Supplementary manuscript for GECo: A Community-based Graph Neural Network Explainer

Domenico Amato[1], Salvatore Calderaro[1,2*], Giosué Lo Bosco[1], Riccardo Rizzo[3], Filippo Vella[3]

[1*]Department of Mathematics and Computer Science, University of Palermo, Via Archirafi 34, Palermo, 90123, Italy.
[2*]Department of Physics and Chemistry - Emilio Segrè, University of Palermo, Viale Delle Scienze Ed. 18, Palermo, 90128, Italy.
[3]Institute for high performance computing and networking, National Research Council of Italy, Via Ugo La Malfa 153, Palermo, 90146, Italy.

*Corresponding author(s). E-mail(s): salvatore.calderaro01@unipa.it;
Contributing authors: domenico.amato01@unipa.it;
giosue.lobosco@unipa.it; riccardo.rizzo@icar.cnr.it;
filippo.vella@icar.cnr.it;

Keywords: Graph Neural Networks, Explainability, XAI

1 Community Detection Algorithms

In 2002, Girvan and Newman introduced an algorithm based on the computation of the edge betweenness [1]. Other methods are based on modularity optimization. The most popular approaches use greedy algorithms, such as Newman (2004) [2] and Clauset et al. (2004) [3], or heuristic search such as the Louvain method introduced by Blondel et al. in 2008 [4].

Our approach employs the algorithm proposed by Clauset et al. [3] based on a greedy algorithm. We selected this algorithm because it works well for large graphs and uses sparse matrices as data structures, which reduces the algorithm's computational complexity compared to Newman's original implementation in 2004 [2]. The

modularity is defined as:

$$Q = \frac{1}{2|\mathcal{E}|} \sum_{ij} \left(A_{ij} - \gamma \frac{d_i d_j}{2|\mathcal{E}|} \right) \delta(c_i, c_j) \tag{1}$$

where $|\mathcal{E}|$ is the number of edges of the graph, A_{ij} is an element of the adjacency matrix A of the graph, d_i and d_j are the nodes degree and $\delta(c_i, c_j) = 1$ if the two nodes belong to the same community, 0 vice versa. The value γ is called the resolution parameter, and it is an arbitrary tradeoff between intra-group edges and inter-group edges. It is widespread to use $\gamma = 1$. If $\gamma < 1$, the modularity favors larger communities, and vice versa, smaller ones. The goal is to find the partition that maximizes Q. The pseudocode of this algorithm is in Algorithm 1.

Algorithm 1 Community Detection Greedy Algorithm [3]

Require: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $n = |\mathcal{V}|$ nodes
Ensure: Set of communities C
Step 1: Initialize each node in its own community, resulting in n communities
$C \leftarrow \{\{v\} \mid v \in \mathcal{V}\}$
while $|C| > 1$ **do**
    Step 2: Compute the modularity variation ΔQ for each pair of communities connected by at least one edge
    for each pair of communities $(C_i, C_j) \in C$ connected by at least one edge **do**
        Compute the modularity variation ΔQ if C_i and C_j are merged
    end for
    Step 3: Identify the community pairs (C_i, C_j) for which ΔQ is the largest and merge them
    $C \leftarrow (C \setminus \{C_i, C_j\}) \cup \{C_i \cup C_j\}$
    Step 4: Record the modularity Q for the current partition
    Note that the modularity Q is computed for the whole graph
end while
Step 5: Select the partition for which Q is maximal
return C

The community detection algorithm has little influence on the results of the GECo algorithm: in Figure 1a and 1b there are two examples of the results obtained with different community detection algorithms. The community detection algorithms used are:

- Clauset, Newman and Moore (CNM) [3] (the one used in our implementation): it is a greedy modularity-based algorithm that efficiently detects communities in large networks by iteratively merging node pairs;
- Girvan and Newman (GN) [1]: it reduces the network by progressively removing the edges with the highest betweenness to reveal community structures;

- Infomap [5]: it uses random walks and information theory to map the flow of information, optimizing for minimum description length;
- Louvain [4]: it is a fast modularity optimization method that detects hierarchical community structures in large networks;
- Leiden [6]: it is an improvement over Louvain, ensuring well-connected communities with better modularity and partition quality;
- Walktrap [7]: it utilizes short random walks to capture local structure and merges nodes into communities based on walk-based similarity.

