Li-Guided Ensemble Learning for Few-Shot Node Classification in Any Graphs (LGEL)

SEOKMIN LEE
jhss4475@dgist.ac.kr
Ulm University
Ulm, Germany

HAWEON JEON
wmjhw0624@dgist.ac.kr
Ulm University
Ulm, Germany

ABSTRACT

Node classification in graph-structured data presents unique challenges, particularly in few-shot learning scenarios where only a small number of labeled nodes are available. This paper introduces a novel ensemble learning method for node classification in any graphs with limited labeled data. Our approach leverages the concept of Label Informativeness(LI), a measure that quantifies how much information a neighbor's label provides about a node's label.

We make two key observations: (1) Li is preserved even in few-labeled settings, (2) We observe that the optimal model choice depends on the graph's Label Informativeness (LI). Based on this finding, we propose an ensemble model that dynamically selects the best-performing architecture depending on the computed LI of the input graph.

Importantly, our method provides intuition about which model will perform the best without requiring actual test results, making it particularly valuable in few-shot scenarios where extensive model comparison is not feasible. Additionally, this ensemble model is designed to be adaptable, meaning that if a new state-of-the-art (SOTA) node classification model emerges, our ensemble approach can incorporate this new model. As long as the new SOTA model is influenced by the LI of the input graph, our method can dynamically select it based on the LI threshold.

Experimental results across 15 diverse datasets demonstrate that our ensemble approach consistently outperforms individual models (DNN, GCN, GAT, GIN and GraphSAGE) in few-shot learning scenarios. This work contributes to the field by proposing a novel ensemble method suitable for few-labeled cases where traditional Mixture of Experts approaches are challenging. Our findings highlight the importance of graph structural properties in model selection for few-node classification tasks.

KEYWORDS

Graph Neural Network, Label Informativeness, Node Classification

1 INTRODUCTION

1.1 Background on graph-structured data and node classification

Graph-structured data is ubiquitous in real-world applications, ranging from social networks to molecular structures. Node classification, the task of predicting labels for nodes in a graph, is a fundamental problem in graph analysis. Recent advances in Graph Neural Networks(GNNs) have significantly improved performance on this task, with models such as Graph Convolutional Networks(GCN) [2], Graph Attention Networks(GAT) [3] and GraphSAGE [5] achieving state-of-the-art results.

1.2 Chllenges in few-shot learning for graph neural networks

While GNNs have shown remarkable success, they typically require a substantial amount of labeled data for training. In many real-world scenarios, however, obtaining labeled data can be expensive, time-consuming, or sometimes impossible. This has led to increased interest in few-shot learning for graphs, where models must learn to classify nodes with only a small amount of labeled examples.

1.3 Brief overview of our approach and its novelty

We propose a novel ensemble learning method that leverages the concept of LI to dynamically select the best-performing GNN architecture. Our approach first trains a GCN on the limited labeled data, uses the predictions to compute the graph's Label Informativeness (LI) [12], and then selects either GCN or GraphSAGE as the final prediction model based on an LI threshold of 0.35. This method allows us to intuitively determine the best-performing model without requiring extensive test results, which is particularly valuable in few shot scenarios.

1.4 Main contributions of our work

- We demonstrate that LI is preserved even in few-labeled settings, enabling its use as a reliable indicator of graph structure.
- We establish a relationship between LI and the performance of different GNN architectures, specifically GCN and GraphSAGE.
- We propose a novel ensemble method for few-shot node classification that outperforms individual GNN models across diverse datasets.
- We provide a simple yet effective strategy for model selection in few-shot learning scenarios, where traditional approaches like MoE are challenging to implement.

1.5 Outline of the rest of the paper

The remainder of this paper is organized as follows: Section 2 reviews related work in few-shot learning for graphs , Label Informativeness (LI) and ensemble methods. Section 3 describes our methodology, including the computation of LI and our ensemble approach. Section 4 presents our experimental setup and results across 15 diverse datasets. Finally, Section 5 concludes the paper and discuss future directions for research.

2 RELATED WORK

2.1 Few-shot Learning in Graph Neural Networks

Recent studies have begun to explore few-shot learning on graphs, adapting meta-learning approaches to graph-structured data.

