Interpretability Methods for Graph Neural Networks

Arijit Khan
Aalborg University
Aalborg, Denmark
arijitk@cs.aau.dk

Ehsan B. Mobaraki
Aalborg University
Aalborg, Denmark
ebmo@cs.aau.dk

***Abstract*—The emerging graph neural network models (GNNs) have demonstrated great potential and success for downstream graph machine learning tasks, such as graph and node classification, link prediction, entity resolution, and question answering. However, neural networks are "black-box" – it is difficult to understand which aspects of the input data and the model guide the decisions of the network. Recently, several interpretability methods for GNNs have been developed, aiming at improving the model's transparency and fairness, thus making them trustworthy in decision-critical applications, leading to democratization of deep learning approaches and easing their adoptions. The tutorial is designed to offer an overview of the state-of-the-art interpretability techniques for graph neural networks, including their taxonomy, evaluation metrics, benchmarking study, and ground truth. In addition, the tutorial discusses open problems and important research directions.**

***Index Terms*—graph neural networks, interpretability, explainable AI**

I. Introduction

Graph neural networks (GNNs) [1] have demonstrated great potential in numerous downstream applications including graph and node classification, link prediction, entity resolution, question answering, recommendation, fraud, and anomaly detection over many real-world domains such as social networks, knowledge graphs, bioinformatics, transportation, and finance [2]–[7]. GNNs are a type of deep learning models designed to tackle graph-related tasks in an end-to-end manner. Therefore, it remains a desirable yet nontrivial task to explain the results of high-quality GNNs.

Recently, interpretability methods for GNNs are gaining rapid attention [8]. The state-of-the-art interpretability methods discover critical nodes, edges, subgraphs, and their features that are responsible for GNN outcomes. However, GNN interpretability methods have not been thoroughly benchmarked against each other. Importantly, the evaluation frameworks, datasets, metrics, downstream tasks, and GNNs employed were often not consistent across past works. Lack of ground truths and errors introduced while evaluating the performance of GNN interpretability methods are other concerns [9], [10].

To this end, our tutorial focuses on categorization, algorithms, and outputs of the state-of-the-art GNN interpretability techniques, their challenges, evaluation metrics, and underlying graph neural networks, in the context of revealing latent decision-making by black-box models for various downstream tasks. Our comprehensive coverage will provide valuable insights into the strengths and weaknesses of different GNN interpretability approaches, their performance and efficiency, as well as their applicability in real-world scenarios.

The work presented in this article is supported by the Novo Nordisk Foundation grant NNF22OC0072415.

II. Motivation of the Tutorial

With the proliferation of deep learning in a wide range of applications, their interpretability methods, or explainable AI is experiencing rapid growth to tackle the black-box nature of deep learning approaches. To safely and trustfully deploy deep models, it is critical to provide both state-of-the-art performance and human-intelligible explanations, especially for data scientists and end-users from interdisciplinary, or even non-machine-learning domains, e.g., biologists, chemists, social scientists, journalists, policymakers, etc. While there have been many tutorials and surveys [11]–[20] about interpretability tools for deep neural networks, peer-reviewed surveys and benchmarking studies on interpretability methods for graph neural networks (GNNs) and the coverage therein are rather limited [8], [21]–[26].

We aim at bridging this gap by discussing 19 recent GNN interpretability methods published in the last five years, together with their classification into different categories, evaluation metrics, performance study, and ground truth. Our tutorial has potentials such as attracting interdisciplinary research and designing human-in-the-loop systems and interpretable data science pipelines for various graph machine learning tasks.

