A study of small sample node classification based on graph data augmentation

Abstract: Graph data is widely found in the real world. However, it often faces a shortage of labeled data in practical applications. Many methods for few-shot learning on graphs aim to classify data with fewer labeled samples. Although good performance has been achieved in few-shot node classification tasks, there are still the following problems. High-quality labeled data are difficult to obtain; generalization ability is insufficient in the parameter initialization process of few-shot node classification (Few-Shot Node Classification, FSNC) methods; the topology structure information in the graph has not been fully mined for the existing FSNC methods. To address these problems, a new Few-Shot Node Classification model based Graph Data Augmentation (GDA-FSNC) was proposed. There are four main modules in GDA-FSNC: a graph data preprocessing module based on structural similarity, a parameter initialization module, a parameter fine-tuning module, and an adaptive pseudo-label generation module. In the graph data preprocessing module, an adjacency matrix enhancement method based on structural similarity was used to obtain more graph structural information. To enhance the diversity of information during the model training process, a mutual teaching data augmentation method was used in the parameter initialization module, by which different patterns and features were learned from the other model. In the adaptive pseudo-label generation module, appropriate pseudo-label generation techniques were automatically selected according to the characteristics of different datasets, then high-quality pseudo-label data was generated. Experiments were conducted on seven real datasets. The experimental

results show that the proposed model performed better than state-of-the-art few-shot learning models such as Meta-GNN, GPN, and IA-FSNC in classification accuracy. On small datasets, it achieved an average improvement of 3.40 percentage points over the baseline IA-FSNC model, and on large datasets, the average improvement was 2.47 percentage points. GDA-FSNC shows better classification performance and generalization ability than state-of-the-art methods in few-shot learning scenarios.

Keywords: node classification; Graph Convolutional Network (GCN); data augmentation; meta-learning; Few-shot Learing (FSL).

0 Introduction

With the rapid development of data science, attribute networks play an important role in scenarios such as citation networks and social media networks, where node classification is a fundamental task. Existing studies have shown that graph neural networks can effectively learn node low-dimensional embeddings and improve classification performance. However, traditional methods rely on a large amount of labeled data, while most node categories in real-world scenarios have only a limited number of labeled instances, which makes these methods prone to overfitting when dealing with small-sample categories.In addition, the process of manually labeling data is both time-consuming and expensive. As a result, Few-Shot Learning (FSL) has gradually received extensive attention from the academic community due to its excellent performance with limited labeled data. In recent years, meta-learning based methods have emerged to address the challenges of FSL, especially in the context of processing Euclidean data. These methods have been widely used in many fields, such

as text and image, to achieve better results. Inspired by meta-learning methods, graph meta-learning methods for graph-structured data have gradually emerged to solve the FSL problem in the graph domain. For example, GPN (Graph Prototypical Network) classifies new instances by learning a set of prototypes for each category and utilizing the similarity between new instances and these prototypes.Meta-GNN combines Graph Neural Network (GNN) with the meta-learner MAML (Model-Agnostic Meta-Learning) to find initial model parameters that perform well in multiple meta-tasks.

Although existing graph meta-learning methods have achieved good results in the field of small-sample learning of graph-structured data, there are several limitations: 1) High-quality labeled data is difficult to obtain. In practical application scenarios, even the small amount of labeled data available may have quality problems; and in specific domains, such as medical image analysis, bioinformatics, etc., the acquisition of high-quality labels is not only costly but also requires specialized knowledge; which further leads to the increased difficulty of small-sample learning on graphs.2) Based on the Graph Convolutional Network (GCN) based traditional Few-Shot Node Classification (FSNC) method needs to perform a large number of tasks under the GCN model, and this process affects its learning efficiency and generalization ability. The main reason is that in GCN, every time a task is performed, information needs to be aggregated and transferred to the whole graph, which will consume a lot of computational resources on large-scale graph data; while small-sample learning itself is easily limited by the insufficient number of samples,

and if multi-tasks are performed on top of this, the model may be over-optimized on the training samples, which leads to a decrease in the generalization ability.3) Existing FSNC methods fail to fully mine the topological structure information in the graph, which leads to the difficulty of the model to accurately capture key features when analyzing complex graph data, which in turn affects its performance. To address the above problems, this paper proposes a Few-Shot Node Classification model based Graph Data Augmentation (GDA-FSNC) model.

1 Related work

In recent years, GNNs have been widely used in machine learning tasks, and enhancing graph data augmentation has become a research focus. For example, FLAG (Free Large-scale Adversarial augmentation on Graphs) introduces learnable gradient-based masks to enhance node features through adversarial perturbation.QI scholars et al [1] proposed DropEdge and its layer independent variant to alleviate the problem of over-smoothing of node representations, the former randomly deleted edges generate perturbed adjacency matrices and share them to all layers, while the latter generates perturbation matrices independently for each layer.

Although GNN has achieved good results in node classification tasks, its performance is largely dependent on the number of labeled nodes in each category. In practice, the problem of insufficient labeled nodes often occurs for emerging categories. For example, in biological networks, newly discovered protein nodes require specialized knowledge for annotation. Therefore, it is particularly important for GNNs to perform effective node classification under the condition of limited

labeled nodes, i.e., the small-sample node classification problem.

Recent research has focused on extracting transferable knowledge from base classes and applying it to new classes, and these methods place special emphasis on meta-learning, learning by performing a meta-training task on the base class and evaluating the model by performing a meta-testing task on the new class. In this paper, we collectively refer to meta-training and meta-testing tasks as meta-tasks. In the field of graph data analytics, important progress has been made in a number of representative models based on the situational meta-learning paradigm. the GPN model classifies new samples based on the similarity between them and these prototypes by constructing a collection of prototypes for each category. the Meta-GNN model combines GNN and MAML with the aim of finding a set of initial model parameters that can exhibit excellent generalization performance across a variety of meta-tasks. The GMeta (Graph Meta-learning model [2] employs structural features of subgraphs to generate representations of nodes, thus effectively addressing the challenge of classifying nodes with small samples.The IA-FSNC (Information Augmentation for Few-Shot Node Classification) model incorporates support augmentation and sample augmentation strategies to improve the performance of the small-sample node classification task.The TENT model [3] efficiently handles the problem of inter-task variability by constructing a graph structure based on category prototypes and corresponding GNN parameters.

