AutoMSR: Auto Molecular Structure Representation Learning for Multi-label Metabolic Pathway Prediction

Jiamin Chen, Jianliang Gao*, Tengfei Lyu, Babatounde Moctard Oloulade, Xiaohua Hu

Abstract—It is significant to comprehend the relationship between metabolic pathway and molecular pathway for synthesizing new molecules, for instance optimizing drug metabolization. In bioinformatics fields, multi-label prediction of metabolic pathways is a typical manner to understand this relationship. Graph neural networks (GNNs) have become an effective method to extract molecular structure's features for multi-label prediction of metabolic pathways. Though GNNs can effectively capture structural features from molecular structure graph, building a well-performed GNN model for a given molecular structure data set requires the manual design of the GNN architecture and fine-tuning of the hyperparameters, which are time-consuming and rely on expert experience. To address the above challenge, we design an end-to-end automatic molecular structure representation learning framework named AutoMSR that can design the optimal GNN model based on a given molecular structure data set without manual intervention. We propose a multi-seed age evolution (MSAE) search algorithm to identify the optimal GNN architecture from the GNN architecture subspace. For a given molecular structure data set, AutoMSR first uses MSAE to search the GNN architecture, and then it adopts a tree-structured parzen estimator to obtain the best hyperparameters in the hyperparameters subspace. Finally, AutoMSR automatically constructs the optimal GNN model based on the best GNN architecture and hyperparameters to extract the molecular structure features for multi-label metabolic pathway prediction. We test the performance of AutoMSR on the real data set KEGG. The experiment results show that AutoMSR outperforms baseline methods on different multi-label classification evaluation metrics.

Index Terms—Metabolic Pathway, Multi-label Prediction, Graph Neural Architecture Search, Graph Neural Network

1 Introduction

It is an important and effective manner to synthesize new drugs [1] based on studying the mappings relationship between molecular structure and metabolic pathway. The metabolic pathway is a chain process involving a series of chemical reactions, where chemical substances form the next stage of metabolites under the action of enzymes in biological cells. Researching drug molecule metabolic pathways can help us understand toxic metabolites and the metabolism of a new drug [2]. Simultaneously, evaluating drug metabolism and pharmacokinetic effects is the necessary step for discovering new drugs [3].

In recent decades, a lot of influential research focuses on metabolic pathway prediction tasks, such as TrackSM [4], biosynthesis pathway finding tool [5], PathPred [6], Pathway Tools [7], UM-BBD Pathway Prediction System [8] and PathComp [9]. Traditional machine learning methods have made some breakthroughs in metabolic pathway prediction tasks, involving Support Vector Machine [10], K Nearest Neighbor Algorithm [11], Naive Bayes [12] and Decision Trees [13]. Although these approaches can implement single-label and multi-label prediction tasks, traditional machine learning methods fail to use molecular structure information to improve prediction performance, especially for multi-label prediction tasks. With the development of graph neural networks, it has been widely used in the bioinformatics fields and has achieved great success [14] [15] [16]. Graph neural networks can effectively mine graph structural and semantic features by using the message passing network to achieve graph convolution operation for molecule structure [17]. For multi-label of metabolic pathway prediction tasks, Baranwal *et al.* [18] proposes a multi-label metabolic pathway prediction framework based on graph convolutional network (GCN) [19]. Yang *et al.* [20] trains a two-layer graph attention network [21] to encode molecular structure features as the input of the downstream metabolic pathway prediction task.

Although graph neural networks have achieved high performance in multi-label prediction tasks on the metabolic pathway, building a graph neural network for a given molecular structure data set requires designing graph neural network architecture and fine-tuning hyperparameters, which is a time-consuming process that relies on expert experience. To solve this challenge, many studies have focused on graph neural architecture search (GNAS) [22] [23] [24] [25] [26]. It can automatically design the optimal graph neural network architecture for different graph data to achieve better performance than manual graph neural network on different graph tasks, such as node classification, link prediction, and graph classification.

In order to better extract the molecular structure features for the multi-label prediction tasks on the metabolic

- *Jiamin Chen, Jianliang Gao, Tengfei Lyu and Babatounde Moctard Oloulade are with the School of Computer Science and Engineering, Central South University, Changsha 410083, Hunan, China. E-mail:{ chenjiamin, gaojianliang, olouladem octard, tengfeilyu }@csu.edu.cn.*
- *Xiaohua Hu is with the College of Computing and Informatics, Drexel University, Philadelphia, PA 19104. E-mail: xh29@drexel.edu.*

(Corresponding Author: Jianliang Gao.)

pathway, we propose an automatic molecular structure representation Learning framework named AutoMSR. Our framework consists of two main building blocks. The first building block automatically identify the optimal GNN architecture and the best hyperparameters based on given molecular structure and property features to construct the GNN best model for getting molecular structure representation. In this module, we design an effective search space and performance evaluation strategy for the multi-label metabolic pathway prediction task. Inspired by age evolution search [27], we combine our work Auto-GNAS [26], a general parallel search graph neural network architecture framework, and propose a multi-seed age evolution search algorithm (MSAE) to search the optimal GNN architecture. Then AutoMSR adopts tree-structured parzen estimator approach [28] to get the best hyperparameter based on the optimal GNN architecture. Finally, AutoMSR uses them to construct the best GNN model and retrain AutoMSR to complete the multi-label metabolic pathway prediction task. AutoMSR can automatically design the optimal GNN model to get effective molecular structure representation for different molecule data sets, which is time-consuming and relies on expert experience. Briefly, our main contributions to this work can be summarized as follows:

- We propose a framework named AutoMSR[1] that can automatically build the optimal GNN model to get the effective molecular structure representation for the downstream task. To the best of our knowledge, this work is the first attempt to use graph neural architecture search method to achieve molecular structure representation for the multi-label metabolic pathway prediction task.
- We design a multi-seed age evolution search algorithm for obtaining the optimal GNN architecture, which uses multiple search seeds with different strengths to search the GNN architectures simultaneously based on the age evolution mechanism.
- We conduct extensive experiments based on the real molecular data set KEGG with different multi-label evaluation metrics. The results of the experiment demonstrate that AutoMSR outperforms state-of-the-art baseline methods on the multi-label pathway prediction task.

Compared to our published preliminary work [29], this work has the following important improvements: **(a).** We propose a multi-seed age evolution search algorithm to identify the optimal GNN architecture with better search performance and fewer search hyperparameters. **(b).** We use graphnorm operation [30] to each layer of GNNs, which is the effective manner to extract graph structural features for graph representation. **(c).** We train AutoMSR using the multi-label soft margin loss and evaluate multiple performances of AutoMSR using the multi-label classification evaluation metrics [31], which can better measure the performance of the AutoMSR on multi-label prediction tasks. The rest of this paper is organized as follows: Section 2 introduces the related work. Section 3 illustrates the material and preprocessing of the data set. Section 4 introduces our methodology. Section 5 presents experimental results and discussion. We conclude this work in section 6.