III. Outline of the Tutorial

```
1 Introduction (15 minutes)
  1.1 Graph neural networks (GNNs) and applications
  1.2 Interpretability of GNNs
      - Definitions, importance, and challenges
2 Taxonomy of interpretability methods for GNNs (15 minutes)
  2.1 Post-hoc vs. intrinsic
  2.2 Global/ class-specific vs. local/ instance-specific
  2.3 Model-specific vs. model-agnostic
  2.4 Forward vs. backward
  2.5 Node-level vs. edge-level vs. subgraph-level
  2.6 Perturbation vs. gradient vs. decomposition
      vs. surrogate models vs. counterfactuals
3 Recent interpretability methods for GNNs (30 minutes)
4 Benchmark & ground truth for GNN interpretability methods
  (15 minutes)
  4.1 Interpretability evaluation metrics
  4.2 Benchmarking results
  4.3 Ground truth datasets
5 Future directions (15 minutes)
```

IV. Description and Covered Works

We provide a brief description of the topics and categorization of important covered works. Due to the space limit, we

only mention the most relevant papers. However, this is not an exhaustive list of papers that are related and will be discussed during the tutorial.

• **Background.** We shall introduce background materials focusing on **(i)** *graph neural networks (GNNs) and applications:* GNNs generally follow recursive neighborhood aggregation or message passing scheme. We shall discuss recent GNNs designed for node and graph classification (e.g., GCN [27], DGCNN [28], DiffPool [29], GIN [30]), link prediction (e.g., SEAL [31], LGLP [32]), and entity resolution (e.g., GraphER [33]). **(ii)** *Definitions and importance of interpretability methods*, including common notions of explanation and interpretation, their usefulness in deriving insights about the model and data, causality, ensuring trust, fairness, safety, and deploying GNNs in decision-critical applications. **(iii)** *Challenges of interpretability methods for GNNs*, such as not grid-like data, bias, redundant evidence, weak GNN model, and misaligned GNN architecture [9].

• **Taxonomy of interpretability methods for GNNs.** We categorize recent interpretability techniques for GNNs as follows [8], [34]. *Intrinsic* approaches construct self-explanatory models that incorporate interpretability directly into their structures, e.g., graph attention networks (GATs) [35]. *Post-hoc* methods create a separate model to provide explanations for an existing GNN. In *global* interpretability methods, users understand how the model works globally by inspecting the structures and parameters of a GNN model, or by generating graph patterns which maximize a certain prediction of the model. In contrast, *local* interpretability methods examine an individual prediction of a model, figuring out why the model makes the decision on a specific test instance.

Forward interpretability methods are *GNN model-agnostic*, they learn evidence about graphs or nodes passed through the GNN. They can be *perturbation-based*, that is, masking some node features and/or edge features and analyzing the changes when the modified graphs are passed through the GNN model. They might also employ a simple, interpretable *surrogate model* to approximate the predictions of a complex GNN, or *counterfactuals-based*, i.e., finding a subgraph whose information is necessary which if removed will result in different predictions. On the other hand, *backward* interpretability methods are *GNN model-specific* and can be either *gradient-based* – backpropagating importance signal backward from the output neuron of the model to the individual nodes of the input graph, or *decomposition-based* – distributing the prediction score in a backpropagation manner until the input layer. Thus, one identifies which nodes, edges, and features contribute the most to the specific output label in the GNN.

• **Recent interpretability methods for GNNs.** We shall discuss and compare 19 recent GNN interpretability methods from the aforementioned categories: GNNExplainer [36], PGExplainer [37], GraphMask [38], SubgraphX [39], PGM-Explainer [40], RelEx [41], GraphLime [42], RCExplainer [43], DnX [44], GCFExplainer [45], CF^2 [46], SA [26], GuidedBP [26], CAM [21], Grad-CAM [47], LRP [48], GNN-LRP [49], ExcitationBP [21], and XGNN [50].

Sensitivity Analysis (SA) considers the squared values of gradients as the importance of different input features.

GuidedBP follows a similar measure as SA, but it only relies on positive gradients, while masking negative gradients.

CAM requires global average pooling (GAP) to the final convolutional feature maps. It uses the weights of the target classes in the classifier layer and maps them to the output embeddings of the last convolutional layer.

Grad-CAM, unlike CAM, does not rely on a GAP layer. It combines feature maps in the last convolutional layer with the gradients in the input layer.

