Influence of graph neural network architecture on explainability

Hubert Janczak (H.T.Janczak@student.tudelft.nl)
Dr. Megha Khosla, Dr. Jana Weber

TU Delft

Introduction & Problem Statement

- GNNSubNet is an XAI tool developed to **disease-inducing protein subnetworks** in protein-protein-interaction graphs analysed by Graph Neural Networks[5,7]
- Explanations can be obtained on a global and local level, where the global explanation optimizes a mask for all graphs in a dataset at once, while a local one creates one for every graph and then aggregates
- The architecture of the model being analysed by GNNSubNet can impact its ability to correctly identify disease subnetworks
- **BAGEL** is a benchmarking tool created to assess the results of explainers, which can be used to compare the explainer efficiency as underlying architecture changes[6]
- The paper which introduces GNNSubNet **only reports on its performance with one model**, leaving a knowledge gap about its performance on other models

Research Questions

How does the explainer performance vary with change in architectures of training models?

1. How do different GNN architectures (GIN, GCN and GraphSAGE) functionally differ between one another?
2. How does GNNSubNet perform with GCN and GraphSAGE as compared to GIN using selected BAGEL metrics?
3. How does GNNSubNet performance change when evaluated with local explanations instead of global explanations?

Methodology

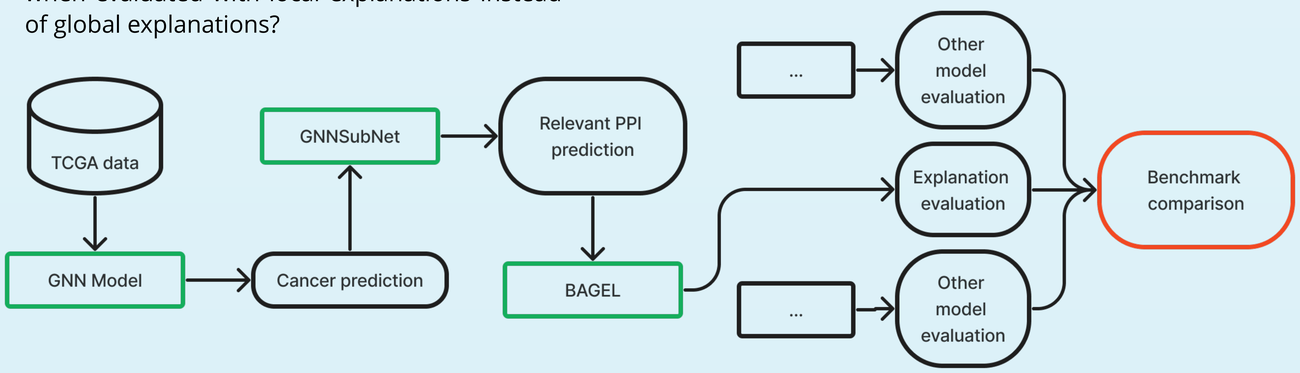
Training: Three models (GIN, GCN, GraphSAGE) were implemented and trained 10 times with the KIRC dataset.
Subnetwork detection: Obtained cancer-relevant subnetworks with GNNSubNet with both global and local explanations.
Evaluation: Assessed the predictions using four explanation performance metrics: RDT-Fidelity, Validity-, Validity+ and Sparsity.

Chosen architecture overview

Graph Isomorphism Network:[2] Model originally chosen by the authors of GNNSubNet. Proven to have the highest possible expressive power of all GNN models.
Graph Convolutional Network:[1] A simple GNN model. Averages information about the neighbors of a node and calculates ReLU of the result.
GraphSAGE:[3] Stands for SAmple and aggreGatE. For aggregation, GraphSAGE samples a subset of the neighborhood. Here, the entire neighbourhood is sampled.

Explainer evaluation metrics

Validity: How well the important nodes were distinguished. Assessed by evaluating model's resilience to changing node features to their average. Validity+ perturbs important nodes, Validity- contrariwise.
RDT-Fidelity:[6] How well the explanation shows the model's decision process. Evaluated by inducing randomized change in unimportant node values.
Sparsity:[6] How few nodes does the explainer need to explain the model. The score is logarithmic.



