Cold Start Problem in Citation Link Prediction

Yi Chen[1,*]

[1]College of Information Engineering, Nanjing University of Finance and Economics, Nanjing, China
[*]Corresponding author: 995730767@qq.com

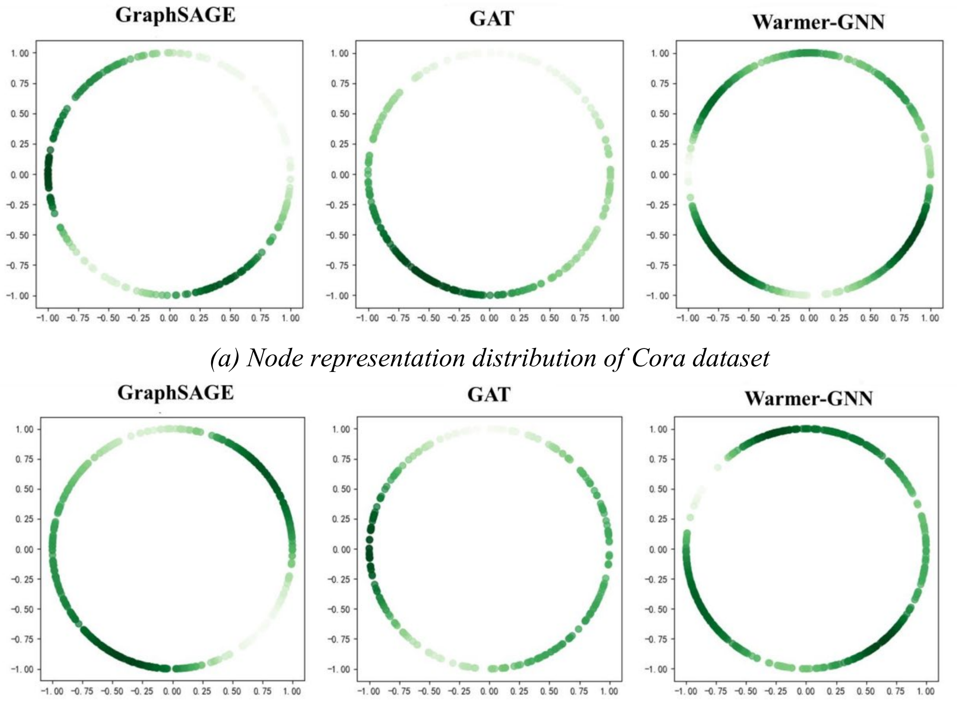
Abstract: *The citation network is a complex network that describes the scientific research results of scholars, the development of scientific fields, and the relationships between various academic fields. In the exploration and analysis of citation networks in actual scenarios, the data in the citation network is often incomplete or contains noise. Therefore, the purpose is to determine and predict whether there is an interaction or relationship between two documents in the citation network. The citation link prediction task has very significant research value. Among them, the link prediction method based on graph neural network has developed rapidly due to its excellent graph structure learning ability. However, there are problems in traditional citation link prediction methods based on graph neural networks that cannot efficiently utilize document attribute features and are similar to the cold start problem in recommendation systems. Aiming at these problems that arise in current graph neural network methods, this paper proposes a graph neural network link prediction model Warmer-GNN based on literature semantic information enhancement (Warmer refers to the problem of solving the cold start problem, and GNN refers to the graph neural network) . The model first establishes the document attribute feature relationship graph and the document citation relationship graph from the perspective of semantic information enhancement, then obtains the information of nodes in the feature map and citation map through the self-attention mechanism, and finally uses a mixture of positive and negative samples to target the feature map Perform negative sampling optimization.*

Keywords: *Citation Network; Graph Neural Network; Link Prediction; Attention Mechanism*

1. Introduction

A citation network is a type of social network in which each vertex represents a document and the edges represent citation relationships between documents. Citation link prediction refers to researchers using techniques such as data mining and machine learning to use known citation network structure information to predict the future citation relationships of papers in the citation network. The application scenarios of citation link prediction are very wide, including promoting academic cooperation, discovering future academic innovation points, measuring the contribution of academic achievements, etc. Graph neural networks are good at mining data in graph structures. Research on graph neural networks has made great achievements in recent years, so they can be used in citation link prediction tasks.At present, research on tasks such as node classification and graph classification based on graph neural networks has been very rich, and the citation link prediction task can be regarded as a classification task to determine whether there is a citation relationship between document nodes. Therefore, the graph neural network is very suitable for the citation link prediction task. Therefore, the research on citation link prediction based on graph neural network is of great significance.