LRP backpropagates the prediction score of a GNN to all nodes by using weights and activation values.

ExcitationBP employs a similar approach as LRP. However, the final scoring techniques and the rules applied in backpropagation of outputs are different.

GNNExplainer learns masks for edges and node features that generate an evidence subgraph of the input graph. The masks are optimized to maximize the mutual information between the predictions of the input graph and that of the evidence subgraph.

PGExplainer performs a similar process as GNNExplainer that maximizes the mutual information between the predictions of the input graph and that of the evidence subgraph; however, it only focuses on the graph structure by using a deep neural network to parameterize the generation of the evidence subgraph. PGExplainer can explain multiple instances collectively and also works in an inductive setting.

GraphMask, similar to PGExplaienr, trains a parameterized classifier that predicts if an edge can be dropped without sufficiently changing the prediction of the model, but performs it for every edge at every GNN layer.

SubgraphX uses Monte Carlo tree search to select the most important subgraph with Shapley value-based formulation.

PGM-Explainer uses an interpretable Bayesian network generated from node features perturbation.

RelEx follows two steps — perturbation-based learning of a local differentiable approximation for GNN, and then learning an interpretable mask over the local approximation.

GraphLime trains a nonlinear surrogate model to the local dataset surrounding a node to explain node classification.

Distill n' Explain (DnX) learns a surrogate GNN via knowledge distillation, and then extracts explanations by solving a convex program.

CF^2, by using causal inference theory, generates both necessary and sufficient (counterfactual and factual) explanations.

RCExplainer partitions the logic of a GNN into a set of decision regions, then by exploring a common decision logic for samples in the same class, it generates robust counterfactual explanations for them.

GCFExplainer conducts node-reinforced random walks on input graphs to generate counterfactual candidates as greedily-summarized global explanations.

GNN-LRP proposes higher-order explanations by scoring sequences of edges, i.e., walks in a graph, instead of individual nodes or edges.

XGNN, unlike instance-level explanations, generates graphs to maximize a final prediction. In particular, XGNN trains a graph generator to generate graphs that can optimize model-level interpretation.

• **Benchmark and ground truth for GNN interpretability.** Several works [9], [21], [23], [24], [26], [39], [51]–[53] theoretically and empirically compared GNN-based interpretability methods. The following open-source libraries implemented many GNN explainability methods: DIG [54], DGL (https://docs.dgl.ai/index.html), and Pytorch-Geometric (https://pytorch-geometric.readthedocs.io/en/stable/). We shall discuss **(i)** *evaluation metrics* for GNN interpretability methods, e.g., fidelity, sparsity, contrastivity, accuracy, and stability [8], [21]; **(ii)** *benchmarking results* for GNN interpretability methods, and **(iii)** *ground-truth datasets*, such as BA-shapes, BA-Community, BA-2Motifs, etc. Specifically, fidelity [8] computes the decrease in accuracy by masking important (or salient) input features. Contrastivity [21] computes the normalized difference of saliency maps across different classes, reporting how class-specific the explanations are. Sparsity [21] measures the size of the explanation set. Stability reports changes in attribution by feeding perturbed input graphs [25]. Besides individual test instance-specific results, one can demonstrate GNN model-specific aggregate results by visualizing important frequent subgraphs induced by salient nodes from a set of test instances [24]. When ground-truth interpretability results are available, we can compute accuracy, precision, recall, and F1-measures of the employed interpretability methods. We shall summarize which interpretability method is more suitable under what evaluation metrics, datasets, GNNs, and downstream tasks.

• **Future Directions.** Future work can be in several directions.

– Past benchmarking efforts mainly considered graph and node classifications as downstream tasks. The effectiveness of the interpretability methods in other graph machine learning computations such as link prediction, entity resolution, and question answering are yet to be properly investigated.

– Bulk of the literature generally considered static, attributed graphs as the underlying dataset. The performance of the interpretability methods on other types of graphs, e.g., knowledge graphs, spatio-temporal networks, multi-modal networks, etc. need to be evaluated.

