Enhancing Explanations of Graph Neural Networks via Bridging Model-level and Instance-level Explainers

Youmin Zhang[1], Qun Liu[1]✉, Guoyin Wang[2], Lili Yang[3], and Li Liu[1]

[1] Key Laboratory of Big Data Intelligent Computing, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China
[2] National Center for Applied Mathematics in Chongqing, Chongqing Normal University, Chongqing, 401331, China
[3] Department of Statistics and Data Science, Southern University of Science and Technology, Shenzhen, 518000, China
Corresponding Author: Qun Liu. `liuqun@cqupt.edu.cn`

Abstract. Current studies on explaining graph neural networks (GNNs) are proposed separately at the model or instance levels, each offering unique insights into GNNs' prediction behaviors. Few studies have explored the potential of bridging multiple-level explainers to enhance explanation quality. To fill this gap, we propose IMOE, a simple yet effective framework that Incorporates Model-level explanations into the Optimization of instance-level Explainers. IMOE learns model-level explanations using GFlowNet. It further extracts representative prototype graph patterns with appropriate diversity via graph clustering, mitigating noise and computational overhead associated with large amounts of graph patterns. Moreover, IMOE leverages the prototype graph patterns to guide the optimization of the instance-level explainer, offering global information for learning faithful explanations. Experiments conducted on four datasets demonstrate the effectiveness and generalization of IMOE. Qualitative studies emphasize IMOE's ability to generate explanations that align with human intuition and domain knowledge. Data and code are available at https://anonymous.4open.science/r/IMOE-0FFC.

Keywords: Graph Data Mining · Graph Neural Networks · Model Explainability · Model-level Explanation · Instance-level Explanation.

1 Introduction

Graph neural networks (GNNs) have emerged as powerful tools for learning rich representations of ubiquitous graph data [5]. Despite GNNs' impressive performance, they face issues in their expandability [12]. The black-box nature of GNNs hinders their adoption in high-stakes domains. Recently, there has been significant interest in developing GNN explainers. Studies can be categorized into model-level and instance-level approaches.

Instance-level explainers offer fine-grained understandings of the prediction for specific input instance [12, 16, 10], identifying the most influential subset of a

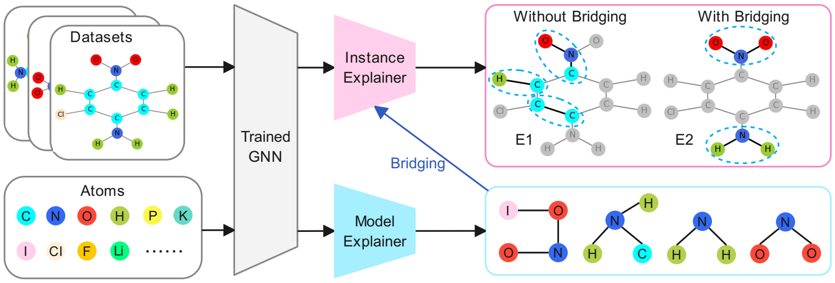


Fig. 1: Motivation of the proposed IMOE.

graph for the GNN's prediction. While they provide valuable local insights, they lack a broader understanding of class-aware characteristics at model-level. Without capturing the global patterns learned by GNNs, the instance-level explainers may learn explanations with noise and spurious relationships (E1 in Fig. 1).

Model-level approaches generate graph patterns with the highest predicted probability for a specific class [13, 3]. As shown in the bottom right of Fig. 1, model-level explanations offer a general perspective on GNN predictions through subgraph patterns. However, existing model-level explanation techniques often provide coarse-grained graph patterns, limiting their ability to offer detailed insights into predictions made for individual graphs in the datasets.

Model-level explanations provide "global" graph patterns for different classes, while instance-level explainers can adopt these patterns as clues to learn fine-grained explanations for each instance. By bridging multiple-level explanations, we aim to refine the instance explanation from E1 to E2 (upper right of Fig. 1). However, challenges arise in ensuring an appropriate level of diversity in graph patterns to enhance instance-level explainers. Insufficient diversity may fail to capture the full range of subgraph patterns within input graphs, while excessive diversity can introduce redundant noise and incur computational overhead. Additionally, designing objectives to optimize instance-level explainers given multiple subgraph patterns remains an open question. Balancing the trade-off between the generality of model-level explanations and the specificity of instance-level explanations is essential for generating faithful and comprehensive explanations.