1. https://github.com/AutoMachine0/Auto-MSR

2 Related Work

In this section, we first briefly introduce the history of graph neural networks and the characteristics of different types of graph neural networks. Then, we introduce the multi-label metabolic pathway prediction based on the graph neural network. Finally, we present the related work of graph neural architecture search and its concept.

2.1 Graph Neural Network and Pathway Prediction

Sperduti et al. [32] first adopted neural networks on directed acyclic graphs. The concept of GNNs was initially discussed in Gori et al. [33]. Motivated by the success of CNNs in the computer vision domain, much work is concentrated on graph convolutional networks(GCNs). There are two types of GCNs, spectral-based [34] and spatial-based [35]. Graph neural network uses message passing to realize graph convolution operation, which can aggregate the features of neighbor nodes to get the representation of the central node. As the spectral-based method needs to operate on the entire graph, it is not easy to parallel and hardly scale to big graphs. However, the spatial-based method is flexible to aggregate feature information between neighbor nodes. The GNNs mentioned in this paper represent spatial-based graph convolutional neural networks.

Since the GNNs can effectively extract the graph structural features, the molecular structure graph contains important information about the molecule. Some researchers have used GNNs to extract the molecular structure features to achieve multi-label metabolic pathway prediction tasks. Baranwal $et\ al.$ [18] proposes a multi-label metabolic pathway prediction framework based on graph convolutional network (GCN) [19]. In this work, the framework first constructs a two-layer GCN to encode molecular structural features for getting the molecular structure representation. And then, concatenating molecular property features and structure representation for obtaining the final molecular representation. Finally, the framework builds a single-layer perceptron for the multi-label prediction task based on the final molecular representation. Since each layer of the graph convolutional network has only one learnable convolution kernel, it will limit the molecular structure representation based on the GCN. To overcome this problem, Yang $et\ al.$ [20] uses a two-layer graph attention network (GAT) [21] to achieve the molecular structure representation. The advantage of the GAT is to calculate different correlation coefficients between the central node and its different neighbors, which can enhance the graph representation for the downstream task. Compared with traditional machine learning, the methods based on GNNs have achieved better prediction results.

2.2 Graph Neural Architecture Search

As shown in Figure 1, the framework of graph neural architecture search contains four processes. First, building a GNN search space S involving GNN architecture subspace and hyperparameter subspace. Different architecture

components consist of GNN architecture subspace, such as attention function, aggregation function, activation function, etc. For one architecture component, it has different operators, and a combination of these operators constructs a GNN architecture s, for example for $\{const, sum, relu\}$. The hyperparameter subspace includes learning rate, mini-batch size, embedding dimension, and so on, these parameters consist of discrete values. Various parameter values constitute the GNN model training hyperparameter, for instance, $\{lr2, mb1, ed2\}$. Next, using search algorithms to sample a GNN architecture and a set of hyperparameters. After, training the GNN model built by sampled GNN architecture and hyperparameters using graph data, and producing the prediction result on the validation data set. Finally, Evaluating the prediction result based on a performance estimation strategy to generate the evaluation feedback for search algorithms iteration.

In the research field of graph neural architecture search, there are mainly two different mechanisms for designing GNN architecture search algorithms, which are based on reinforcement learning, based on evolutionary learning. GraphNAS [22] and AutoGNN [23] use reinforcement learning to design the GNN architecture search algorithm, they use LSTM as the agent to sample different GNN architectures. And the LSTM is trained based on policy gradient to maximize the expected validation accuracy of the sampled GNN architecture. GeneGNN [24], GraphPAS [25] and Auto-GNAS [26] construct the GNN architecture search algorithm based on evolution mechanism. GeneGNN proposes a search framework that can simultaneously search for GNN architecture and hyperparameters. GraphPAS and Auto-GNAS combine parallel computing with GNN architecture search algorithm for the first time, which greatly improves the efficiency of GNN architecture search process.

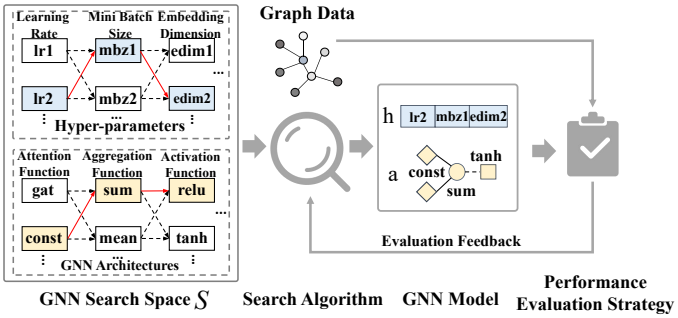

Fig. 1. The framework of graph neural architecture search. It includes four steps. The first step is to construct GNN search space S based on GNN architecture and hyperparameters candidate set. The second step is to design a search algorithm to sample GNN architecture a and hyperparameters h to build the GNN model. The third step is to use the evaluation strategy to estimate the GNN model and generate the evaluation feedback. Finally, the last step is to update the search algorithm based on the evaluation feedback for the next search epoch.

3 Material and Preprocessing

In this section, we first introduce the real metabolic pathway data set DEGG used in this work. Then, we introduce the preprocessing method for the data set in this paper.

3.1 Data Set

Among the reliable and open biological metabolic pathway data sets, One of the most widely used available data sets is the Kyoto Encyclopedia of Genes and Genome (KEGG) database [36]. KEGG consists of graphical diagrams of biochemical pathways involving metabolic pathways and some of the known regulatory pathways. To make the experiment convincing, we conduct our multi-label metabolic pathway prediction task based on the KEGG. The Simplified Molecular Input Line Entry Specification (SMILES) [37] is an important method to express structural information with a small number of characters. We use the RDKit toolkit to obtain the molecular property and structure based on the molecular SMILES representation. To focus on the ability of AutoMSR to encode molecular structures compared with manual GNN architectures, we construct a multi-label metabolic pathway data set based on the KEGG database for the multi-label prediction task. Our data set contains 4192 molecules, where the number of atoms is greater than or equal to 10 for each molecule. In the data set, 3883 molecules belong to only one metabolic pathway class, and the remaining 309 molecules belong to multiple classes. We first randomly shuffle the data set, and then the data set is divided into three parts, 80% for training, 10% for validation, and 10% for testing. As shown in Table 1, we summarized the number of molecules contained in each metabolic pathway class and the corresponding proportion in our data set.

TABLE 1
Description of The Data Set.

