GraphFramEx: Towards Systematic Evaluation of Explainability Methods for Graph Neural Networks

Anonymous Author(s)
Anonymous Affiliation
Anonymous Email

Abstract

As one of the most popular machine learning models today, graph neural networks (GNNs) have attracted intense interest recently, and so does their explainability. Unfortunately, today's evaluation frameworks for GNN explainability often rely on few inadequate synthetic datasets, leading to conclusions of limited scope due to a lack of complexity in the problem instances. As GNN models are deployed to more mission-critical applications, we are in dire need for a common evaluation protocol of explainability methods of GNNs. In this paper, we propose, to our best knowledge, the first systematic evaluation framework for GNN explainability GRAPHFRAMEX, considering explainability on three different "user needs". We propose a unique metric, the characterization score, which combines the fidelity measures and classifies explanations based on their quality of being sufficient or necessary. We scope ourselves to node classification tasks and compare the most representative techniques in the field of input-level explainability for GNNs. We found that personalized PageRank has the best performance for synthetic benchmarks, but gradient-based methods outperform for tasks with complex graph structure. However, none dominates the others on all evaluation dimensions and there is always a trade-off. We further apply our evaluation protocol in a case study for frauds explanation on eBay transaction graphs to reflect the production environment.

1 Introduction

As machine learning models are being deployed to mission critical applications and are having increasingly profound impact on our society, interpreting machine learning models has become crucially important [1, 2]. At the same time, graph neural networks (GNNs) are of growing interest and are ubiquitous in many learning systems across various areas[3–8]). Due to the complex data representation and non-linear transformation, explaining decisions made by GNNs is challenging. The past decade has witnessed the rise of new methods to explain GNN predictions [9–24].

How do these GNN explanation methods compare with each other? How should we evaluate these GNN explanation methods? These two questions, unfortunately, are still open today. Today's GNN explainability methods are often evaluated on the inadequate synthetic datasets introduced by [10], later referred as *type 1* (see AppendixA.6 for the types of synthetic data) - where groundtruth is available and often on different grounds — as shown in Table 1. Furthermore, they only consider a small subset of metrics to evaluate their method and this choice is very *different* from method to method. Most papers do not consider the aspect of computing time. They also evaluate their method on an almost accurate GNN model, without considering the influence of GNN accuracy on explainability. As a result, insights obtained in these different papers often *do not reflect their performance on real-world applications!* Most method papers (see upper section of Table 1) have inconsistent rankings when evaluation the methods on type 1 synthetic datasets or on real datasets.

Submitted to the First Learning on Graphs Conference (LoG 2022). Do not distribute.

Table 1: XAI LITERATURE FOR GNN NODE CLASSIFICATION. "Acc" defines the accuracy (F1-score) measured with respect to the groundtruth, "Fid+" and "Fid-" refer to the fidelity metrics as defined in [26] (see Appendix A.4). The column "Time" indicates if the paper has run a comparative analysis of the computation time of the explainability methods. The final column "GNN accuracy" shows if the authors have reported the testing accuracy of their model.

Paper Type	Year	Explainer	Use type 1 syn data**	Synthetic			Real			Time	GNN Accuracy
				Acc	Fid-	Fid+	Acc	Fid-	Fid+		
Method [9]	2019	LRP	✓	✓							
Method [10]	2019	GNNExplainer	✓	✓							> 0.90
Method [11]	2020	PGExplainer	✓	✓						✓	$0.92 - 1.00$
Method [12]	2020	RelEx	✓	✓							
Method [13]	2020	PGM-Explainer	✓	✓			✓				$0.85 - 1.00$
Method [14]	2021	RG-Explainer	✓	✓							
Method [15]	2021	ZORRO						✓*			$0.48 - 0.79$
Method [16]	2021	SubgraphX	✓			✓				✓	$0.86 - 0.99$
Method [17]	2021	CF-GNNExplainer	✓	✓	✓						> 0.87
Method [18]	2021	RCExplainer	✓	✓		✓				✓	$0.84 - 0.99$
Method [19]	2021	Gem	✓		✓*					✓	
Taxonomy [26] (Yuan et al.)	2020	GNNExplainer,PGExplainer SubgraphX,DeepLift GNN-LRP,Grad-CAM,XGNN	✓	✓	✓	✓					
Taxonomy [25] (Faber et al)	2021	Saliency,Occlusion,IntegratedGrad GNNExplainer,PGM-Explainer		✓						✓	0.81-1.00
Taxonomy [27] (Li et al)	2022	GraphMask GNNExplainer,PGExplainer						✓*			
Taxonomy [28] (Agarwal et al)	2022	VanillaGrad,IntegratedGrad GraphMask,GraphLIME GNNExplainer,PGExplainer PGMExplainer						✓*			

* Different denomination in the paper, but the same evaluation mechanism as ours.
** Type 1: [10]; Type 2: [25]; Type 3: MUTAG [29], MoleculeNet [30],... See Appendix A.6 for the full synthetic data classification.

Only the taxonomy survey [25] that proposes three novel synthetic benchmarks - *type 2* - has consistent results with real data.

Evaluation Framework. In this paper, we aim at overcoming these limitations and propose GRAPHFRAMEX, the first systematic framework for evaluating explainability methods in the context of node classification. We consider three aspects of *users' needs* in our evaluation protocol. Our framework further distinguishes two types of explanations, according to whether they are *necessary* or *sufficient*. For evaluation, we combine the two fidelity measures, Fid+ and Fid-, that capture the two explanation types, into one single performance metric: the *characterization* score. Our evaluation method does not require groundtruth from synthetic datasets and can be applied to any graph datasets in practice. This paper is the first to study the relation between accuracy and explainability. We evaluate a variety of explainability methods on type 1 synthetic datasets of [10] and ten real datasets. We show the limitations of these specific synthetic datasets. To reflect the production environment, we run a fraud explanation study for eBay transaction graphs. Because runtime is also important, our analysis further compares methods on their average mask computation time. This is also the first paper interested in explaining inaccurate GNN models and the first to investigate the influence of GNN accuracy on the explainer performance.