Existing link prediction methods based on graph neural networks still have some shortcomings. In the traditional graph neural network link prediction method based on the information transfer mechanism, node attribute feature information is only used as input features, and then the network structure features of the node are obtained through the neighborhood aggregation strategy. The disadvantage of this method is that it cannot train the model for nodes with missing neighbors. However, in the actual citation network, since a document can only cite published documents, the citation relationships of documents are very sparse. Documents that cite sufficient relational data in the information transfer mechanism appear more frequently in domain aggregation and supervision losses, making the graph neural network more likely to bias such documents, thus sacrificing the link prediction performance of documents that cite sparse relational data. . However, these methods only look at the problem from the perspective of citation links, and from the perspective of document link prediction, these methods are not fair to the high proportion of documents lacking citation information. Since documents with no citation information cannot be measured in the citation link prediction task, we uses unpopular documents lacking citation information as substitutes for analysis. Because in the citation link prediction task, all citation information of these documents can be easily divided into the test set to achieve the same link prediction effect as documents without citation information in actual experiments. In order to reflect the problems that arise in the processing of link prediction based on graph neural networks for documents lacking citation information, we refers to another visualization work that represents the learning uniformity on the unit hypersphere to evaluate the node feature uniformity of different graph neural network models. was visualized[1]. First, sort the Cora data set and Citeseer data set in descending order according to the number of citation information of the document. Randomly sample 250 popular documents with rich citation information from the top 15% of the document collection and randomly sample 250 unpopular documents with lack of citation information from the bottom 30% of the document collection. The purpose of random sampling is, on the one hand, to more clearly reflect the data distribution characteristics in the graph, and on the other hand, to make the sampled sample more representative. Then use t-SNE to map the sampling document node representations learned by different models and the corresponding initial features to the two-dimensional normalized vector of the unit hypersphere (that is, a circle with a radius of 1), and use Gaussian kernel density estimation to draw two dimensional feature distribution[2]. As shown in Figure 1, the horizontal and vertical axes are node coordinates, and the color depth represents the probability density value. The darker the color, the more points fall in this area. From (a) and (b) in Figure 1, we can see the comparison between two link prediction benchmark models based on graph neural networks and the Warmer-GNN model proposed in this article under two data sets. In the baseline model, the sampled document collection shows highly clustered node features, while the node representation trained by Warmer-GNN is relatively more uniform.



(a) Node representation distribution of Cora dataset

(b) Node representation distribution of Citeseer dataset

Figure 1: Gaussian kernel density estimate of node representation distribution

2. Method

The overall framework of Warmer-GNN is shown in figure 2. The input sources of the model include citation graph and feature graph created by the citation network. The citation diagram contains all the document nodes in the citation network as well as the citation information of the document. The feature graph contains all the document nodes in the citation network as well as the explicit attribute feature information of the document. In this paper, feature graph is used to replace the initial feature embedding of document nodes to provide attribute feature information for the model. The model is divided into citation graph training and feature graph training. we refers to the DeepWalk[3] model and obtains the positive sample set corresponding to the citation graph and the feature graph by random walk algorithm to form node pairs, and takes the two graphs and their node pairs as model inputs to obtain the final node embedded representation through graph neural network module training. Because the data sources of the citation graph and the feature graph come from the same citation network, and the nodes in the two graphs correspond to each other, the graph neural network parameters in the graph neural network module can be shared.

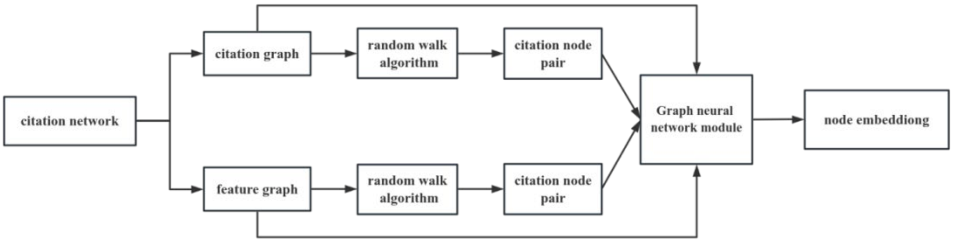


Figure 2: Warmer-GNN framework

2.1. Construction of citation graph and feature graph

The citation graph contains all document nodes in the citation network and the corresponding citation relationships, which can be expressed as$G_{cite} = (V_{cite}, E_{cite})$.$V_{cite}$ is a node collection of the citation graph, which contains all document nodes in the citation network. It is an edge set in the citation graph, E_{cite}representing the citation relationship between documents, which contains the citation relationships corresponding to all document nodes in the citation network. The establishment of the citation graph is similar to the general graph neural network method. The establishment of the feature graph is introduced in detail below.