Labels	Metabolic Pathway Classes	Number	Ratio
0	Other Secondary Metabolites	1084	25.8%
1	Terpenoids and Polyketides	898	21.4%
2	Xenobiotics	661	15.8%
3	Lipid	562	13.4%
4	Amino Acid	416	9.9%
5	Cofactors and Vitamins	407	9.7%
6	Carbohydrate	253	6.0%
7	Nucleotide	108	2.6%
8	Other Amino Acids	103	2.5%
9	Energy	102	2.4%
10	Glycan	96	2.3%

3.2 Preprocessing

3.2.1 Molecular Property Features

As SMILES sequence is widely used in bioinformatics, which can retain original molecular structure information, we use it to represent molecule sequence based on the data set KEGG. In this paper, the molecular property feature is a 20-dimensional vector composed of two parts. Part 1 is the 13-dimensional molecular fingerprint vector, we use the molecule SMILES sequence and the RDKit toolkit to get the molecular fingerprint based on MACCS keys, which was designed by MDL Information Systems with 166 molecular characteristics [38]. Part 2 is the 7-dimensional molecular description vector generated by the molecule SMILES sequence and the RDKit toolkit.

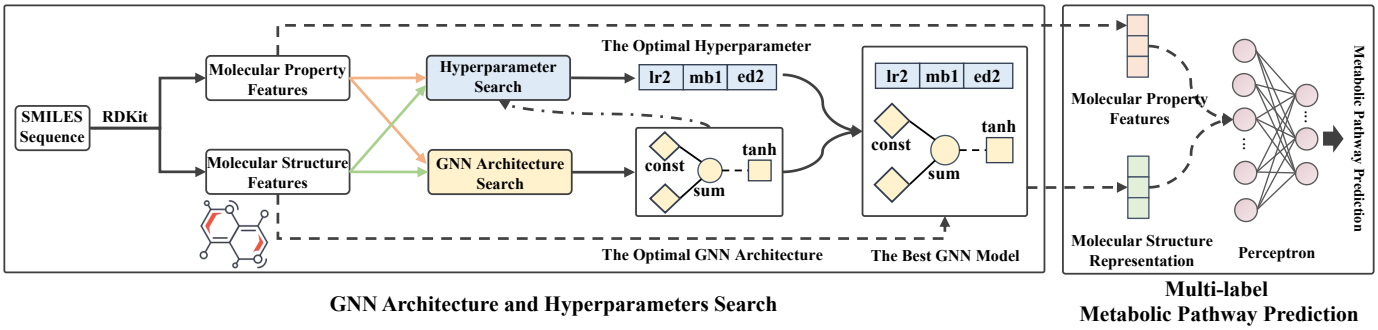

Fig. 2. The Auto-MSR framework. The multi-label metabolic pathway prediction based on AutoMSR contains two steps. Step 1 is GNN architecture and hyperparameters search, this process is to automatically identify the optimal GNN architecture and hyperparameters using molecular property and structure features based on the target task. Step 2 is multi-label metabolic pathway prediction, this process is to automatically construct the best GNN model to encode the molecular structure for getting the molecular structure representation, and the AutoMSR concatenates molecular property feature and structure representation as perceptron input for the multi-label metabolic pathway prediction task.

3.2.2 Molecular Structure Features

In order to better use the AutoMSR to encode molecular structure representation based on molecular structure information. We first use the RDKit toolkit and the molecule SMILES sequence to achieve the original molecular topology structure, in which each node represents an atom and each edge represents chemical bonds. Then we further process the molecular topology structure and use a 2-radius subgraph to represent the original node in the molecular topology structure graph. We assign a unique index to each kind of 2-radius subgraph to construct a learnable node embedding vector dictionary. Finally, we use the original molecular topology structure graph and its corresponding 2-radius subgraph embedding matrix as the molecular structure feature for AutoMSR.

4 Methodology

To better understand our proposed framework AutoMSR, we will illustrate the workflow of AutoMSR by Figure 2. The multi-label metabolic pathway prediction based on AutoMSR contains two steps. Step 1 is to automatically identify the optimal GNN architecture and hyperparameters based on the multi-label metabolic pathway prediction task. Step 2 is to automatically retrain the AutoMSR based on the best GNN model using molecular structure and property features, the best GNN model will encode the molecular structure to get the molecular structure representation. Finally, AutoMSR concatenates molecular property feature and structure representation as perceptron input for the multi-label metabolic pathway prediction task.

4.1 GNN Architecture And Hyperparameters Search

In this section, we first describe problem formulation for achieving the optimal GNN model. And then, we present the design of the GNN search space in this work. Finally, we introduce the detailed progress of GNN architecture and hyperparameter search algorithms.

4.1.1 Problem Formulation

Given a GNN search space S involving GNN architecture components and hyperparameters, a graph data set G including a training set G_t and validation set G_v, a performance evaluation strategy E. The target firstly is to identify the optimal graph neural architecture A_{opt} from the GNN architecture subspace A based on fixed hyperparameters, where the GNN model m is constructed by A and trained on set G_t and gets the best performance on set G_v. And then, taking the same way to find the best hyperparameters H_{opt} from the hyperparameter subspace H based on the A_{opt}. The mathematical expression is as follows:

$$\begin{aligned} A_{opt} &= \arg\max_{A \in S} E(m(A, G_t), G_v) \\ H_{opt} &= \arg\max_{H \in S} E(m(H, A_{opt}, G_t), G_v) \end{aligned} \tag{1}$$

4.1.2 GNN Search Space

The GNN search space consists of two subspaces, namely the GNN architecture subspace and the hyperparameter subspace. We build our architecture subspace based on the typical GNN structure space [25], which is composed of five GNN architecture components.

- **Attention Function (Att).** Attention coefficient a_{ij} is generated by an attention function that relies on the features of the central node and its neighbor nodes. Table 2 shows the operators of attention functions in this work.

TABLE 2
Attention Functions of GNN Architecture Subspace.

Attention Type	Function
$gene-linear$	$a_{ij}^{gen} = W_b * tanh(W_c * h_i + W_n * h_j)$
gat	$a_{ij}^{gat} = reaky\_relu(W_c * h_i + W_n * h_j)$
cos	$a_{ij}^{cos} =< W_c * h_i, W_n * h_j >$
$linear$	$a_{ij}^{lin} = tanh(sum(W_c * h_i))$
$sym-gat$	$a_{ij}^{sym} = r_{ij}^{gat} + r_{ji}^{gat}$
gcn	$a_{ij}^{gcn} = 1/\sqrt{d_i d_j}$
$const$	$a_{ij}^{const} = 1$

- **Aggregation Function (Agg).** The neighbor nodes' features are merged by aggregation function with the corresponding attention coefficient a_{ij} for a better

representation of the central node. This process is crucial for the node to learn the characteristics of the graph structure. We design three aggregation functions, sum, $mean$, max in GNN architecture subspace.