Moving Forward. As an early attempt to systematically investigate evaluation of GNN explainability, this paper also aims to facilitate the assessment of future explainability methods and shed light on how to build more effective explainability methods that would incorporate the advantages of existing methods. We have created an online platform for people to compete and compare their method to a standard leaderboard with our proposed evaluation and a selected set of representative methods. They also have the possibility to integrate their method to the final leaderboard. It also opens new doors to create synthetic datasets that better reflect the complexity of real ones, which we will discuss in Section 5.2.4. Our code is *anonymously* available at `https://anonymous.4open.science/r/GraphFramEx-E054/`.

2 Related work

Confronted to a rapid increase of XAI methods, researchers have tried to identify a list of properties desired of explainable systems and developed concrete tools to help compare and evaluate all of the methods [31, 32]. Following these systematic XAI evaluation reviews, recent studies have proposed to systematically evaluate the performance of explainability methods for GNNs [25–28]. [25] evaluates explainability methods on three new benchmarks for which groundtruth is available to alleviate five pitfalls observed in the widely used type 1 synthetic datasets. But methods are only evaluated with the accuracy metric. Our framework evaluates explainers regardless of the existence of groundtruth. The first attempt to construct an evaluation framework without groundtruth explanations is the paper of Yuan et al. [26]. They evaluate diverse explainability methods on two fidelity scores at different sparsity levels. But simple baselines such as distance and PageRank and gradient-based methods are omitted, while we show their superiority in some settings. [27] adopts the same methodology as [26], but normalizes one of the fidelity scores. Authors of [28] are the first ones to carry out a theoretical study and derive upper bounds on three evaluation metrics: unfaithfulness, instability and fairness mismatch. Like [25], we consider stability and fairness to be optional criteria and not general quality measures. None of the papers studies the relation between accuracy and explainability. Moreover, they do not consider other mask transformation than sparsity.

3 Problem setup

Let $G = (\mathcal{V}, \mathcal{E})$ represent the graph with $\mathcal{V} = \{v_1, v_2...v_N\}$ denoting the node set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ as the edge set. Edges may be directed or undirected. The numbers of nodes and edges are denoted by N and M, respectively. A graph can be described by an adjacency matrix $\mathbf{A} \in \{0,1\}^{N \times N}$, with $a_{ij} = 1$ if there is an edge connecting node i and j, and $a_{ij} = 0$ otherwise. In addition, nodes in $\mathcal{V}$ are associated with d-dimensional features, denoted by $\mathbf{X} \in \mathbb{R}^{N \times d}$.

In the context of node classification, a GNN can be written as a function $f : \mathcal{V} \longrightarrow \mathcal{Y}$, which assigns to nodes in $\mathcal{V}$ labels from a finite set $\mathcal{Y}$. The GNN model is trained with an objective function $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ that computes a cross-entropy loss $s = \mathcal{L}(y, \hat{y})$ by comparing the model's prediction $\hat{y}$ to a ground-truth label y. To fuse the information of both node features and graph structure in node representation vectors, GNN models utilize a message passing scheme to aggregate information from neighboring nodes.

Given a pre-trained classifier f, our objective is to obtain an explanation model. An "explanation" in the domain of GNNs is a mask or a subgraph of the initial graph, i.e., a set of weighted nodes, edges and possibly node features. The weights on those graph entities relate to their inherent importance for explaining the model outcomes. The explainer model usually performs a feature attribution operation which associates each feature of a computation graph G_C with a weight or relevance score for the classifier's prediction. The computation graph G_C might be the initial graph G or a subgraph around the target node v_t since some methods only look at a k-hop neighbourhood to do predictions. We focus on the contribution of the structural features, namely the edges. To explain each node v_t, all the methods compared in this paper generate a mask $\mathbf{M}_E(\mathcal{E}, f, v_t, y_t) \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$, each element of which is the importance score of the edges to the prediction class y_t of the target node v_t. The more complex methods also generate a mask $\mathbf{M}_{NF}(\mathcal{V}, f, v_t, c_t)$ on the node features (see Table 5 in Appendix B). At the end, an explanation corresponds to a mask $\mathbf{M}_E$ on the edges and sometimes a mask $\mathbf{M}_{NF}$ on the node features, that operate on the initial graph to form a subgraph G_S with adjacency matrix $\mathbf{A}_S = \mathbf{M}_E \odot \mathbf{A}$ and features $\mathbf{X}_S = \mathbf{M}_{NF} \odot \mathbf{X}$, where $\odot$ denotes elementwise multiplication. We denote by $y_t^{G_S}$ and $y_t^{G_{C \setminus S}}$ the model's predictions for node v_t when taking as input respectively the explanatory or masked graph G_S and its complement or masked-out graph $G_{C \setminus S}$.

Scope. Our framework only compares *post-hoc* explainability methods since our focus is on explaining any GNN model. We restricted our study to *input-level* methods because there are currently limited model-level explainability methods [10, 20]. We evaluate both *model-aware* and *model-agnostic* methods in the context of node classification tasks. See Appendix A for the full definitions and taxonomy.

4 Method

This section presents the three design choices made by the users and the evaluation metrics used to assess explainers performance.