Meta-GNN (Zhou et al. 2019) [10] proposes a graph meta-learning framework that incorporates the MAML algorithm with graph neural networks for few-shot node classification. Similarly, RALE (Liu et al. 2021) [11] introduces a method that captures node dependencies through relative and absolute location embeddings, leveraging hub nodes to model long-range dependencies in graph data.

However, these approaches face several limitations. Firstly, they often do not perform consistently well across different types of graph structures, particularly struggling to generalize between homophilous and heterophilous graphs. This inconsistency highlights the need for models or model selection processes that can adapt to varied graph properties.

Secondly, these methods typically require a k-shot setting, where exactly k labeled samples per class are available for the support set. For instance, Meta-GNN and RALE evaluate on 1-shot, 3-shot, and 5-shot scenarios across different datasets. This constraint limits the models' flexibility in handling real-world scenarios where the number of available labeled samples can vary widely across different classes, especially for novel or rare categories.

The limitations of existing approaches underscore the need for more versatile models that can perform well on both homophilous and heterophilous graphs, while also relaxing the strict k-shot assumption. Ideally, such models should be able to handle variable and potentially very small numbers of labeled samples per class, reflecting the diversity of real-world data distributions.

Furthermore, with only a small number of labeled examples available in few-shot learning scenarios, traditional model selection processes become unreliable. This makes it challenging to choose or adapt the most appropriate model for a given graph structure, especially when the underlying graph properties are unknown or diverse. These challenges highlight the gaps in current research and motivate the development of more adaptive and robust approaches to few-shot learning on graphs.

2.2 Ensemble Methods for Graph Data

Recent studies have explored ensemble techniques to improve the performance and robustness of graph-based models. GraphMerge (Hou et al. 2021) [7] proposes a graph ensemble technique that combines dependency trees from different parsers to mitigate parsing errors in sentiment analysis tasks. Reliable Data Distillation (RDD) (Zhang et al. 2020) [8] introduces a framework that captures node and edge reliability in Graph Convolutional Networks (GCNs) to make better use of high-quality data. Graph Ensemble Learning (GEL) [9] (Lin et al. 2022) presents a knowledge-passing strategy to construct an ensemble model with interactions among base models.

However, these ensemble methods face significant challenges in extremely few-labeled scenarios. GraphMerge, while effective for combining multiple parse trees, is designed for sentiment analysis tasks with sufficient labeled data. RDD and GEL, though aimed at semi-supervised learning, still require a certain amount of labeled data to function effectively. For instance, RDD uses 20 instances per class as labeled data for citation networks, which may not be available in extreme few-shot scenarios.

In essence, while these ensemble methods show promise for graph data, their effectiveness is significantly hampered in scenarios with very few labeled samples. Developing ensemble techniques that can operate reliably and effectively with highly variable and extremely limited labeled data remains an open challenge in graph-based few-shot learning.

2.3 Label Informativeness (LI)

Label Informativeness (LI) [12] is a measure that quantifies how much information a neighbor's label provides about a node's label in a graph. LI has emerged as an important concept in understanding the predictive power of local graph neighborhoods. Unlike traditional measures such as homophily, LI provides insights into the performance potential of different GNN architectures.

Previous study [12] has primarily shown that higher LI correlates with improved performance of GCNs. However, this study has not fully leveraged LI to dynamically enhance model selection and prediction accuracy, particularly in few-shot learning scenarios. Our work addresses this gap by proposing an ensemble method that utilizes LI to inform model selection, thereby optimizing performance even with limited labeled data.

2.4 Summary/Reflection

Our review of the literature reveals several key limitations in current approaches to graph-based learning with limited labeled data. Existing few-shot learning methods for graphs, such as Meta-GNN and RALE, often struggle to perform consistently across different graph structures, particularly between homophilous and heterophilous graphs. These methods typically require a k-shot setting, constraining their applicability in real-world scenarios where the number of labeled samples may vary widely across classes.

Current ensemble methods for graphs, including GraphMerge, RDD, and GEL, while promising, face significant challenges in extremely label-scarce environments. These methods often require a substantial amount of labeled data to function effectively, which may not be available in extreme few-shot scenarios. Additionally, traditional model selection processes become unreliable with very few labeled examples, making it difficult to choose the most appropriate model for a given graph structure.