2 Problem definition

Formally, let $G = (A, X, V, E, C)$ denote an attribute graph,

where $V = \{v_1, v_2, \ldots v_n\}$ denotes the set of nodes; E denotes the set of edges; C denotes the feature matrix, d denotes the node feature dimension, and n denotes the number of nodes in the graph; A denotes the adjacency matrix, which represents the topology of the graph, and C denotes the set of categories to which the nodes belong.

In a small sample node classification setup there is $C = C_b \bigcup C_n$, where C_b denotes the base class and C_n denotes the new class, and a sufficient number of labeled nodes in the base class C_b are used to train the model, and subsequently, given an arbitrary subset of the N new classes C_n, the goal is to train classifiers for the new class C_n with only K (K is usually taken as 3, 5) labeled nodes per class that can predict the labels of the remaining unlabeled nodes in the N classes, known as the N-way K- shot node classification problem.

A meta-learning approach is also used to deal with the small sample node classification problem, with the base class C_b as the meta-training category set and the new class C_n as the meta-testing category set. A meta-training task t is constructed by sampling N classes from the base class C_b. t consists of a support set S_t and a query set Q_t, S_t consists of K randomly selected nodes from each class in the category N, denoted as A, and Q_t consists of b randomly selected residual nodes from each class in the category N, denoted as B. The support set S_t is the labelled nodes in the task t, and b denotes the number of nodes evaluated in the query set Q_t contained in the query set Q_t, and b is set as 12. The goal of each meta-training task is to minimise the classification loss between the predicted probability and the true label of the query set Q_t.

Also given a series of meta-training tasks, the aim is to learn a classifier model that is able to utilise transferable prior knowledge in meta-testing tasks. Similar to the meta-training N-way K-shot task, the meta-testing task t¢ is sampled in a new class C_n , t' consists of the support set $S_{t'}$ and the query set $Q_{t'}$, and the labels of the query set $Q_{t'}$ can be predicted by the classifier model.

3 GDA-FSNC model

In this paper, we propose a small sample node classification model GDA-FSNC based on graph data enhancement technique, which consists of four main modules: graph data preprocessing based on structural similarity, parameter initialisation, parameter fine-tuning and adaptive pseudo-label generation module. The design idea of the model is as follows:

1) By analysing the number of common neighbours of node pairs as well as the degree of individual nodes, the adjacency matrix of the original dataset was enhanced to contain more information about the structure of the nodes, which helps to better represent the inter-relationships between nodes.

2) In accordance with the small-sample learning paradigm, the augmented entire dataset is divided into base and new classes, and the base and new class datasets are further subdivided into support and query sets in order to provide the corresponding training and testing data at different stages.

3) On the base class data, the parameters are initialised using a two-layer graph convolutional neural network (GCN), which is structured to fully learn the structural information of the graph dataset. In addition, the learning ability of the model is

enhanced using a graph data augmentation method based on the idea of mutual teaching to accelerate the convergence of the model loss.

4) When the GCN converges after training in the base class, the linear layer weight parameters in the first convolutional layer of the GCN model with better classification performance are selected and passed to the new class classification task as initialisation parameters. These parameters contain the a priori knowledge and structural information of the base class, which is an important guide for the learning of the new class.

5) Fine-tuning the initialisation parameters of the GCN using the support set in the new class and the high-quality pseudo-labelled data generated by the adaptive pseudo-labelling generation module, a process aimed at better adapting the model to the data distribution and structural characteristics of the new class, allowing it to get better performance in the new class task.

3.1 Data preprocessing module based on structural similarity

The similarity of graph nodes is an important basis for knowledge discovery and pattern recognition on graphs and networks, and the adjacency matrix is the most direct method used to measure the degree of structural similarity between nodes in a graph. In order to present the structural information of a graph more efficiently, the neighbourhood matrix can be enriched by calculating the structural similarity between nodes using different metrics such as Salton[4], Dice and Jaccard. In order to investigate the performance impact of different metrics on the downstream small-sample node classification task, different node similarity metrics are used to

augment the neighbour-joining matrix, and then a 2-way 1-shot node classification experiment is carried out on two citation datasets[5], Cora and Citeseer, using the GDA-FSNC model with the structural augmentation removed. The classification results are shown in Table 1. (where Origin denotes that no similarity metrics are used for enhancement and only the original neighbourhood matrix is used, and Salton&Dice denotes that the neighbourhood matrix is enhanced using the linear fusion of Salton and Dice metrics).

Table 1 Classification accuracy under different similarity measures

measures	Cora	Citeseer
Origin	72.08	72.08
Jaccard	73.61	73.61
Dice	74.44	74.44
Salton	75.13	75.13
Salton&Dice	75.50	75.50

From the above table, it can be observed that there are differences in the effectiveness of different metrics for neighbourhood matrix enhancement.Salton, Dice, Jaccard and Salton&Dice similarity metrics all improve the classification accuracy of the model to a certain extent, with the linear fusion approach of Salton&Dice being the most effective in terms of improvement in classification accuracy. This is because the Salton metric takes into account the number of common neighbours and is normalised by the square root of the node degree, so augmenting the adjacency matrix with the Salton metric reveals more effective information about the details of the

structural similarity between the nodes in the graph; whereas with the Dice and Jaccard metrics, both of them emphasise on the overall size and differences in the set of nodes' neighbours, so that when using either the Dice or the Jaccard metrics, on the other hand, better reflect the global information of structural similarity between nodes[6-7].