4.1 Multi-objectives for explainability

To build GraphFramEx, we start from the perspective of the data subject. Users design the framework based on their expectations on the produced explanations. They can make choices on three dimensions: the explanation focus, the mask nature and the mask transformation strategy.

Aspect 1: the focus of explanation. Some users want to explain why a certain decision has been returned for a particular input. In this case, the task of explaining has a more applied nature: they are interested in the *phenomenon* itself and try to reveal findings in the data, i.e. explain the true labeling of the nodes. The model's predictions are ignored in the explanation process. Others prefer to explain how the model works. In this case, they are interested in the GNN *model* behavior and try to explain the logic behind the model, i.e. the predicted labels. These equally complementary and important reasons demand different analysis methods. The choice of explanation focus determines the explanation objective and evaluation.

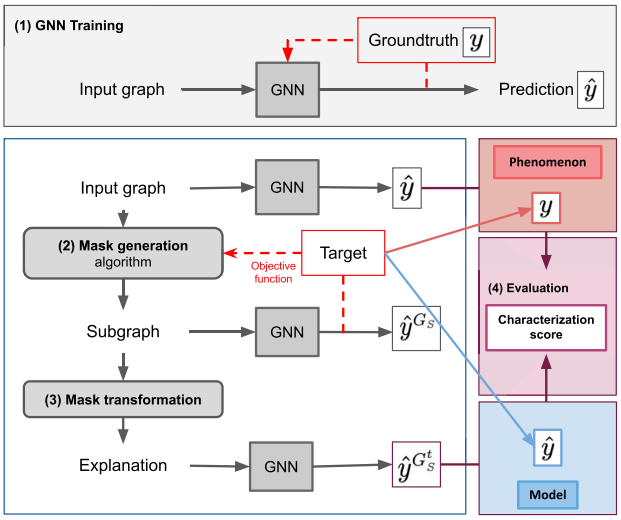


Figure 1: General protocol. The explanation focus is the **phenomenon** or the **model**. (1) A GNN model learns to predict the label $\hat{y}$ of each node in the input graph. For the explanation of node labels (true or predicted), we use this pre-trained model. The explainability method generates a soft mask M_E, which operates on the input graph to return a subgraph G_S. (2) The goal is to reproduce a target label: y or $\hat{y}$. (3) The mask is transformed to output the final explanatory subgraph G_S^t. (4) We evaluate G_S^t by comparing its predicted label to our target.

Aspect 2: mask nature: hard or soft mask. Edge masks $\mathbf{M}_E$ are normalized so that each weight lies between 0 and 1. To convey human-intelligible explanation, we can directly operate the initial *soft mask*, $\mathbf{M}_E^{soft} \in [0,1]^{M\times M}$ on G_C and return an explanatory subgraph G_S^{soft}, where the edge weights reflect the relative importance of edges. But, users might prefer a non-weighted subgraph G_S^{hard} as explanation. In this case, once the mask has been transformed (Aspect 3), we convert the mask into a *hard mask*, $\mathbf{M}_E^{hard} \in \{0,1\}^{M\times M}$ by setting every positive values to 1.

Aspect 3: the mask transformation. Because there is no such thing as a "good" size for an explanation, it is even harder to compare explainability methods. Existing explainability methods return different sizes of explanations by default. To make them comparable, most papers propose to fix a sparsity level to apply to all explanations and compare the same-sized explanations [16, 18, 33]. We define three strategies to reduce explanation size: sparsity, threshold and topk (see Appendix B), which transform the edge mask M_E into a sparser version M_E^t. We decide to use the topk strategy because it is the only strategy that enforces a maximum number k of edges independently of the size of the graph and the explainer methodology. This independence property is important as human-intelligible explanations cannot exceed a certain number of graph entities. Too small explanations omit important elements and will not be sufficient, while too big explanations contain irrelevant nodes and edges and will not be necessary.

4.2 Evaluation

Phenomenon	**Model**
$fid_+ = \frac{1}{N}\sum_{i=1}^{N}\left\lvert \mathbb{1}(\hat{y}_i = y_i) - \mathbb{1}(\hat{y}_i^{G_{C\setminus S}} = y_i)\right\rvert$	$fid_+ = 1 - \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}(\hat{y}_i^{G_{C\setminus S}} = \hat{y}_i)$
$fid_- = \frac{1}{N}\sum_{i=1}^{N}\left\lvert \mathbb{1}(\hat{y}_i = y_i) - \mathbb{1}(\hat{y}_i^{G_S} = y_i)\right\rvert$	$fid_- = 1 - \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}(\hat{y}_i^{G_S} = \hat{y}_i)$

We define multiple dimensions on which we can evaluate explanations. If we have the ground-truth explanations, we can use the accuracy metric. In most of the cases, ground-truth explanations are unknown and explanatory subgraphs are evaluated on their contribution to the initial prediction.

Fidelity. To be independent from any ground-truth explanations, we suggest using the fidelity measures. We extend the definitions in [26] by considering in addition the explanation focus. We make some adjustments: for the phenomenon focus, the fidelity is measured with respect to the ground-truth node label y; for the model focus, it is measured with respect to the outcome of the GNN model $\hat{y}$. In the context of node classification, the indicator function certifies whether the predicted class of a subgraph corresponds to the desired class defined as the true label y in the phenomenon focus or the predicted label for the whole graph $\hat{y}$ in the model focus.

Typology. Considering the large spectrum of possible explanations, we propose to classify explanations in two categories based on their fidelity scores. Each category defines the role of the explanation in producing the observed outputs: the explanation can be necessary and/or sufficient.