The concept of Label Informativeness (LI) has shown potential in understanding the predictive power of local graph neighborhoods and correlating with the performance of GNNs. However, existing research has not fully leveraged LI for dynamic model selection and performance enhancement, particularly in few-shot learning scenarios.

These gaps in the current research landscape highlight the need for more versatile and robust approaches to few-shot learning on graphs. There is a clear demand for methods that can perform well on both homophilous and heterophilous graphs, handle variable numbers of labeled samples per class, and operate effectively in extremely label-scarce environments. Furthermore, the potential of LI in guiding model selection and improving prediction accuracy in few-shot scenarios remains largely unexplored, presenting an opportunity for novel contributions in this field.

3 METHODS

3.1 Problem Statement and Assumptions

In this paper, we address the challenge of node classification in graphs with extremely limited labeled data, specifically with randomly selected labeled nodes [5,10,20,30]. Unlike traditional few-shot learning approaches, which often assume a fixed number of samples per class, our situation does not require such assumptions.

Let $G = (V, E)$ be an undirected graph, where V is the set of nodes and E is the set of edges. Each node $v_i \in V$ has a feature vector $\mathbf{x}_i \in \mathbb{R}^d$ and belongs to one of C classes. We define the following:

- $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ is the set of all node features, where $n = |V|$.
- $Y = \{y_1, y_2, ..., y_n\}$ is the set of all node labels, where $y_i \in \{1, 2, ..., C\}$.
- $V_l \subset V$ is the set of labeled nodes, where $|V_l| \ll |V|$.
- $V_u = V \setminus V_l$ is the set of unlabeled nodes.
- X_l and Y_l are the features and labels of the nodes in V_l, respectively.
- X_u are the features of the nodes in V_u.

Our goal is to learn a function $f : \mathbb{R}^d \to \{1, 2, ..., C\}$ that can accurately predict the labels of nodes in V_u, given the graph structure G, all node features X, and the labeled set (X_l, Y_l).

3.2 Definition of Label Informativeness (LI)

Label Informativeness (LI) [12] is defined as:

$$LI := \frac{I(y_\xi, y_\eta)}{H(y_\xi)} \tag{1}$$

where y_ξ and y_η are the class labels of two connected nodes ξ and η, $I(y_\xi, y_\eta)$ is the mutual information between y_ξ and y_η, and $H(y_\xi)$ is the entropy of y_ξ.

More specifically, LI_edge is defined as:

$$LI\_edge = -\frac{\sum_{c_1,c_2} p(c_1, c_2) \log \left(\frac{p(c_1,c_2)}{p(c_1) \cdot p(c_2)} \right)}{\sum_c p(c) \log p(c)} \tag{2}$$

where $p(c_1, c_2)$ is the joint distribution of labels for connected nodes, and $p(c)$ is the degree-weighted distribution of class labels.

LI ranges from 0 to 1, with higher values indicating that neighboring labels are more informative about a node's label. The paper [12] proposes LI as a complementary measure to homophily for characterizing graph datasets, particularly for distinguishing different types of heterophilous graphs.

3.3 LI preservation in Few-Shot Settings

To verify the preservation of Label Informativeness (LI) in few-shot learning scenarios, we conducted a comprehensive analysis across 10 diverse datasets. For each dataset, we computed LI under varying numbers of labeled nodes: 5, 10, 15, 20, 25, 30, and the full ground truth.

Our methodology for this analysis was as follows:

1. For each labeling scenario (5 to 30 labeled nodes), we trained a Graph Convolutional Network (GCN) using only the available labeled data.
2. We used the trained GCN to predict labels for all unlabeled nodes.
3. Using these predicted labels, we computed the LI of the graph.
4. We repeated this process 10 times for each scenario to ensure robustness and averaged the results.
5. Finally, we compared these LI values to the LI computed using the full ground truth labels.

Figure 1 illustrates the results of this analysis.

