Yale

XAI for Graphs

Guest Lecture at Stanford CS 224W: Machine Learning with Graphs

Rex Ying

Readings

- Readings are updated on the website (syllabus page)
- **Readings**:
  - LIME (local interpretation)
  - SHAP (attribution)
  - GNNExplainer
  - GNN Explainability Taxonomy
  - Trustworthy Graph Neural Networks
  - GraphFramEx Evaluation

Trustworthy Graph Learning

- **Trustworthy** AI/GNN includes many components
  - Explainability, fairness, robustness, privacy, …
  - Algorithms to tackle combination of these aspects
- **Challenges**
  - Role of graph topology is previously unexplored in these problems
  - Comprehensive quantiative evaluation

Big Picture: Aspects of Trustworthy GNNs

- **Robustness**
- **Explainability**
- Privacy
- Fairness
- Accountability
- Environmental well-being
- Others

Each aspect can play a role in gaining trust from users of deep learning models

Challenges in GNN context

- **Role of graph topology is previously unexplored in these problems**
- **Quantiative evaluation is often difficult**

Outline of Today's Lecture

1. **Explainability and its Problem Settings**

2. **GNNExplainer**

3. **Explainability Evaluation**

Outline of Today's Lecture

1. Explainability and its Problem Settings

Motivation, goals and settings

2. GNNExplainer

3. Explainability Evaluation

Explainability

- The **black-box** nature of deep learning makes it a **major challenge** to:
  - Understand what is learned by the ML model
  - Extract insights of the underlying data we are trying to model
- **Explainable Artificial Intelligence (XAI)** is an umbrella term for any research trying to solve the **black-box problem for AI**
- Why is it useful?
  - Enable users to **understand the decision-making** of the model
  - **Gain trust from human users** of the deep learning system
- Simple-to-read guide: [2004.14545.pdf (arxiv.org)]

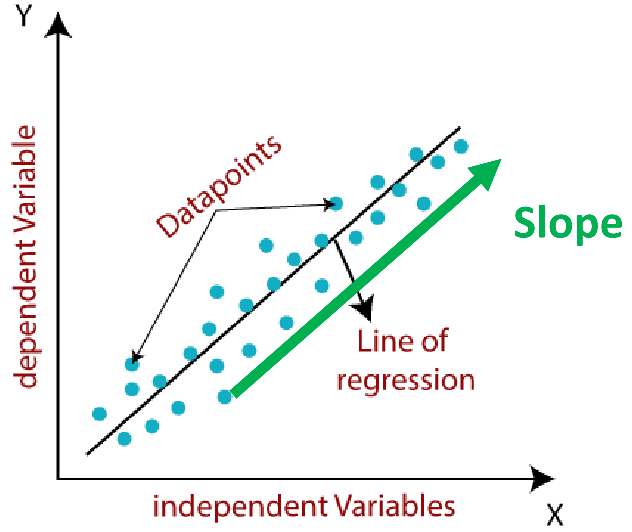
What was explainable about previous ML models?

Explainable Models: Linear regression

- **Linear regression**
  - **Slope is explainable** (how much does one variable affects a prediction)
  - $y = w_1 x_1 + w_2 x_2 + w_3 x_3 + \cdots$

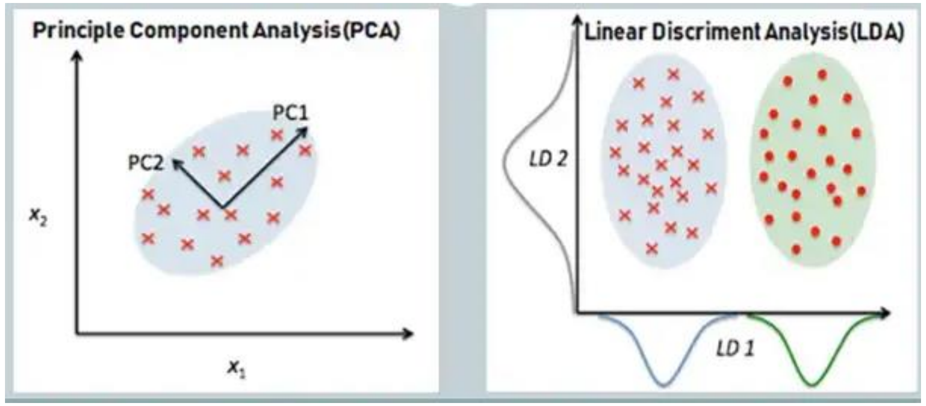
prediction weights features

- Each feature has an associated **weights**, indicating its **importance**
  - "A change of Δx amount to feature x_1 will result in increase of prediction by Δy