- **Attention-head (Head).** To stabilize the learning process, computing multiple independent attention coefficients for the attention operator is practical to achieve this target. The set of attention-head is $\{1, 2\}$ in this paper.
- **Hidden Dimension (Dim).** Reducing and transforming the dimension of the original feature is an effective way to enhance the hidden representation by a learnable matrix W. In this work, the selection of hidden dimensions includes 8, 16, 32, and 64.
- **Activation Function (Act).** The activation function makes the model have a nonlinear fitting ability, and it is important to smooth the hidden representation. In this work, the activation functions are listed as follows: $\{tanh, sigmoid, relu, linear, softplus, leaky\_relu, relu6, elu\}$.

We construct a hyperparameter subspace involving four different parameters, which can improve the GNN model performance built by the optimal GNN architecture.

- **Learning Rate (LR).** This parameter determines the speed of model learning based on the gradient descent optimization algorithm. The candidate set of learning rate in this work is $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$
- L_2 **Regularization Strength (RS).** The L_2 regularization strength controls the constraint of L_2 regularization on the loss function, which can effectively prevent the model from overfitting. We use four different values involving 0, 10^{-3}, 10^{-4}, 10^{-5} to construct the choice of regularization strength
- **Mini Batch Size (BS).** Dividing the training set into different mini-batches for model training is an effective way to achieve model training on a large-scale dataset. The set of mini batch size is $\{5, 10, 15, 20, 25, 30\}$ in this work.
- **Embedding Dimension (EDim).** In this paper, different 2-radius subgraph node representations construct the learnable embedding dictionary. And the embedding dimension of this dictionary is essential for AutoMSR to achieve superior performance in multi-label metabolic pathway prediction tasks. The choice of embedding dimensions in hyperparameter subspace includes $8, 16, 32, 64, 128, 256$.

4.1.3 GNN Architecture Search

The GNN layers are fixed to two, and the GNN architecture is designed as a stack, each GNN layer has six different architecture components. we use graphnorm operation [30] for each GNN layer. GraphNorm can effectively suppress the noise existing in the molecular graph and enhance the representation of the molecular graph. At the same time, this operation can speed up the convergence process of GNN model training. The size of the GNN architecture subspace and the hyperparameter subspace are 1344^2 and 720 respectively in this work. If we search for GNN architecture and hyperparameters at the same time, the scale of the search is $1344^2 \times 720$ and is greater than 10^9. It will result in too much time overhead to get a satisfactory GNN model. Taking into account the efficiency factor, we identify the optimal GNN architecture and hyperparameters based on two efficient and independent search processes. We propose a multi-seed age evolution (MSAE) search algorithm to obtain the optimal GNN architecture in the GNN architecture subspace on the default hyperparameters.

Compared with GraphPAS [25] used in preliminary work, MSAE has the following improvements: **(a).** It is simpler in design and has fewer search parameters to tune. **(b).** It has better search performance to get the best GNN architecture in the multi-label metabolic pathway prediction task. MSAE constructs multiple search seeds with different mutation strengths to search GNN architectures in parallel based on the age evolution mechanism. MSAE includes the following four processes:

- **Population Initialization.** Randomly generating K GNN architectures as the population $popK$. Then, the GNN architectures in $popK$ are evaluated in parallel, and the evaluation feedback is obtained as the fitness. Finally, using a list to maintain different GNN architectures and their corresponding fitness in the population $popK$. In the GNN architecture search process, we select the recall of the multi-label classification as the evaluation strategy to produce the evaluation feedback.
- **Parent Selecting.** Randomly selecting N GNN architectures from the population $popK$ as the parent P for producing new child GNN architectures. This process is called the exploitation step in the search algorithm, and it can provide search direction based on the information of the existing search results.
- **Multi-seed Mutation.** Using M mutation searchers with different mutation intensities to generate new GNN architectures based on parent P, and getting the fitness by evaluation strategy. In this process, the parent P will sort the GNN architectures based on the fitness and then distribute the top M GNN architectures to different mutation searchers. After, each mutation searcher uses different mutation intensities, which m architecture components will mutate randomly for one GNN architecture, to generate M new child architectures.
- **Age Evolution Updating.** For the traditional evolution updating process, the population will add child individuals if the fitness of the child individual is better than the worst individual in the population, and the population will delete the worst individual. In order to explore the search space more efficiently, the age evolution updating process will remove the oldest individuals in the population instead of the worst individuals. In each search epoch, the $popK$ list will add the M newest child architectures to the far left and delete the M GNN architectures on the far right. All GNN architectures generated during the search process will be maintained using a history

list. Algorithm 1 presents the MSAE search algorithm logic in detail.

4.1.4 Hyperparameters Search

The choice of hyperparameters plays a crucial role in getting a good model performance. The model with appropriate hyperparameters can often get excellent effects. On the contrary, the model performance will not reach expectations. To get better molecular structure representation, we design an effective hyperparameter subspace based on the multi-label metabolic pathway prediction task. We use the tree-structured parzen estimator (TPE) algorithm [28], which is one of the most practical hyperparameter search algorithms, to identify the optimal hyperparameters based on the optimal GNN structure. Since the TPE algorithm has many open-source and reliable implementation libraries, we choose the Hyperopt library [39] to implement hyperparameters search from the hyperparameter subspace we designed. In the hyperparameters search process, we select the precision of the multi-label classification as the evaluation strategy to generate the evaluation feedback.

Algorithm 1 MSAE search algorithm

Input:
GNN architecture subspace A, validation set G_v, training set G_t, search epoch j, population size K, parent size N, number of mutation searchers M, mutation intensity m.

Output:
The optimal GNN architecture A_{opt}.

// initialization
$pop_K \leftarrow random\ initialization(K, A)$
$fitness_K \leftarrow evaluation(pop_K, G_t, G_v)$
$history_g.append(pop_K)$
$history_f.append(fitness_K)$
// multi-seed searching
for $i \leftarrow 0$ to j **do**
  $parent_P$, $fitness_P \leftarrow select(N, pop_K, fitness_K)$
  $children \leftarrow select\_top(M, parent_P, fitness_P)$
  $children \leftarrow mutation(m, children)$
  $fitness_c \leftarrow evaluation(children, G_t, G_v)$
  $history_g.append(children)$
  $history_f.append(fitness_c)$
  // age evolution updating
  for $child$, $fitness \leftarrow children$,$fitness_c$ **do**
    $pop_K.insert(0, child)$
    $fitness_K.insert(0, fitness)$
  end for
  for $i \leftarrow 0$ to M **do**
    $pop_K.pop()$
    $fitness_K.pop()$
  end for
end for
// architecture deriving
Select top T GNN architectures from $history_g$ based on $history_f$.
Re-train the T GNN architectures for R times to get the average evaluation performance.
Select the best A_{opt} from the T GNN architectures based on average evaluation performance.

