GNN Counterfactual Specifications

05. October 2021

1 Introduction

Current xAI platform constraints influence the GNN counterfactuals corresponingly. Only GNNs for homogeneous graphs (all nodes same number and type of features) and no multi-graphs (no self-edges or multiple edges between nodes). The software package that is used is Pytorch-Geometric (PyG): `https://pytorch-geometric.readthedocs.io/en/latest/`, so the architectures that are already implemented in this package have priority. It needs to be noted that this package (as well as another well-known based on that topic `https://www.dgl.ai/`) can be used to create new custom types of GNN architetures.

Hierarchy levels:

- Graph tasks
- Typical state-of-the-art architectures
- Possible actions

The change of number of classes is out of scope. Same goes for the number of hidden units (or so-called hidden channels).

2 GNN Architectures

`https://pytorch-geometric.readthedocs.io/en/latest/modules/nn.html#models`

GCN is the most prominent, GIN for graph classification

The possible layers are listed here: `https://pytorch-geometric.readthedocs.io/en/latest/modules/nn.html#convolutional-layers`

3 Actions

- `add_node`
- `remove_node`
- `add_edge`
- `remove_edge`
- `add_feature_all_nodes`
- `remove_feature_all_nodes`
- `add_feature_all_egdes`
- `remove_feature_all_egdes`

4 Quality Management

The aforementioned logic needs to be tested after each series of actions is “submited”, i.e. after the buttons “Predict” or “Retrain” are pressed. Property-based Testing: [**?**].

5 Future Work

References