Based on the above experimental findings, this paper adopts the linear fusion method of Salton and Dice metrics in order to capture both local and global information about the similarity of nodes in the graph when calculating the node similarity matrix, which in turn effectively enhances the adjacency matrix A, and the augmented matrix is denoted as $\widetilde{A} \in R^{n\times n}$. Let a_{ij} * denote the similarity between the node pairs $(v_1,\ v_2)$, and then a_{ij} * is defined as:

$$a_{ij}^{*} = \begin{cases} a_{ij} + \alpha_1\theta_1 + \alpha_2\theta_2, & if \left|ne_i \cap ne_j\right| \geq 1 \\ a_{ij}, & otherwise \end{cases} \quad (1)$$

Where, $\theta_1 = \left|ne_i \cap ne_j\right| / \sqrt{deg_i \times deg_j}$ is the Salton metric and $\theta_2 = 2\left|ne_i \cap ne_j\right| / (\left|ne_i\right| + \left|ne_j\right|)$ is the Dice metric. n_{ei} is the set of neighbours of node v_i and $\left|ne_i \cap ne_j\right|$ is the number of common neighbour nodes between node v_i and node v_j. degi denotes the degree of node v_i. In linear fusion methods, determining the weights 1 a and 2 a for the Salton metric and the Dice metric usually depends on experiments and data analysis. Therefore, in this paper, we use a grid search algorithm to determine the weights of Salton metric and Dice metric using classification accuracy, which is used to evaluate the effectiveness of the linear fusion method with different combinations of weights[8-9]. The combination of weights that results in the optimal performance evaluation metrics is selected as the final weights of Salton and Dice

metrics. (Initially, the weights of the Salton and Dice metrics are set to be equal, 0.5 each, and the weight parameter ranges from 0 to 1, discretised, so that the sum of their weights is 1. A grid search algorithm is performed to find the optimal combination of weights in the interval from 0 to 1.)

According to the Salton and Dice metric formula, it is known that the greater the number of common neighbours between two nodes, the closer they are to each other, and the greater the value of a_{ij}^{*} [10], which can be used to quantify the structural similarity between two nodes in a graph. For large graphs, a_{ij}^{*} can also be computed efficiently because it only requires the set of neighbour nodes of the two nodes rather than the global information of the whole graph.

3.2 Initialisation of GDA-FSNC model parameters

Parameter initialisation plays an important role in meta-learning methods, and for traditional meta-learning methods, the process of parameter initialisation is accomplished by constructing multiple related node classification tasks. The specific steps are as follows:

1) Construct multiple node classification tasks, each of which randomly selects a combination of subsets of different classes from the same dataset to obtain training samples.

2) Each task will initialise the model parameters independently, usually using the same network structure. This is because each task has its own unique combination of training samples and categories, so the model parameters need to be initialised independently to suit the needs of each task[11].

3) In the meta-training phase, model parameters are adjusted for model training for each task. Specifically, the model is iteratively trained according to the node categories of each task and the model parameters are gradually adjusted to optimise the performance metrics (e.g. classification accuracy) of the downstream tasks[12].

4) By training across related tasks, a representation of model parameters that is valid for different tasks is learnt. This generalised parameter can be used as an initialisation parameter for new tasks to help them better adapt to the dataset, thus improving the performance of the model on a variety of different tasks.

Traditional meta-learning generalises model parameters through multiple task training. Inspired by the above ideas, IA-FSNC can substantially reduce the overhead caused by repeated training by passing the linear layer weight parameters and the generative layer weight parameters in the first convolutional layer of the GCN in the meta-training phase to the parameter fine-tuning phase as the initialisation parameters of the first layer of the GCN in the parameter fine-tuning.

However, the parameter initialisation strategy of the IA-FSNC model still has some shortcomings.The linear layer weight parameters of IA-FSNC are derived from the layers in the GCN that perform linear transformations.Although these parameters are trained and optimised, due to the intrinsic nature of the linear transformations, they have limited learning ability and are difficult to adequately capture the complex graph structure and node feature relationships. In addition, the generative layer weight parameters of the IA-FSNC model come from additional linear layers, which are weakly correlated with the GCN model itself, resulting in poor quality of the acquired

parameters.The parameter initialisation method of the IA-FSNC model uses parameters from a single GCN model, which restricts the model's ability to generalise and learn. In addition, IA-FSNC directly transmits the parameters of the meta-training phase without evaluating the strengths and weaknesses of the effects of these parameters, and may also suffer from poor quality of the propagated parameters.

To solve the above problems, this paper proposes a new parameter initialisation strategy to optimise the parameter initialisation process of the model. In the meta-training phase, a graph data enhancement method based on mutual teaching is proposed to address the problems of insufficient learning capability of a single GCN and insufficient data information in small sample scenarios. Specifically, two GCN models are allowed to train on the same dataset and use each other's generated labels or predictions as training targets, thus enhancing the diversity of information during model training. The relationship between the mutual teaching enhancement method and model loss design will be discussed in detail in the subsequent sections of this paper.

In order to pass the parameters of meta-training more efficiently to obtain a priori knowledge, the weight parameters of the first convolutional layer of the GCN model, which has the highest classification accuracy, are selected to be passed to the meta-testing stage GCN. The linear layer is usually located at the beginning of the graphical convolutional network, which is responsible for the aggregation of the node features, and its weight parameters determine how to combine the node features without considering the complex spatial structure information is not taken into

account. The weight parameter of the convolutional layer determines how to perform the convolution operation on the input feature map, which is able to capture more complex spatial structure information, so the weight parameter of the convolutional layer has better generalisation performance.

In order to effectively evaluate the effect of passing parameters and ensure the quality of the parameters, this paper introduces the classification accuracy threshold constraint, which selects the parameters of the GCN model to be passed only when the classification accuracy of the model is in the set threshold interval. The specific settings of the classification accuracy threshold constraint will be described in detail in Chapter 4, Experiments.

By adopting the above strategy, the aim is to optimise the parameter initialisation process in order to be able to obtain better GCN parameters, thus providing better initialisation conditions for the meta-testing phase.

3.3 Fine-tuning of GDA-FSNC model parameters

Meta-testing phase evaluates the performance of the model on an unseen test set In this phase, parameter fine-tuning is usually used to optimise the model. Given the initialisation parameters in the parameter initialisation process, a two-layer GCN is used to output the embedded representations of all the nodes in the new class and train a classifier between the nodes and their labels in the support set, which is prone to overfitting due to the limited number of samples. In this paper, we propose an improved adaptive pseudo-labelling generation strategy for parameter fine-tuning to reduce the model's dependence on high-quality labelled data and thus solve the

overfitting problem. The two pseudo-label generation techniques included in this strategy are first analysed in the following, i.e., the high-confidence pseudo-label generation algorithm and the label propagation algorithm. These pseudo-label generation techniques are part of the data augmentation approach, which aims to efficiently expand the training dataset and improve the generalisation ability and robustness of the model.