- SUFFICIENT EXPLANATION An explanation is sufficient if it leads by its own to the initial prediction of the model explanation. Since other configurations in the graph may also lead to the same prediction, it is possible to have multiple sufficient explanations for the same prediction. A sufficient explanation has a fid_- score close to 0. We later report $(1 - fid_-)$ in our experiments.
- NECESSARY EXPLANATION An explanation is necessary if the model prediction changes when you remove it from the initial graph. Necessary explanations are similar to counterfactual explanations [34]. A necessary explanation has a fid_+ score close to 1.

An explanation is a characterization of the prediction if it is both necessary and sufficient. It can be interpreted as the certificate for a specific class or label. Explainability methods should aim at returning this type of explanations as they are the most informative and complete.

General performance metrics. A variety of functions exists to combine Fidelity+ and Fidelity- measures into a single metric on the overall quality of the explanation such as the area under Fid+/(1-Fid-) curve (AUC). For users interested in only one aspect of an explanation, *i.e.* necessary or sufficient, we suggest to use the fidelity scores independently, *i.e.* Fid- or Fid+, and compare the performance of explainability methods with Fid+@K or (1-Fid-)@K metrics.

Characterization score. In this paper, we recommend the characterization score as a global evaluation metric, due to its ability to balance the sufficiency and necessity requirements. This approach is analogous to combining precision and recall in the Micro-F1 metric. The $charact$ score is the *weighted harmonic mean* of Fid+ and 1-Fid- as defined in Equation 1:

$$charact = \frac{w_+ + w_-}{\frac{w_+}{fid_+} + \frac{w_-}{1-fid_-}} = \frac{(w_+ + w_-) \times fid_+ \times (1 - fid_-)}{w_+ \cdot (1 - fid_-) + w_- \cdot fid_+} \tag{1}$$

where $w_+, w_- \in [0, 1]$ are respectively weights for $fid+$ and $1 - fid_-$ and satisfy $w_+ + w_- = 1$. In the context of explainability, it is important to know that the explanation is leading to the prediction, i.e. sufficient, but also essential for this output, i.e. necessary. As seen in Equation 1 and Fig. 2, the characterization score with equal weights on Fid+ and (1-Fid-) is low as soon as one of the two terms is low. It reflects the strong simultaneous dependency of the characterization score to both fidelity measures. In addition, it is possible to vary the weights w_+ and w_- to compare explainers more on one aspect rather than the other.

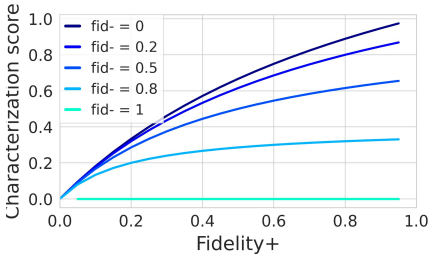


Figure 2: Characterization score for $w_+ = w_- = 0.5$

Efficiency. Efficiency relates to the trade-off between performance, assessed by the characterization score, and computation time of an explanation. A method is very efficient if it quickly generates explanations that well characterize the GNN predictions. This is an important criteria as users might want rapid answers to their why-questions.

5 Results

We evaluate existing methods on their efficiency, characterization power, and type of explanations. No method is dominating the others in all aspects. We also discuss here the limitations of previous evaluation protocols.

5.1 Experimental settings

We describe the setup and implementation details for the explainability procedure. See Appendix B for more details on the datasets statistics, the methods and the experimental protocol.

Datasets.

- **Synthetic datasets** We use type 1 synthetic datasets introduced by [10]. We refer the reader to Appendix A.6 to learn more about the 3 classes of existing synthetic datasets in explainability for GNNs. Ground truth explanations are available.
- **Real datasets** We use 10 publicly available datasets to evaluate our framework on real graphs: the citation network datasets [35], the Facebook Page-Page network dataset [36], the actor-only induced subgraph of the film-director-actor-writer network [37], the WebKB datasets [37], and the Wikipedia networks [36]. We use the code accessible in Pytorch geometric.
- **eBay** We test our evaluation framework on a real-world eBay transaction graph dataset. This is a binary node classification task where transaction nodes are labeled as legit or fraudulent. The objective is to explain fraudulent nodes. The eBay graph dataset is a very large sampled real-world dataset with 289k nodes (208k transaction nodes) and 1% of all nodes (1.48% of transaction nodes) are fraudulent. This is a typical example of a rare event detection task.

GNN models. By default, we use the graph convolutional networks (GCN) [38]. Besides GCN, we also evaluate explainability methods on graph attention networks (GAT) [39] and graph isomorphism networks (GIN) [40]. Results using GAT and GIN models are presented in Appendix C.

Explainers. To explain the decisions made by the GNNs, we adopt different classes of explainers including structure-based methods, gradient/feature-based methods and perturbation-based methods. We refer the reader to Appendix A.3 for the full taxonomy and to Appendix B.2 for more details on the explainability methods. In our experiments, we compare the following methods: **Random** gives every edge and node feature a random value between 0 and 1; **Distance** assigns higher importance to edges that have lower distance to the target node; **PageRank** measures the importance of edges following the personalized PageRank strategy with automatic restart on the target node [41, 42]; **Saliency (SA)** measures node importance as the weight on every node after computing the gradient of the output with respect to node features [9]; **Integrated Gradient (IG)** avoids the saturation problem of the gradient-based method Saliency by accumulating gradients over the path from a baseline input (zero-vector) and the input at hand [43]; **Grad-CAM** is a generalization of class activation maps (CAM) [44]; **Occlusion** attributes the importance of an edge as the difference of the model initial prediction prediction on the graph after removing this edge [25]; **GNNExplainer** computes the importance of graph entities (node/edge/node feature) using the mutual information [10]; We also try **Basic GNNExplainer** that considers only edge importance; **PGExplainer** is very similar to GNNExplainer, but generates explanations only for the graph structure (nodes/edges) using the reparameterization trick to overcome computation intractability [11]; **PGM-Explainer** perturbs the input and uses probabilistic graphical models to find the dependencies between the nodes and the output [13]; and **SubgraphX** explores possible explanatory subgraphs with Monte Carlo Tree Search and assigns them a score using the Shapley value [16].