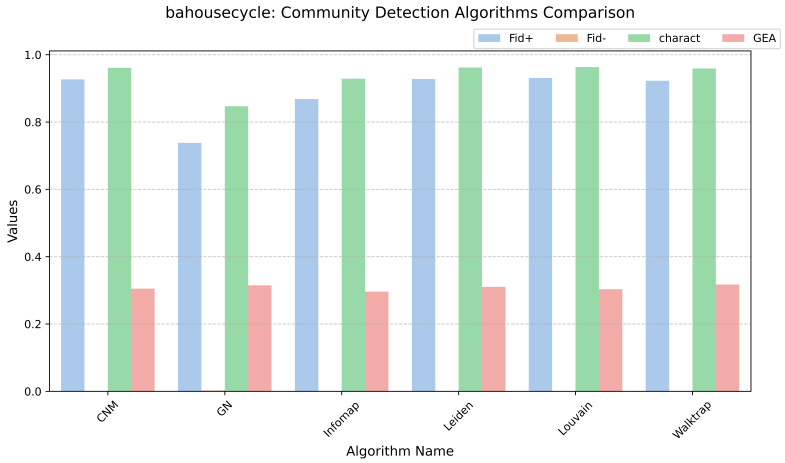
An in-depth analysis of GECo's performance across both synthetic (ba_house_cycle) and real-world (Mutagenicity) datasets highlights CNM and Louvain as consistently strong community detection algorithms, with Walktrap emerging as the top performer on the Mutagenicity dataset. Both CNM and Louvain deliver robust and stable performance across all evaluation metrics, including high fidelity to the original model predictions, effective characterization of key graph substructures, and competitive graph explanation accuracy (GEA). On the ba_house_cycle dataset, Louvain slightly outperforms CNM in terms of Fid^+ and substructure characterization, while CNM achieves better results in Fid^-. In contrast, on the Mutagenicity dataset, Walktrap surpasses both CNM and Louvain, achieving the best performance across all metrics, with marked improvements in Fid^+, Fid^-, and GEA. Walktrap proves particularly effective in generating precise explanations and accurately identifying relevant communities. These findings emphasize the importance of selecting an appropriate community detection algorithm tailored to the dataset's characteristics. While Louvain and CNM maintain reliable performance across diverse scenarios, Walktrap shows that certain algorithms may excel under specific conditions. Nevertheless, the overall variation in GECo's performance remains limited, confirming that the choice of community detection algorithm has little influence on the final results of the GECo algorithm.

2 Evaluating the Impact of Sparsity on Graph Explanation Quality

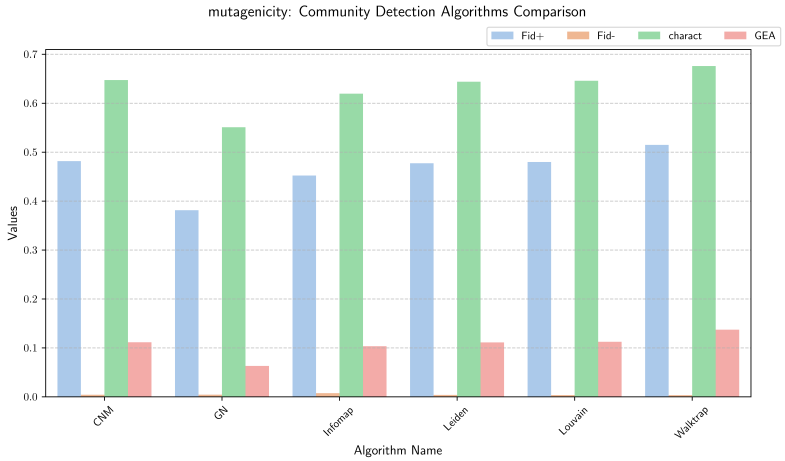
A good explanation should be sparse, i.e., capture only the relevant features and discard the irrelevant ones. The **sparsity** [8] studies this property, and it is defined as:

$$Sparsity = \frac{1}{N}\sum_{i=1}^{N}\left(1 - \frac{|m_i|}{|M_i|}\right) \tag{2}$$

where $|m_i|$ is the number of important features of the mask (nodes/edges/node features), and $|M_i|$, is the total number of features in M_i. The mask identified by the proposed methodology consists of a set of nodes. In this case, m_i represents the number of nodes of the mask, while M_i is the total number of nodes in the graph. To evaluate the robustness and explanation quality of various graph explanation methods under different levels of sparsity, we conducted an experiment using the Mutagenicity dataset. The mask's sparsity directly impacts the quality and interpretability of the



(a) The results on the ba_house-cycle synthetic dataset.



(b) The results obtained on the Mutagenicity dataset.