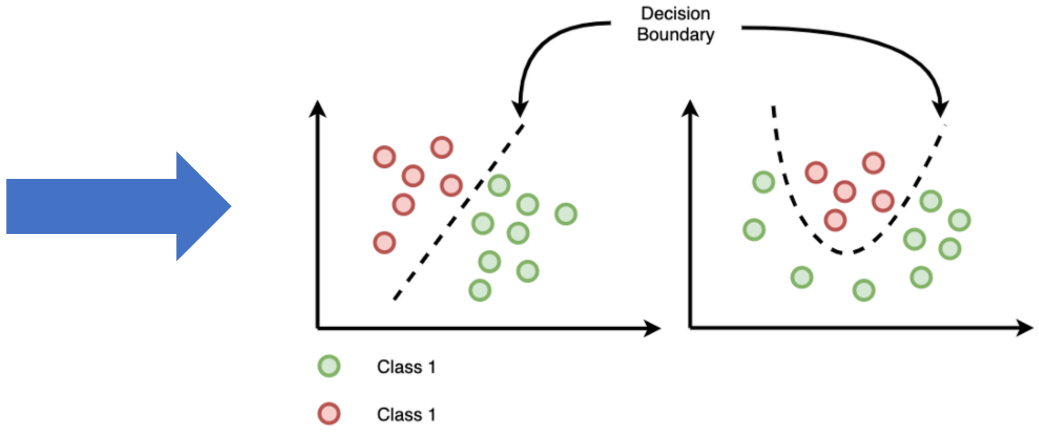


Explainable Models: Dimension Reduction

- **Dimension reduction**
  - Dimension reduction allows us to visualize the training data distribution