To this end, we propose IMOE that Incorporates Model-level explanations into the Optimization of instance-level Explainers. IMOE first uses GFlowNet to learn diverse subgraph pattern candidates as model-level explanations. Second, the DBSCAN model is applied to identify prototype graph patterns, mitigating noise and computational overhead associated with highly diverse explanations. Finally, we introduce a loss function that leverages the prototype graph patterns to guide the optimization of the instance-level GNN explainer, ensuring more faithful learned explanations. By leveraging prototype graph patterns to optimize instance-level explainers, IMOE bridges the multi-level explainers, enabling faithful explanations for GNNs. Experiments conducted on four datasets demonstrate that IMOE enhances the quality of explanations across various metrics, and the generalization of several instance-level explainers. Qualitatively, IMOE produces explanations that align with human intuition and domain knowledge.

2 Related Works

Model-level explainers generate subgraph patterns with high predicted probability for a specific class. PAGE [9], GCFExplainer [2], and Glocal-Explainer [8] learn explanations from datasets. Methods such as clustering and KNN [9], high-frequent subgraphs mining [8], and vertex-reinforced random walks [2] are adopted. Generally, they lack generalization to unseen instances. In contrast, XGNN [13] constructs a graph generator and employs reinforcement learning to learn explanations. XInsight [3] adopts GFlowNets to learn diverse explanations proportional to a reward distribution. However, without properly managing the diversity of explanations, XInsight may introduce noise and computational costs.

Instance-level explainers identify significant subgraphs for the prediction of a specific instance. The masking strategy is widely adopted, including GNNExplainer [12], and PGExplainer [7]. Other approaches learn explanations using graph generators such as SubgraphX [14]. Furthermore, GNN-MOExp [6] and CF^2 [10], balance factual and counterfactual subgraphs, treating them as necessary and sufficient conditions. Recently, studies leverage "global" subgraph patterns to guide the optimization of instance-level explainers, including MOE [18], Gem [4], and ReFine [11]. High-frequency subgraphs [18], knowledge distillation [4], and class-aware tailored PGExplainer [11] are adopted for generating "global" subgraph patterns.

Studies [18, 4, 11] demonstrate the effectiveness of using "global" subgraph patterns to guide instance-level explainer learning. However, these approaches derive "global" patterns from specific data, which may not capture global information and can cause the instance-level explanations to overfit the patterns of specific instances. Our approach learns subgraph patterns solely from model-level explainers, independent of datasets. We apply clustering algorithms to derive more general prototype subgraph patterns, obtaining robust supervision.

3 Model Framework

The IMOE aims to construct an explainer $g(A, X; \Theta)$ that produces a mask matrix M and considers the subgraph $M \odot A$ as an explanation, where A is the adjacency matrix and X represents the features. IMOE consists of a model-level explainer that generates diverse subgraphs, which are subsequently refined as prototype explanations via clustering (Fig. 2.A). An instance-level explainer g identifies explanations for each instance (Fig. 2.B), with its optimization guided by the prototype explanations (Fig. 2.C).

3.1 Learning Model-level Explanations

We use GFlowNet, known for effectively learning diverse subgraphs [3], to produce model-level explanations. GFlowNet consists of initial state s_0 (a random

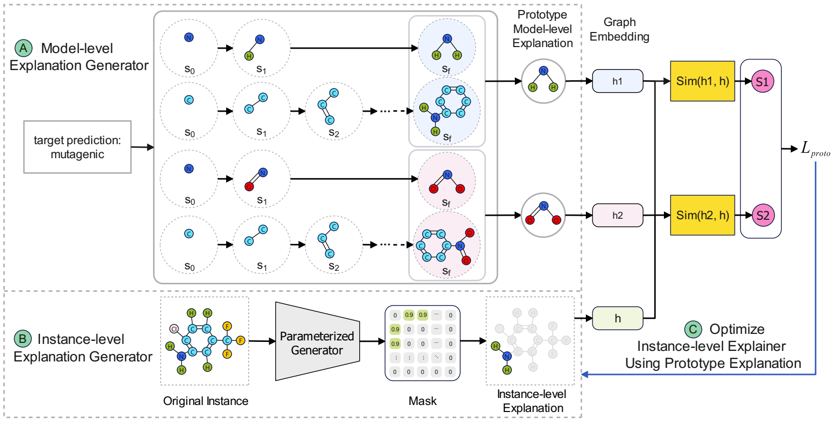

A Model-level Explanation Generator
target prediction: mutagenic
s_0
s_1
s_2
s_f
Prototype Model-level Explanation
Graph Embedding
h1
h2
h
Sim(h1, h)
Sim(h2, h)
S1
S2
L_{proto}
B Instance-level Explanation Generator
Original Instance
Parameterized Generator
Mask
Instance-level Explanation
C Optimize Instance-level Explainer Using Prototype Explanation