3.3.1 High-confidence pseudo-label generation algorithm

In high confidence pseudo-labelling generation algorithm, this paper uses a classifier to classify the nodes in the set of unselected nodes. Some of them are then labelled to increase the number of nodes selected based on high confidence. Specifically, it is assumed that the dataset has c classes , using the following way to denote the set of labelled nodes: $D_L = \{(x_i, y_{ij}),\ \forall i \in V_L, j \in [1,c]\}$, and the set of unlabelled nodes is denoted as $D_U = \{x_i, \forall_i \in V_U\}$, where V_L is the set of labelled vertices, V_U is the set of unlabelled vertices, and $V = V_L \bigcup V_U$. The pseudo-labelled vector $\widetilde{y}_i, i \in D_U$ of node v_i is obtained from the vector pi of the ith row of the predicted probability matrix P obtained from the learning of GCN by one one-hot operation as given by the equation as follows:

$$\begin{cases} j = argmax([p_{i1}, p_{i2}, \ldots p_{ic}]) \\ \widetilde{y}_{ij} = 1 \end{cases} \quad (2)$$

Let the prediction confidence r_i of node v_i be the maximum value of the i-th row vector p_i of P.

$$r_i = max(max([p_{i1}, p_{i2}, \ldots, p_{ic}])) \quad (3)$$

Then, the index set is returned for the column vector $r = [r_1,\ r_2, \ldots,\ r_n]^T$, sorted

in descending order, then:

$$q = argsort(r) \tag{4}$$

where q is the index vector returning the ordered confidence level. A number of nodes with high confidence pseudo-labels for each class can be obtained using q. For each GCN, the set of these nodes is V^{pl} , and the support set is expanded using V^{pl} . The number of nodes with high confidence pseudo-labels for each class is determined by the hyperparameter *topk*.

3.3.2 Label Propagation Algorithm

The label propagation algorithm iteratively propagates existing labels through the neighbourhood of the graph to generate pseudo-labels and expand the support set. In order to capture both local and global information of node similarity in the graph data during the label propagation process and to better represent the graph structure, this paper adopts the augmented adjacency matrix $\widetilde{A}$ obtained in the previous subsection 3.1 to represent the adjacency of the graph. In the iterative process, the enhanced adjacency matrix $\widetilde{A}$ is multiplied with the current pseudo-labelling matrix to update the labels of each node, in which graph structure-based label propagation is performed; for nodes with existing labels, onehot coding is used to recover their correct labels.

During the iteration of the label propagation algorithm, the pseudo-label of each node is updated by the propagation of the labels of its neighbouring nodes. To improve the accuracy of the pseudo-labels, the effects of the original labels (e.g., labels in the training data) and label propagation are balanced by the parameter σ in

each iteration.

After analysing the two pseudo-label generation techniques in Sections 3.3.1 and 3.3.2, this paper proposes a metric for pre-computing the graph-level feature similarity for each dataset. This metric is used to measure the consistency of node features across the whole graph; the higher the graph-level feature similarity, the more consistent the node features are across the whole graph. Based on this, appropriate pseudo-label generation strategies are automatically selected to expand the sample set based on the graph-level feature similarity. The node similarity is calculated by combining the node feature matrix and the adjacency matrix, and the average node feature similarity (avg_sim) of the whole graph is calculated using the weighted normalisation method based on the edge information in the adjacency matrix, which is used as a measure of the graph-level feature similarity.

4 Experiments and analysis of results

4.1 Experimental set-up

The experimental environment of this paper is based on 12-core Intel Xeon Platinum 8255C CPU, RTX 2080 Ti GPU with 11GB of video memory (Gigabyte), the programming language is Python3.9, and the experimental model is PyTorch1.10.0.To ensure the fairness and accuracy of the experiments, the results of the experiments in this paper are taken as the average value of the model's 10 runs. runs of the results. The query set size b in the meta-test task is uniformly set to 12. The following describes the dataset, baseline model and experimental parameter settings respectively.

4.1.1 Data set and baseline methodology

In this paper, seven real-world datasets are selected, including four small-scale datasets: Cora, Citeseer, Computers, and Coauthor-CS, and three larger-scale datasets: Cora-full, Amazon Electronics, and Amazon Clothing.This paper summarises key statistical information of the selected datasets, including the number of nodes, the number of edges, the number of features, the number of categories, and the similarity of graph-level features proposed in this paper, as shown in Table 2. statistics of the selected datasets, including the number of nodes, the number of edges, the number of features, the number of categories and the graph-level feature similarity proposed in this paper, and the related data are shown in Table 2.

To validate the effectiveness of GDA-FSNC, it is analysed in this paper in comparison with six mainstream methods, including: GCN, Meta-GNN, G-Meta, GPN, IA-FSNC and TENT.

Table 2 Statistics of datasets

data set	Number of nodes	number of sides	characteristic number (math.)	Number of categories	Graph level feature similarity
Cora	2708	10556	1433	7	0.650
Citeseer	3327	9104	3703	6	0.596
Computers	13381	491722	767	10	0.727
Coauthor-CS	18333	163788	6805	15	0.728
AmazonElectronics	42318	43556	8669	167	0.814

Cora-full	19793	65311	8710	70	0.607
Amazon Clothing	24919	91680	9034	77	0.762

4.1.2 Experimental parameter settings

The GDA-FSNC model in this paper is implemented by a two-layer GCN model. The model optimisation is done using Adam with a learning rate of 0.01 and the weight decay rate is set to 10-4. Since the GDA-FSNC model automatically selects the appropriate pseudo-label generation technique to generate pseudo-labels based on the graph-level feature similarity avg_sim of the dataset, the graph-level feature similarity threshold δ is set to 0.6 by default in this experiment, and when avg_sim is greater than δ, the GDA-FSNC model will use the label propagation strategy to generate pseudo-labels to expand the support set. otherwise, the high-confidence pseudo-label generation algorithm is used. FSNC model will use the label propagation strategy to generate pseudo-labels to expand the support set, otherwise the high-confidence pseudo-label generation algorithm is used. (The default setting of δ is 0.6, which is based on experimental experience, and the subsequent content will give the corresponding experimental analysis for this value.) In the high-confidence pseudo-label generation algorithm, the topk parameter is set to 30 by default. in addition, in order to effectively evaluate the effect of passing parameters in the parameter fine-tuning stage and ensure the quality of parameters, this paper introduces a threshold constraint on the classification accuracy. Experimental experience shows that when the classification accuracy of the model is within [0.45,0.95], the model in

the meta-training phase is able to learn richer a priori knowledge, and passing such parameters can improve the classification performance in the meta-testing phase. Therefore, in this experiment, the classification accuracy threshold constraint is set to be within [0.45,0.95].