Source

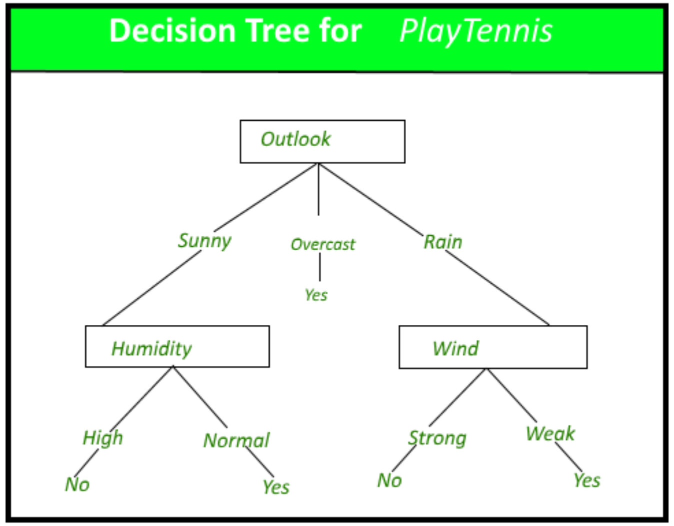


Source

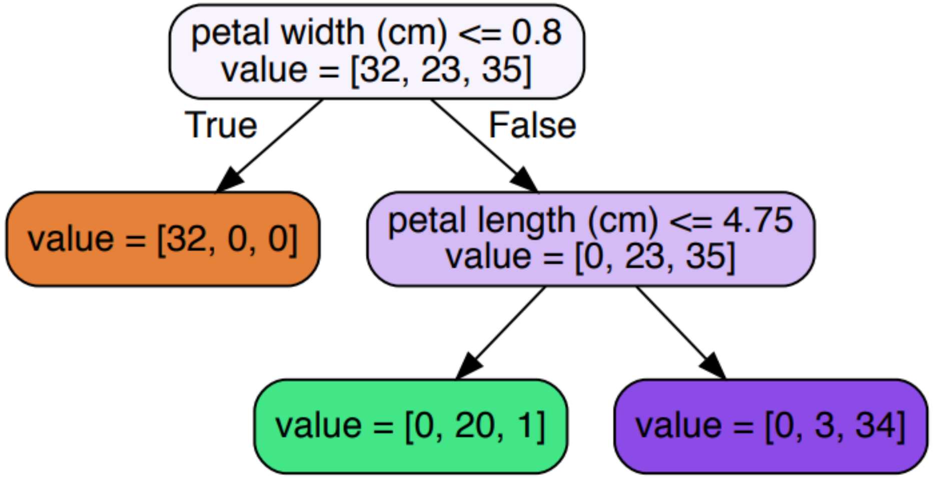
- Decision boundary can be visualized and understood
  - Instances at the boundary characterizes how different classes are different

Explainable Models: Decision Tree

- **Decision trees** are very explainable!
- On every node of the decision tree, we understand a criteria for prediction
- We can perform statistics for each decision node
  - E.g. if the condition of the node is met, **80% of the instances will be classified as being positive**