Protocol. In this work, we focus on node classification tasks and compare local, that is input-level, explainability methods. We train one of the three GNN models. Once trained, we use the GNN to do predictions on a testing set. Explanations are then eventually transformed with the topk strategy. We evaluate the methods with the fidelity measures and the characterization score with equal weights $w_+ = w_- = 0.5$ in four different settings defined as the combinations of the two possible focus, *phenomenon* and *model*, and mask nature, *hard* or *soft* masks.

5.2 Main results

5.2.1 Explainer efficiency and type of explanation on real datasets

The legend of figure 3 shows the overall ranking of each explainability method. We rank them on their characterization score averaged on all real datasets for explanations of size 10 edges in the four settings (*phenomenon* / *model*, *hard* / *soft* mask). Saliency has the highest overall characterization score. More generally, gradient/feature-based methods are better than perturbation-based methods.

The overall characterization score of the twelve explainers on the real datasets is also evaluated against their average computation time of an explanatory mask. Left plot of Figure 3 shows that, in addition to having the best characterization score, Saliency is also the most efficient. In the setting

where we explain the model with a hard mask, we observe that Occlusion has the best overall score but is 10^4 times slower than Saliency.

We compare the methods on the type of explanation they return. On the right plot of Figure 3, methods scoring high on the x-axis return necessary explanations, while those scoring high on the y-axis return good sufficient explanations. We observe that Saliency is by far the best one to return necessary explanations. But, for sufficient explanations, Occlusion, Grad-CAM and PageRank are better choices. As a general remark, we observe that most of the methods are able to return very good sufficient explanations as their explanations have a fidelity- score close to 0. But very few generate necessary explanations: only Saliency, Distance and Occlusion reach a fidelity+ score greater than 0.6 in at least one of the four settings.

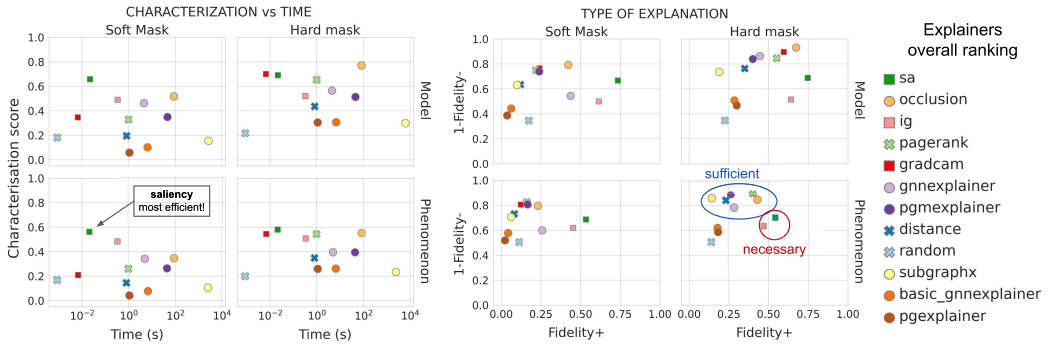


Figure 3: Results on real datasets. (left) Performance and computation time. (right) Type of explanation returned by each explainability method. *sa* - Saliency. *ig* - Integrated Gradient.

5.2.2 Explaining wrong predictions

Most of the papers report GNN testing accuracy greater than 80% and all of them test their explainers on a mixture of correct and wrong predictions (see Table 1). But when ignoring this distinction, they unknowingly take a different focus. When they explain correct predictions, they target the true label and explain both the phenomenon and the GNN model. When they explain wrong predictions, the predictions by the GNN do not correspond to the true label and, therefore, they can only get an insight of the GNN logic. We decide to study what happens to our explainers ranking if we separate correct from wrong predictions. Figure 4 shows a general drop of performance of the explainers when the predictions do not match the true label. So, mixing wrong and correct nodes will necessarily reduce the scores. We also see that the gradient-based method Saliency is the only method able to explain the model logic when the predictions are wrong. This is not surprising as model-aware explainability methods focus on the model's internal working and will always explain the logic before the phenomenon. Therefore, all current papers that generate explanations when the model is not 100% accurate, are naturally biased towards gradient-based methods. This small study also encourages using Saliency to produce good explanations of a wrong GNN as it can also serve users to have an easier acceptance of bad models if they can actually explain them.

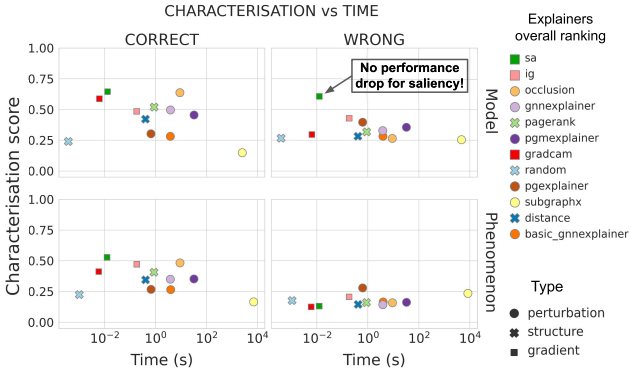


Figure 4: Average performance when explaining only correct (left) or only wrong (right) predictions on 5 real datasets. *sa* - Saliency. *ig* - Integrated Gradient.