The experimental citation data set includes the processed keyword information of the document. First, the keyword information amount is calculated for all the keywords in the citation data set. For keywords with a high amount of information, they cannot express the characteristics of the document very well and have little relationship with the citations between the documents. Instead, they will interfere with the calculation results of the keyword similarity of the article. Assuming that the set of all keywords in the data set is$K = \{k_1, \dots k_n\}$,then the information content of the keywords can be defined as:

$$I(k_a) = -\log\big(p(k_a)\big) \tag{1}$$

$p(k)$is the probability of the keyword appearing in all articles. Therefore, this article first defines the collection of documents as$V = \{v_1, \dots v_n\}$in the data set. where n represents the number of documents in the training set. The keyword similarity between two documents$v_m$and$v_n$between two document collections can be defined as:

$$\mathrm{sim}(v_m, v_n) = \frac{\sum_{k_1 \in K_{v_m} \cap K_{v_n}} \frac{1}{I(k_1)}}{\sum_{k_2 \in K_{v_m}} \frac{1}{I(k_2)} + \sum_{k_3 \in K_{v_n}} \frac{1}{I(k_3)}} \tag{2}$$

In the above formula,$K_{v_m}$and$K_{v_n}$are the keyword sets of the document$v_m$and$v_m$respectively. After obtaining the keyword similarity of the documents in the data set, the following describes how to extract the feature information required in the model for documents without citations and other documents to create a feature graph.

The feature graph created by the model can be expressed as$G_{feature} = (V_{feature}, E_{feature})$,$V_{feature}$is a node set of feature graph, including all document nodes in the citation network.$E_{feature}$is the directed edge set of the feature graph, which represents the feature relationship set of the document nodes.

Because the documents without citations has no citation information in the citation graph, it is trained as a single feature graph in the graph neural network. Therefore, in order to link the documents without citations to predict its feature information in the feature graph, it needs to be more accurate. Because documents with citations have citation information in the citation graph at the same time, the feature information in the feature graph can be retained more. Therefore, we use different methods to establish the edge sets of nodes for documents without citations and nodes for documents with citations in the feature graph.

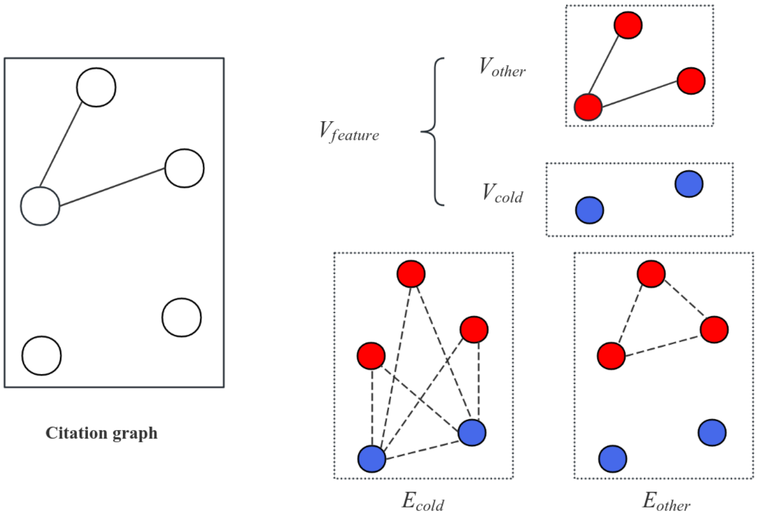


Figure 3: Related concepts of feature graph

As shown in figure 3, firstly, the node set in the feature graph is divided, where $V_{feature} = \{V_{cold}, V_{other}\}$,$E_{feature} = \{E_{cold}, E_{other}\}$.$V_{cold}$is the set of non-citation document nodes in the data set and the set of other nodes of documents in the data set.E_{cold} is a set of edges that contain both $V_{cold}$and $V_{feature}$.After obtaining the keyword similarity between $V_{cold}$and$V_{feature}$, the number of the edges $N_{feature}$with the highest keyword similarity is extracted to form the E_{cold}. E_{other}Is a set of edges that contain only the nodes inV_{other}. For each node,we select at most 5 nodes in $V_{feature}$with the highest keyword similarity to form the E_{other}. The calculation formula of $N_{feature}$is as follows:

$$N_{feature} = p * N_{cite} \tag{3}$$

2.2. Citation Link Prediction Model based on Graph Neural Network

The details of the graph neural network module are shown in Figure 4. The node edge embedding of the feature graph and citation graph aggregates neighbor information through the sage pooling operation, and finally updates the node information through the gated filtering gate mechanism. The node information of the citation graph and the feature graph is fused through the self-attention mechanism and the final node embedding representation is obtained through linear layer transformation. The details in the graph neural network module are introduced in detail below.

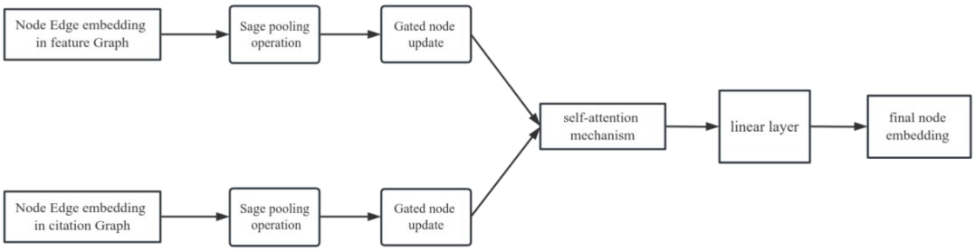


Figure 4: Graph neural network module

First, a fixed number of neighbor samples of each order of each node in the feature graph and citation graph are obtained by random sampling to form the neighbor set $N_{i,cite}$ and $N_{i,feature}$ of the node $v_i$to be trained. Then the edge embedding of the node $v_i$is defined as $U_i = \{u_{i,cite}, u_{i,feature}\}$, (C is the edge embedding size). Where $u_{i,cite} \in R^{1\times C}$ is the edge embedding representation on the citation graph of v_i; $u_{i,feature} \in R^{1\times C}$ is the edge embedding representation on the feature graph of v_i. In order to aggregate the neighbor edge embedding information of the nodev_i,we refer the maximum pooling aggregation function in GraphSAGE[4] to aggregate the k-th order neighbor edge embedding of the node v_i in the citation graph and the feature graph respectively:

$$u_{i,cite}^{(k)} = \mathrm{Max}\left(\{w_{cite}^{(k)} u_{j,cite}^{(k-1)}, \forall v_j \in N_{i,cite}\}\right) \tag{4}$$

$$u_{i,feature}^{(k)} = \mathrm{Max}\left(\{w_{feature}^{(k)} u_{j,feature}^{(k-1)}, \forall v_j \in N_{i,feature}\}\right) \tag{5}$$

In Formula 4, $u_{i,cite}^{(k)}$ is the k-th order edge embedding of v_i in the citation graph,$w_{cite}^{(k)}$ is the weight matrix of the aggregation function in the citation graph, and $u_{j,cite}^{(k-1)}$ is the edge embedding of the k-1th order aggregation neighbor of the node v_i in the citation graph. The definition in formula 5 is similar to formula 4 and will not be repeated here. Maximum pooling, as the aggregation function in the graph neural network model, is mainly used to retain the main features of neighbors, reduce the calculation amount of the calculation model, while suppressing noise and reducing information redundancy.

In order to better capture the homogeneity phenomenon in the citation graph and feature graph, this paper refers to the structural idea of gated-GNN[5] for node information update learning. This idea uses a filter gate mechanism to filter out information in the target node v_i that is inconsistent with neighbor information, and finally obtains the edge embedding representation $u_{i,cite}$and $u_{i,feature}$ of the nodev_i:

$$f_{gate,cite} = \mathrm{sigmoid}\left(W_{f,cite} \cdot \mathrm{concat}(u_{i,cite}, u_{i,cite}^{(k)}) + b_{f,cite}\right) \tag{6}$$