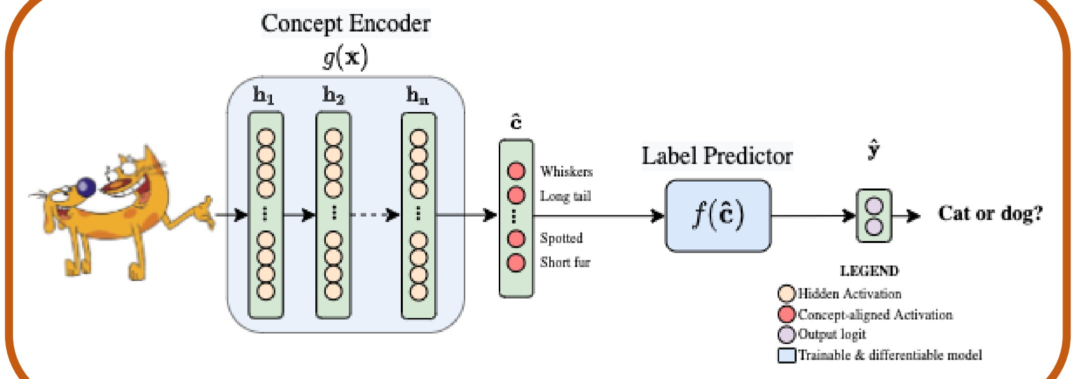
Source

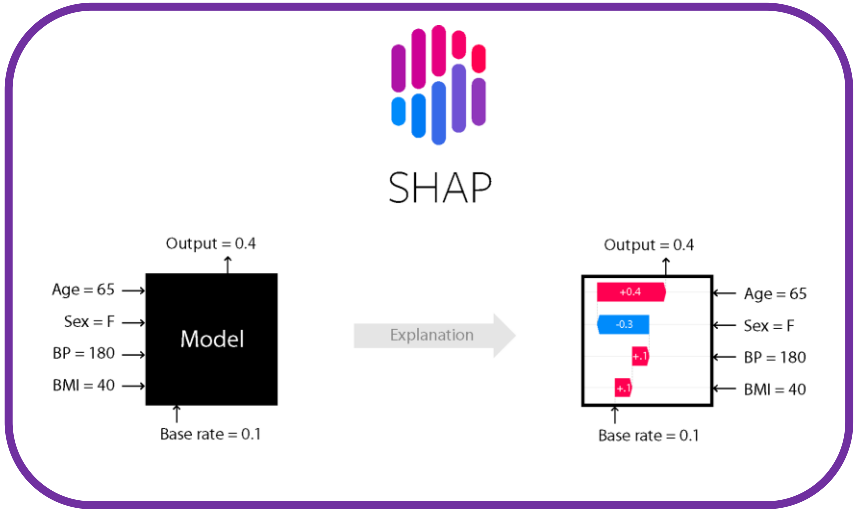


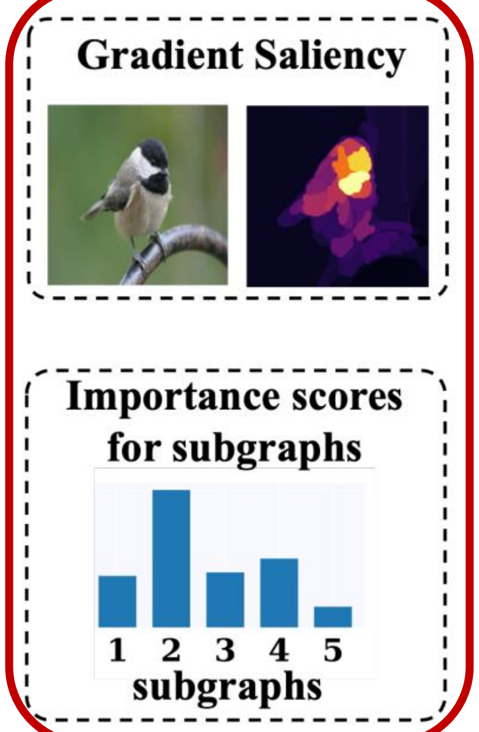
Source

Explainable Characteristics

- What makes model explainable?
  - **Importance** values (for pixels, features, words, nodes in graphs …)
  - **Attributions**: straightforward relationships between prediction and input features
  - Encourage **concepts** and [prototypes]





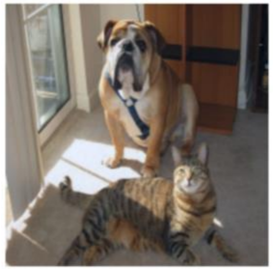


Example: Computer Vision

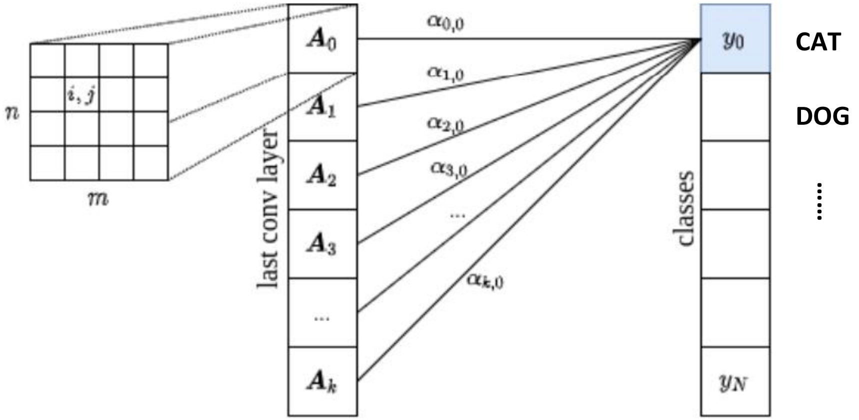
Explanation in Computer Vision:

A particular region of the image **displays the predicted class of objects** (cat / dog in this example)

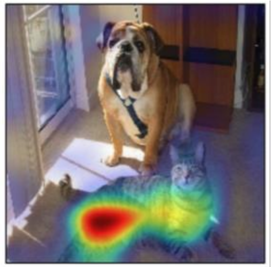
Importance scores on pixels



original graph

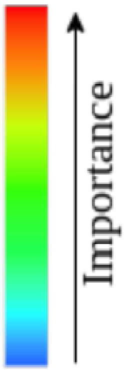


computation process of **CNN** and the prediction



explanation of "cat"