4.2 Node classification experiments and analysis

In this paper, three small datasets Cora, Citeseer, and Computers are used for experimental analyses of 2-way K-shot node classification.Three large datasets, Cora-full, Amazon Electronics, and Amazon Clothing, are used for experimental analyses of 5-way K-shot node classification. analysis. In order to comprehensively evaluate the performance of the model under different experimental settings, the Coauthor-CS dataset is used to conduct 2-way K-shot and 5-way K-shot node classification experiments. The experimental results are shown in Tables 3 and 4.

Combining all datasets and settings, the GDA-FSNC model improved classification accuracy by an average of 3.40 percentage points on the small dataset compared to the baseline model IA-FSNC, and the average improvement on the large dataset was 2.47 percentage points. In the 2-way 1-shot setting of the Coauthor CS dataset, the classification accuracy of the GDA-FSNC model improves by only 0.67 percentage points relative to the baseline model IA-FSNC, while in the 5-way 1-shot setting of this dataset, the classification accuracy improves by 7.73 percentage points over the IA-FSNC model. The experimental results show that the GDA-FSNC model exhibits better performance in the small-sample learning node classification task with good generalisation ability, especially for datasets with more categories.

For the baseline models: the GCN, G-Meta, Meta-GNN and GPN; these models do not perform as well as GDA-FSNC in most datasets and settings; the reason may be that these models have insufficient learning and generalisation capabilities when dealing with limited samples, making it impossible to accurately capture complex distributions and structures in graph data. Experimental results show that IA-FSNC and TENT perform better in the small-sample learning node classification task. However, the parameter initialisation method of the IA-FSNC model adopts the parameters of a single GCN model, which limits its generalisation and learning ability to a certain extent; the IA-FSNC model directly passes the parameters of the meta-training stage, but does not evaluate the adaptability and advantages and disadvantages of these parameters, which is also a constraint on the performance improvement of the model.The TENT model mainly takes into account the differences between tasks but does not The TENT model mainly considers the differences between tasks, but does not fully consider the differences between different datasets, which affects its performance on multiple datasets.

Table 3 Classification accuracy (mean and standard deviation) of different models on small datasets

mould	Cora			Citeseer		
	2-way 1-shot	2-way 3-shot	2-way 5-shot	2-way 1-shot	2-way 3-shot	2-way 5-shot
GCN	62.92±3.72	75.08±3.21	82.21±2.63	53.25±4.71	65.0±1.02	72.33±4.15

Meta-GNN	67.73±0.12	76.16±0.16	83.05±0.17	55.10±0.12	68.46±0.09	75.69±0.10
G-Meta	65.43±0.16	76.31±0.13	81.75±0.10	54.48±0.10	66.46±0.11	73.44±0.12
GPN	64.32±0.12	77.43±0.20	82.45±0.15	59.46±0.16	67.31±0.15	75.73±0.09
IA-FSNC	74.78±0.17	80.68±0.13	85.95±0.10	69.83±0.14	78.23±0.12	81.33±0.35
TENT	55.51±0.11	62.79±0.13	62.14±0.12	53.01±0.11	54.21±0.11	56.17±0.11
GDA-FSNC	75.05±0.15	83.71±0.43	87.95±0.06	75.53±0.15	80.35±0.12	82.20±0.11
mould	Computers			Coauthor-CS		
	2-way 1-shot	2-way 3-shot	2-way 5-shot	2-way 1-shot	2-way 3-sho	2-way 5-shot
GCN	71.13±16.2	84.79±12.64	88.75±10.67	82.33±18.73	92.58±7.83	93.10±6.93
Meta-GNN	73.92±0.21	87.66±0.64	89.99±0.14	86.85±0.12	91.93±0.11	93.69±0.15
G-Meta	72.50±0.15	85.95±0.19	89.63±0.23	85.59±0.10	90.56±0.43	92.89±0.32
GPN	72.87±0.	86.55±0.1	90.62±0.0	91.99±0.1	94.25±0.	93.37±0.

	47	3	8	0	07	08
IA-FSNC	80.24±0.12	87.71±0.52	91.04±0.07	91.43±0.12	95.70±0.05	96.65±0.05
TENT	86.12±0.16	92.47±0.10	94.58±0.09	90.81±0.06	93.04±0.03	95.36±0.04
GDA-FSNC	90.16±0.14	96.58±0.06	97.64±0.04	92.10±0.14	96.47±0.07	96.70±0.06

Table 4 Classification accuracy (mean and standard deviation) of different models on big datasets

mould	Cora-full			Coauthor-CS		
	5-way 1-shot	5-way 3-shot	5-way 5-shot	5-way 1-shot	5-way 3-shot	5-way 5-shot
GCN	31.85±2.20	38.33±1.38	42.89±3.23	45.05±0.31	53.48±2.15	59.37±1.68
Meta-GNN	50.57±1.87	56.19±0.57	61.66±3.85	53.18±0.49	61.18±1.73	63.47±2.46
G-Meta	42.71±1.63	52.64±1.24	55.68±3.28	50.97±0.67	62.83±0.91	64.65±1.02

