On Explainability of Graph Neural Networks via Subgraph Explorations: Appendix

A. Datasets and Experimental Settings

A.1. Datasets and GNN Models

We employ different GNN variants to fit these datasets and explain the trained GNNs. Note that these models are trained to obtain reasonable performance. Specifically, we report the architectures and performance of these GNNs as below:

- **MUTAG (GCNs)**: This GNN model consists of 3 GCN layers. The input feature dimension is 7 and the output dimensions of different GCN layers are set to 128, 128, 128, respectively. We employ max-pooling as the readout function and ReLU as the activation function. The model is trained for 2000 epochs with a learning rate of 0.005 and the testing accuracy is 0.92. We study the explanations for the whole dataset.
- **MUTAG (GINs)**: This GNN model consists of 3 GIN layers. For each GIN layer, the MLP for feature transformations is a two-layer MLP. The input feature dimension is 7 and the output dimensions of different GIN layers are set to 128, 128, 128 respectively. We employ max-pooling as the readout function and ReLU as the activation function. The model is trained for 2000 epochs with a learning rate of 0.005 and the testing accuracy is 1.00. We study the explanations for the whole dataset.
- **BBBP (GCNs)**: This GNN model consists of 3 GCN layers. The input feature dimension is 9 and the output dimensions of different GCN layers are set to 128, 128, 128, respectively. We employ max-pooling as the readout function and ReLU as the activation function. The model is trained for 800 epochs with a learning rate of 0.005 and the testing accuracy is 0.863. We randomly split this dataset into the training set (80%), validation set (10%), and testing set (10%). We study the explanations for the testing set.
- **Graph-SST2 (GATs)**: This GNN model consists of 3 GAT layers. The input feature dimension is 768 and all GAT layers have 10 heads with 10-dimensional features. We employ max-pooling as the readout function and ReLU as the activation function. In addition, we set the dropout rate to 0.6 to avoid overfitting. The model is trained for 800 epochs with a learning rate of 0.005 and the testing accuracy is 0.881. We follow the training, validation, and testing splitting of the original SST2 dataset. We study the explanations for the testing set.
- **BA-2Motifs (GCNs)**: This GNN model consists of 3 GCN layers. The input feature dimension is 10 and the output dimensions of different GCN layers are set to 20, 20, 20, respectively. For each GCN layer, we employ L2 normalization to normalize node features. We employ average pooling as the readout function and ReLU as the activation function. The model is trained for 800 epochs with a learning rate of 0.005 and the testing accuracy is 0.99. We randomly split this dataset into the training set (80%), validation set (10%), and testing set (10%). We study the explanations for the testing set.
- **BA-Shape (GCNs)**: This GNN model consists of 3 GCN layers. The input feature dimension is 10 and the output dimensions of different GCN layers are set to 20, 20, 20, respectively. For each GCN layer, we employ L2 normalization to normalize node features. In addition, we use ReLU as the activation function. The model is trained for 800 epochs with a learning rate of 0.005 and the testing accuracy is 0.957. We randomly split this dataset into the training set (80%), validation set (10%), and testing set (10%). We study the explanations for the testing set.

A.2. Experimental Settings

A.3. Evaluation Metrics

We further introduce the evaluation metrics in detail. First, given a graph G_i, its prediction class y_i, and its explanation, we obtain a hard explanation mask M_i where each element is 0 or 1 to indicate whether the corresponding node is identified as important. For our SubgraphX and MCTS-based baselines, the masks can be directly determined by the obtained subgraphs. For GNNExplainer and PGExplainer, their explanations are edge masks and can be converted to explanation masks by selecting the nodes connected with these important edges. Then by occluding the important nodes in G_i based on M_i, we

can obtain a new graph $\hat{G}_i$. Finally, the Fidelity score can be computed as

$$Fidelity = \frac{1}{N}\sum_{i=1}^{N}(f(G_i)_{y_i} - f(\hat{G}_i)_{y_i}), \tag{10}$$

where N is the total number of testing samples, $f(G_i)_{y_i}$ means the predicted probability of class y_i for the original graph G_i. Intuitively, Fidelity measures the averaged probability change for the predictions by removing important input features. Since simply removing nodes significantly affect the graph structures, we occlude these nodes with zero features to compute the Fidelity. In addition, we also employ Sparsity to measure the fraction of nodes are selected in the explanations. Then it can be computed as

$$Sparsity = \frac{1}{N}\sum_{i=1}^{N}(1 - \frac{|M_i|}{|G_i|}), \tag{11}$$

where $|M_i|$ denotes the number of important nodes identified in M_i and $|G_i|$ means the number of nodes in G_i. Ideally, good explanations should select fewer nodes (high Sparsity) but lead to significant prediction drops (high Fidelity).

