DR-DDA: Double Residual Graph Transformer for Drug-disease Association Prediction

Ronghe Zhou and Yong Zhang*

***Abstract*—Discovering potential drug candidates for diseases is a critical task in drug repositioning. Conventional wet experiments are labor-intensive, so developing a framework for predicting drug-disease associations is essential. In this paper, we designed a new deep learning framework (called DR-DDA) to predict drug-disease associations. First we constructed a new heterogeneous network including drug Gaussian interaction profile kernel similarity, disease Gaussian interaction profile kernel similarity and association matrix of drugs-diseases, and in drug similarity we considered drug frequency weights. Next, we constructed two bi-parallel graph transformers, to better extract drug features and disease features in heterogeneous networks. We then performed separate residual operations on the dual parallel graph transformer with different weights to learn more complex features and improve accuracy. Finally we validated the model on two publicly available datasets and DR-DDA achieves better results compared to other base models, once again showing that DR-DDA can be an effective tool for predicting drug-disease association prediction.**

***Index Terms*—Deep learning, double residual, graph transformer, drug-disease association prediction**

I. Introduction

THE prediction of drug-disease associations is crucial in drug discovery and is also a hot issue. Drug discovery, also called drug repurposing, is the study of whether existing drugs or drugs under development can treat other diseases [1]. Traditional drug development is a long and resource-consuming process, from screening of drug candidates to various pharmacological tests to clinical trials, and successful approval for marketing often takes 10-15 years [2]. What's more, it takes 2 to 3 billion US dollars to successfully develop a new drug, but such a high investment cost is rewarded with a clinical approval rate of less than 10% [3], [4], [5]. Therefore, drug-disease association prediction has become an effective way to discover new drugs by analyzing the information of various features of drugs and diseases, and searching for new associations between drugs and diseases. With localized drug-disease associations, drug scientists can target follow-up studies, greatly shortening the drug development cycle. For example, aspirin was initially widely used as an antipyretic and analgesic, but later researchers found that aspirin could reduce the risk of cardiovascular disease, so now aspirin is one of the most important drugs for the prevention and treatment of cardiovascular disease [6]. Sildenafil was initially developed as a therapeutic drug for cardiovascular diseases, but later, scientists found that it also has significant efficacy on male erectile dysfunction in clinical trials, so now sildenafil has become a commonly used medication for treating male erectile dysfunction [7].

With the rapid development of artificial intelligence technology, the prediction of associations between drugs and diseases has opened up new opportunities. Deep learning is a currently popular technique, which utilizes neural network models to learn features from data for various tasks such as computer vision [8], natural language processing [9], health informatics [10], financial field [11], intelligent Transportation [12] and so on. Deep learning-based drug-disease association prediction is generally modeled using structural features of the network [13]. The current heterogeneous network consists mainly of drug-drug similarities, disease-disease similarities, and drug-disease associations validated by wet experiments [14].

Although a number of deep learning methods have been investigated for drug-disease association prediction, most of them are direct fusions of drug-drug similarity matrix, disease-disease similarity matrix, and drug-disease association matrix to compose a heterogeneous network. This ignores the frequency of drug use, if a drug is used frequently in the drug-disease association matrix, there may be a greater likelihood that it will be able to treat the new disease, so we considered the drug use frequency matrix in constructing the heterogeneous network. Second, most of the previous methods use GNN for drug-disease association prediction, including GCN and GAT. GNN generally updates node representations through the neighbor nodes of a node, and it is difficult to capture the features of distant nodes in the graph for complex graphs [15].

In order to solve the above problem, we added the drug frequency matrix to the feature matrix. Specifically, for the drug-disease association matrix, we counted the total number of drug occurrences in each row and then normalized the resulting drug frequency matrix and multiplied it by the drug similarity matrix. Because if a drug appears more often, it means that the drug is likely to be able to treat more diseases, this also highlights the drugs that stand out in the drug similarity matrix and improves the accuracy of the prediction. Second, to address the problem that GNNs do not convey features of distant nodes well when dealing with complex graph data, we use graph transformer to predict the association between drugs and diseases. Graph transformer uses the attention mechanism to directly couple any two nodes in the graph without limiting the distance between nodes, so it can better capture the full range of features [16]. In addition, in order to solve the problem of vanishing and degrading gradients, we add residuals to the graph transformer so that the network can skip some layers to deliver messages directly, avoiding performance degradation due to the network being too deep [17]. The main contributions of this paper are as follows.

Manuscript received July 19, 2025; revised September 15, 2025.

Ronghe Zhou is a postgraduate student of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, Liaoning 114051, PR China (e-mail: ronghezhou@ustl.edu.cn).

Yong Zhang is a professor of School of Electronic and Information Engineering, University of Science and Technology Liaoning, Anshan, Liaoning 114051, PR China (*Corresponding author, e-mail: zy9091@ustl.edu.cn).

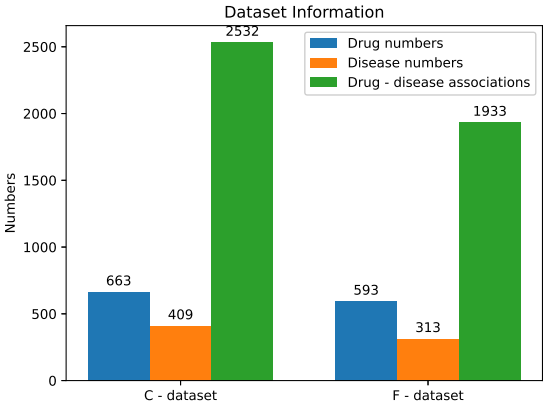


Fig. 1: Information of the dataset.

•We proposed a new deep learning-based drug-disease association prediction model, which well predicts the association between drugs and diseases by obtaining feature embeddings of drugs and diseases through heterogeneous networks.

•The heterogeneous network was constructed by considering not only drug-to-drug similarity, disease-to-disease similarity, and drug-to-disease association, but also drug frequency.

•A bi-residual transformer was constructed to better extract drug features and disease features from heterogeneous networks.

