, Li M, et al. Predicting Human lncRNA-Disease Associations Based on Geometric Matrix Completion[J]. IEEE Journal of Biomedical and Health Informatics, 2020, 24(8): 2420-2429.
15. Li P, Tiwari P, Xu J, et al. Sparse regularized joint projection model for identifying associations of non-coding RNAs and human diseases[J]. Knowledge-Based Systems, 2022, 258: 110044.
16. Lu C, Yang M, Luo F, et al. Prediction of lncRNA–disease associations based on inductive matrix completion[J]. Bioinformatics, 2018, 34(19): 3357-3364.
17. Shi Z, Zhang H, Jin C, et al. A representation learning model based on variational inference and graph autoencoder for predicting lncRNA-disease associations[J]. BMC Bioinformatics, 2021, 22(1): 136.
18. Liu T, Wang H, Yu H, et al. The Long Non-coding RNA HOTTIP Is Highly Expressed in Colorectal Cancer and Enhances Cell Proliferation and Invasion[J]. Molecular Therapy - Nucleic Acids, 2020, 19: 612-618.