Fig. 2: Architecture of the proposed IMOE

node), intermediate states s_i, and the terminal state s_f (explanation). The inflow is actions leading to a state, while the outflow is actions transitioning to the next state. The terminal state is reached when the inflow equals the reward.

The transition from s_t to s_{t+1} depends on action $A(n_s, n_e)$, where n_s is the starting node from s_t, and n_e is the ending node sampled from specific node types. Action $A(n_s, n_e)$ is a policy network $\pi(A(n_s, n_e)|s_t)$, instantiated by a 3-layer GCN (dimensions: 32, 64, 128). To train the policy network, we define the reward as the probability $f(s)$ of target class c, as shown in Eq. (1).

$$R(s) = \begin{cases} f(s) & f(s) = c \\ 0 & f(s) \neq c \end{cases} \tag{1}$$

The reward $R(s)$ is incorporated into the flow consistency condition to form the objective Eq.(2). $F(s_p \to s)$ and $F(s \to s_c)$ represent the inflow from parent state s_p and outflow to child state s_c, respectively. $R(s)$ is 0 for intermediate states and the reward value for terminal states. τ denotes a trajectory.

$$\mathcal{L}_g(\tau) = \sum_{s \in \tau} \left(\sum_{s_p \in Par(s)} F(s_p \to s) - R(s) - \sum_{s_c \in Child(s)} F(s \to s_c) \right) \tag{2}$$

The objective function ensures outflow equaling inflow for intermediate states, and terminal state's inflow matching the graph's reward, allowing the model to learn high-reward explanatory graphs. During training, multiple trajectories τ are sampled, promoting diverse explanatory graph exploration. GFlowNet generates explanations for each class c, denoted as $ME_c^{candidates}$. To filter redundant subgraphs in $ME_c^{candidates}$, DBSCAN is applied to cluster the explanations. Cluster centroids are selected as prototype explanations ME_c^{proto}.

$$ME_c^{proto} = Cluster(ME_c^{candidates}) \quad = (ME_{c1}, ME_{c2}, ..., ME_{cj}) \tag{3}$$

3.2 Optimizing the Instance-level Explainer

For learning instance-level explanations, we design a generator g to produce a mask matrix M. The generator g consists of two components: a graph encoder and a multilayer perceptron that takes the embeddings of concatenated node pairs as input and learns a variable $\omega_i \in \varOmega$ for each edge e_i, as shown in Eq. (4).

$$\varOmega = g(A, X, \varTheta). \tag{4}$$

The $m_{pq} \in M$ is obtained using a concrete function $C(\omega_i)$ defined in Eq. (5).

$$\begin{aligned} m_{pq} = C(\omega_i, \epsilon, \tau_2) = \sigma((\log \epsilon - \log(1-\epsilon) + \omega_i)/\tau_2) \\ where\ \epsilon \sim Uniform(0,1)\ and\ \omega_i \in \varOmega \end{aligned} \tag{5}$$

where the parameter $\tau_2 = \tau_0 \left(\frac{\tau_T}{\tau_0}\right)^{i/\text{epoch}}$, and i is the current iteration. The first objective function of optimizing $\varTheta$ is shown in Eq. (6).

$$\mathcal{L}_{ori} = I(G^0, G^t) + CE(Y_c^0, P(Y_c^t|G^t)) \tag{6}$$

where G^0 is the graph and G^t is the explanation. $CE(Y_c^0, P(Y_c^t|G^t))$ is the cross-entropy of the predictions for G^0 and G^t. $I(G^0, G^t)$ is the sparsity of G^t.