The paper is organized in the following structure, the section I is an introduction to the relevant background knowledge and the limitations that exist in some of the methods. Section II is the dataset information and the proposed methodology. Section III is the analysis and discussion of the experimental results. The conclusion and future work is shown in section IV.

II. Materials and Methods

In this section we will give a specific description of the dataset used, the construction of the heterogeneous network, and the double residual transformer.

A. Datasets

To validate the effectiveness of DR-DDA, we evaluated the model on two datasets (Cdataset [18] and Fdataset [19]). The Cdataset dataset contains 663 drugs, 409 diseases, and 2532 known associations; and the Fdataset dataset contains 593 drugs, 313 diseases, and 1933 known associations. The details of the dataset are displayed in Figure 1. Gaussian interaction profile (GIP) kernel similarity [20] was used to calculate drug-drug and disease-disease similarity. GIP is primarily designed to address the problem of sparsity in the similarity of molecular fingerprints of drugs and the similarity of disease phenotypes, more information can be found in reference Liu et al. [21].

B. Construction of Heterogeneous Networks

Inspired by Liu et al.[14], we constructed heterogeneous networks as initial input features for DR-DDA. Different from their method, we consider the drug frequency matrix in the construction of the heterogeneous network. Our heterogeneous network consists of three main components, which are drug-drug GIP similarity, disease-disease GIP similarity, and known associations that exist between drugs and diseases. The drug-drug GIP similarity is denoted as $A_{drug} \in R^{n_1 \times n_1}$, and the disease-disease GIP similarity is denoted as $A_{dis} \in R^{n_2 \times n_2}$, $n1$ and $n2$ denote the number of drugs and the number of diseases, respectively. The drug-disease association matrix is defined as $A_{dd} \in R^{n_1 \times n_2}$, which is validated by relevant wet experiments, if A_{ij}=1,it means that there is a known association between the corresponding drug and the disease, otherwise it means that there is no known association between the corresponding drug and the disease.The main steps of heterogeneous matrix construction are as follows:

• Step1: The frequency of drug use f was calculated from the drug-disease association matrix A. The formula for f is $f_i = \sum_{j=1}^{n_2} A_{ij}, i = 1, 2, \cdots, n_1$.

• Step2: Scaling operation for drug frequency, the scaled drug frequency g was obtained, $g_i = f_i^{-0.5}, i = 1, 2, \cdots, n_1$.

• Step3: Update the drug similarity matrix, that is, consider drug frequencies in the drug-drug GIP similarity matrix, $C = diag(g) \cdot A_{drug}$, where diag is the diagonal matrix consisting of the vectors g, where g is $\begin{bmatrix} g_1 & 0 & \cdots & 0 \\ 0 & g_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_{n_1} \end{bmatrix}$

• Step4: The middle matrices $R_{n_1 \times (n_1+n_2)}$ and $S_{n_2 \times (n_1+n_2)}$ are obtained, $R = [CD]$, $S = \left[D^T N\right]$

• Step5: The final heterogeneous network matrix H is obtained, $H = \begin{bmatrix} R \\ S \end{bmatrix}$

C. Double residual graph transformer

Transformer [22] has made a big splash in the field of natural language processing (NLP) since it was proposed. Graph transformer [23], [24] is a neural network that applies transformer to graph-structured data, combining the advantages of transformer and graph neural networks (GNN). We constructed a double residual graph transformer inspired by the work in Shi et al. [25], Liu et al. [21] and Li et al. [26].

1) Multi-head attention: The attention mechanism is at the heart of graph transformer and its main idea is to measure the correlation between different elements. In graph-structured data, the source node needs to measure which neighboring nodes are more important to itself. Q_i represents the feature information of the source node and K_j and V_j represent the feature information of the neighboring nodes, respectively, the attention scores of the source node Q_i and each of the neighboring nodes K_j are computed, the higher the attention score, the higher the attention weight will be assigned. After obtaining the attention score, the source node performs a weighted summation of the neighbor nodes V_j based on the attention score to obtain the information of the aggregated neighbor nodes. Here are the specific calculations for the multi-head attention.

For each node i and its neighbor node j, its query matrix Q_i, key matrix K_j and value matrix V_j are defined as