Fig. 1: The effect of different graph communities algorithm detection on the results of the GECo method.

explanation; a good explanation should focus only on the relevant features, while the irrelevant ones are discarded.

In this experiment, we have investigated how explanation sparsity influences the fidelity of various graph explanation methods. Sparsity refers to the proportion of the original graph structure retained in an explanation—lower sparsity values indicate more compact (i.e., sparser) subgraphs. In contrast, higher values retain a more significant portion of the original graph. This allows for assessing the trade-off between explanatory conciseness and fidelity, where fidelity measures how well the explanation preserves the original model's prediction. Specifically, we reported Fid^+, which

quantifies the predictive confidence of the model when restricted to the explanation subgraph. The number of nodes to be kept, m_i, in the mask must be computed to compute the fidelity at a certain level of sparsity. This value can be computed as:

$$m_i = M_i(1 - S) \tag{3}$$

The sparsity values considered are $S = \{0.5, 0.55, 0.6, 0.65, 0.7, 0.75.0.8, 0.85, 0.9\}$. We conducted the experiments, computing the Fid^+ for each sparsity level, repeating the process for 100 runs for each considered method. The obtained results are in Figure 2. GECo consistently outperforms all other methods, achieving the highest Fid^+ across most sparsity levels. It peaks around a sparsity of 0.6–0.65, suggesting that GECo can generate compact yet highly informative explanations. As sparsity increases beyond this point, Fid^+ slightly declines, which is expected due to the inclusion of less relevant nodes and edges. GNNExplainer ranks second in performance but exhibits a steady decrease in Fid^+ as the explanations become sparser. This suggests a dependency on more extensive graph information to maintain fidelity. PGMExplainer and PGExplainer remain relatively stable but underperform compared to GECo and GNNExplainer, indicating less effective selection of relevant substructures. SubgraphX and TAGE achieve the lowest fidelity scores, particularly at higher sparsity levels. Their explanations appear less aligned with the model's decision logic, especially when constrained to small subgraphs. These findings highlight GECo's effectiveness in balancing sparsity and fidelity, making it a strong candidate for interpretable GNN explanations for graph classification tasks.

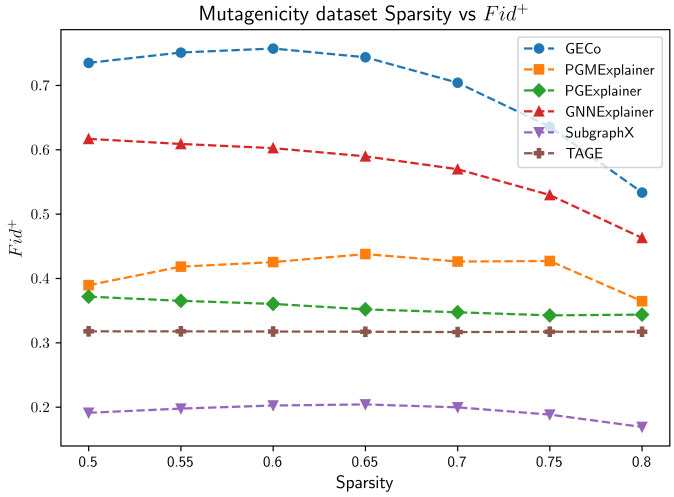


Fig. 2: Fid^+ at different sparsity levels for the Mutagenicity dataset.

References

[1] Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences **99**(12), 7821–7826 (2002)

[2] Newman, M.E.J.: Fast algorithm for detecting community structure in networks. Phys. Rev. E **69**, 066133 (2004)

[3] Clauset, A., Newman, M.E.J., Moore, C.: Finding community structure in very large networks. Phys. Rev. E **70**, 066111 (2004)

[4] Blondel, V.D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment **2008**(10), 10008 (2008)

[5] Rosvall, M., Bergstrom, C.T.: Maps of random walks on complex networks reveal community structure. Proceedings of the national academy of sciences **105**(4), 1118–1123 (2008)

[6] Traag, V.A., Waltman, L., Van Eck, N.J.: From louvain to leiden: guaranteeing well-connected communities. Scientific reports **9**(1), 1–12 (2019)

[7] Pons, P., Latapy, M.: Computing communities in large networks using random walks. In: Computer and Information Sciences-ISCIS 2005: 20th International Symposium, Istanbul, Turkey, October 26-28, 2005. Proceedings 20, pp. 284–293 (2005). Springer

[8] Pope, P.E., Kolouri, S., Rostami, M., Martin, C.E., Hoffmann, H.: Explainability methods for graph convolutional neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)