4.2 Multi-label Metabolic Pathway Prediction

In this step, AutoMSR generates the molecular structure representation vector X_s based on the best GNN model produced in the first step and achieves the molecular property feature vector X_p with the SMILES sequence and the RDKit. Then, concatenating these two feature vectors into $X_m = [X_s, X_p]$ as the final molecular representation. Next, AutoMSR feeds the molecular representation X_m into a single-layer perceptron and gets the final prediction vector Y as follows equation (2). Later, computing the multi-label loss values based on multi-label soft margin loss. Finally, AutoMSR uses the $Adam$ to optimize the trainable parameters of the entire system. In the model testing phase, the prediction value will be set at 1 if it is greater than the threshold of 0.5 otherwise set to 0. The metabolic pathway prediction task is automatically completed by the second part of AutoMSR.

$$Y = sigmoid(W \cdot X_m + b) \tag{2}$$

where W and b are the learnable matrices and bias of single-layer perceptron, and $sigmoid$ is the activation function. Later, Computing the multi-label loss values based on multi-label soft margin loss and prediction vector Y as follows equation (3):

$$\begin{aligned} L = -\frac{1}{C}\sum_{i=1}^{n}[y_i \cdot log(\frac{1}{1+exp(-\hat{y}_i)}) \\ + (1-y_i) \cdot log(\frac{exp(-\hat{y}_i)}{1+exp(-\hat{y}_i)})] \end{aligned} \tag{3}$$

Where C is the number of labels, n is the number of training samples, $\hat{y}_i$ is the prediction vector for one sample, y_i is the corresponding label vector of $\hat{y}_i$. subsequently, AutoMSR uses the $Adam$ to optimize the whole system based on loss values L. In the model testing phase, the $\hat{y}_i$ will be set to 1 if it is greater than the threshold of 0.5 otherwise set to 0.

5 Experiments and Discussion

In this section, we firstly introduce the experimental settings involving experiment configuration, framework implementation, and evaluation metrics. Then, we will describe the baseline methods. Finally, we compare AutoMSR with baseline approaches, and then we discuss the ablation experiment and the search parameter sensitivity experiment of AutoMSR.

5.1 Experimental Settings

The experimental configuration consists of two parts. Part one is the search configuration including GNN architecture and hyperparameters search configuration. Part two is the GNN model training configuration.

- **Search Configuration.** For the search algorithm MSAE, the number of mutation searchers M is 4, the scale of population size $popK$ is 100, the parents P are 40, and the mutation strength for each mutation searcher is the same value 1, the GNN architecture

search epoch is 475. For the initial hyperparameters in the process of GNN architecture search, the learning rate is 0.001, the L_2 Regularization strength is 0.0005, the embedding dimension is 70, and the mini-batch size is 20. In the hyperparameters search process, the search epoch is 500.

- **GNN Training Configuration.** To improve the efficiency of AutoMSR to identify the optimal GNN architecture and hyperparameters, we set the GNN model training epoch as 5 in the search process. Though the GNN model training epoch is small in the search process, the evaluation performance is enough to distinguish whether the GNN model designed by AutoMSR is promising for this task. We set the GNN model training epoch as 100 to ensure the convergence of test results during the testing process.

The optimal GNN architecture and hyperparameters designed by AutoMSR for multi-label metabolic pathway prediction task as shown in Figure 3.

	Att	Agg	Head	Dim	Norm	Act
First GNN Layer:	const,	sum,	2,	64,	graphnorm,	tanh
	Att	**Agg**	**Head**	**Dim**	**Norm**	**Act**
Second GNN Layer:	cos,	mean,	1,	64,	graphnorm,	softplus

Hyperparamters: **RL** = 0.01 **RS** = 0 **BS** = 15 **EDim** = 256

Fig. 3. An example of the optimal GNN architecture and hyperparameters designed by AutoMSR (MSAE) for the multi-label metabolic pathway prediction task.

5.2 Framework Implement

We use Auto-GNAS[2] [26] parallel search interface to realize the parallel search capability of the MSAE search algorithm for AutoMSR. We implement the Graphnorm operation based on PYG [3] [40] for AutoMSR, We design the hyperparameter search module of AutoMSR using Hyperopt[4] [39].

5.3 Evaluation Metrics

In order to better evaluate the multi-label prediction task, we use four kinds of multi-label classification metrics, which are widely used in multi-label metabolic pathway prediction task [31]. The four metrics are the average multi-label classification $Accuracy$, $Precision$, $Recall$ and $F1\ score$ respectively. they are defined as follows:

$$\begin{aligned} Accuracy &= \frac{1}{m}\sum_{i=1}^{m}\frac{|y_i \cap \hat{y}_i|}{|y_i \cup \hat{y}_i|} \\ Precision &= \frac{1}{m}\sum_{i=1}^{m}\frac{|y_i \cap \hat{y}_i|}{|\hat{y}_i|} \\ Recall &= \frac{1}{m}\sum_{i=1}^{m}\frac{|y_i \cap \hat{y}_i|}{|y_i|} \\ F1\ score &= \frac{1}{m}\sum_{i=1}^{m}\frac{2|y_i \cap \hat{y}_i|}{|y_i| + |\hat{y}_i|} \end{aligned} \tag{4}$$

2. https://github.com/AutoMachine0/Auto-GNAS
3. https://pytorch-geometric.readthedocs.io/en/latest/
4. https://github.com/hyperopt/hyperopt

Where m is the number of samples, $\hat{y}_i$ is the prediction vector for one sample, y_i is the corresponding label vector of $\hat{y}_i$.

5.4 Baseline Methods

In order to demonstrate the superiority of AutoMSR in the multi-label metabolic pathway prediction task, we take the following actions to our experiment. First, we use a multi-label soft margin loss function to retrain the state-of-art manual graph neural networks and the optimal GNN model which is sampled by our published preliminary work. Second, we use four multi-label classification metrics mentioned in equation 4 to reevaluate performance.

- **GCN.** Baranwal $et\ al.$ [18] uses the graph convolutional network (GCN) to encode the molecular structure features for getting the molecular structure representation. And then, using a perceptron to predict the metabolic pathway classes based on molecular property features and structure representation. The hyperparameters of this framework are set as follows: the number of GNN layers is 2, the learning rate is 0.001, the embedding dimension is 70, the mini-batch size is 10, and the training epoch is 100.
- **GAT.** To overcome the shortcoming that GCN has only one learnable convolution kernel to learn the structure and semantics features of the graph data. Yang $et\ al.$ [20] constructs a two-layer graph attention network (GAT) to achieve the molecular structure representation. We reproduce this method based on the following hyperparameters: the learning rate is 0.001, the embedding dimension is 70, the mini-batch size is 20, and the training epoch is 100.
- **AutoMSR (GraphPAS).** We use the multi-label soft margin loss function to retrain the optimal GNN model sampled by our preliminary work [29] and use the four multi-label classification metrics to reevaluate it. The optimal GNN architecture and hyperparameters are as shown in Figure 4.