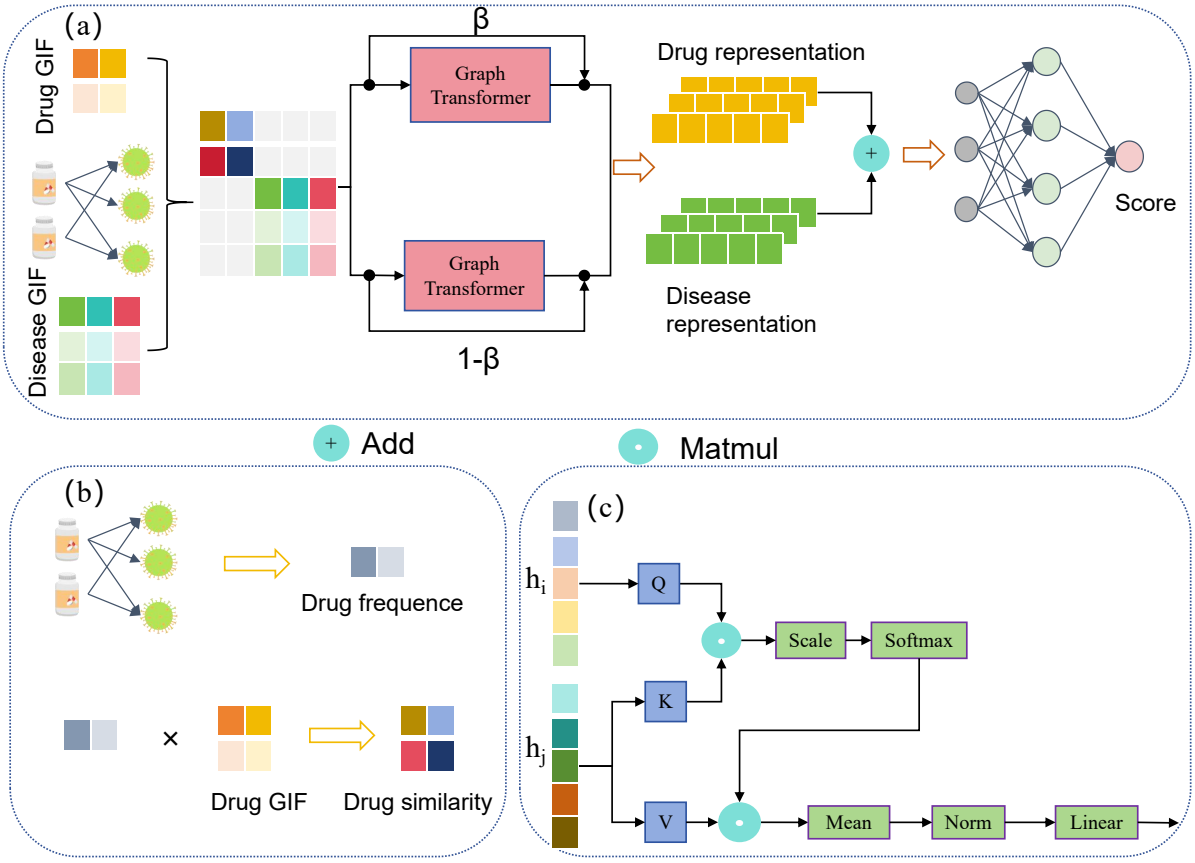


Fig. 2: a: The overall structure of DR-DDA. b: Calculation of drug frequency matrices. c: Multi-head attention in graph transformer.

Equation 1,2 and 3.

$$Q_i = W_q X_i + b_q \tag{1}$$

$$K_j = W_k X_j + b_k \tag{2}$$

$$V_j = W_v X_j + b_v \tag{3}$$

where W_q, W_k and W_v are the learnable parameter matrix of Q_i, K_j and V_j , b_q, b_q and b_q are the bias vector, and X is the feature matrix of the heterogeneous network.

After obtaining Q_i, K_j and V_j, for node i and its neighboring nodes, the attention score α is calculated as shown in 4 and 5.

$$e_{i,j} = \frac{Q_i \cdot K_j}{\sqrt{d_k}} \tag{4}$$

$$\alpha_{i,j} = \frac{exp(e_{i,j})}{\sum_{k \in N(i)} exp(e_{i,k})} \tag{5}$$

where Dim is the dimension of each header, the division by $\sqrt{d_k}$ in is to scale the dot product and prevent the gradient from vanishing or exploding [22]. $N(i)$ denotes the set of neighbors of all nodes and then the attention coefficient α is obtained by softmax.

After getting the attention score, it starts aggregating the information of neighboring nodes. Message passing is the core of graph neural networks, which is where nodes in the graph exchange information with neighboring nodes to update their features [27]. For node i, its message m is the product of the value vectors of the neighboring nodes and the attention. Then calculate the average value of message m over all heads, it is shown in 6 and 7.

$$m_i = \sum_{j \in N(i)} \alpha_{i,j} \cdot V_j \tag{6}$$

$$\bar{m_i} = \frac{1}{H} \sum_{h=1}^{H} m_i^h \tag{7}$$

where V_j denotes the value vector of node i's neighbor node j, H is the number of heads, and m_i^h is the message of node i on the current head.

2) Double residual and prediction: Residual can effectively solve the problem of gradient vanishing caused by deeper network layers and improve the generalization ability of the model [17]. We designed two residuals to address the problem of gradient loss as the network hierarchy gets deeper.

$$X_1 = X_1 + \beta \cdot X \tag{8}$$

$$X_2 = X_2 + (1 - \beta) \cdot X \tag{9}$$

where X_1 and X_2 are the feature matrices of message $\bar{m}$ after Layer-Norm [28], Relu, Dropout and other operations respectively and β is the residual weight.

And then X_1 and X_2 are summed and the number of drugs and the number of diseases are taken as the representation of drugs and the representation of diseases, respectively

$$\begin{bmatrix} x_r \\ x_d \end{bmatrix} = X_1 + X_2 \tag{10}$$

where x_r is a representation of the drug and x_d is a representation of the disease.

Finally, we extract features of specific node pairs from the computed drug representations and disease representations, respectively, and input them into the MLP to obtain the final prediction scores.

$$E_{i,i} = ReLU\left(x_r\left[s_{i1},:\right] + x_d\left[s_{i2},:\right]\right), i = 1, \cdots, M \tag{11}$$

$$Score = MLP(E_{i,i}) \tag{12}$$

where $s \in R^{M\times 2}$ is the matrix corresponding to the samples, M is the number of samples, each row $s_i = [s_{i1}, s_{i2}]$ represents a sample, s_{i1} represents the first column of the sample, which is the index of the drug, and s_{i1} represents the second column of the sample, which is the index of the disease.

D. Optimization and loss function

The Adam optimizer [29] is used to optimize the parameters in our model. Because drug-disease association prediction is a classification problem, for the classification task, we use the binary cross entropy (BCE) loss to evaluate the difference between predicted and true values, it is defined as Eq.(13)

$$Loss_{BCE} = -\frac{1}{M}\sum_{i=1}^{M} y_i \log \widehat{y}_i + (1 - y_i)\log(1 - \widehat{y}_i) \tag{13}$$

where M is the sample size, y_i represents the true label. It is known that y_i = 1 when the drug is associated with the disease, and y_i = 0 when there is no such association, $\widehat{y}_i$ is the predicted label.

E. Overall structure of DR-DDA

Algorithm 1 is a pseudo-code for DR-DDA, which first calculates the drug-to-drug GIP similarity matrix, the disease-to-disease GIP similarity matrix, and then combines it with the drug-to-disease association matrix to form a heterogeneous network. Then, the features of drugs and diseases are obtained using graph transformer and double residual network, and finally feature pairs are obtained from the samples and input into MLP for prediction. Figure 2 shows the general structure of DR-DDA, part a shows the general structure, part b shows the computation of the drug frequency matrix and the updated drug similarity matrix. part c shows the details of the graph transformer layer.