B. Explanations for Graph Classification Models

In this section, we report more visualizations of explanations for graph classification models. The results are reported in Figure 7 and 8. In Figure 7, we show the explanations of real-world datasets BBBP and MUTAG. Obviously, our proposed method can provide more human-intelligible subgraphs as explanations while PGExplainer and GNNExplainer focus on discrete edges. In addition, we also report the results of sentiment dataset Graph-SST2 in Figure 8. The results show that our SubgraphX can provide reasonable explanations to explain the predictions. For example, in the second row, the input sentence is "none of this violates the letter of behan's book, but missing is its spirit, its ribald, full-throated humor", whose label is negative and the prediction is correct. From the human's view, "missing" should be the keyword for the semantic meaning. Our SubgraphX shows that the "missing is its spirit" phrase is important, which successfully captures the keyword. The other methods capture the words and phrases such as "violates", "none of this", which are less related to the negative meaning.

C. Explanations for Node Classification Models

In this section, we report more visualizations of explanations for node classification models. The results are reported in Figure 9 where we show the explanations of node classification dataset BA-Shape. Obviously, our SubgprahX focuses on the whole motifs for correct predictions and captures partial motifs for incorrect predictions. This is reasonable since if the model can capture the whole motif, then it is expected to correctly predict the target node; otherwise, the information of partial motifs is not enough to make correct predictions.