$$u_{i,cite}^{-} = u_{i,cite} \odot (1 - f_{gate,cite}) \tag{7}$$

$$u_{i,cite} = \sigma\left(u_{i,cite}^{-} + u_{i,cite}^{(k)}\right) \tag{8}$$

In Equation 2.6, a filter gate $f_{gate,cite}$is applied to the edge embedding$u_{i,cite}$ of the target node $v_i$in the citation graph to filter out information inconsistent with the aggregated neighbor embedding $u_{i,cite}^{(k)}$, where $W_{f,cite}$ is the weight matrix and $b_{f,cite}$ is the bias term. $\mathrm{concat}()$is a vector concatenation operation. In Equation 7,$u_{i,cite}^{-}$ is the remaining embedding after the filter gate operation of $u_{i,cite}$, where □ is the vector dot multiplication operation. In Formula 8,Combining the remaining embedding $u_{i,cite}^{-}$ and the aggregated neighbor embedding $u_{i,cite}^{(k)}$ obtained previously, we obtain the final edge embedding representation $u_{i,cite}$ of node v_i in the citation graph after the node information is updated.The process of updating the node information of the node$v_i$in the feature graph and finally obtaining the edge embedding $u_{i,feature}$ is the same, so we won't go into details here.

After neighbor aggregation, edge embeddings $u_{i,cite}$ and $u_{i,feature}$ of node v_i are finally obtained, and U_i is obtained through vector splicing. Next, it is necessary to use the self-attention mechanism to obtain the self-attention coefficient $\alpha_i \in R^{1\times 2}$ of node v_i to measure the importance of the two edge embeddings $u_{i,cite}$ and $u_{i,feature}$ respectively:

$$a_i = softmax(\hat{w}^T\, tanh(\, U_i W))^T \tag{9}$$

In Formula 9, $W \in R^{2C\times A}$ and $\hat{w} \in R^{A\times 1}$ (A is the attention feature dimension) are learnable parameters. The final node embedding calculation formula for node v_i is as follows:

$$h_i = ||M^T U_i a_i + b||_2 \tag{10}$$

In Formula 10, h_i is the final node embedding representation of the paper node. $M \in R^{2C\times D}$ and (D is the dimension size of the final embedding representation of the node) are the learnable transformation matrix, which transforms the node edge embedding dimension into the dimension of the final embedding representation of the node, and b is the deviation term.

2.3. Negative sampling of documents without citations

Since there are no real negative samples for nodes of documents without citations in the citation graph, the selection of negative samples for nodes of documents without citations is based on the keyword feature similarity of the paper nodes. This paper selects the N paper nodes with the lowest similarity for each nodes of documents without citations as the negative sample candidate set of the nodes of documents without citations, and randomly selects n<N document nodes as negative samples in each training process. However, there are some problems in using these negative sample information directly: because these negative samples are samples that are easy to be distinguished by the model, although this model does not use initial node embedding, the negative samples may be helpful to model learning in the first few rounds of training. , but in the subsequent training process, the training of negative samples is difficult to learn from the model, and at the same time it increases the training burden of the model. In order to

solve this problem, we uses a mixed embedding method of positive and negative samples to generate difficult negative samples that are close to positive samples, so that the model can better learn the boundary between positive instances and negative instances.

The method of mixing positive and negative samples comes from the mixup algorithm[6].For the training node pair (v_i, v_j) , v_i is the training node, v_j is the positive sample, and $H_{neg} = \{h_{neg,1}, \dots h_{neg,n}\}$ is the embedded set of n negative samples corresponding to v_i nodes. The calculation formula of the new negative sample embedding $h'_{neg,m}$ obtained by the positive and negative sample mixing method is as follows:

$$h'_{neg,m} = \alpha h_{v_j} + (1 - \alpha) h_{neg,m} \tag{11}$$

Where $\alpha \in (0, 1)$ is the mixing coefficient of uniform sampling and h_{v_j} is the node embedded representation of v_j. From the above formula, we can get $H'_{neg} = \{h'_{neg,1}, \dots h'_{neg,n}\}$ after embedding the new negative sample $h'_{neg,m}$, and then score the difficult negative sample in the form of inner product, and select the node with the highest score to embed $h'_{neg,x}$. The calculation formula is as follows:

$$h'_{neg,x} = \underset{h'_{neg,m} \in H'_{neg}}{\operatorname{argmax}} (h'_{neg,m} \cdot h_{v_i}) \tag{12}$$

3. Experiments

In this chapter, in order to verify the effectiveness of the proposed model, we conduct a series of experiments on two public data sets. First of all, we will introduce the information of the data sets used in the experiment and the competitor algorithms to evaluate the performance of the algorithm. Then, we will introduce the baseline models of several citation link prediction methods and the experimental parameter settings for comparison. Finally, the experimental results are analyzed.