Algorithm 1: The pseudo-code of the proposed DR-DDA

Input: Drug-Drug GIP Similarity Matrix, Disease-Disease GIP Similarity Matrix, Drug-Disease Association Matrix, max epoch Max_{epoch}

Output: Predicted score

```
Constructing the feature matrix of the isomorphic graph, used the method in section II-B ;
while the maximum number of epoch is not reached do
    Calculate query matrix Q_i, key matrix K_j and value matrix V_j using Eq.(1),using Eq.(2) and using Eq.(3);
    Calculate the attention score α using Eq.(5);
    Calculate the average message m̄ after multi-head attention, using Eq.(7);
    Calculate X_1 and X_2 using Eq.(8) and Eq.(9);
    Using Eq.(10) to get the representation of drugs and the representation of diseases;
    Feature pairs were extracted from the samples based on drug and disease representations using Eq.(11);
    The predicted scores are obtained by inputting them into the MLP, using Eq.(12);
end
Return predicted score
```

III. Experimental results and discussions

A. Experimental settings

We use the pytorch [30] framework to implement DR-DDA, pytorch version is 2.0.1, torch geometric[31] version is 2.4.0, the maximum training epoch set to 500, the learning rate set to 0.0001, the dropout set to 0.5, and the number of cross-validation folds set to 10. Nine folds at a time as a training set and one fold as a test set.

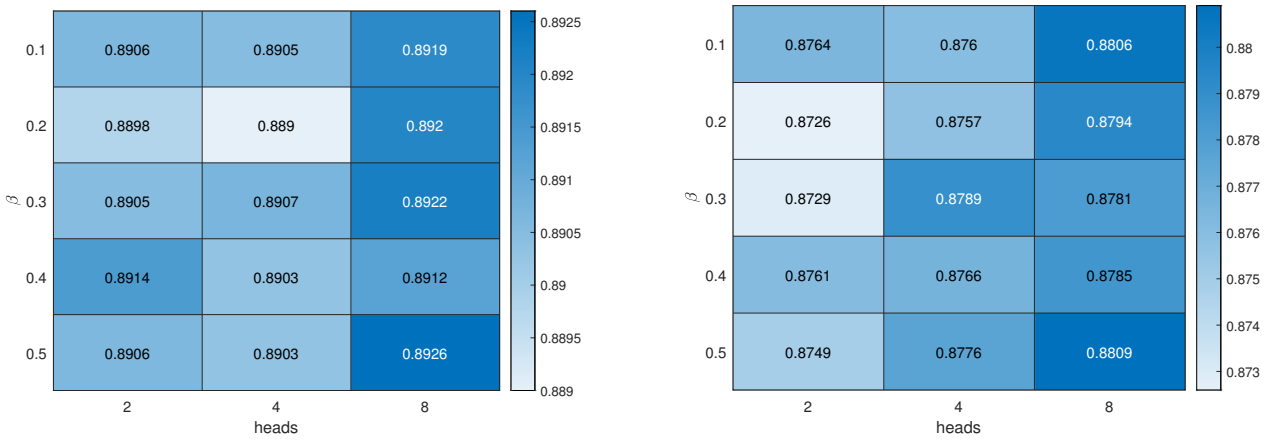
The residual weight α and number of heads will directly affect the performance of feature fusion, and we set the value range of α as follows: $\beta \in \{0.1 \sim 0.5\}$, the number of heads for multi-head attention is set to $\{2, 4, 8\}$. As shown in Figures 3 and 4, when obtaining best values for AUC and AUPR, β = 0.5, heads = 8.

B. Performance of DR-DDA

To further validate the performance of DR-DDA, we compare it with some base models(GCN [32], GAT [33], GATv2 [34], k-GNNs [35] and UniMP [25]), evaluation metrics included AUC, AUPR, Accuracy, Precision, Recall, F1-score and MCC. All metrics results are ten-fold cross-validated means and standard deviations. Tables 2 and 3 show the results of DR-DDA and other comparison models on the two datasets. It is clear from the tables that DR-DDA outperforms the comparison models in most of the evaluation metrics, and although it is not the highest in terms of accuracy, taken together, DR - DDA is still the best performing model. Figure 5 shows the histograms of all metrics for all models on both datasets. From the figure we can see that DR-DDA ranks first in the height of the bars for most of the metrics and the height of the error bars is also close to the other compared models, once again demonstrating the better performance of DR-DDA.

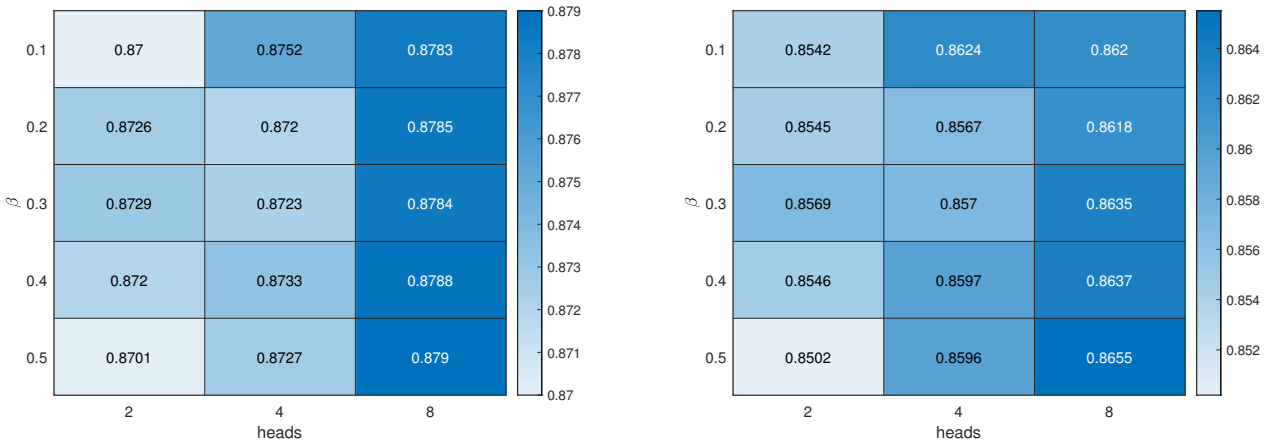
C. Ablation experiment

1) Effect of double residuals graph transformer: To further analyze the role of double residuals graph transformer



(a) (b)

Fig. 3: (a):AUC on C-datasets. (b):AUPR on C-datasets.