– Existing methods assign importance scores to nodes, edges, features, and subgraphs. It is also important to investigate and visualize what higher-order representations are learnt by the intermediate neurons, which would facilitate model comprehension.

– A more direct, human-in-the-loop evaluation of GNN interpretability methods would be useful, for instance, how they assist in improving user's understanding and trust in the GNN model.

V. RELATED TUTORIALS

We have not given a prior tutorial on the topics. Recent related tutorials include explainable AI in data management [ICDE22, SIGMOD22], deep learning interpretation [CIKM22], Counterfactual explanations [KDD21], machine learning explainability and robustness [KDD21], explainable AI in industry [KDD19, AAAI20]. The scope of our tutorial is different from those past tutorials.

VI. AUDIENCE PARTICIPATION

The tutorial is intended for researchers and practitioners in the broad area of deep learning, explainable AI, and graph data models. Familiarity with basic machine learning and neural techniques would be helpful. The tutorial is designed for 40% novice, 30% intermediate, and 30% expert, to maintain a balance between the overview and technical contents.

VII. PRESENTER BIOGRAPHY

Arijit Khan is an IEEE senior member and an associate professor in the department of computer science, Aalborg University, Denmark. He earned his Ph.D. from UC Santa Barbara, USA and did a post-doc at ETH Zurich, Switzerland. He has been an assistant professor at NTU Singapore. Arijit is the recipient of the IBM Ph.D. Fellowship (2012-13). He published several papers in premier data management and mining venues, e.g., SIGMOD, VLDB, TKDE, ICDE, WWW, SDM, EDBT, CIKM, WSDM, and TKDD. Arijit co-presented tutorials on graph queries, systems, and applications at ICDE (2012), VLDB (2014, 2015, 2017), and CIKM (2022); served in the program committee of KDD, SIGMOD, VLDB, ICDE, ICDM, EDBT, SDM, CIKM, AAAI, WWW, and as an associate editor of TKDE and TKDD. More information at https://homes.cs.aau.dk/~Arijit/index.html.

Ehsan B. Mobaraki is a Ph.D. student in the department of computer science, Aalborg University, Denmark. He works on interpretability methods for graph neural networks. He published in the Joint Workshop on Graph Data Management Experiences & Systems (GRADES) and Network Data Analytics (NDA), co-located with SIGMOD 2023. More information at https://emobaraki94.wixsite.com/mywebpage.

REFERENCES

[1] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32, no. 1, pp. 4–24, 2021.

[2] X.-M. Zhang, L. Liang, L. Liu, and M.-J. Tang, "Graph neural networks and their current applications in bioinformatics," *Frontiers in Genetics*, vol. 12, 2021.

[3] H. Ren, W. Hu, and J. Leskovec, "Query2box: reasoning over knowledge graphs in vector space using box embeddings," in *ICLR*, 2020.

[4] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. S. Pande, "Moleculenet: a benchmark for molecular machine learning," *CoRR*, vol. abs/1703.00564, 2017.

[5] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, "Graph neural networks in recommender systems: a survey," *ACM Comput. Surv.*, vol. 55, no. 5, pp. 97:1–97:37, 2023.

[6] J. Wu, Q. Yuan, D. Lin, W. You, W. Chen, C. Chen, and Z. Zheng, "Who are the phishers? phishing scam detection on ethereum via network embedding," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 52, no. 2, pp. 1156–1166, 2022.

[7] T. S. Jepsen, C. S. Jensen, and T. D. Nielsen, "Graph convolutional networks for road networks," in *SIGSPATIAL*, 2019.

[8] H. Yuan, H. Yu, S. Gui, and S. Ji, "Explainability in graph neural networks: A taxonomic survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2022.

[9] L. Faber, A. K. Moghaddam, and R. Wattenhofer, "When comparing to ground truth is wrong: on evaluating gnn explanation methods," in *KDD*, 2021.