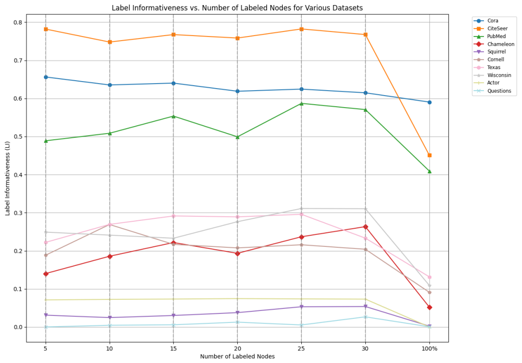

Figure 1: Li - number of labeled data

As evident from Figure 1, the LI values remain relatively stable across different numbers of labeled nodes for most datasets. This stability is particularly notable when comparing the LI computed with few labeled nodes (as low as 5-10) to the LI computed with full ground truth labels.

For instance, datasets like Cora, Actor and Questions show remarkable consistency in their LI values across all labeling scenarios. Even for datasets with more variability, such as Chameleon or Squirrel, the overall trend of LI remains consistent, and the values computed with few labels are generally indicative of the full-label LI.

This preservation of LI in few-shot settings is crucial for our methodology, as it allows us to reliably estimate the graph's structural properties and guide model selection even when only a small fraction of nodes are labeled. We hypothesize that LI remains stable for several reasons:

1. Even with few labeled nodes, the GCN used to predict labels for unlabeled nodes may be capturing enough of the graph's structure to maintain a consistent LI score. This suggests that GCN's predictions, even with limited training data, reflect the inherent label relationships in the graph.
2. The stability of LI across different labeling scenarios indicates that the predicted labels from GCN maintain consistent patterns of label relationships, even with few-labeled case.

These factors likely contribute to the observed stability of LI across different labeling scenarios, making it a reliable metric for model selection in few-shot learning contexts.

3.4 Model Selection Based on LI

Our analysis revealed a significant relationship between Label Informativeness (LI) and the performance of various Graph Neural Network (GNN) architectures. To investigate this relationship, we conducted a comprehensive study across multiple datasets:

- We split data into labeled data and unlabeled data. Only 10 labels were randomly assigned, with the remaining nodes used exclusively for testing.
- We computed the ground truth LI score for each dataset.
- We trained and evaluated five different models (GCN, GraphSAGE, GAT, GIN and DNN) on each dataset.
- Each model was trained and evaluated 10 times per dataset, and the results were averaged to ensure reliability.
- We compared the test accuracy of each model against the LI score of the corresponding dataset.

Figure 2: Model performance comparison across datasets with varying LI scores.

Dataset	LI Score	GCN	DNN	GraphSAGE	GAT	GIN	Avg Acc	Winner	2nd Place
PATTERN	0.0020	0.6852	0.6477	0.6795	0.6545	0.5386	0.6411	GCN (0.6852)	GraphSAGE (0.6795)
Squirrel	0.0023	0.2273	0.2535	0.2926	0.1964	0.2542	0.2448	GraphSAGE (0.2926)	GIN (0.2542)
CLUSTER	0.0377	0.6350	0.2933	0.4133	0.3367	0.4083	0.4173	GCN (0.6350)	GraphSAGE (0.4133)
Wisconsin	0.1092	0.3348	0.7577	0.6040	0.3423	0.2970	0.4672	DNN (0.7577)	GraphSAGE (0.6040)
Roman-empire	0.2108	0.2085	0.4461	0.4627	0.1858	0.1635	0.2933	GraphSAGE (0.4627)	DNN (0.4461)
PubMed	0.4093	0.7886	0.7300	0.7609	0.7714	0.7412	0.7584	GCN (0.7886)	GAT (0.7714)
Computers	0.5279	0.7858	0.5745	0.6780	0.7786	0.0793	0.5792	GCN (0.7858)	GAT (0.7786)
Cora	0.5904	0.8074	0.5712	0.8006	0.7984	0.7490	0.7453	GCN (0.8074)	GraphSAGE (0.8006)
Photo	0.6662	0.8446	0.7010	0.8016	0.8560	0.2283	0.6863	GAT (0.8560)	GCN (0.8446)

Figure 2 illustrates the results of this analysis. As evident from Figure 2, several key observations emerge:

- For datasets with higher LI scores (approximately > 0.35), GCN tends to perform better than or comparably to other models. This is particularly noticeable for datasets like PubMed, Computers, and Cora.
- For datasets with lower LI scores (approximately < 0.35), GraphSAGE often outperforms or matches the performance of other models. This is visible for datasets such as Roman-empire, Squirrel and PATTERN.
- GAT and DNN show variable performance across different LI scores, but generally do not consistently outperform GCN or GraphSAGE in their respective LI ranges.
- There is a general trend of increasing model accuracy as the LI score increases, regardless of the model used. This suggests that graphs with higher LI are generally easier for all models to learn from.