(a) (b)

Fig. 4: (a):AUC on F-datasets. (b):AUPR on F-datasets.

TABLE I: Prediction performance of DR-DDA on the C-dataset.

Model	AUC	AUPR	Accuracy	Precision	Recall	F1	MCC
GCN	0.8796 (0.024)	0.8638 (0.026)	0.7842 (0.030)	0.8082 (0.067)	0.7677 (0.102)	0.7788 (0.033)	0.5813 (0.049)
GAT	0.8625 (0.022)	0.8392 (0.030)	0.7139 (0.048)	0.8419 (0.023)	0.5278 (0.116)	0.6412(0.088)	0.4631 (0.078)
GATv2	0.8646 (0.019)	0.8427 (0.026)	0.7307 (0.036)	**0.8441 (0.026)**	0.5672 (0.087)	0.6743 (0.061)	0.4899 (0.060)
k-GNNs	0.8752 (0.024)	0.8589 (0.031)	0.7591 (0.042)	0.8316 (0.029)	0.6493 (0.091)	0.7260 (0.063)	0.5328 (0.075)
UniMP	0.8642 (0.019)	0.8407 (0.026)	0.7326 (0.021)	0.8441 (0.032)	0.5723 (0.047)	0.6805 (0.035)	0.4922 (0.040)
DR-DDA	**0.8926 (0.027)**	**0.8809 (0.033)**	**0.8092 (0.036)**	0.8041 (0.027)	**0.8191 (0.088)**	**0.8088 (0.050)**	**0.6223 (0.069)**

TABLE II: Prediction performance of DR-DDA on the F-dataset.

Model	AUC	AUPR	Accuracy	Precision	Recall	F1	MCC
GCN	0.8646 (0.026)	0.8448 (0.034)	0.7693 (0.035)	0.8322 (0.032)	0.6803 (0.108)	0.7417 (0.066)	0.5524 (0.056)
GAT	0.8426 (0.025)	0.8173 (0.023)	0.6769 (0.044)	0.8242 (0.042)	0.4548 (0.122)	0.5749 (0.098)	0.3982 (0.069)
GATV2	0.8453 (0.024)	0.8207 (0.029)	0.6811 (0.040)	0.8439 (0.035)	0.4459 (0.096)	0.5772 (0.078)	0.4115 (0.064)
k-GNNs	0.8519 (0.025)	0.8319 (0.029)	0.7331 (0.043)	**0.8445 (0.036)**	0.5716 (0.098)	0.6771 (0.072)	0.4941 (0.076)
UniMP	0.8551 (0.031)	0.8300 (0.028)	0.6943 (0.060)	0.8292 (0.042)	0.4942 (0.163)	0.6023 (0.123)	0.4293 (0.100)
DR-DDA	**0.8790 (0.030)**	**0.8655 (0.029)**	**0.7964 (0.039)**	0.7714 (0.047)	**0.8484 (0.052)**	**0.8066 (0.035)**	**0.5990 (0.079)**

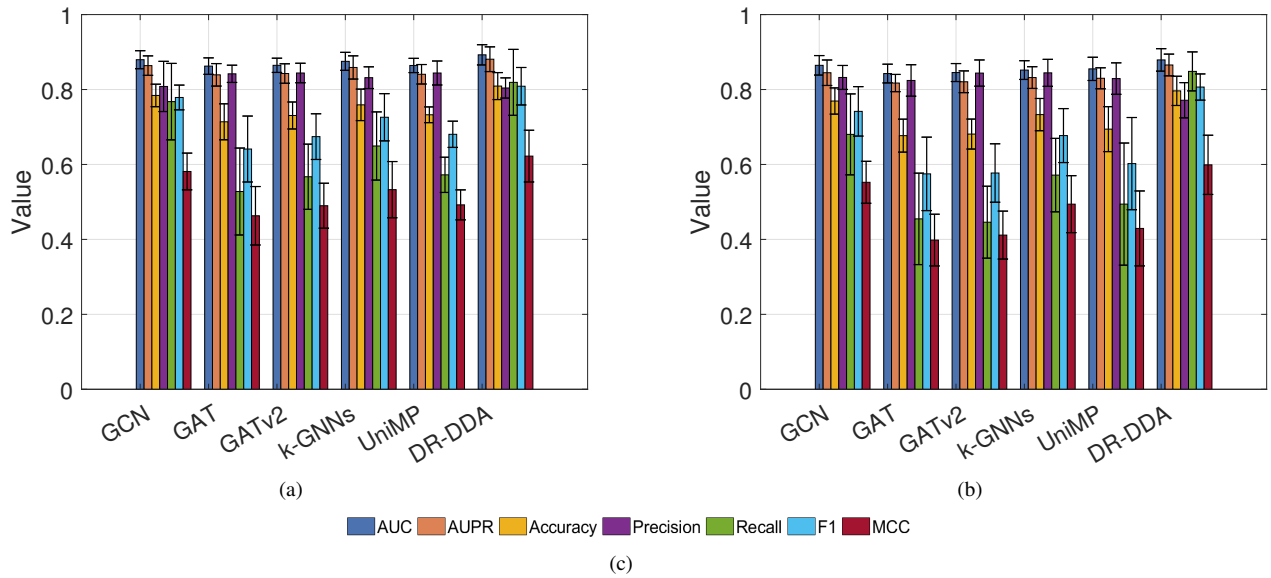


Fig. 5: (a):All models on C-datasets. (b):All models on F-datasets.

TABLE III: The effect of double residual graph transformer on the C-dataset.

Metrics	Single-DDA	DR-DDA
AUC	0.8641 (0.020)	**0.8926 (0.027)**
AUPR	0.8447 (0.026)	**0.8809 (0.033)**
Accuracy	0.7310 (0.033)	**0.8092 (0.036)**
Precision	**0.8453 (0.031)**	0.8041 (0.027)
Recall	0.5695 (0.095)	**0.8191 (0.088)**
F1	0.6747 (0.064)	**0.8088 (0.050)**
MCC	0.4915 (0.053)	**0.6223 (0.069)**

TABLE IV: The effect of double residual graph transformer on the F-dataset.