To incorporate the prototype explanation into the optimization of $\varTheta$, we define Eq. (7) as a regularization term. $Sim(G^t, ME_{cj})$ represents the cosine similarity between the embeddings of G^t and ME_{cj}. As one instance may contain multiple explanatory subgraphs, we define the $Top^l(\cdot)$ function to optimize the similarity between G^t and the l most similar prototypes among ME_c^{proto}.

$$\mathcal{L}_{proto} = Top^l\Big(Sim(G^t, ME_{c1}), Sim(G^t, ME_{c2}), ..., Sim(G^t, ME_{cj})\Big) \tag{7}$$

Based on the defined regularization term, we form the objective of the instance-level explainer as Eq. (8). Parameters are optimized using Adam optimizer.

$$\mathcal{L}_{exp} = \mathcal{L}_{ori} + \mathcal{L}_{proto} \tag{8}$$

4 Experiment and Analysis

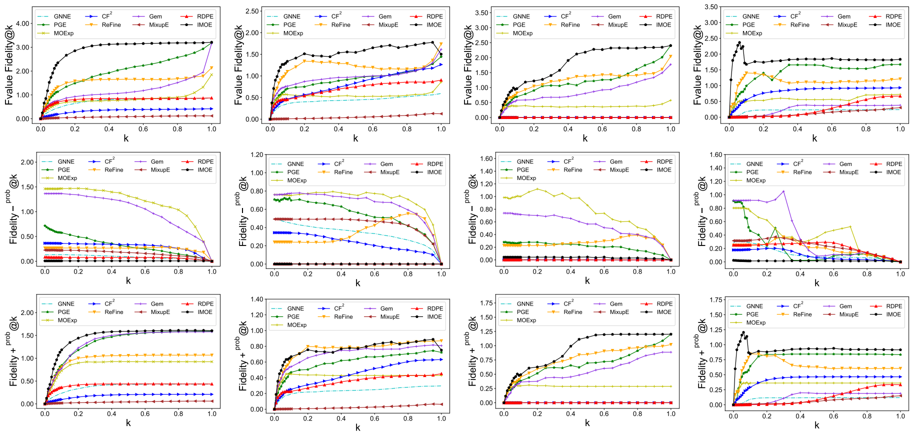
We conduct experiments on three real-world datasets and one synthetic dataset, including **Mutagenicity** [12, 7], **NCI1** [10], **PROTEINS** [2], and **BA-4Motifs**. The GNN to be explained consists of three GCN layers with dimensions of 20, 20, and 20, respectively. Table 1 lists the statistics of the datasets and the corresponding prediction accuracy. The baselines included in our experiments are: **GNNExplainer (GNNE)** [12], **PGExplainer (PGE)** [7], **GNN-MOExp (MOExp)** [6], **CF^2** [10], **ReFine** [11], **Gem** [4], **MixupExplainer (MixupE)** [15], and **RDPE** [16]. We use the *Fidelity* [12, 7, 16, 17] as metrics, including $Fidelity -^{prob}$ @k, $Fidelity +^{prob}$ @k and $Fvalue\ Fidelity$@k [16, 17].

Table 1: Statistics of the datasets and prediction accuracy of GNN

Datasets	Graphs	avg. Nodes	avg. Edges	Labels	Acc.
Mutagenicity	4337	30.32	30.77	2	0.821
NCI1	4110	29.87	32.30	2	0.752
PROTEINS	1113	39.1	145.6	2	0.696
BA-4Motifs	2000	25	83.25	2	0.913

4.1 Performance Comparison

The *Fidelity* comparisons in Fig.3 are obtained by adjusting the sparsity across two ranges: 0.01 to 0.1 in 0.01 increments, and 0.1 to 1 in 0.05 increments. IMOE achieves the best $FvalueFidelity@k$ compared to baselines without prototype model-level explanation constraints, with notable improvement in $Fidelity+^{prob}$ $@k$, especially when retains ≤10% edges, indicating more precise explanations. For $Fidelity-^{prob}@k$, IMOE performs best on Mutagenicity and second-best on others, particularly for small k, showing high consistency between explanatory subgraph and original graph predictions, demonstrating higher fidelity under high sparsity. IMOE's effectiveness is attributed to incorporating model-level explanations, enabling it to capture essential patterns contributing to the model's decisions, resulting in more faithful explanations.