We hypothesize that the observed relationship between LI and model performance can be explained as follows:

(1) For graphs with higher LI (> 0.35), GCN's performance is superior likely due to its effective aggregation of information from immediate neighbors. When LI is high, a node's label is strongly related to its neighbors' labels, making this local aggregation particularly effective.
(2) For graphs with lower LI (< 0.35), GraphSAGE's better performance could be attributed to its sampling techniques that aggregate information from a wider neighborhood. When immediate neighbors' labels are less informative (low LI), this broader perspective may be more beneficial.
(3) The different architectures of these GNNs (e.g., how they aggregate and transform node features) may be more or less suited to graphs with different LI values, depending on how much they rely on local vs. global information.

These observations not only explain our empirical results but also provide insights into the relationship between graph properties and GNN architecture performance. This understanding guides our model selection strategy and opens avenues for future research in developing more adaptive graph learning algorithms.

Based on these observations, we propose an LI-guided model selection strategy:

- Compute the graph's LI using the limited labeled data and GCN predictions.
- If LI > 0.35, select GCN as the final prediction model.
- If LI <= 0.35, select GraphSAGE as the final prediction model.

While there might be more sophisticated selection strategies, we chose the simplest approach to demonstrate that simplicity can be effective. This strategy allows us to choose the most suitable model for a given graph structure without requiring extensive testing or validation data, which is particularly valuable in few-shot learning scenarios. By leveraging the LI score, we can make an informed decision about which model is likely to perform best, even when we have very limited labeled data to work with.

It is important to note that more variations in train/test splits may be required to fully confirm the robustness of this strategy. These variations can be thoroughly examined in the experimental apparatus (next) section.

3.5 Ensemble Learning Algorithm

Our proposed ensemble learning algorithm combines the insights from LI preservation and LI-based model selection. The algorithm proceeds at Algorithm 1.

Algorithm 1 Model Selection and Prediction for Graph Nodes

```
Input: Graph $G$, labeled nodes $V_L$, labels $Y_L$
Output: Predicted labels for all unlabeled nodes

Train GCN: Train a Graph Convolutional Network (GCN) using the labeled nodes $V_L$ and their corresponding labels $Y_L$.
Predict using GCN: Use the trained GCN to predict labels for all unlabeled nodes in the graph.
Compute Label Information (LI): Calculate the Label Information (LI) using the predicted labels.
Check LI value:
if LI > 0.35 then
    Select GCN: Select the trained GCN as the final model.
else
    Train GraphSAGE: Train a GraphSAGE model using the labeled nodes $V_L$ and their corresponding labels $Y_L$.
    Select GraphSAGE: Select the trained GraphSAGE model as the final model.
end if
Final Predictions: Use the selected model (either GCN or GraphSAGE) to make final predictions for all unlabeled nodes in the graph.
```

This ensemble approach leverages the strengths of both GCN and GraphSAGE, adapting to the structural properties of the input graph as captured by LI. By doing so, we can achieve superior performance across a wide range of graph types, even in extremely label-scarce scenarios. Additionally, this ensemble model is designed to be adaptable, meaning that if a new state-of-the-art (SOTA) node classification model emerges, our ensemble approach can incorporate this new model. As long as the new SOTA model is influenced by the LI of the input graph, our method can dynamically select it based on the LI threshold.

3.6 Summary

We confirm that Label Informativeness (LI) remains stable in few-labeled scenarios across 10 diverse datasets, making it a reliable measure for guiding model selection. Our analysis reveals a relationship between LI and the performance of different Graph Neural Network (GNN) architectures: GCN performs better for datasets with LI > 0.35, while GraphSAGE is more suitable for datasets with LI < 0.35. Based on these findings, we propose an ensemble learning method that first trains a GCN on the limited labeled data, computes the graph's LI using GCN predictions, and then selects either GCN or GraphSAGE as the final prediction model depending on the LI threshold of 0.35. This approach leverages the strengths of both models, ensuring optimal performance in few-shot learning scenarios.