Results

RDT-Fidelity: High across all models. GNNSubNet correctly distinguishes nodes of importance. Models are resilient to random perturbations.
Validity+: Varied, with high standard deviation. once all important nodes are perturbed, the model is guessing at random.
Validity-: Very high for all models, again showing GNNSubNet's accuracy. The scores are higher for most models due to Validity- causing lower perturbations on average than RDT-Fidelity.

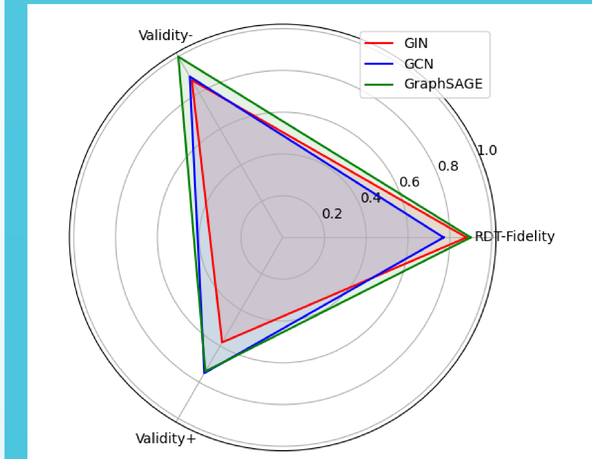


Figure 1: Radar comparison of mean Validity-, Validity+ and RDT-Fidelity calculated over global explanations. The scores are similar across all models.

Table 1: Bagel metric evaluation of each model using global and local explanations over 10 training attempts (mean/stddev).

Global Explanations	GIN	GCN	GraphSage
RDT-Fidelity	0.88/0.10	0.77/0.04	0.90/0.10
Normalized Validity+	0.58/0.26	0.75/0.13	0.74/0.10
Validity-	0.87/0.11	0.89/0.07	1.0/0.0
Sparsity	0.034/0.014	0.025/0.006	0.018/0.006

Local Explanations	GIN	GCN	GraphSage
RDT-Fidelity	0.89/0.05	0.75/0.04	0.83/0.13
Normalized Validity+	0.49/0.23	0.57/0.10	0.75/0.12
Validity-	0.90/0.05	0.96/0.06	1.0/0.0
Sparsity	0.035/0.008	**0.0036**/0.0020	0.01/0.004

Table 2: GNN model accuracy. Each model was trained 10 times.

Model	Min	Mean	Max	δ
GIN	0.5	0.69	0.85	0.12
GCN	0.5	0.61	0.72	0.09
GraphSAGE	0.80	0.87	0.92	0.03

Sparsity: Variance between models is significant, with GIN obtaining the best score out of all models. The way a model aggregates data has significant impact on the explanation density.
Global and local explanations: Global and local explanations are very similar, within one standard deviation from each other's counterparts. Only difference occurs for GCN's Sparsity (bold) which is significantly worse for local explanations.

Conclusions & Future work

Conclusions
- Evaluation of explanation accuracy (i.e. Fidelity and Validity) was similar across all models
- Sparsity score differs between models, with GIN being the best out of all models compared
- Global and local explanations are similarly performant on GNNSubNet
- Overall, due to its superior explanation density, GIN is the best model for subnetwork detection out of the models chosen for evaluation

Future work
- Other model types, such as attentional models could prove easier for GNNSubNet to evaluate
- Impact of sampling size on explainability should be investigated

References

[1] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. 2016.
[2] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful are Graph Neural Networks? 2018.
[3] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive Representation Learning on Large Graphs. 2017.
[4] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. AI Open, 1:57–81, 2020.
[5] Bastian Pfeifer, Anna Saranti, and Andreas Holzinger. GNN-SubNet: disease subnetwork detection with explainable graph neural networks. Bioinformatics, 38(Supplement 2):ii120–ii126, September 2022.
[6] Mandeep Rathee, Thorben Funke, Avishek Anand, and Megha Khosla. BAGEL: A Benchmark for Assessing Graph Neural Network Explanations. 2022.
[7] Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. GNNExplainer: Generating Explanations for Graph Neural Networks. 2019..