[10] L. Faber, A. K. Moghaddam, and R. Wattenhofer, "Contrastive graph neural network explanation," in *ICML workshop on Graph Representation Learning and Beyond*, 2020.

[11] G. Ras, N. Xie, M. v. Gerven, and D. Doran, "Explainable deep learning: a field guide for the uninitiated," *J. Artif. Intell. Res.*, vol. 73, pp. 329–396, 2022.

[12] Z. C. Lipton, "The mythos of model interpretability," *Commun. ACM*, vol. 61, no. 10, 2018.

[13] M. Du, N. Liu, and X. Hu, "Techniques for interpretable machine learning," *Commun. ACM*, vol. 63, no. 1, 2020.

[14] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine learning interpretability: a survey on methods and metrics," *Electronics*, vol. 8, no. 8, 2019.

[15] R. Pradhan, A. Lahiri, S. Galhotra, and B. Salimi, "Explainable AI: foundations, applications, opportunities for data management research," in *ICDE*, 2022.

[16] Z. Yang, N. Liu, X. B. Hu, and F. Jin, "Tutorial on deep learning interpretation: a data perspective," in *CIKM*, 2022.

[17] C. Wang, X. Li, H. Han, S. Wang, L. Wang, C. C. Cao, and L. Chen, "Counterfactual explanations in explainable AI: a tutorial," in *KDD*, 2021.

[18] A. Datta, M. Fredrikson, K. Leino, K. Lu, S. Sen, and Z. Wang, "Machine learning explainability and robustness: connected at the hip," in *KDD*, 2021.

[19] K. Gade, S. C. Geyik, K. Kenthapadi, V. Mithal, and A. Taly, "Explainable AI in industry," in *KDD*, 2019.

[20] C. Agarwal, E. Saxena, S. Krishna, M. Pawelczyk, N. Johnson, I. Puri, M. Zitnik, and H. Lakkaraju, "OpenXAI: towards a transparent evaluation of model explanations," *CoRR*, vol. abs/2206.11104, 2022.

[21] P. E. Pope, S. Kolouri, M. Rostami, C. E. Martin, and H. Hoffmann, "Explainability methods for graph convolutional neural networks," in *CVPR*, 2019.

[22] C. Agarwal, O. Queen, H. Lakkaraju, and M. Zitnik, "Evaluating explainability for graph neural networks," *Sci Data*, vol. 10, no. 1, 2023.

[23] C. Agarwal, M. Zitnik, and H. Lakkaraju, "Probing GNN explainers: a rigorous theoretical and empirical analysis of GNN explanation methods," in *AISTATS*, 2022.

[24] K. T. T. Shun, E. E. Limanta, and A. Khan, "An evaluation of backpropagation interpretability for graph classification with deep learning," in *IEEE Big Data*, 2020.

[25] B. Sánchez-Lengeling, J. Wei, B. Lee, E. Reif, P. Wang, W. Qian, K. McCloskey, L. Colwell, and A. Wiltschko, "Evaluating attribution for graph neural networks," in *NeurIPS*, 2020.

[26] F. Baldassarre and H. Azizpour, "Explainability techniques for graph convolutional networks," in *ICML Workshop on Learning and Reasoning with Graph-Structured Representations*, 2019.

[27] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017.

[28] M. Zhang, Z. Cui, M. Neumann, and Y. Chen, "An end-to-end deep learning architecture for graph classification," in *AAAI*, 2018.

[29] Z. Ying, J. You, C. Morris, X. Ren, W. L. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," in *NeurIPS*, 2018.

[30] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *ICLR*, 2019.

[31] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," in *NeurIPS*, 2018.

[32] L. Cai, J. Li, J. Wang, and S. Ji, "Line graph neural networks for link prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 9, pp. 5103–5113, 2022.

[33] B. Li, W. Wang, Y. Sun, L. Zhang, M. A. Ali, and Y. Wang, "Grapher: token-centric entity resolution with graph convolutional neural networks," in *AAAI*, 2020.