	Att	Agg	Head	Dim	Norm	Act
First GNN Layer:	linear,	sum,	2,	16,	none,	relu6
	Att	**Agg**	**Head**	**Dim**	**Norm**	**Act**
Second GNN Layer:	const,	sum,	1,	32,	none,	sigmoid

Hyperparamters: **RL** = 0.0001 **RS** = 0.0001 **BS** = 5 **EDim** = 128

Fig. 4. The optimal GNN architecture and hyperparameters designed by AutoMSR (GraphPAS).

5.5 Comparison Results

To distinguish our previous work, the method of the previous work is called AutoMSR (GraphPAS), and the method of this work is called AutoMSR (MSAE). For evaluating model performance more reliably, we run the multi-label metabolic pathway prediction experiments 10 times and compute the average performance for each method.

TABLE 3
Performance of AutoMSR Against Comparative Approaches (multi-label classification metrics)

Methods	Accuracy	Precision	Recall	F1 Score
GCN	77.94%±0.90%	80.44%±0.89%	79.70%±0.11%	79.31%±0.95%
GAT	79.70%±0.61%	81.97%±0.53%	81.72%±0.87%	81.08%±0.66%
AutoMSR (GraphPAS)	81.07%±0.83%	83.91%±0.98%	82.75%±0.71%	82.52%±0.83%
AutoMSR (MSAE)	**84.32%±0.81%**	**86.88%±0.88%**	**87.58%±0.77%**	**86.19%±0.75%**

TABLE 4
Ablation Experiment of AutoMSR (multi-label classification metrics)

Methods	Accuracy	Precision	Recall	F1 Score
AutoMSR (w/o n&p)	80.02%±0.96%	82.48%±1.02%	82.04%±0.96%	81.44%±0.95%
AutoMSR (w/o n)	80.44%±1.34%	82.88%±1.54%	82.73%±1.51%	81.95%±1.45%
AutoMSR (w/o p)	84.02%±1.11%	86.62%±1.11%	86.65%±0.97%	85.66%±1.03%
AutoMSR (all)	**84.32%±0.81%**	**86.88%±0.88%**	**87.58%±0.77%**	**86.19%±0.75%**

5.5.1 Performance Experiment

The performance of AutoMSR (MSAE) and other existing manual baseline models are shown in Table 3. As shown in Table 3, our framework achieves the best performance in the multi-label metabolic pathway prediction task on the real-world data set KEGG. Compared to all the baselines, our framework outperforms baseline methods by a significant margin and achieves state-of-the-art performance in all evaluation metrics including $Accuracy$, $Precision$, $Recall$ and $F1\ score$. Specifically, AutoMSR (MSAE) outperforms GAT by 4.62% on $Accuracy$, 4.91% on $Precision$, 5.86% on $Recall$, and 5.11% on $F1\ score$. For handcrafted GNN models, we need to design the GNN architecture and fine-tune the hyperparameters of the GNN model in a manual manner for a given biological graph data set, which is a time-consuming task that relies on expert experience. Moreover, it is difficult for non-professionals to obtain a model that meets expectations based on a manual way in the field of bioinformatics. On the contrary, the AutoMSR (MSAE) can automatically sample the GNN architecture and the hyperparameters that can make the model perform better based on the downstream task. This is an automatic optimization process with litter human intervention, and it can automatically construct the optimal GNN model for different biological data sets.

There are three reasons why the performance of AutoMSR (MSAE) is better than that of AutoMSR (GraphPAS) as follows: **(a).** Compared with the traditional GNN search space [22] [23] [24], the GNN search space we designed based on the multi-label metabolic pathway task is smaller in scale. AutoMSR (GraphPAS) uses information entropy to constrain the search direction, which may lead to premature convergence of the search process in the small-scale GNN search space, and it is detrimental for finding better GNN architecture. However, AutoMSR (MSAE) uses the age evolution mechanism to update the population, which makes the search process more exploratory. Moreover, AutoMSR (MSAE) uses multiple mutation searchers with different mutation strengths to search simultaneously. this search process can enhance the diversity of the population, which is advantageous to find better-performing GNN architectures in the small-scale GNN search space. **(b).** We use graphnorm operation for each GNN layer, it can effectively suppress the noise existing in the molecular graph and enhance the representation of the molecular graph. At the same time, this operation can speed up the convergence process of GNN model training. This operation is helpful to get better performance for the GNN model sampled by AutoMSR (MSAE). **(c).** Different from the same evaluation strategy in the two search processes of AutoMSR (GraphPAS), we use different evaluation metrics as the evaluation strategy in two processes. In GNN architecture search process, we use the $Recall$ metric as the evaluation strategy to produce the evaluation feedback for MSAE algorithm updating. In the hyperparameters search process, we use the $Precision$ metric as the evaluation strategy to produce the evaluation feedback for TPE algorithm updating. This is a bidirectional optimization strategy that can effectively improve the performance of the optimal GNN model designed by AutoMSR (MSAE).

5.5.2 Ablation Experiment

To explore the impact of different modules on the performance of the optimal GNN model designed by AutoMSR, we conduct the ablation study. The AutoMSR (w/o n&p) represents that we delete the graphnorm operation and the input of molecular property features. The AutoMSR (w/o n) denotes that we delete the graphnorm operation. The AutoMSR (w/o p) stands for deleting the input of molecular property features. The AutoMSR(all) indicates that all modules are reserved for the optimal GNN model. As shown in Table 3, the variant of AutoMSR (w/o p) is better than AutoMSR (w/o n) and AutoMSR (w/o n&p) in all performances, it shows that graphnorm operation contributes significantly to improving the performance of GNN models. Because graphnorm operation can effectively suppress the noise existing in the molecular graph and enhance the representation of the molecular graph. At the same time, this operation can speed up the convergence process of GNN model training. Comparing the performance of AutoMSR (all) and AutoMSR (w/o p), we can find their results are close. It shows that the molecular property

features have limited influence on the performance, and proves that the GNN model designed by AutoMSR can reduce the dependence of molecular property features for getting excellent performance in the multi-label metabolic pathway task. This is advantageous to researchers with limited expertise in the field of bioinformatics. In short, the GNN model designed by AutoMSR can only obtain sufficient efficient features from molecular structure features to complete downstream tasks.

5.5.3 Search Parameter Sensitivity Experiment

To study the influence of search parameters on the performance of AutoMSR to obtain the GNN architectures, we design a search parameter sensitivity experiment. The result is shown in Figure 5. The x-axis marks the search epoch, and the y-axis represents the average recall on the validation data set of the top 10 GNN architectures in the history list. In the evaluation step of GNN architecture, the training epoch of the GNN model is 5. In the experiment, we set the size of the population $popK$ as 100, and the search epoch is 100.