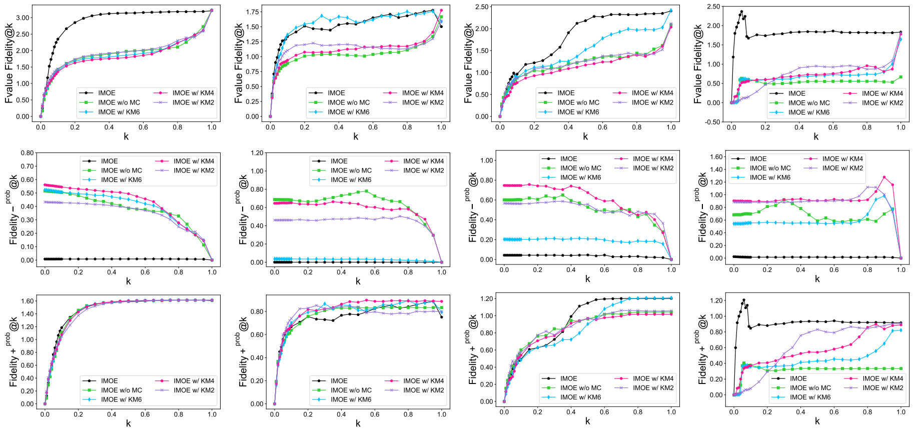
(a) Mutagenicity (b) NCI1 (c) PROTEINS (d) BA-4Motifs

Fig. 3: Performance Comparison of $Fidelity@k$

Among baselines, ReFine and Gem perform comparably for small k, showing the benefits of bridging multi-level explanations. However, their "global" explanations, generated using a class-aware tailored PGExplainer and distilling process, lack the comprehensiveness of true model-level explanations, as they are derived from individual instances. IMOE outperforms them, emphasizing the potential of integrating pure model-level explanations to enhance explanations, particularly when explanatory subgraphs are highly sparse.

4.2 Ablation Study

Ablation studies are conducted between IMOE and its variant without model-level explanation constraints, *IMOE w/o MC*. IMOE is also compared with prototype patterns learned by K-Means:*IMOE w/ KM6*, *IMOE w/ KM4*, and *IMOE w/ KM2*, where the number of clustering centers is set to 6, 4, and 2, respectively. Fig. 4 shows IMOE consistently outperforming *IMOE w/o MC* in $Fvalue-Fidelity@k$ across all datasets, indicating the critical role of prototype model-level explanation constraints. The improvement in $Fidelity\ -^{prob}\ @k$ is more pronounced than $Fidelity\ +^{prob}\ @k$, especially for small k, providing additional evidence supporting the effectiveness of IMOE's idea. K-Means underperforms DBSCAN on three datasets, only matching IMOE's performance on NCI1 with 6 clusters. Reducing clusters to 4 or 2 drops performance on NCI1. K-Means fails to achieve optimal performance and requires manual cluster number tuning. In contrast, DBSCAN automatically determines the number of clusters based on data point density. IMOE's superior performance with DBSCAN across multiple datasets demonstrates the chosen clustering algorithm's effectiveness.



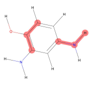
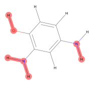
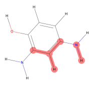
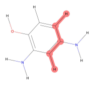
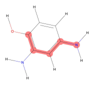
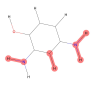
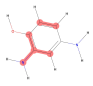
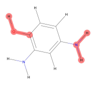
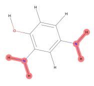
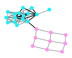
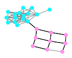
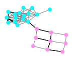
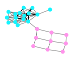
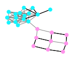
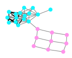
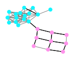
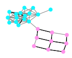
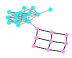
(a) Mutagenicity (b) NCI1 (c) PROTEINS (d) BA-4Motifs

Fig. 4: Ablation Study of $Fidelity@k$

4.3 Case Study

We present qualitative studies to demonstrate IMOE's effectiveness on Mutagenicity and BA-4Motifs datasets. Table 2 shows one instance per dataset, with

Table 2: Comparative Cases

Mutagenicity								
GNNE	PGE	MOExp	CF^2	ReFine	Gem	MixupE	RDPE	IMOE
								
BA-4Motifs								
GNNE	PGE	MOExp	CF^2	ReFine	Gem	MixupE	RDPE	IMOE
								

explanations highlighted in bold. For Mutagenicity with ground truths, carbon rings combined with NO_2 or NH_2 are ground truths [7, 12]. While baselines only identify one NH_2, IMOE identifies all NH_2, providing a more comprehensive understanding of mutagenicity factors. For BA-4Motifs, generated by a Barabasi-Albert graph with a ground truth motif, IMOE accurately identifies the entire "grid" motif, highlighting its ability to identify motifs under prototype model-level constraints. Qualitative observations demonstrate IMOE's effectiveness in identifying the underlying prediction rationale under prototype model-level explanation constraints.