3.1. Datasets

The experiment uses two real-world citation network datasets Cora and Citeseer[7]. The attribute features of each node in the Cora dataset correspond to the keywords represented by the word bag of the document. Each node in the Citeseer dataset has the attribute features extracted from the paper content corresponding to the removal of stop words in the document and the words that appear less than 10 times in the document. The specific statistics for the two datasets are shown in Table 1:

Table 1: Descriptions of datasets

Statistical data	nodes	edges	attribute features	Paper category
Cora	2708	5429	1433	7
Warmer-GNN	3327	4732	3703	6

3.2. Competitors

We compare the proposed model with five competitors and set the best parameters according to the original paper.

(1)GCN[8]: GCN (Graph Convolution Network) is a semi-supervised learning method that can deal with nodes with and without tagged information.

(2)GraphSAGE[4]: GraphSAGE is an unsupervised inductive graph representing framework on a large graph.

(3)GAE[9]: TGAE (Graph Autoencoder) is a graph embedding method based on autoencoders, which transforms the graph embedding problem into a reconstruction problem and learns model parameters by minimizing the error between the observed data and the model.

(4)VGAE[9]:VGAE (Graph Variational Autoencoder) is a graph embedding method based on variational autoencoders.

(5)GAT[10]: GAT (Graph Attention Network) is a graph neural network model based on the attention mechanism.

Then we will use the above competitors to do cold start link prediction experiment respectively and compare it with our method.In order to test the link prediction effect of our method on documents without citation, the cold start link prediction experiment randomly selects 80% of the document nodes and their edge sets in the dataset as the training set, and the remaining 20% of the documents will be regarded as documents without citation, and their nodes and edge sets will be used as the test set. In both datasets, the p ratio value in the feature graph is set to 0.7.

3.3. Experimental Results

Under the above experimental conditions, the link prediction experimental results are shown in Table 2:

Table 2: Comparison of experimental results on cold start link prediction experiment

	Cora		Citeseer	
	AUC	F1	AUC	F1
GraphSAGE	0.68	0.66	0.65	0.64
GCN	0.71	0.65	0.72	0.66
GAE	0.78	0.72	0.83	0.77
VGAE	0.72	0.68	0.79	0.72
GAT	0.78	0.71	0.84	0.77
Warmer-GNN	**0.86**	**0.77**	**0.93**	**0.84**

4. Conclusions

In this study, we propose a citation link prediction method for the cold start problem, namely Warmer-GNN. This algorithm constructs the feature map and citation map of the citation network, introduces a positive and negative sample mixing strategy to generate high-quality negative samples for the feature map, and combines the attention mechanism to achieve citation link prediction. We conducted cold-start citation link prediction experiments on two different data sets, and the results proved that Warmer-GNN has great performance advantages.

References

[1] Gilmer J, Schoenholz S S, Riley P F, et al. Neural message passing for quantum chemistry[C]. International conference on machine learning. PMLR, 2017: 1263-1272.

[2] Botev Z I, Grotowski J F, Kroese D P. Kernel density estimation via diffusion[J].Ann Stat 2010, 38(5): 2916-2957

[3] Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations[C].Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. 2014: 701-710.

[4] Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[J]. Advances in neural information processing systems, 2017, 30.

[5] Qian T, Liang Y, Li Q. Solving cold start problem in recommendation with attribute graph neural networks [J]. arXiv preprint arXiv:1912.12398, 2019.

[6] Huang T, Dong Y, Ding M, et al. Mixgcf: An improved training method for graph neural network-based recommender systems[C].Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 665-674.

[7] Sen P, Namata G, Bilgic M, et al. Collective classification in network data[J]. AI magazine, 2008, 29(3): 93-93.

[8] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.

[9] Kipf T N, Welling M. Variational graph auto-encoders[J]. arXiv preprint arXiv:1611.07308, 2016.

[10] Veličković P, Cucurull G, Casanova A, et al. Graph attention networks [J]. arXiv preprint arXiv:1710.10903, 2017.