Metrics	Single-DDA	DR-DDA
AUC	0.8463 (0.027)	**0.8790 (0.030)**
AUPR	0.8208 (0.029)	**0.8655 (0.029)**
Accuracy	0.7010 (0.029)	**0.7964 (0.039)**
Precision	**0.8333 (0.031)**	0.7714 (0.047)
Recall	0.5039 (0.068)	**0.8484 (0.052)**
F1	0.6250 (0.052)	**0.8066 (0.035)**
MCC	0.4385 (0.050)	**0.5990 (0.079)**

in the model, we designed a variant of DR-DDA that only a graph transformer and residual, denoted as (Single-DDA). Tables III and IV show the results of Single-DDA and DR-DDA for all metrics on both datasets. It is obvious from the table that DR-DDA, compared to Single-DDA, has better performance in AUC, AUPR, Accuracy, Recall, F1, and MCC metrics, except for Precision, which once again proves the excellent performance of the double residual structure. Figures 7 and 8 show boxplots of Single-DDA and DR-DDA for multiple assessment metrics on both datasets. It is clear from the graphs that the median of the boxes is larger for DR-DDA on the other metrics except Precision, indicating that DR-DDA has better classification performance. In addition, DR-DDA has shorter boxes, indicating that DR-DDA also has excellent stability with a small range of data fluctuations.

2) Effect of residuals: In order to verify the role of residual structure in the model, we designed a variant that removes the residual structure from the model and retains only the double graph transforer,called Nores-DDA, and the results are shown in Tables V and VI. As can be seen from the tables, in most of the metrics, the performance is significantly worse than the original model after de-pointing the residual structure, which once again demonstrates the excellent performance of DR-DDA.

TABLE V: The role of residual structure on Cdataset in the parallel graph transformer structure.

Metrics	Nores-DDA	DR-DDA
AUC	0.8872(0.026)	**0.8926 (0.027)**
AUPR	0.8680(0.028)	**0.8809 (0.033)**
Accuracy	0.8069(0.037)	**0.8092 (0.036)**
Precision	0.7790 (0.043)	**0.8041 (0.027)**
Recall	**0.8622 (0.050)**	0.8191 (0.088)
F1	**0.8171(0.033)**	0.8088 (0.050)
MCC	0.6200(0.071)	**0.6223 (0.069)**

TABLE VI: The role of residual structure on Fdataset in the parallel graph transformer structure.

Metrics	Nores-DDA	DR-DDA
AUC	0.8737(0.029)	**0.8790 (0.030)**
AUPR	0.8597(0.027)	**0.8655 (0.029)**
Accuracy	0.7907 (0.037)	**0.7964 (0.039)**
Precision	0.7698 (0.043)	**0.7714 (0.047)**
Recall	0.8339 (0.034)	**0.8484 (0.052)**
F1	0.7998(0.031)	**0.8066 (0.035)**
MCC	0.5847(0.071)	**0.5990 (0.079)**

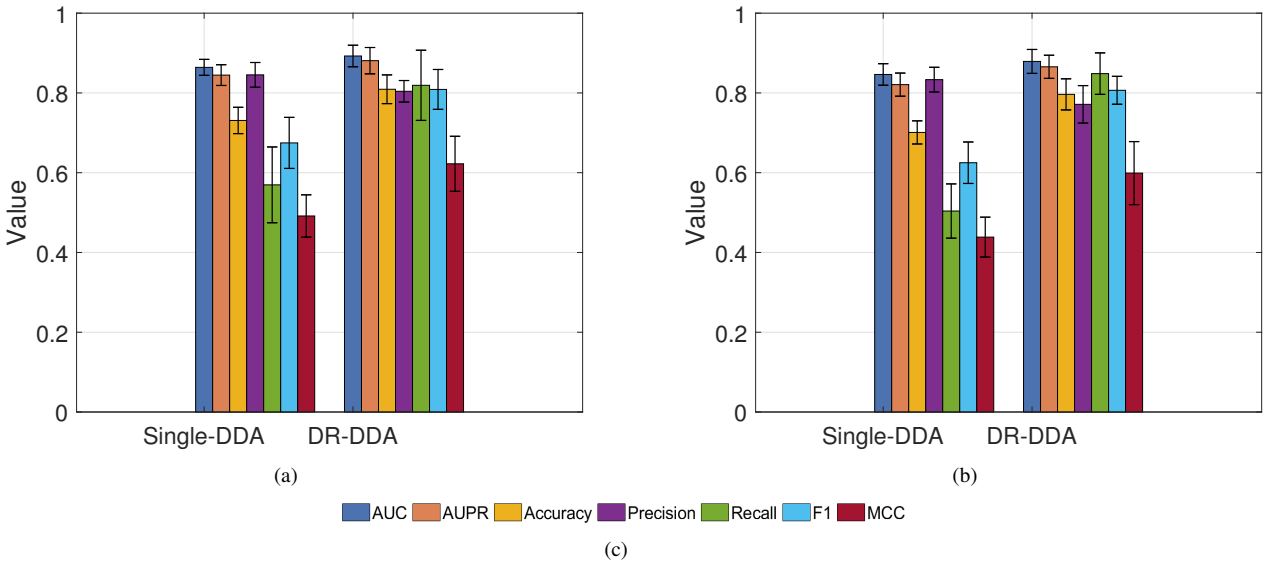


Fig. 6: (a):Single-DDA and DR-DDA on C-datasets. (b):Single-DDA and DR-DDA on F-datasets.