4 EXPERIMENTAL APPARATUS

4.1 Datasets

Our experiments were conducted on a diverse set of 15 graph datasets to evaluate the performance of our proposed method across various graph structures and applications. Figure 3 provides an overview of the datasets used in this study, along with their computed Label Informativeness (LI) scores.

These datasets represent a broad spectrum of graph types and properties, allowing us to thoroughly evaluate the effectiveness and generalizability of our proposed method. The datasets include:

- Citation networks (Cora, CiteSeer, PubMed) representing scientific publications and their citations.
- Co-purchase networks (Computers, Photo) from Amazon, showing product relationships.
- Wikipedia networks (Chameleon, Squirrel) based on Wikipedia pages and their links.
- University website graphs (Cornell, Texas, Wisconsin) from the WebKB project.
- An actor co-occurrence network.
- Social and historical networks (Questions, Roman-empire) from the HeterophilousGraphDataset.
- Synthetic benchmark graphs (CLUSTER, PATTERN) for controlled experiments.

The Label Informativeness (LI) scores provided in Table 3 range from very low (0.0020 for PATTERN) to relatively high (0.6662 for Photo), indicating a wide variety of graph structures in our dataset collection. This diversity allows us to test our method's performance across various graph structures, sizes, and domains, providing a comprehensive evaluation of its effectiveness and robustness.

4.2 Procedure

The primary objective of our experiments is to demonstrate the effectiveness of our ensemble model in dynamically selecting the optimal graph neural network architecture based on the calculated Label Informativeness (LI) score, particularly in few-labeled scenarios.

Our experimental procedure is as follows:

1. **Data Splitting:** For each dataset, we randomly split the data into labeled and unlabeled sets. The labeled set is used for training, while the unlabeled set serves as our test set.
2. **Few-Shot Learning Scenarios:** To simulate few-shot learning conditions, we vary the number of labeled nodes. We conduct separate experiments with 5, 10, 20, and 30 labeled nodes for each dataset.
3. **Model Training:** We train five different models using only the labeled data:
   - Graph Convolutional Network (GCN)
   - Graph Isomorphism Network (GIN)
   - Graph Attention Network (GAT)
   - GraphSAGE
   - Our proposed ensemble model (LGEL)
4. **LI Computation:** For our ensemble model, we compute the Label Informativeness (LI) score using the method described in Section 3.5.
5. **Model Selection:** Based on the computed LI score, our ensemble model dynamically selects either GCN or GraphSAGE as described in Section 3.5.
6. **Evaluation:** We evaluate the test accuracy of all models on the unlabeled data.
7. **Cross-Dataset Comparison:** We compute the average performance of each model across all 15 datasets for each few-shot scenario (5, 10, 20, and 30 labeled nodes).
8. **Repetition and Averaging:** Steps 1-7 are repeated 10 times with different random splits to ensure robustness. The results are averaged across these runs.

This procedure allows us to compare the performance of our ensemble model against individual GNN architectures (GCN, GIN, GAT, and GraphSAGE) across various datasets and few-shot learning scenarios. By doing so, we aim to demonstrate that our ensemble approach can effectively adapt to different graph structures and consistently outperform individual models, especially in situations with very limited labeled data.

4.3 Evaluation metrics

To evaluate the performance of our models across different datasets and few-shot learning scenarios, we use the following metrics:

- **Test Accuracy:** Our primary metric is the classification accuracy on the test set (unlabeled nodes). It is calculated as:

$$\text{Accuracy} = \frac{\text{Number of correctly classified nodes}}{\text{Total number of nodes in the test set}} \tag{3}$$

- **Standard Deviation:** We report the standard deviation of accuracies across 10 attempts to measure the consistency of each model's performance.

These metrics allow us to comprehensively evaluate our ensemble model's performance, not just in terms of overall accuracy, but also in its ability to adapt to different graph structures and its efficiency in few-shot learning scenarios.