GPN	49.75±2.10	61.78±0.66	65.77±2.83	58.61±0.54	69.70±0.81	72.66±0.49
IA-FSNC	62.32±0.71	70.42±0.39	75.29±0.57	80.43±012	91.65±0.05	94.13±1.53
TENT	52.64±0.08	64.33±0.48	67.74±1.27	54.59±0.17	70.16±0.38	73.12±0.08
GDA-FSNC	65.44±0.10	70.56±0.20	77.28±2.18	88.16±0.38	93.06±0.45	95.28±1.04
mould	Amazon Electronics			Amazon Clothing		
	5-way 1-shot	5-way 3-shot	5-way 5-shot	5-way 1-shot	5-way 3-shot	5-way 5-shot
GCN	41.47±0.97	51.87±1.84	61.92±2.81	48.60±3.15	59.82±2.52	66.88±0.39
Meta-GNN	54.23±1.29	62.19±1.48	68.08±3.16	67.42±1.66	74.62±2.35	75.38±1.78
G-Meta	44.14±1.24	55.75±0.52	60.06±2.98	57.71±0.67	64.44±1.68	71.28±1.34
GPN	46.79±1.40	61.41±0.79	66.48±2.35	59.39±1.87	72.32±2.27	74.40±1.25
IA-FSNC	68.80±0.86	79.78±0.45	83.77±0.23	75.53±0.32	83.39±0.85	85.26±0.48
TENT	66.51±0.1	77.33±0.5	80.42±0.0	69.17±0.1	80.07±0.4	82.29±0.6

	3	6	5	0	8	1
GDA-FSNC	70.86±0.73	81.76±1.16	84.61±2.09	78.31±0.97	86.86±0.45	88.24±1.98

4.3 Ablation experiments and analysis of the GDA-FSNC model

Six datasets, Cora, Citeseer, Computers, Corafull, Amazon Electronics and Amazon Clothing, were selected for the ablation experiments in this paper, with the aim of evaluating the impact of the individual modules in the GDA-FSNC model on the model performance. The model variant that expands the support set using only the label propagation algorithm to generate pseudo-labels is named GDA-FSNC\C; whereas the variant that expands the sample set using only the high-confidence pseudo-label generation technique is called GDA-FSNC\L; the model variant that removes the structural similarity-based preprocessing module for graph data is called GDA-FSNC\S; and the removal of the mutual pedagogical data augmentation method that The model variant that removes the mutual teaching data enhancement method and is trained using only a single GCN model is referred to as GDA-FSNC\MT. the meta-test task experiments were all set up as a 2-way 3-shot. the results of the overall ablation study are shown in Table 5. the results of the overall ablation study are shown in Table 5. the results of the ablation study are shown in Table 5.

The experimental results show that the GDA-FSNC model, by integrating multiple data enhancement techniques (e.g., label propagation, high-confidence pseudo-label generation, and inter-teaching training methods), outperforms the other

variants with a single strategy or module culling on all datasets. Specifically, the GDA-FSNC\C model variant is able to propagate pseudo-labelling information more accurately in datasets with high feature similarity (e.g., Cora, Electronics, and Clothing), effectively improving classification accuracy. The GDA FSNC\L model variant, on the other hand, performs well on graph data with lower feature similarity (e.g., Citeseer), showing that a high-confidence pseudo-labelling generation strategy is an effective data augmentation tool in graph data with more dispersed node feature distribution. In addition, preprocessing of graph data based on structural similarity also plays a role in enhancing the adjacency matrix and improving model performance. It is worth mentioning that after removing the mutual teaching data enhancement method, the model performance decreases on all datasets, which verifies the importance of this method in improving the model classification performance. In summary, by integrating these data enhancement strategies, the GDA-FSNC model not only demonstrates good generalisation ability, but also achieves better classification performance on multiple types of datasets.

Table 5 The classification results of GDA-FSNC and Its variant models

mould	Cora	Citeseer	Computers	Amazon Electronics	Amazon Clothing	Cora-full
GDA-FSNC\L	82.73	78.74	84.72	50.00	54.17	80.00
GDA-FSNC\C	83.65	75.40	94.44	93.61	94.37	85.12
GDA-FSNCMT	82.23	72.08	84.03	82.64	87.11	77.22
GDA-FSNC\S	82.89	78.47	94.75	93.33	92.96	85.87

GDA-FSNC	84.05	79.71	95.14	95.83	95.14	87.43

4.4 Parameter sensitivity experiments and analysis of the GDA-FSNC model

This subsection focuses on analysing the impact of graph-level feature similarity (avg_sim) threshold δ, support set size |S| in the meta-testing phase, query set size |Q| in the meta-training phase and meta-task parameters (N-way, K-shot) in the meta-training phase on the performance of the GDA-FSNC model and its variants. Evaluate the impact of the threshold δ setting in the adaptive pseudo-label generation module of the GDA-FSNC model on the model performance on six datasets, Cora, Citeseer, Computers, Corafull, Amazon Electronics, and Amazon Clothing. Analytical experiments on support set size |S| and query set size |Q| were conducted on Cora and Citeseer, two moderately sized datasets with a small variety of data, to investigate the impact of these parameters on model performance. Sensitivity experiments on meta-task parameters (N-way, K-shot) in the meta-training phase were conducted on three larger datasets, Cora-full, Amazon Electronic and Amazon Clothing. These larger datasets have more categories and provide richer information for parameter (N-way, K-shot) sensitivity studies.

4.4.1 Graph-level feature similarity (avg_sim) threshold δ

In the adaptive pseudo-label generation process, the threshold δ of graph-level feature similarity (avg_sim) plays a crucial role in selecting an appropriate pseudo-label generation strategy. Therefore, this section aims to analyse how different settings of the graph-level feature similarity threshold δ affect the model performance.

Given that the graph-level feature similarities of the selected datasets are all over 0.5, the model will uniformly adopt the label propagation algorithm if δ is set below 0.5, which is not conducive to evaluating the impact of different threshold settings on the performance. In order to deeply explore the specific impact of threshold settings on model performance, the initial experimental threshold in this study was set at 0.5, and the threshold range was set between 0.5 and 1, with discrete values taken every 0.1, i.e., values of 0.5, 0.6, 0.7, 0.8, 0.9, and the meta-test tasks were set as 2-way 1-shot.The experimental results are shown in Table 6.