[34] J. Kakkad, J. Jannu, K. Sharma, C. C. Aggarwal, and S. Medya, "A survey on explainability of graph neural networks," *CoRR*, vol. abs/2306.01958, 2023.

[35] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *ICLR*, 2018.

[36] Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, "GNNExplainer: generating explanations for graph neural networks," in *NeurIPS*, 2019.

[37] D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen, and X. Zhang, "Parameterized explainer for graph neural network," in *NeurIPS*, 2020.

[38] M. S. Schlichtkrull, N. D. Cao, and I. Titov, "Interpreting graph neural networks for NLP with differentiable edge masking," in *ICLR*, 2021.

[39] H. Yuan, H. Yu, J. Wang, K. Li, and S. Ji, "On explainability of graph neural networks via subgraph explorations," in *ICML*, 2021.

[40] M. N. Vu and M. T. Thai, "PGM-Explainer: probabilistic graphical model explanations for graph neural networks," in *NeurIPS*, 2020.

[41] Y. Zhang, D. DeFazio, and A. Ramesh, "RelEx: a model-agnostic relational model explainer," in *AIES*, 2021.

[42] Q. Huang, M. Yamada, Y. Tian, D. Singh, D. Yin, and Y. Chang, "GraphLIME: local interpretable model explanations for graph neural networks," *CoRR*, vol. abs/2001.06216, 2020.

[43] M. Bajaj, L. Chu, Z. Y. Xue, J. Pei, L. Wang, P. C. Lam, and Y. Zhang, "Robust counterfactual explanations on graph neural networks," in *NeurIPS*, 2021.

[44] T. A. Pereira, E. Nasciment, L. E. Resck, D. Mesquita, and A. H. Souza, "Distill n' Explain: explaining graph neural networks using simple surrogates," in *AISTATS*, 2023.

[45] Z. Huang, M. Kosan, S. Medya, S. Ranu, and A. K. Singh, "Global counterfactual explainer for graph neural networks," in *WSDM*, 2023.

[46] J. Tan, S. Geng, Z. Fu, Y. Ge, S. Xu, Y. Li, and Y. Zhang, "Learning and evaluating graph neural network explanations based on counterfactual and factual reasoning," in *WWW*, 2022.

[47] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: visual explanations from deep networks via gradient-based localization," in *ICCV*, 2017.

[48] R. Schwarzenberg, M. Hübner, D. Harbecke, C. Alt, and L. Hennig, "Layerwise relevance visualization in convolutional text graph classifiers," in *TextGraphs@EMNLP*, 2019.

[49] T. Schnake, O. Eberle, J. Lederer, S. Nakajima, K. T. Schütt, K. Müller, and G. Montavon, "Higher-order explanations of graph neural networks via relevant walks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7581–7596, 2022.

[50] H. Yuan, J. Tang, X. Hu, and S. Ji, "XGNN: towards model-level explanations of graph neural networks," in *KDD*, 2020.

[51] A. Longa, S. Azzolin, G. Santin, G. Cencetti, P. Liò, B. Lepri, and A. Passerini, "Explaining the explainers in graph neural networks: a comparative study," *CoRR*, vol. abs/2210.15304, 2022.

[52] E. B. Mobaraki and A. Khan, "A demonstration of interpretability methods for graph neural networks," in *GRADES & NDA@SIGMOD*, 2023.

[53] P. Li, Y. Yang, M. Pagnucco, and Y. Song, "Explainability in graph neural networks: an experimental survey," *CoRR*, vol. abs/2203.09258, 2022.

[54] M. Liu, Y. Luo, L. Wang, Y. Xie, H. Yuan, S. Gui, H. Yu, Z. Xu, J. Zhang, Y. Liu, K. Yan, H. Liu, C. Fu, B. Oztekin, X. Zhang, and S. Ji, "DIG: a turnkey library for diving into graph deep learning research," *J. Mach. Learn. Res.*, vol. 22, pp. 240:1–240:9, 2021.