5.2.3 Select a pertinent explainability method

Based on the experiments, we outline how the design dimensions of GraphFramEx enables domain-specific users to quickly find best explanability models for their GNN prediction tasks.

GraphFramEx finds the most appropriate method according to the 3 aspects described in section 4.1 and the accuracy level of the trained model, and can be shown as a decision tree. Figure 5 presents one decision tree when we set the mask transformation as the *topk* strategy with 10 edges ($k = 10$), for brevity purposes. It guides users to select the optimal method according to their multi-objectives and suggest explainers that are the best at returning necessary (red box), sufficient (green box) or both necessary and sufficient explanations (orange box). Other design considerations such as runtime can also be easily included based on the experiments. Note that additional explainability methods can be easily incorporated in our evaluation framework and be considered in the decision tree for general users.

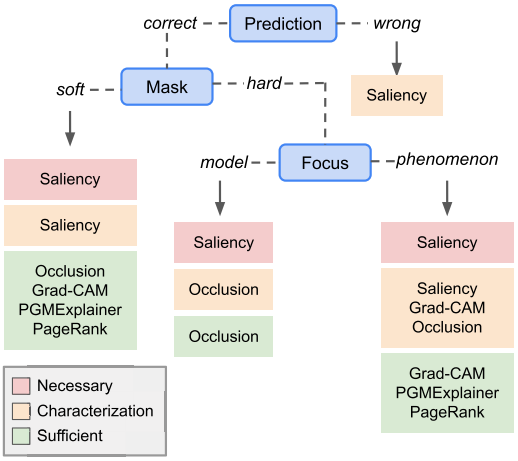


Figure 5: GraphFramEx decision tree for a mask transformation $topk = 10$.

5.2.4 Further Analysis

Trade-off. As observed in the two previous sections, Saliency seems to outperform the other methods except when we want sufficient explanations. In this case, Occlusion is the most appropriate one. We investigate if Saliency dominates the other methods. Figure 6 compares Saliency and Occlusion, respectively the first and second best methods on each dataset. Even though Saliency seems to dominate Occlusion to explain both model and phenomenon, we observe that it actually underperforms for Wisconsin, Actor and Facebook datasets when the focus is the model. We also observe that Occlusion is better at returning sufficient explanations, while Saliency is more appropriate for necessary explanations. This trade-off study shows that there is no existing explainability method that dominates others in all aspects.

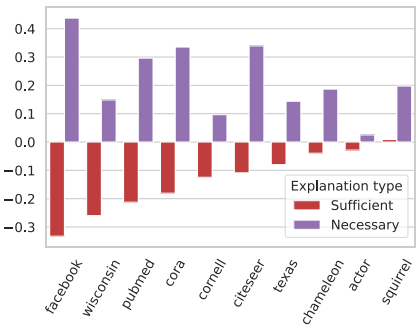


Figure 6: Trade-off between Occlusion and Saliency. Relative $fid+$ and $(1\text{-}fid-)$. Positive scores: superiority of Saliency.

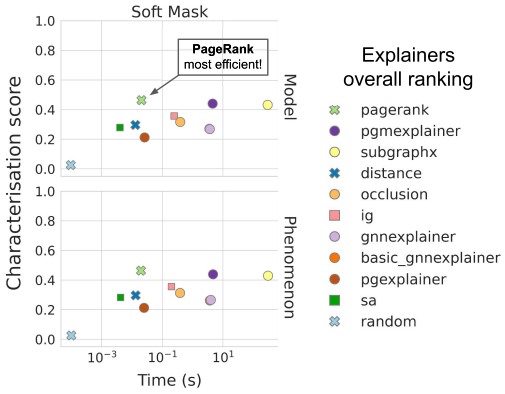
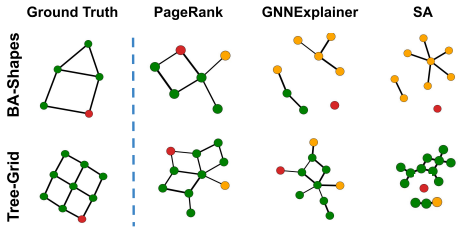


Figure 7: Performance vs computation time for synthetic data. The explanation is a soft mask, *i.e.* edges are weighted by their importance.

Limit of synthetic benchmarks. We further reveal the limitations of evaluating explainability methods on type 1 synthetic datasets. We show inconsistency between the method rankings on real and those widely used synthetic datasets [10]. While PageRank returns the most accurate explanations (right table on Figure 8), and has the best time-performance trade-off and characterization score (see Figure 7) on synthetic data, this structure-based method is not able to highlight the important entities of real graphs (see Figure 3). In addition, Saliency has one of the lowest accuracies on every synthetic dataset, while it is the most optimal method to explain GCNs on real graphs (see Fig. 3). Method assessment on synthetic datasets eludes the power of gradient-based methods and their ability to extract decisive graph features when node dependency is not elementary and node features are meaningful. These examples demonstrate that evaluation on type 1 synthetic datasets gives only poor informative insight.



Explainer	BA-Shapes syn1	BA-Grid syn2	Tree-Cycle syn3	Tree-Grid syn4	BA-Bottles syn5	CC-ratio
Random	0.005	0.007	0.003	0	0	0.486
Distance	0.678	0.731	0.251	0.385	**0.733**	**0.1**
PageRank	**0.862**	**0.938**	0.323	**0.485**	0.658	0.128
SA	0.043	0	0.071	0.222	0.101	0.2
IG	0.447	0.186	0.244	0.435	0.400	0.18
Occlusion	0.172	0	0.220	0.268	0.074	0.206
GNNExplainer	0.310	0.664	0.233	0.328	0.127	0.176
PGM-Explainer	0.543	0.663	**0.350**	0.473	0.394	0.174
PGExplainer	0.652	0.801	0.204	0.236	0.455	0.262
SubgraphX	0.613	0.758	0.218	0.238	0.660	0.217

Figure 8: Accuracy on synthetic data. Explanations are generated to have the same number of edges than the expected groundtruth motif. (left) Explanatory subgraphs are drawn next to the expected ground truth. They contain the **target** node, **explanatory** nodes and **other** nodes. (right) F1-score indicates the similarity between the explanatory subgraph and the motif and CC-ratio the connectivity.