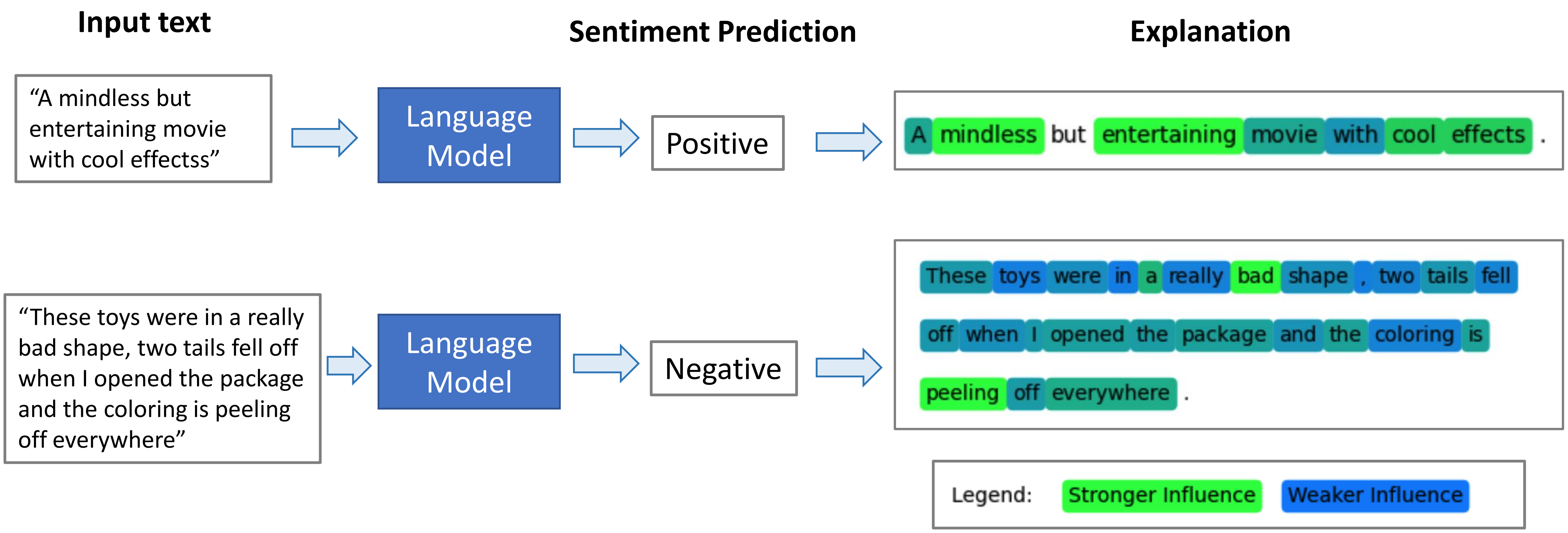
explanation of "dog"



Selvaraju, Ramprasaath R., et al. "Grad-cam: Visual explanations from deep networks via gradient-based localization."

Example: Natural Language Processing

Explanation in Natural Language Processing: important tokens that lead to the prediction



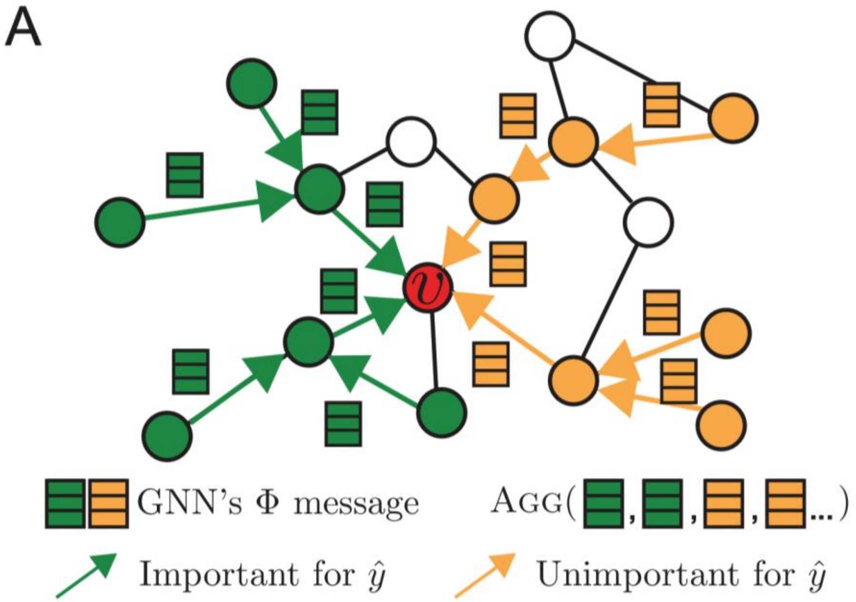
Dunn, Andrew, Diana Inkpen, and Răzvan Andonie. "Context-Sensitive Visualization of Deep Learning Natural Language Processing Models."