4.4 Generalization to Current State-of-the-art Explainers

To evaluate IMOE's generalization, we optimize other instance-level explainers using prototype graph patterns, including GNNExplainer, PGExplainer, CF^2, MixupExplainer, and RDPE. Figure 5 compares the performance of instance-level explainers with and without prototype model-level explanation constraints ("*Model w/ MC*"). In most cases, incorporating these constraints improves performance or remains comparative. This indicates that IMOE's idea helps optimize explainers and is not harmful when it cannot improve the vanilla version's performance. Improvements in GNNE and PGE are more significant than other baselines, as they are explainers with fewer objectives and are easier to optimize. For RDPE and CF^2, balancing multiple objectives during optimization is difficult, limiting the benefit of incorporating the model-level constraint.

Table 3: Performance on different values of the hyperparameter l

l	Mutagenicity		NCI1		PROTEINS		BA-4Motifs	
	k=5	k=10	k=5	k=10	k=5	k=10	k=5	k=10
1	0.67	1.24	**1.12**	**1.34**	**0.75**	**0.96**	0.46	0.88
2	**1.52**	**2.35**	0.86	0.99	0.45	0.60	**2.28**	**1.67**
3	1.04	2.02	0.91	1.21	0.43	0.63	1.52	1.65
4	1.07	1.89	0.84	1.13	0.62	0.89	1.49	1.45

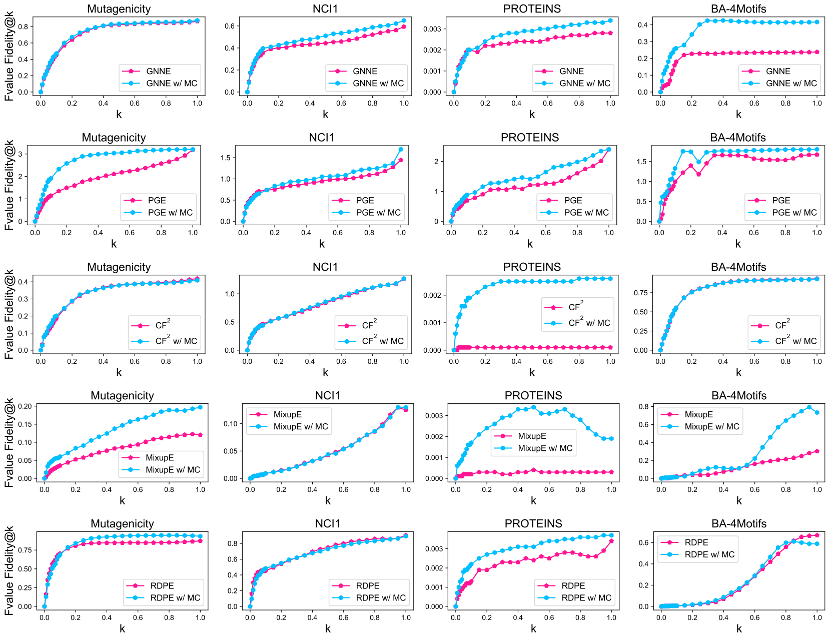


Fig. 5: Generalization of IMOE on $FvalueFidelity@k$

4.5 Study of Hyperparameter l

We investigate the impact of the hyperparameter l in the regularization (Eq. (7)). Table 3 presents the $Fvalue\ Fidelity@k$ on four datasets with k set to 5 and 10. For Mutagenicity, which contains two mutagenic factors (NO_2 and NH_2), and BA-4Motifs, constructed by attaching two types of motifs to the BA graph, the highest fidelity explanations are obtained using the two most similar prototype graphs ($l = 2$), consistent with the number of motifs in these datasets. In contrast, for NCI1 and PROTEINS, which lack explicit ground truth motifs, optimal explanations are obtained using only one most similar prototype graph.