The experimental results show that on most datasets, the model performance is optimal or near-optimal when δ is set around 0.6. For example, on the Citeseer dataset, when δ is set to 0.6, the classification accuracy of the model increases from 59.38% to 75.34%, showing a large improvement[13]. For most of the datasets, the model performance starts to degrade when the threshold is set above 0.6, especially on the Amazon Electronics and Amazon Clothing datasets, the performance degradation is very significant when δ is set to 0.9, dropping to 51.23% and 55.21%, respectively, which can be attributed to the fact that when δ is set to 0.9, it means that only the when the average feature similarity between nodes in the graph is very high (greater than or equal to 0.9), the label propagation algorithm is enabled. If all the actual avg_sim values are below 0.9, even if they are very close to 0.9, the model will use the high confidence pseudo-label generation algorithm by default. The high-confidence pseudo-labelling generation algorithm may be more effective in datasets where the spatial distribution of features varies significantly, as it relies on

the model's confidence in the predictions of individual nodes. In such datasets, even if the nodes are connected by edges, the features or labels of each node may have more independent predictive attributes due to the obvious feature differences. In contrast, in datasets with highly similar features, the use of this algorithm leads to the omission of important global structural information, making it possible for the model to fail to effectively capture structural and feature similarities in the graph, which can affect performance.

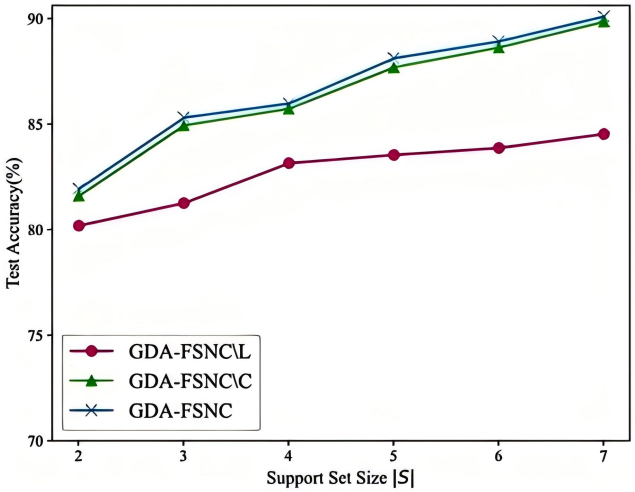
Table 6 The classification results of GDA-FSNC and Its variant models

mould	Cora	Citeseer	Computers	Amazon Electronics	Amazon Clothing	Cora-full
GDA-FSNC\L	82.73	78.74	84.72	50.00	54.17	80.00
GDA-FSNC\C	83.65	75.40	94.44	93.61	94.37	85.12
GDA-FSNCMT	82.23	72.08	84.03	82.64	87.11	77.22
GDA-FSNC\S	82.89	78.47	94.75	93.33	92.96	85.87
GDA-FSNC	84.05	79.71	95.14	95.83	95.14	87.43

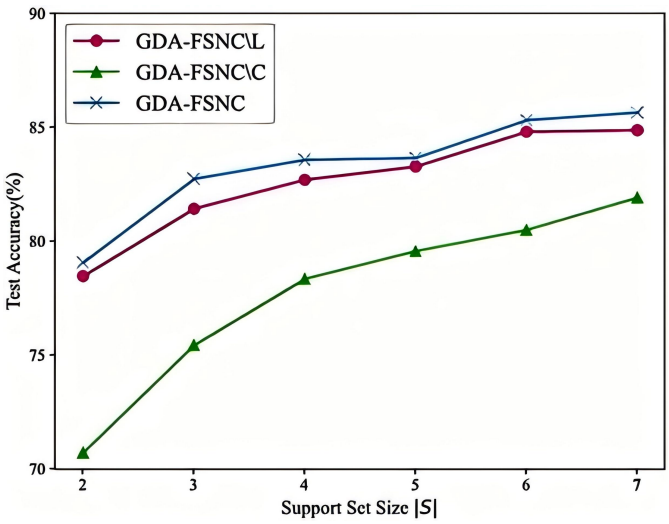
4.4.2 Supporting Set Size|S|

This section investigates the effect of support set size|S| on the performance of GDA-FSNC and its variants during the meta-testing phase. The corresponding experimental results are shown in Fig. 1, where the horizontal coordinate indicates the size of the support set |S| and the vertical coordinate indicates the classification accuracy of the model.

The experimental results show that the classification accuracy of the model shows an upward trend as the support set size increases, which is due to the fact that a larger support set facilitates the model's learning of the potential distribution of the data, and at the same time enhances the model's ability to generalise to the new categories. However, as the support set size increases, the accuracy improvement shows a slowing down trend, which indicates that when the support set is too large, the information gain of the newly added samples for the model starts to decrease, which may be due to the higher consistency between the new samples and the existing samples in the feature space. In summary, an appropriate increase in the support set can improve the performance of the meta-learning model, but too large a support set may cause a slowdown in the model performance improvement.



(a) Cora



(b) Citeseer

Fig. 1 The classification results of GDA-FSNC and its variants at different |S| values

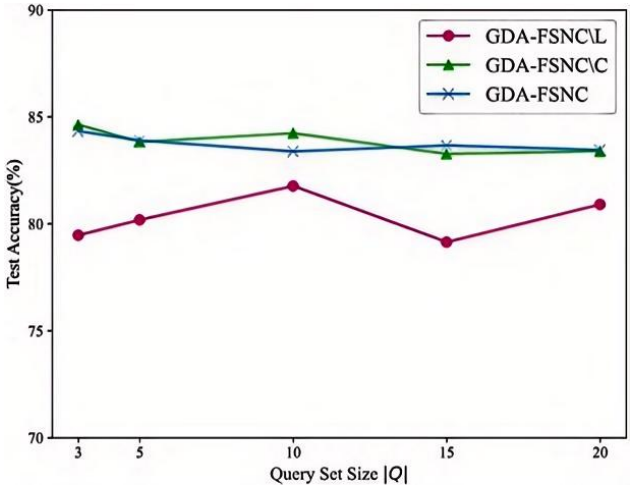
4.4.3 Query Set Size |Q|

In this section, the effect of query set size|Q| on the performance of the model GDAFSNC and its variants during the meta-training phase is investigated, and the Cora and Citeseer datasets are selected for the related experiments. The experimental results are displayed in Fig. 2, where the horizontal coordinate indicates the query set size |Q| and the vertical coordinate indicates the classification accuracy of the model.