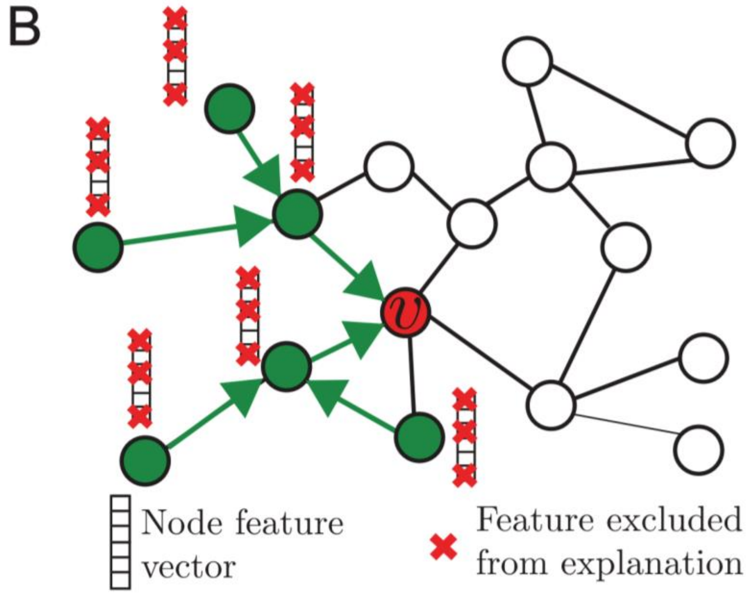
Example: Graph Learning

Explanation in Graph Learning: an important **subgraph structure** and a small **subset of node features** that play a crucial role in GNNs prediction

Explanations for prediction at **node v**

A: Import subgraph structure **B: important subset of features**

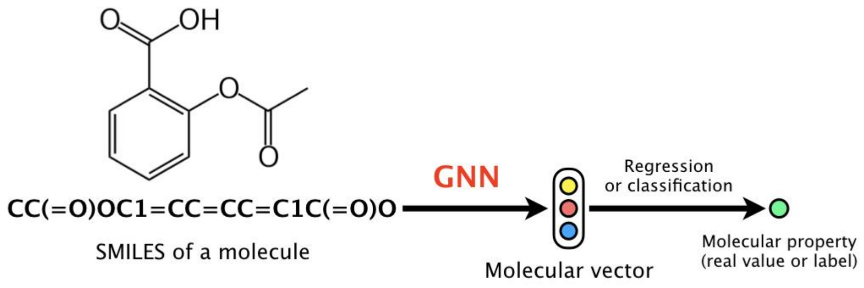




Ying, et al. "Gnnexplainer: Generating explanations for graph neural networks."

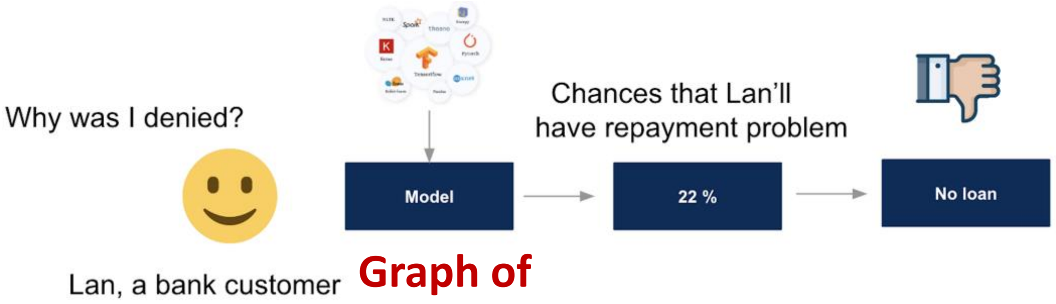
Goal of GNN Explainability

- Model's behavior might be different from the underlying phenomenon
- **Explaining ground truth phenomenon**



What are the characteristics of toxic molecules?