5.3 Case study: explaining frauds in the real-world e-commerce graph

We test our systematic evaluation framework on a production use case: explaining fraudulent transactions in the e-commerce scenario at eBay. In the scope of our research, we only explain correct predictions[1]. GNNexplainer is by far the most effective method (see Figure 9). It also returns not only sufficient explanations like most of the methods, but also necessary explanations. While the edge mask is directly deduced from the node feature mask in Saliency and Integrated Gradient, GNNExplainer has the particularity to compute edge and node features importance independently when solving the optimization problem. This explains the superiority of GNNExplainer in this production case where node features and edges bring both different insights to understand fraudulent nodes. Overall, we notice that perturbation-based methods are better than structure-based and gradient-based methods in this production use case.

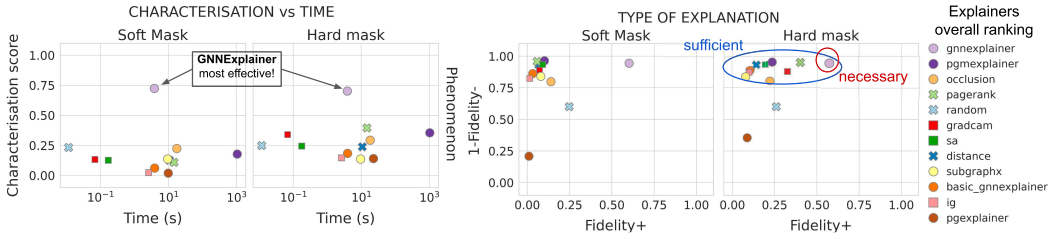


Figure 9: Results on eBay graph to explain correctly predicted fraudulent nodes. Results for the model focus are omitted as they correspond to the phenomenon. Explanation size is $topk = 10$. *sa* stands for Saliency; *ig* stands for Intergrated Gradient.

6 Conclusion

In this paper, we propose GraphFramEx, a systematic evaluation framework for explainability methods for GNNs. By deliberately choosing methods from all categories, our comparison covers the full spectrum of input-level explainers for node classification tasks. Taking as model a GCN, we show the limits of a traditional evaluation on type 1 synthetic data. Our evaluation with the characterization score allows us to fairly evaluate all sorts of explainability methods in real-world scenarios. With our trade-off study, we however want to raise awareness that users should not rely on one single method to explain and trust their decision-making algorithm. Our case study on eBay graph shows the outstanding performance of GNNExplainer for explaining correctly predicted fraudulent nodes.

GraphFramEx is intended to help users navigate through the increasing number of explainability methods for GNNs. We encourage people to evaluate new explainability methods on real data and/or the 3 synthetic benchmarks [25] - *type 2* synthetic data - as they better reflect real-world complexity. While our work interprets explanations as positive weights masking the existing graph entities, we also aim at exploring new definitions that also involve non-adjacent pairs of nodes and assess the negative impact of edges and node features on the predicted outcomes.

[1]To circumvent the classification error of the trained GNN (Appendix B)

References

[1] Michaela Benk and Andrea Ferrario. Explaining interpretable machine learning: Theory, methods and applications. *SSRN Electron. J.*, 2020. 1, 13

[2] Davinder Kaur, Suleyman Uslu, Kaley J Rittichier, and Arjan Durresi. Trustworthy artificial intelligence: A review. *ACM Comput. Surv.*, 55(2):1–38, January 2022. 1

[3] Yongji Wu, Defu Lian, Yiheng Xu, Le Wu, and Enhong Chen. Graph convolutional networks with markov random field reasoning for social spammer detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1054–1061, 2020. 1

[4] Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin Riedmiller, Raia Hadsell, and Peter Battaglia. Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning*, pages 4470–4479. PMLR, 2018.

[5] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. *Advances in neural information processing systems*, 29, 2016.

[6] Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur. Protein interface prediction using graph convolutional networks. *Advances in neural information processing systems*, 30, 2017.

[7] Takuo Hamaguchi, Hidekazu Oiwa, Masashi Shimbo, and Yuji Matsumoto. Knowledge transfer for out-of-knowledge-base entities: A graph neural network approach. *arXiv preprint arXiv:1706.05674*, 2017.

[8] Elias Khalil, Hanjun Dai, Yuyu Zhang, Bistra Dilkina, and Le Song. Learning combinatorial optimization algorithms over graphs. *Advances in neural information processing systems*, 30, 2017. 1

[9] Federico Baldassarre and Hossein Azizpour. Explainability techniques for graph convolutional networks. May 2019. 1, 2, 6, 15, 16, 21

[10] Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. GNNExplainer: Generating explanations for graph neural networks. *Adv. Neural Inf. Process. Syst.*, 32:9240–9251, December 2019. 1, 2, 3, 6, 8, 15, 16, 18

[11] Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network. November 2020. 2, 6, 21

[12] Yue Zhang, David Defazio, and Arti Ramesh. RelEx: A Model-Agnostic relational model explainer. May 2020. 2, 21

[13] Minh N Vu and My T Thai. PGM-Explainer: Probabilistic graphical model explanations for graph neural networks. October 2020. 2, 6, 16, 21