5 Conclusion

This paper explores enhancing GNN explanation quality by bridging multi-level explainers. We propose IMOE, a framework combining the distinct characteristics of model-level and instance-level explainers. IMOE integrates GFlowNet and DBSCAN to learn diverse prototype subgraph patterns, guiding the instance-level explainer through a specially designed loss function. Experiments on four datasets demonstrate IMOE's effectiveness in enhancing explainer performance across various metrics, highlighting the potential of integrating multiple-level GNN explainers to improve the explanability and trustworthiness of GNNs.

6 Acknowledgements

This work is supported by the National Natural Science Foundation of China (Nos.62221005, 62476032, 61936001), Project of Chongqing Municipal Education Commission, China (No.HZ2021008).

References

1. Ester, M., Kriegel, H., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD. pp. 226–231 (1996)
2. Huang, Z., Kosan, M., Medya, S., Ranu, S., Singh, A.K.: Global counterfactual explainer for graph neural networks. In: WSDM. pp. 141–149. ACM (2023)
3. Laird, E., Madushanka, A., Kraka, E., Clark, C.: XInsight: revealing model insights for gnns with flow-based explanations. In: Explainable Artificial Intelligence, xAI. vol. 1902, pp. 303–320. Springer (2023)
4. Lin, W., Lan, H., Li, B.: Generative causal explanations for graph neural networks. In: ICML. vol. 139, pp. 6666–6679. PMLR (2021)
5. Liu, Q., Tan, H., Zhang, Y., Wang, G.: Dynamic heterogeneous network representation method based on meta-path. Acta Electronica Sinica **50**(8), 1830 (2022)
6. Liu, Y., Chen, C., Liu, Y., Zhang, X., Xie, S.: Multi-objective explanations of GNN predictions. In: ICDM. pp. 409–418. IEEE (2021)
7. Luo, D., Zhao, T., Cheng, W., Xu, D., Han, F., Yu, W., Liu, X., Chen, H., Zhang, X.: Towards inductive and efficient explanations for graph neural networks. IEEE TPAMI. **46**(8), 5245–5259 (2024)
8. Lv, G., Chen, L., Cao, C.C.: On glocal explainability of graph neural networks. In: DASFAA. vol. 13245, pp. 648–664. Springer (2022)
9. Shin, Y., Kim, S., Shin, W.: PAGE: prototype-based model-level explanations for graph neural networks. IEEE TPAMI. **46**(10), 6559–6576 (2024)
10. Tan, J., Geng, S., Fu, Z., Ge, Y., Xu, S., Li, Y., Zhang, Y.: Learning and evaluating graph neural network explanations based on counterfactual and factual reasoning. In: WWW. pp. 1018–1027. ACM (2022)
11. Wang, X., Wu, Y., Zhang, A., He, X., Chua, T.: Towards multi-grained explainability for graph neural networks. In: NeurIPS. pp. 18446–18458 (2021)
12. Ying, Z., Bourgeois, D., You, J., Zitnik, M., Leskovec, J.: GNNExplainer: generating explanations for graph neural networks. In: NeurIPS. pp. 9240–9251 (2019)
13. Yuan, H., Tang, J., Hu, X., Ji, S.: XGNN: towards model-level explanations of graph neural networks. In: KDD. pp. 430–438. ACM (2020)
14. Yuan, H., Yu, H., Wang, J., Li, K., Ji, S.: On explainability of graph neural networks via subgraph explorations. In: ICML. vol. 139, pp. 12241–12252. PMLR (2021)
15. Zhang, J., Luo, D., Wei, H.: MixupExplainer: generalizing explanations for graph neural networks with data augmentation. In: KDD. pp. 3286–3296. ACM (2023)
16. Zhang, Y., Cheung, W.K., Liu, Q., Wang, G., Yang, L., Liu, L.: Towards explaining graph neural networks via preserving prediction ranking and structural dependency. Inf. Process. Manag. **61**(2), 103571 (2024)
17. Zhang, Y., Liu, Q., Wang, G., Cheung, W.K., Liu, L.: GEAR: Learning graph neural network explainer via adjusting gradients. Knowledge-Based Systems **302**, 112368 (2024)
18. Zhao, Y., Xu, Y., Zhang, Y., He, W., Cui, L.: Multi-objective graph neural network explanatory model with local and global information preservation. In: DASFAA. vol. 14855, pp. 311–320. Springer (2024)