In Figure 5 (a), we discuss the impact of the number of mutation searchers and mutation strength to the AutoMSR performance. In the experiment, the size of the parent is 10 for each curve. The $ms = [1]$ represents one mutation searcher with the slight mutation strength 1, $ms = [1, 1, 1, 1]$ stands for four mutation searchers with the same slight mutation strength 1, $ms = [1, 2, 3, 4]$ indicates four mutation searchers with the different mutation strengths 1, 2, 3, and 4 respectively, $ms = [5, 5, 5, 5]$ represents four mutation searchers with the same dramatic mutation strength 5. The curve of $ms = [1, 1, 1, 1]$ represents that increasing the number of mutation searches can effectively improve the performance of AutoMSR. Because multiple mutation searchers can enhance the AutoMSR exploration rate and the diversity of the population, it is beneficial for searching for better GNN architectures in the GNN search space. The curve of $ms = [5, 5, 5, 5]$ shows that dramatic mutation strength will hurt the stability of the mutation searcher for getting better GNN architecture in the search process. The unstable search process is detrimental for AutoMSR to find better GNN architecture in the small GNN search space.

We analyze the influence of the size of the population on AutoMSR performance as shown in subfigure 5 (b). We use four mutation searchers with different mutation strengths 1, 2, 3, and 4 respectively. The $ps = 5$ represents the size of the parent is 5 in the search process. We can observe that increasing the size of the parent is beneficial to the performance of AutoMSR in the search process. Because increasing the scale of parents can increase the parent diversity for the mutation searchers to explore. However, when the parameter of mutation searchers remains unchanged, the performance of AutoMSR will not improve when the size of the parent increases to a certain threshold. The reason is that increasing the diversity of parents will also introduce more GNN structure individuals with poor performance into the system, which will limit the search efficiency of mutation searchers.

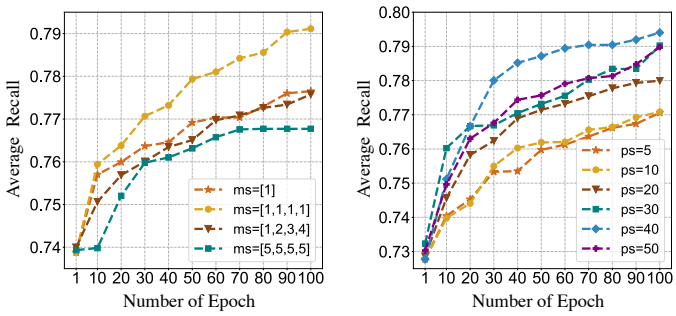

Fig. 5. The search performance of AutoMSR with different mutation searchers and the size of the parent. In Figure (a), the $ms = [1]$ represents one mutation searcher with the slight mutation strength 1, $ms = [1, 1, 1, 1]$ stands for four mutation searchers with the same slight mutation strength 1. In Figure (b), $ps = 5$ indicates the size of the parent is 5.

6 Conclusions

In this work, we proposed an efficient multi-seed age evolution search algorithm to search for the optimal GNN architecture. It is simpler in design and has fewer search hyperparameters to tune, and it has better search performance to get better GNN architecture in a small GNN search space. We add grahpnorm for each GNN layer, which can effectively suppress the noise existing in the molecular graph and enhance the representation of the molecular graph. At the same time, this operation can speed up the convergence process of GNN model training. We use the bidirectional optimization evaluation strategy for the GNN architecture and hyperparameters search process. It is beneficial to improve the performance of AutoMSR. In order to better solve the problem of multi-label prediction, we use multi-label soft margin loss as the loss function to train AutoMSR and use a variety of multi-label classification metrics for system evaluation. Experimental results based on datasets KEGG demonstrate that our proposed framework can obtain better performance than baseline methods.

References

[1] Ayoun Cho, Hongseok Yun, Jin Hwan Park, Sang Yup Lee, and Sunwon Park. Prediction of novel synthetic pathways for the production of desired chemicals. *BMC Systems Biology*, 4(1):1–16, 2010.

[2] Zhoupeng Zhang and Wei Tang. Drug metabolism in drug discovery and development. *Acta Pharmaceutica Sinica B*, 8(5):721–732, 2018.

[3] Bertil B Fredholm, William W Fleming, Paul M Vanhoutte, and Théophile Godfraind. The role of pharmacology in drug discovery. *Nature Reviews Drug Discovery*, 1(3):237–238, 2002.

[4] Mai A Hamdalla, Sanguthevar Rajasekaran, David F Grant, and Ion I Mandoiu. Metabolic pathway predictions for metabolomics: a molecular structure matching approach. *Journal of chemical information and modeling*, 55(3):709–718, 2015.

[5] Hiroyuki Kuwahara, Meshari Alazmi, Xuefeng Cui, and Xin Gao. Mre: a web tool to suggest foreign enzymes for the biosynthesis pathway design with competing endogenous reactions in mind. *Nucleic acids research*, 44(s1):W217–W225, 2016.

[6] Yuki Moriya, Daichi Shigemizu, Masahiro Hattori, Toshiaki Tokimatsu, Masaaki Kotera, Susumu Goto, and Minoru Kanehisa. Pathpred: an enzyme-catalyzed metabolic pathway prediction server. *Nucleic acids research*, 38(s2):W138–W143, 2010.

[7] Peter D Karp, Suzanne M Paley, Markus Krummenacker, Mario Latendresse, Joseph M Dale, Thomas J Lee, Pallavi Kaipa, Fred Gilham, Aaron Spaulding, Liviu Popescu, et al. Pathway tools version 13.0: integrated software for pathway/genome informatics and systems biology. *Briefings in bioinformatics*, 11(1):40–79, 2010.

[8] Lynda BM Ellis, Junfeng Gao, Kathrin Fenner, and Lawrence P Wackett. The university of minnesota pathway prediction system: predicting metabolic logic. *Nucleic acids research*, 36(s2):W427–W432, 2008.

[9] Minoru Kanehisa, Susumu Goto, Masahiro Hattori, Kiyoko F Aoki-Kinoshita, Masumi Itoh, Shuichi Kawashima, Toshiaki Katayama, Michihiro Araki, and Mika Hirakawa. From genomics to chemical genomics: new developments in kegg. *Nucleic acids research*, 34(s1):D354–D357, 2006.

[10] Kouta Toshimoto, Naomi Wakayama, Makiko Kusama, Kazuya Maeda, Yuichi Sugiyama, and Yutaka Akiyama. In silico prediction of major drug clearance pathways by support vector machines with feature-selected descriptors. *Drug Metabolism and Disposition*, 42(11):1811–1819, 2014.