- **Explaining model predictions**

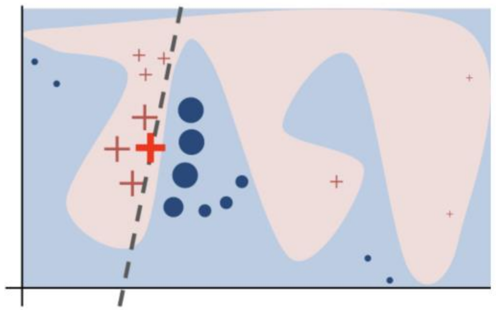


Graph of bank transactions

Why does the model recommend no loan for Person X?

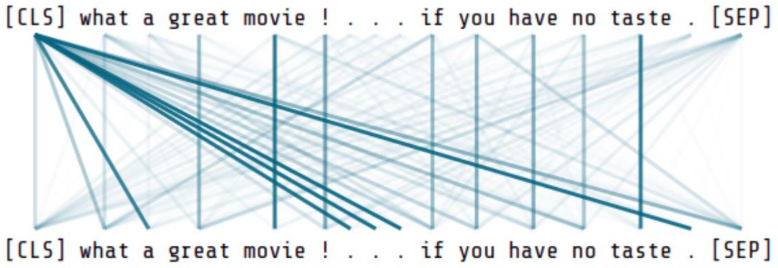
Deep Learning Explainability Methods: Examples

- **Proxy Model**
  - Learn an interpretable model that locally approximates the original model. (Example: SHAP)



- **Saliency Maps**
  - Compute the gradients of outputs with respect to inputs (example: Grad-CAM)

- **Attention Mechanisms**
  - Visualize attention weights in attention models, such as transformer and GAT architectures.



Reasons for Explainability

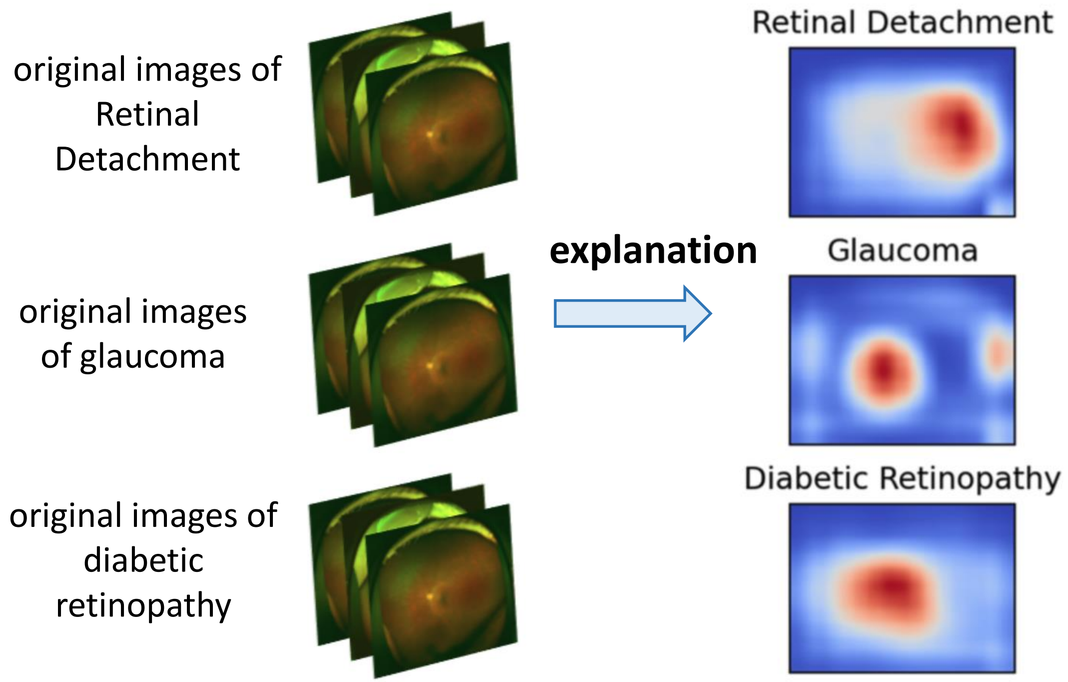
Why do we need Explainability?