5 RESULTS

5.1 Overall Performance Comparison

We present the overall performance of our ensemble model compared to individual GNN architectures across different few-shot

Figure 3: Overview of datasets used in the experiments with their LI scores

Dataset	LI Score	Category	Source	Description
PATTERN	0.0020	Synthetic	GNNBenchmarkDataset	Synthetic patterned graph
Squirrel	0.0023	Wikipedia Network	WikipediaNetwork	Wikipedia pages and links
CLUSTER	0.0377	Synthetic	GNNBenchmarkDataset	Synthetic clustered graph
Wisconsin	0.1092	University Website	WebKB	University webpages
Roman-empire	0.2108	Historical Network	HeterophilousGraphDataset	Roman Empire political relations
PubMed	0.4093	Citation Network	Planetoid	Medical publications
Computers	0.5279	Co-purchase Network	Amazon	Amazon product co-purchasing
Cora	0.5904	Citation Network	Planetoid	Computer science publications
Photo	0.6662	Co-purchase Network	Amazon	Amazon product co-purchasing
CiteSeer	0.4508	Citation Network	Planetoid	Computer science publications
Chameleon	0.0522	Wikipedia Network	WikipediaNetwork	Wikipedia pages and links
Cornell	0.0911	University Website	WebKB	University webpages
Texas	0.1316	University Website	WebKB	University webpages
Actor	0.0003	Co-occurrence Network	Actor	Actor co-occurrences in movies
Questions	0.0007	Social Network	HeterophilousGraphDataset	Stack Exchange question tags

learning scenarios. Table 1 shows the average accuracy and standard deviation for each model across all 15 datasets.

Table 1: Average accuracy (%) and standard deviation for different models across all datasets

Model	5 Labels	10 Labels	20 Labels	30 Labels
GCN	0.4884 ± 0.0142	0.5190 ± 0.0150	0.5411 ± 0.0173	0.5858 ± 0.0244
GIN	0.3820 ± 0.0361	0.4125 ± 0.0241	0.4363 ± 0.0279	0.4672 ± 0.0254
GAT	0.4562 ± 0.0381	0.4937 ± 0.0305	0.5422 ± 0.0177	0.5853 ± 0.0143
GraphSAGE	0.5009 ± 0.0226	0.5588 ± 0.0119	0.6114 ± 0.0122	0.6557 ± 0.0235
DNN	0.4755 ± 0.0217	0.5207 ± 0.0098	0.5609 ± 0.0187	0.6148 ± 0.0220
Our Ensemble	**0.5213 ± 0.0234**	**0.5754 ± 0.0081**	**0.6203 ± 0.0094**	**0.6606 ± 0.0233**

Our ensemble model consistently outperforms individual GNN architectures across all few-shot learning scenarios, with the highest improvement observed in the most label-scarce condition (5 labels).

6 DISCUSSION

6.1 Key Scientific Insights

Our results provide several key insights:

- Our ensemble model, which dynamically selects between GCN and GraphSAGE based on LI, consistently outperforms individual GNN architectures across various few-shot learning scenarios.
- The performance improvement of our ensemble model is most pronounced in extremely label-scarce conditions (5 labels), where it achieves an average accuracy of 52.13% compared to 50.09% for the next best model (GraphSAGE).

6.2 Threats to Validity

While our results are promising, we acknowledge potential threats to validity:

- Dataset Bias: Our 15 datasets, while diverse, may not represent all possible graph structures. However, the inclusion of synthetic datasets (PATTERN, CLUSTER) helps mitigate this concern.
- Random Seed Sensitivity: To address this, we repeated each experiment 10 times with different random splits and reported average results with standard deviations.
- LI Threshold Sensitivity: The 0.35 threshold for model selection was determined empirically and may not be optimal for all datasets. Future work could explore adaptive thresholding techniques.

6.3 Generalization

We believe our findings generalize well for several reasons:

- Our datasets span various domains (citation networks, social networks, co-purchase networks, etc.) and exhibit a wide range of LI scores (0.0020 to 0.6662).
- The consistency of our results across different few-shot scenarios (5, 10, 20, 30 labels) suggests robustness to varying levels of label scarcity.
- Our method's reliance on LI, a fundamental graph property, rather than domain-specific features, enhances its potential for generalization to new datasets and domains.