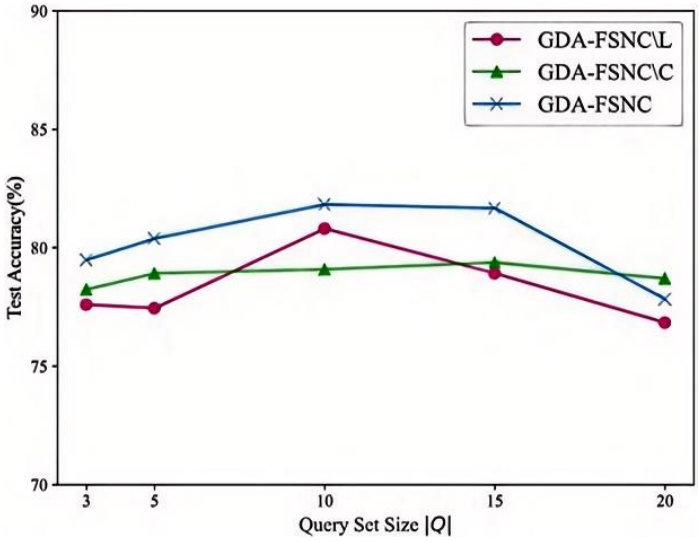
On the Cora dataset, GDA-FSNC and its variants have different sensitivities to the query set size |Q|. In particular, GDA-FSNC\L shows more significant performance improvement when |Q| is small and then fluctuates. The reason is its high-confidence pseudo-label generation strategy, which is unable to accurately capture the differential features between categories on datasets with high feature distribution consistency. As |Q| changes, the query set may introduce more noise and uncertainty. In contrast, GDA-FSNC\C exhibits good stability under different sizes of query sets. This is due to its adoption of a label propagation strategy that effectively

captures the category features in datasets with high consistency. In addition, the GDA FSNC model is stable under different |Q| values and has good generalisation ability.

On the Citeseer dataset, the relationship between the three models and the query set size is complex; initially, increasing the query set improves the performance, but too large a query set may lead to noise and overfitting. Therefore, optimising the query set size |Q| to balance the learning efficiency and prediction ability of the models is crucial to improve the model performance in the meta-learning phase.



(a) Cora



(b) Citeseer

Fig. 5 The classification results of GDA-FSNC and its variants at different |Q| values

5 Conclusion

In this paper, a new small-sample node classification model (GDA-FSNC) based on graph data augmentation technique is proposed by studying and analysing the problems of current small-sample learning methods. The experimental results of the conducted node classification experiments, ablation experiments and parameter sensitivity analyses show that the GDA-FSNC model significantly improves the classification accuracy of the model in small-sample learning scenarios, and has a good generalisation ability on a wide range of datasets. The model effectively extracts node topology information by augmenting the adjacency matrix using structural similarity; the mutual teaching data augmentation method adopted not only improves the efficiency of parameter initialisation, but also enhances the model's ability to generalise across tasks; and the application of the adaptive pseudo-labelling generator module reduces the model's dependence on high-quality labelled data and further optimises the model's adaptation to a variety of datasets. Future research can focus on the quantitative analysis of different data enhancement strategies in small sample scenarios and optimisation strategies for multiple graph data.

Reference

[1] QI G J, AGGARWAL C, TIAN Q, et al. Exploring context and content links in social media: A latent space method[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 34(5): 850-862.

[2] YUAN Z, SANG J, LIU Y, et al. Latent feature learning in social media network[C]// Proceedings of the 21st ACM International Conference on Multimedia.

New York: ACM, 2013: 253-262.

[3] DING K, WANG J, LI J, et al. Graph prototypical networks for few shot learning on attributed networks [C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York: ACM, 2020: 295-304.

[4] Si Yachao, Liu Ziqi, Zhao Mingzhan. Research on Node Classification Based on Graph Convolutional Networks and Graph Data Enhancement Techniques [J]. Journal of Hebei Institute of Architecture and Technology, 2024, 42 (02): 236-240.

[5] Chen Pengpeng, Xing Chengguang, Liu Bo, et al. Aircraft Recognition and Detection Algorithm Based on Significant Graph Data Enhancement [J]. Aerospace Science and Technology, 2023, 34 (11): 118-124.

[6] Yang Ying, Hao Xiaoyan, Yu Dan, et al. Graph Data Generation Method for Extracting Attacks from Graph Neural Network Models [J]. Computer Applications, 2024, 44 (08): 2483-2492.

[7] Chen Zhuomin, Wang Li, Zhu Xiaofei, et al. False news detection based on adversarial graph enhanced contrastive learning [J]. Chinese Journal of Information Science, 2023, 37 (06): 137-146.

[8] Zhang Jiajie, Guo Yi, Wang Jiahui. Multi teacher Learning Graph Neural Network Based on Feature and Graph Structure Information Enhancement [J]. Computer Application Research, 2023, 40 (07): 2013-2018.

[9] Li Y ,Xu L ,Yamanishi K .GMMDA: Gaussian mixture modeling of graph in latent space for graph data augmentation[J].Knowledge and Information Systems,2024,(prepublish):1-29.

[10] Xiaojun L ,Guanjun L ,Jian L .Heterogeneous graph neural network with graph-data augmentation and adaptive denoising[J].Applied Intelligence,2024,54(5):4411-4424.

[11] Shaowu X ,Luo W ,Xibin J .Graph Contrastive Learning with Constrained Graph Data Augmentation[J].Neural Processing Letters,2023,55(8):10705-10726.

[12] Gang H ,Lajos P ,Christos H .Data augmentation based on waterfall plots to increase value of response data generated by small single arm Phase II trials[J].Contemporary Clinical Trials,2021,110106589-106589.

[13] Jamil A ,Khan M ,Wook S B .Data augmentation-assisted deep learning of hand-drawn partially colored sketches for visual search.[J].PloS one,2017,12(8):e0183838.