[11] Yu-Dong Cai, Ziliang Qian, Lin Lu, Kai-Yan Feng, Xin Meng, Bing Niu, Guo-Dong Zhao, and Wen-Cong Lu. Prediction of compounds' biological function (metabolic pathways) based on functional group composition. *Molecular diversity*, 12(2):131–137, 2008.

[12] Michelle L Green and Peter D Karp. A bayesian method for identifying missing enzymes in predicted metabolic pathway databases. *BMC bioinformatics*, 5(1):1–16, 2004.

[13] Wray Buntine. Learning classification trees. *Statistics and computing*, 2(2):63–73, 1992.

[14] Zhe Quan, Yan Guo, Xuan Lin, Zhi-Jie Wang, and Xiangxiang Zeng. Graphcpi: Graph neural representation learning for compound-protein interaction. In *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine*, pages 717–722, 2019.

[15] Lvxing Zhu, Zhaolin Hong, and Haoran Zheng. Predicting gene-disease associations via graph embedding and graph convolutional networks. In *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine*, pages 382–389, 2019.

[16] Xin Chen, Xien Liu, and Ji Wu. Drug-drug interaction prediction with graph representation learning. In *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine*, pages 354–361, 2019.

[17] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of International Conference on Machine Learning*, pages 1263–1272, 2017.

[18] Mayank Baranwal, Abram Magner, Paolo Elvati, Jacob Saldinger, Angela Violi, and Alfred O Hero. A deep learning architecture for metabolic pathway prediction. *Bioinformatics*, 36(8):2547–2553, 2020.

[19] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of International Conference on Learning Representations*, pages 1–12, 2017.

[20] Zhihui Yang, Juan Liu, Zeyu Wang, Yufan Wang, and Jing Feng. Multi-class metabolic pathway prediction by graph attention-based deep learning method. In *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine*, pages 126–131, 2020.

[21] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *Proceedings of International Conference on Learning Representations*, pages 1–12, 2018.

[22] Yang Gao, Hong Yang, Peng Zhang, Chuan Zhou, and Yue Hu. Graphnas: Graph neural architecture search with reinforcement learning. *arXiv preprint arXiv:1904.09981*, 2019.

[23] Kaixiong Zhou, Qingquan Song, Xiao Huang, and Xia Hu. Auto-gnn: Neural architecture search of graph neural networks. *arXiv preprint arXiv:1909.03184*, 2019.

[24] Min Shi, David A Wilson, Xingquan Zhu, Yu Huang, Yuan Zhuang, Jianxun Liu, and Yufei Tang. Evolutionary architecture search for graph neural networks. *arXiv preprint arXiv:2009.10199*, 2020.

[25] Jiamin Chen, Jianliang Gao, Yibo Chen, Moctard Babatounde Oloulade, Tengfei Lyu, and Zhao Li. Graphpas: Parallel architecture search for graph neural networks. In *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2182–2186, 2021.

[26] Jiamin Chen, Jianliang Gao, Yibo Chen, Babatounde MOCTARD Oloulade, Tengfei Lyu, and Zhao Li. Auto-gnas: A parallel graph neural architecture search framework. *IEEE Transactions on Parallel and Distributed Systems*, pages 1–12, 2022.

[27] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of AAAI Conference on Artificial Intelligence*, pages 4780–4789, 2019.

[28] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Proceedings of International Conference on Neural Information Processing Systems*, pages 2546–2554, 2011.

[29] Jiamin Chen, Jianliang Gao, Tengfei Lyu, Babatounde Moctard Oloulade, and Xiaohua Hu. Multi-label metabolic pathway prediction with auto molecular structure representation learning. In *Proceedings of IEEE International Conference on Bioinformatics and Biomedicine*, pages 171–176, 2021.

[30] Tianle Cai, Shengjie Luo, Keyulu Xu, Di He, Tie-yan Liu, and Liwei Wang. Graphnorm: A principled approach to accelerating graph neural network training. In *Proceedings of International Conference on Machine Learning*, pages 1204–1215, 2021.

[31] Shantanu Godbole and Sunita Sarawagi. Discriminative methods for multi-labeled classification. In *Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 22–30, 2004.

[32] Alessandro Sperduti and Antonina Starita. Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, 8(3):714–735, 1997.

[33] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *Proceedings of IEEE International Joint Conference on Neural Networks*, pages 729–734, 2005.

[34] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.

[35] David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional networks on graphs for learning molecular fingerprints. *arXiv preprint arXiv:1509.09292*, 2015.

[36] Minoru Kanehisa and Susumu Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30, 2000.

[37] David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.

[38] Eli Fernández-de Gortari, César R García-Jacas, Karina Martinez-Mayorga, and José L Medina-Franco. Database fingerprint (dfp): an approach to represent molecular databases. *Journal of cheminformatics*, 9(1):1–9, 2017.

[39] James Bergstra, Brent Komer, Chris Eliasmith, Dan Yamins, and David D Cox. Hyperopt: a python library for model selection and hyperparameter optimization. *Computational Science & Discovery*, 8(1):014008, 2015.

[40] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *Proceedings of International Conference on Learning Representations Workshop*, 2019.

Jiamin Chen received the B.S. degree in applied physics from Nanchang University, Nanchang, Jiangxi, China in 2014, and the M.S. degree in radio physics from Nanchang University, Nanchang, Jiangxi, China in 2017. He is currently a PhD candidate in the School of Computer Science and Engineering at Central South University, Changsha, Hunan, China. His research interests are automatic machine learning and graph neural networks.

Jianliang Gao received the PhD degree from the Institute of Computing Technology (ICT), Chinese Academy of Sciences, China. He is currently a professor with the School of Computer Science and Engineering, Central South University, China. He is the general chair of the 2016 IEEE Conference on Big Data. His main research interests include machine learning and graph data mining.

Tengfei Lyu is a master student in the School of Computer Science and Engineering, Central South University, Changsha, Hunan, China. He received a B.S. degree from Bohai University, Jinzhou, Liaoning, China, in 2019. His research interests are machine learning and graph neural networks.

Babatounde Moctard Oloulade received a B.S. degree in computer science from the National School of Applied Economics and Management, University of Abomey-Calavi, Cotonou, Benin in 2012. He received his M.S. degree in Computer Science and Technology from Wuhan University of Technology, Wuhan, China in 2020. He is currently a Ph.D. student at the School of Computer Science and Engineering, Central South University, Changsha, China. His current research interests include big data and graph mining.

Xiaohua Hu Xiaohua Hu is a full professor and the founding director of the Data Mining and Bioinformatics Lab at the College of Computing and Informatics, Drexel University. He is also serving as the founding co-director of the US National Science Foundation (NSF) Center (I/U CRC) on Visual and Decision Informatics (NSF CVDI), IEEE Computer Society Bioinformatics and Biomedicine Steering Committee Chair, and IEEE Computer Society Big Data Steering Committee Chair.