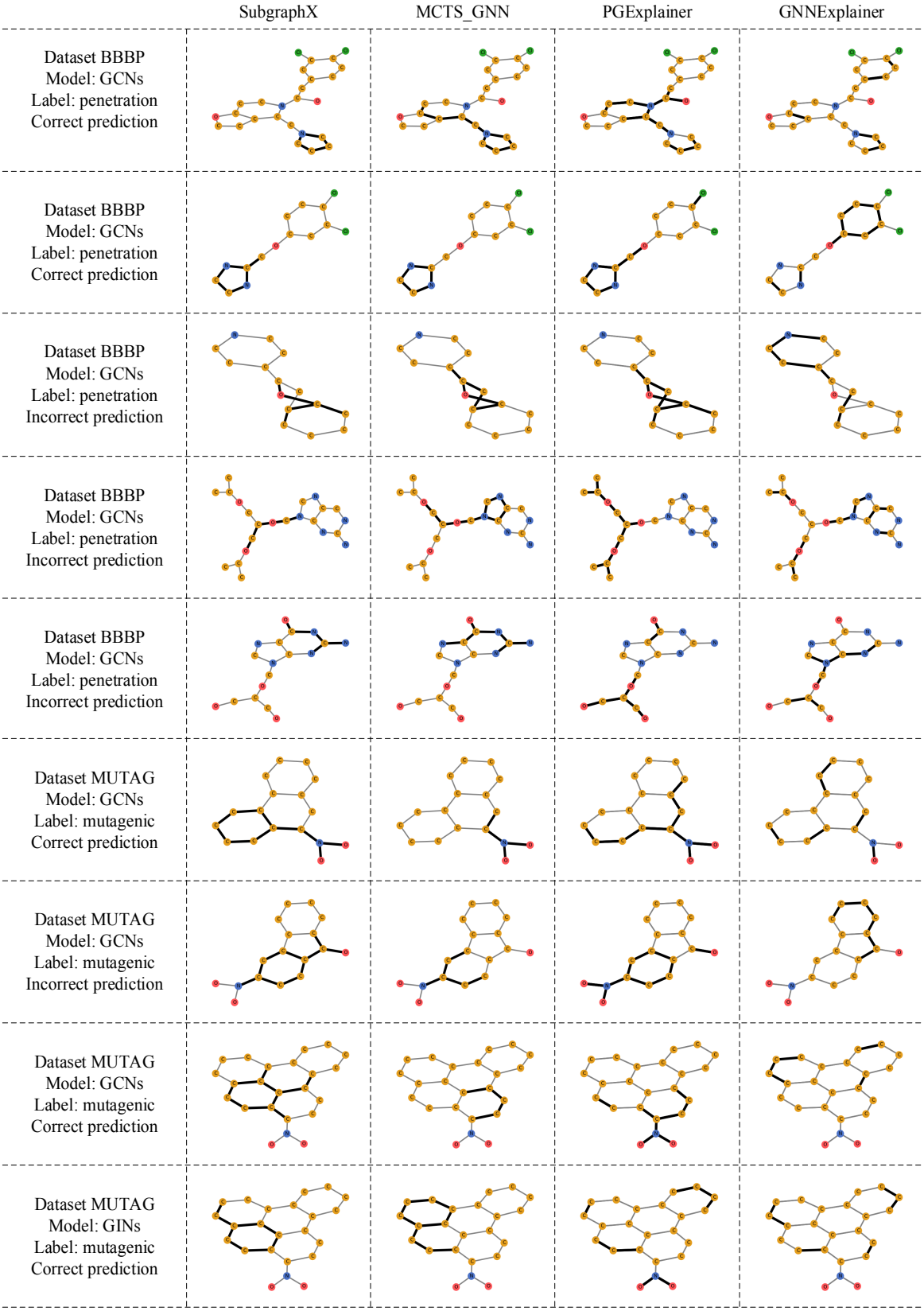

Figure 7. Explanation results of the BBBP and MUTAG datasets. Here Carbon, Oxygen, Nitrogen, and Chlorine are shown in yellow, red, and blue, green respectively.

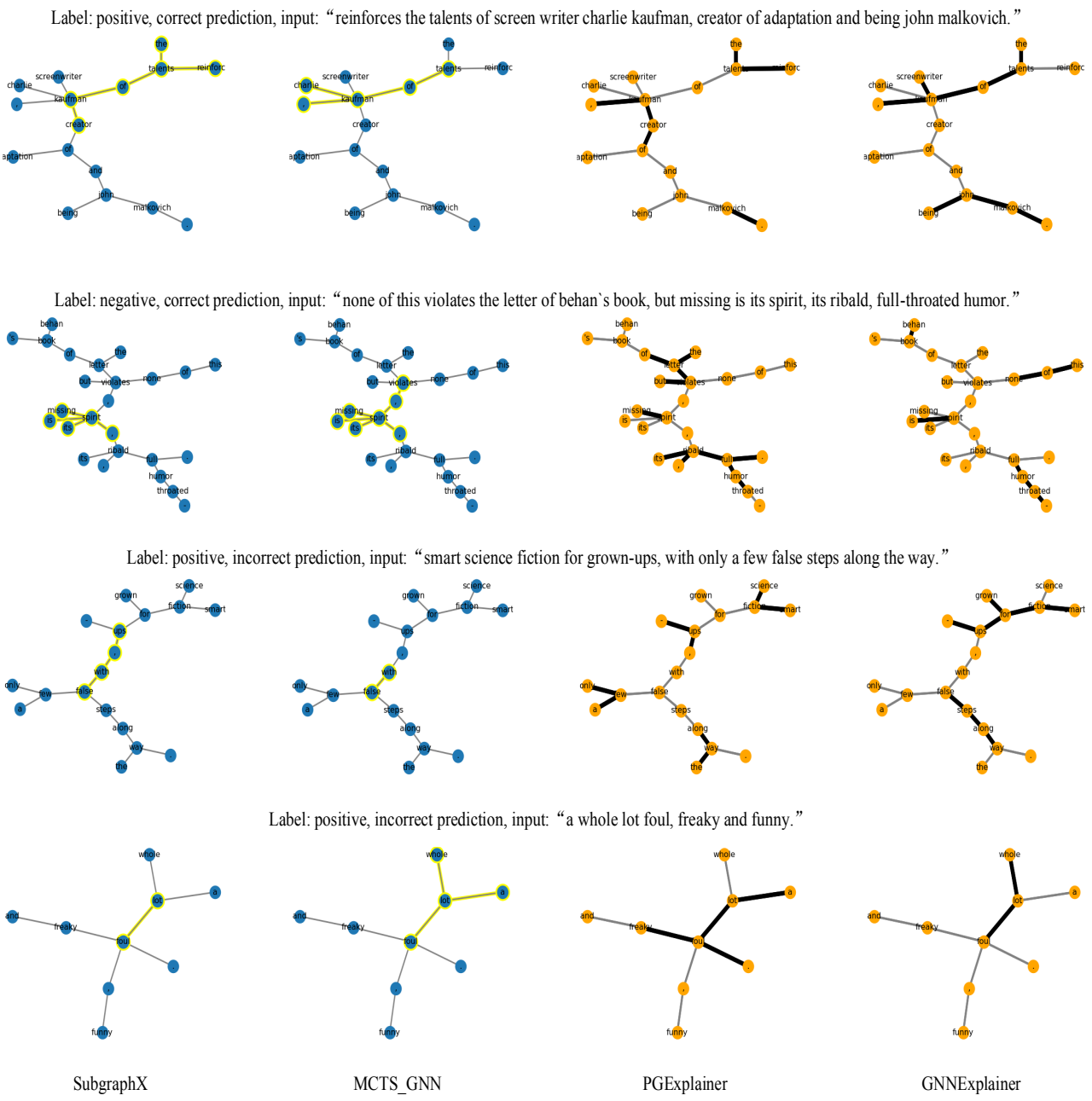

Figure 8. Explanation results of Grpah-SST2 dataset.