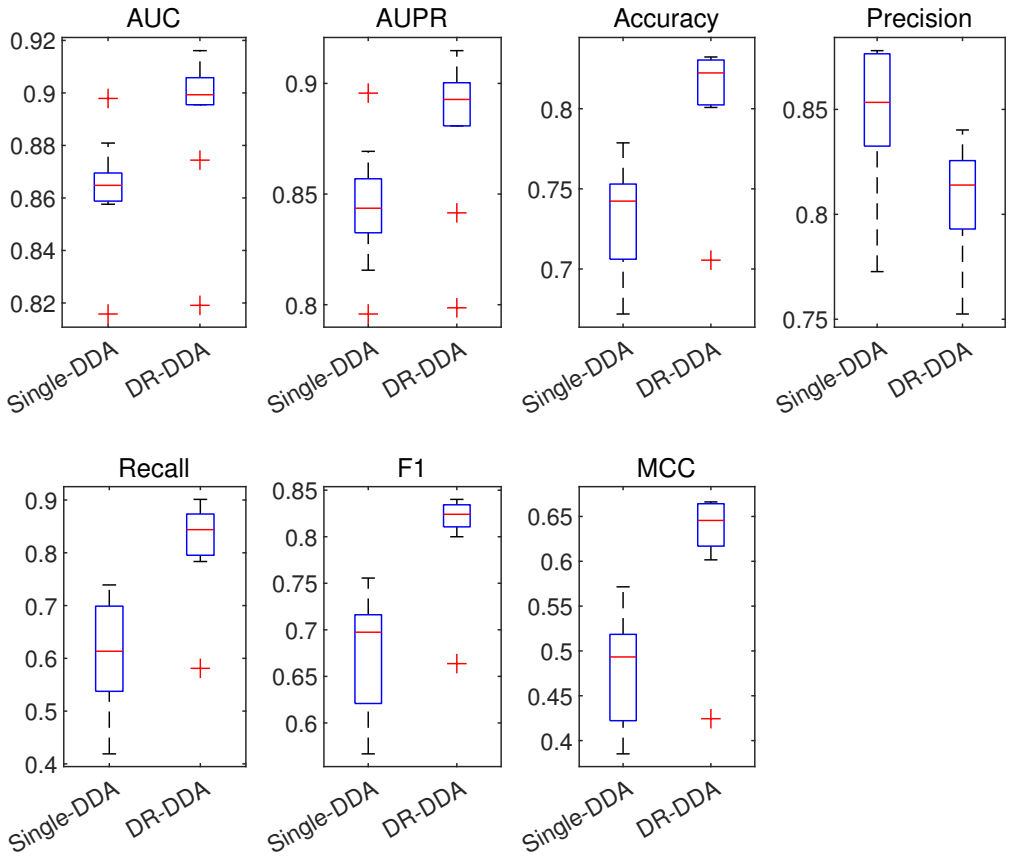


Fig. 7: Boxplots of Single-DDA and DR-DDA on C-datasets.

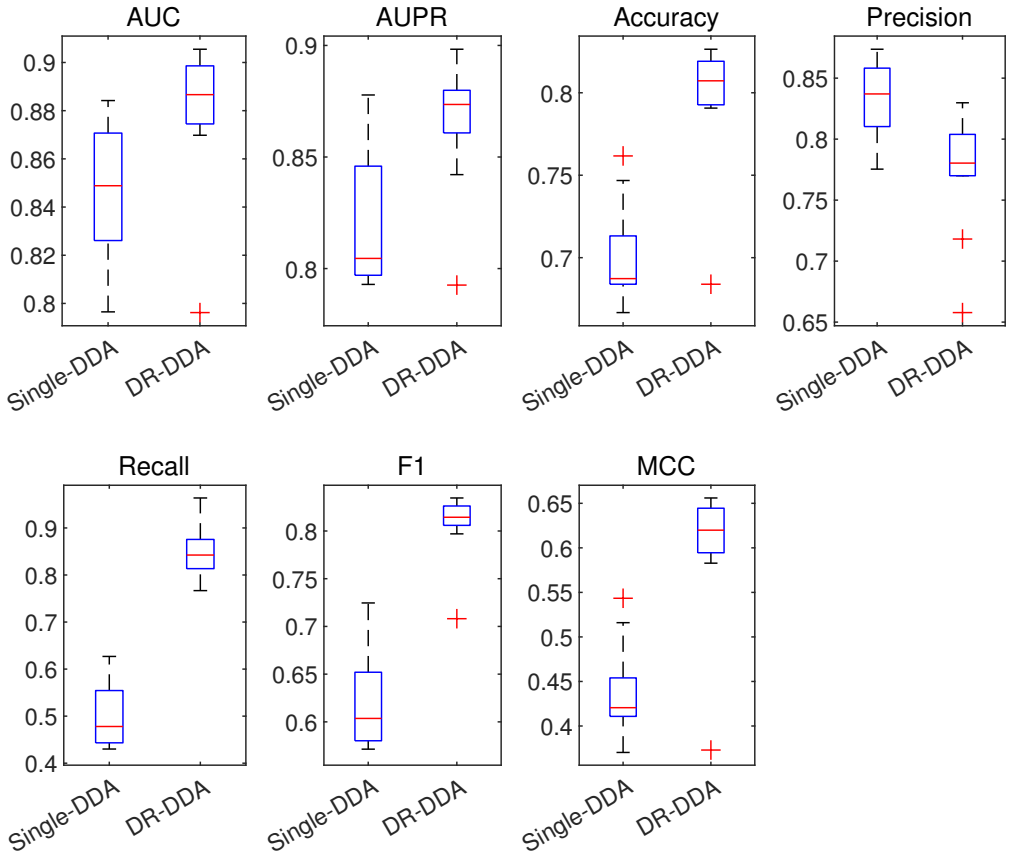


Fig. 8: Boxplots of Single-DDA and DR-DDA on F-datasets.

IV. Conclusions

In this paper, we proposed a deep learning framework to predict drug-disease association. To overcome the situation that all drugs in the drug-drug similarity matrix have the same importance and that neighboring nodes are not well aggregated in traditional GNNs, we design a double residual graph transformer (DR-DDA) to predict the association between drugs and diseases. First, we processed the drug-drug similarity matrix to consider drug frequencies. Then, in order to better extract the features in the heterogeneous network, we designed the double residual graph transformer. We validated the performance of DR-DDA on two publicly available datasets and show that DR-DDA is able to outperform the comparison models on most of the evaluation metrics. In order to further validate the effect of the double residual graph transformer, we designed an ablation experiment by keeping only one graph transformer and removing the double residuals in it, and the results show that the performance of the model significantly decreases after removing the double residuals, which again validates the performance of the double residual graph transformer.