[14] Caihua Shan, Yifei Shen, Yao Zhang, Xiang Li, and Dongsheng Li. Reinforcement learning enhanced explainer for graph neural networks. https://papers.nips.cc/paper/2021/file/be26abe76fb5c8a4921cf9d3e865b454-Paper.pdf. Accessed: 2021-11-24. 2, 15, 16, 21

[15] Anonymous Authors. Hard masking for explaining graph neural networks. https://openreview.net/pdf?id=uDN8pRAdsoC, . Accessed: 2022-6-4. 2

[16] Hao Yuan, Haiyang Yu, Jie Wang, Kang Li, and Shuiwang Ji. On explainability of graph neural networks via subgraph explorations. February 2021. 2, 4, 6, 13, 15, 16, 21

[17] Ana Lucic, Maartje ter Hoeve, Gabriele Tolomei, Maarten de Rijke, and Fabrizio Silvestri. CF-GNNExplainer: Counterfactual explanations for graph neural networks. February 2021. 2

[18] Mohit Bajaj, Lingyang Chu, Zi Yu Xue, Jian Pei, Lanjun Wang, Peter Cho-Ho Lam, and Yong Zhang. Robust counterfactual explanations on graph neural networks. July 2021. 2, 4

[19] Wanyu Lin, Hao Lan, and Baochun Li. Generative causal explanations for graph neural networks. April 2021. 2, 15, 16, 21

[20] Hao Yuan, Jiliang Tang, Xia Hu, and Shuiwang Ji. XGNN: Towards Model-Level explanations of graph neural networks. June 2020. 3, 21

[21] Thomas Schnake, Oliver Eberle, Jonas Lederer, Shinichi Nakajima, Kristof T Schutt, Klaus-Robert Mueller, and Gregoire Montavon. Higher-Order explanations of graph neural networks via relevant walks. *IEEE Trans. Pattern Anal. Mach. Intell.*, PP, September 2021. 13, 21

[22] Anonymous Authors. Causal screening to interpret graph neural networks. https://openreview.net/pdf?id=nzKv5vxZfge, . Accessed: 2022-6-5. 13, 21

[23] Shirley (ying-Xin) Wu. ReFine: Towards Multi-Grained explainability for graph neural networks. 21

[24] Phillip E Pope, Soheil Kolouri, Mohammad Rostami, Charles E Martin, and Heiko Hoffmann. Explainability methods for graph convolutional neural networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2019. 1, 13, 21

[25] Lukas Faber, Amin K Moghaddam, and Roger Wattenhofer. When comparing to ground truth is wrong: On evaluating GNN explanation methods. 2, 3, 6, 9, 15

[26] Hao Yuan, Haiyang Yu, Shurui Gui, and Shuiwang Ji. Explainability in graph neural networks: A taxonomic survey. December 2020. 2, 3, 5, 13, 14, 15, 16, 21

[27] Peibo Li, Yixing Yang, Maurice Pagnucco, and Yang Song. Explainability in graph neural networks: An experimental survey. March 2022. 2, 3

[28] Chirag Agarwal, Marinka Zitnik, and Himabindu Lakkaraju. Probing GNN explainers: A rigorous theoretical and empirical analysis of GNN explanation methods. June 2021. 2, 3, 21

[29] Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020. 2, 15

[30] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. *Chemical science*, 9(2):513–530, 2018. 2, 15

[31] Kacper Sokol and Peter Flach. Explainability fact sheets: A framework for systematic assessment of explainable approaches. December 2019. 3

[32] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. January 2022. 3

[33] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, New York, NY, USA, January 2020. ACM. 4

[34] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. November 2017. 5

[35] Carl Yang. HNE: Heterogeneous network embedding: Survey, benchmark, evaluation, and beyond. 6

[36] Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. September 2019. 6

[37] Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-GCN: Geometric graph convolutional networks. February 2020. 6

[38] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. September 2016. 6, 13, 18, 20

[39] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. October 2017. 6, 13, 20

[40] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? October 2018. 6, 13, 20

[41] Sergey Bring. The PageRank citation ranking: bringing order to the web. *Proceedings of ASIS, 1998*, 98:161–172, 1998. 6

[42] Hanzhi Wang, Zhewei Wei, Junhao Gan, Sibo Wang, and Zengfeng Huang. Personalized PageRank to a target node, revisited. June 2020. 6

[43] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. March 2017. 6

[44] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. October 2016. 6

[45] Qiang Huang, Makoto Yamada, Yuan Tian, Dinesh Singh, Dawei Yin, and Yi Chang. GraphLIME: Local interpretable model explanations for graph neural networks. January 2020. 13

[46] Anonymous Authors. Zorro: Hard masking for explaining graph neural networks. . 13

[47] Benjamin Sanchez-Lengeling, Jennifer Wei, Brian Lee, Emily Reif, Peter Y Wang, Wesley Wei Qian, Kevin Mc Closkey, Lucy Colwell, and Alexander Wiltschko. Evaluating attribution for graph neural networks. https://papers.nips.cc/paper/2020/file/417fbbf2e9d5a28a855a11894b2e795a-Paper.pdf. Accessed: 2021-11-22. 13

[48] Christoph Molnar. Interpretable machine learning. https://christophm.github.io/interpretable-ml-book/index.html, January 2022. Accessed: 2022-1-7. 13

[49] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008. 16

[50] Susie Xi Rao, Shuai Zhang, Zhichao Han, Zitao Zhang, Wei Min, Zhiyao Chen, Yinan Shan, Yang Zhao, and Ce Zhang. xfraud. *Proceedings VLDB Endowment*, 15(3):427–436, November 2021. 17

[51] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 18

[52] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. December 2014. 18