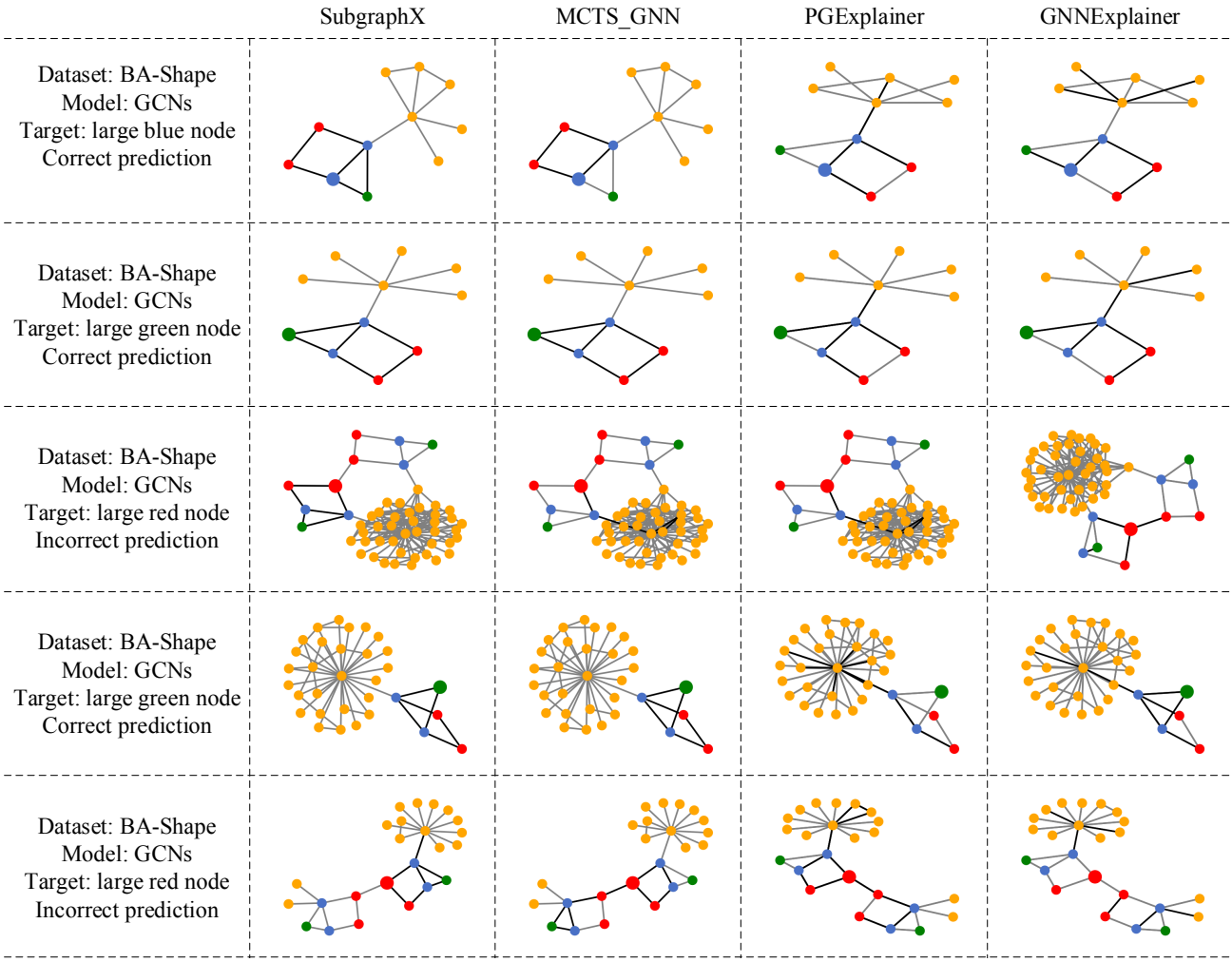

Figure 9. Explanation results of BA-Shape dataset. The target node is shown in a larger size.