In the future, we will develop some more advanced deep learning methods to predict drug-disease associations, and at the same time, consider more biochemical information to build heterogeneous networks for richer representation of drugs and diseases.

References

[1] J.-P. Jourdan, R. Bureau, C. Rochais, and P. Dallemagne, "Drug repositioning: a brief overview," *Journal of Pharmacy and Pharmacology*, vol. 72, no. 9, pp. 1145–1151, 2020.

[2] K. Park, "A review of computational drug repurposing," *Translational and Clinical Pharmacology*, vol. 27, no. 2, p. 59, 2019.

[3] Z. Huang, Z. Xiao, C. Ao, L. Guan, and L. Yu, "Computational approaches for predicting drug-disease associations: a comprehensive review," *Frontiers of Computer Science*, vol. 19, no. 5, pp. 1–15, 2025.

[4] Z. Tanoli, U. Seemab, A. Scherer, K. Wennerberg, J. Tang, and M. Vähä-Koskela, "Exploration of databases and methods supporting drug repurposing: a comprehensive survey," *Briefings in Bioinformatics*, vol. 22, no. 2, pp. 1656–1678, 2021.

[5] T. T. Ashburn and K. B. Thor, "Drug repositioning: identifying and developing new uses for existing drugs," *Nature Reviews Drug Discovery*, vol. 3, no. 8, pp. 673–683, 2004.

[6] G. Pasero, P. Marson *et al.*, "A short history of anti-rheumatic therapy. ii. aspirin," *Reumatismo*, vol. 62, no. 2, pp. 148–156, 2010.

[7] H. A. Ghofrani, I. H. Osterloh, and F. Grimminger, "Sildenafil: from angina to erectile dysfunction to pulmonary hypertension and beyond," *Nature Reviews Drug discovery*, vol. 5, no. 8, pp. 689–702, 2006.

[8] A. Voulodimos, N. Doulamis, A. Doulamis, and E. Protopapadakis, "Deep learning for computer vision: A brief review," *Computational Intelligence and Neuroscience*, vol. 2018, no. 1, p. 7068349, 2018.

[9] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 2, pp. 604–624, 2020.

[10] C. TAOUSSI, I. HAFIDI, and A. METRANE, "Machine learning and deep learning in health informatics: Advancements, applications, and challenges." *Engineering Letters*, vol. 33, no. 5, pp. 1448–1461, 2025.

[11] A. M. Ozbayoglu, M. U. Gudelek, and O. B. Sezer, "Deep learning for financial applications: A survey," *Applied Soft Computing*, vol. 93, p. 106384, 2020.

[12] Y. Ma, A. Halik, and X. Quan, "Flow-attention based dynamic graph convolutional recurrent network for traffic forecasting." *Engineering Letters*, vol. 33, no. 5, pp. 1543–1557, 2025.

[13] Y. Kim, Y.-S. Jung, J.-H. Park, S.-J. Kim, and Y.-R. Cho, "Drug-disease association prediction using heterogeneous networks for computational drug repositioning," *Biomolecules*, vol. 12, no. 10, p. 1497, 2022.

[14] B.-M. Liu, Y.-L. Gao, F. Li, C.-H. Zheng, and J.-X. Liu, "Slgcn: Structure-enhanced line graph convolutional network for predicting drug–disease associations," *Knowledge-Based Systems*, vol. 283, p. 111187, 2024.

[15] U. Alon and E. Yahav, "On the bottleneck of graph neural networks and its practical implications," *arXiv preprint arXiv:2006.05205*, 2020.

[16] B. Wang, L. Cui, L. Bai, and E. R. Hancock, "Graph transformer: Learning better representations for graph neural networks," in *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshops, S+ SSPR 2020, Padua, Italy, January 21–22, 2021, Proceedings*. Springer, 2021, pp. 139–149.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[18] H. Luo, J. Wang, M. Li, J. Luo, X. Peng, F.-X. Wu, and Y. Pan, "Drug repositioning based on comprehensive similarity measures and bi-random walk algorithm," *Bioinformatics*, vol. 32, no. 17, pp. 2664–2671, 2016.

[19] A. Gottlieb, G. Y. Stein, E. Ruppin, and R. Sharan, "Predict: a method for inferring novel drug indications with application to personalized medicine," *Molecular Systems Biology*, vol. 7, no. 1, p. 496, 2011.

[20] J. Ha, "Smap: Similarity-based matrix factorization framework for inferring mirna-disease association," *Knowledge-Based Systems*, vol. 263, p. 110295, 2023.

[21] J. Liu, S. Guan, Q. Zou, H. Wu, P. Tiwari, and Y. Ding, "Amdgt: Attention aware multi-modal fusion using a dual graph transformer for drug–disease associations prediction," *Knowledge-Based Systems*, vol. 284, p. 111329, 2024.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[23] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[24] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," in *Proceedings of the Web Conference 2020*, 2020, pp. 2704–2710.

[25] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, "Masked label prediction: Unified message passing model for semi-supervised classification," *arXiv preprint arXiv:2009.03509*, 2020.

[26] G. Li, S. Li, C. Liang, Q. Xiao, and J. Luo, "Drug repositioning based on residual attention network and free multiscale adversarial training," *BMC Bioinformatics*, vol. 25, no. 1, p. 261, 2024.

[27] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1263–1272.

[28] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[29] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[30] A. Paszke, "Pytorch: An imperative style, high-performance deep learning library," *arXiv preprint arXiv:1912.01703*, 2019.

[31] M. Fey and J. E. Lenssen, "Fast graph representation learning with pytorch geometric," *arXiv preprint arXiv:1903.02428*, 2019.

[32] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[33] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[34] S. Brody, U. Alon, and E. Yahav, "How attentive are graph attention networks?" *arXiv preprint arXiv:2105.14491*, 2021.

[35] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and leman go neural: Higher-order graph neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 4602–4609.