6.4 Future Work and Impact

Future research directions include:

- Exploring the integration of other GNN architectures into our ensemble framework.
- Investigating adaptive thresholding techniques for model selection.
- Extending our approach to other graph learning tasks, such as link prediction or graph classification.

The impact of our work extends beyond academic interest. By enabling effective learning from extremely limited labeled data, our method could significantly reduce the cost and effort of data annotation in real-world applications of graph machine learning, from social network analysis to biological network studies.

7 CONCLUSION

In this paper, we introduced a novel ensemble learning method for few-shot node classification in graphs, leveraging the concept of Label Informativeness (LI). Our approach demonstrates that even with extremely limited labeled data (as few as 5 nodes), it is possible to make informed decisions about model selection and achieve superior performance compared to individual GNN architectures.

The preservation of LI in few-shot settings and its correlation with the performance of different GNN architectures provide valuable insights into the nature of graph learning. These findings not only contribute to the theoretical understanding of GNNs but also offer practical guidelines for model selection in real-world scenarios where labeled data is scarce.

Our work opens up new avenues for research in few-shot learning on graphs and has the potential to significantly impact applications where data annotation is costly or challenging. As graph-structured data becomes increasingly prevalent across various domains, methods that can learn effectively from limited labeled data will be crucial in harnessing the full potential of this rich data structure.

8 LIMITATIONS

Despite the promising results, our work has several limitations:

- Scalability: Our current implementation may face challenges with extremely large graphs due to the computational complexity of LI calculation. Future work should explore more efficient approximation methods for LI in large-scale graphs.
- Model Diversity: Our ensemble currently considers only GCN and GraphSAGE. While these represent distinct approaches to graph learning, incorporating a wider range of GNN architectures could potentially yield further improvements.
- Task Specificity: Our method is currently focused on node classification tasks. Its effectiveness for other graph learning tasks, such as link prediction or graph classification, requires further study.
- Theoretical Guarantees: While we provide empirical evidence for the effectiveness of our approach, theoretical guarantees on the performance improvement are yet to be established.

Addressing these limitations presents exciting opportunities for future research and the further development of few-shot learning methods for graph-structured data.

REFERENCES

[1] LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. *Nature*, 521(7553), pp.436-444.

[2] Kipf, T.N. and Welling, M., 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR).*

[3] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. and Bengio, Y., 2018. Graph attention networks. In *International Conference on Learning Representations (ICLR).*

[4] Xu, K., Hu, W., Leskovec, J. and Jegelka, S., 2018. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR).*

[5] Hamilton, W., Ying, Z. and Leskovec, J., 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems* (pp. 1024-1034).

[6] Kipf, T.N. and Welling, M., 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR).*

[7] Hou, X., Qi, P., Wang, G., Ying, R., Huang, J., He, X. and Zhou, B., 2021. Graph ensemble learning over multiple dependency trees for aspect-level sentiment classification. *arXiv preprint arXiv:2103.11794.*

[8] Zhang, W., Miao, X., Shao, Y., Jiang, J., Chen, L., Ruas, O. and Cui, B., 2020. Reliable data distillation on graph convolutional network. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data* (pp. 1399-1414).

[9] Lin, Q., Yu, S., Sun, K., Zhao, W., Alfarraj, O., Tolba, A. and Xia, F., 2022. Robust graph neural networks via ensemble learning. *Mathematics*, 10(8), p.1300.

[10] Zhou, F., Cao, C., Zhang, K., Trajcevski, G., Zhong, T. and Geng, J., 2019. Meta-GNN: On Few-shot Node Classification in Graph Meta-learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 2357-2360).

[11] Liu, Z., Fang, Y., Liu, C. and Hoi, S.C., 2021. Relative and Absolute Location Embedding for Few-Shot Node Classification on Graph. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 5, pp. 4267-4275).

[12] Platonov, O., Kuznedelev, D., Babenko, A. and Prokhorenkova, L., 2023. Characterizing Graph Datasets for Node Classification: Homophily–Heterophily Dichotomy and Beyond. *arXiv preprint arXiv:2209.06177.*