- **Trust**: Explainability is a prerequisite for humans to **trust and accept** the model's prediction.
- **Causality**: Explainability can sometimes imply **causality** for the target prediction: attribute X causes the data to be Y
- **Transferability**: The model needs to convey an understanding of decision-making by humans before it can be **safely deployed to unseen data**.
- **Fair and Ethical Decision Making**: Knowing the reasons for a certain decision is a societal need, in order to perceive if the prediction **conforms to ethical standards.**

Explainability Settings (1)

By target:

- **Instance-level:** a **local** explanation for a single input x and the prediction $\hat{y}$
  - identify the important components of individual instances
- **Model-level:** a **global** explanation for a specific dataset D or classes of D
  - provide **high-level insights** into the model's decision-making behaviors



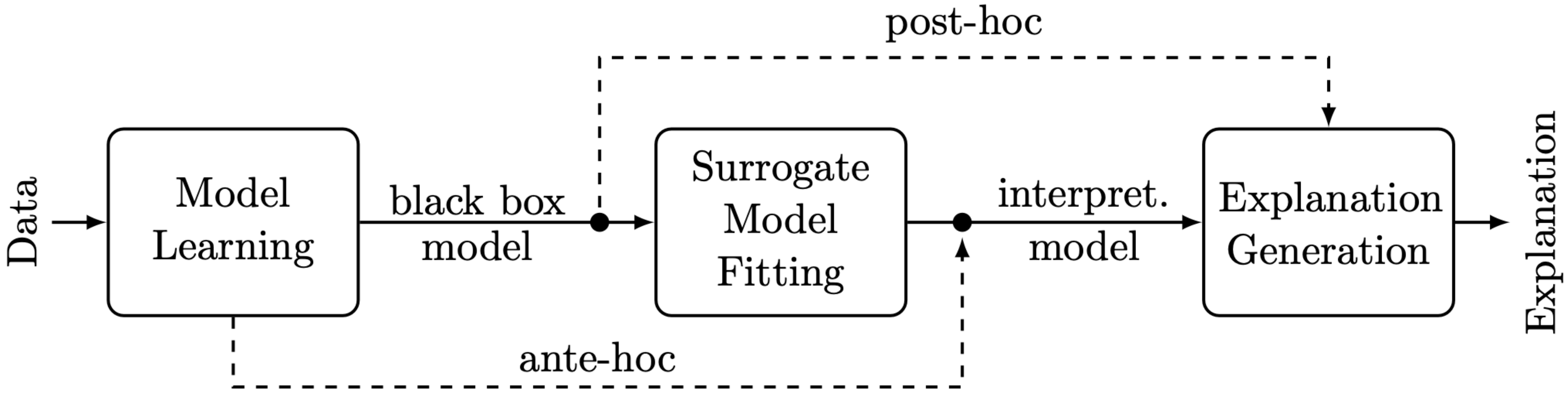
Example: model-level explanations for each class

Engelmann, Justin, Amos Storkey, and Miguel O. Bernabeu. "Global explainability in aligned image modalities."

Explainability Settings (2)

By stages:

post-hoc: Explainability is created after model creation **(for black-box models)**



ante-hoc: Explainability is **built-in** from the beginning of the model creation **(for intrinsically interpretable / white-box models)**

By applicability of the method:

model-specific: the machanism for generating explanation is **model-dependent** and works only for a specific model.

model-agnostic: the machanism for generating explnation is **applicable** for many or even all model classes

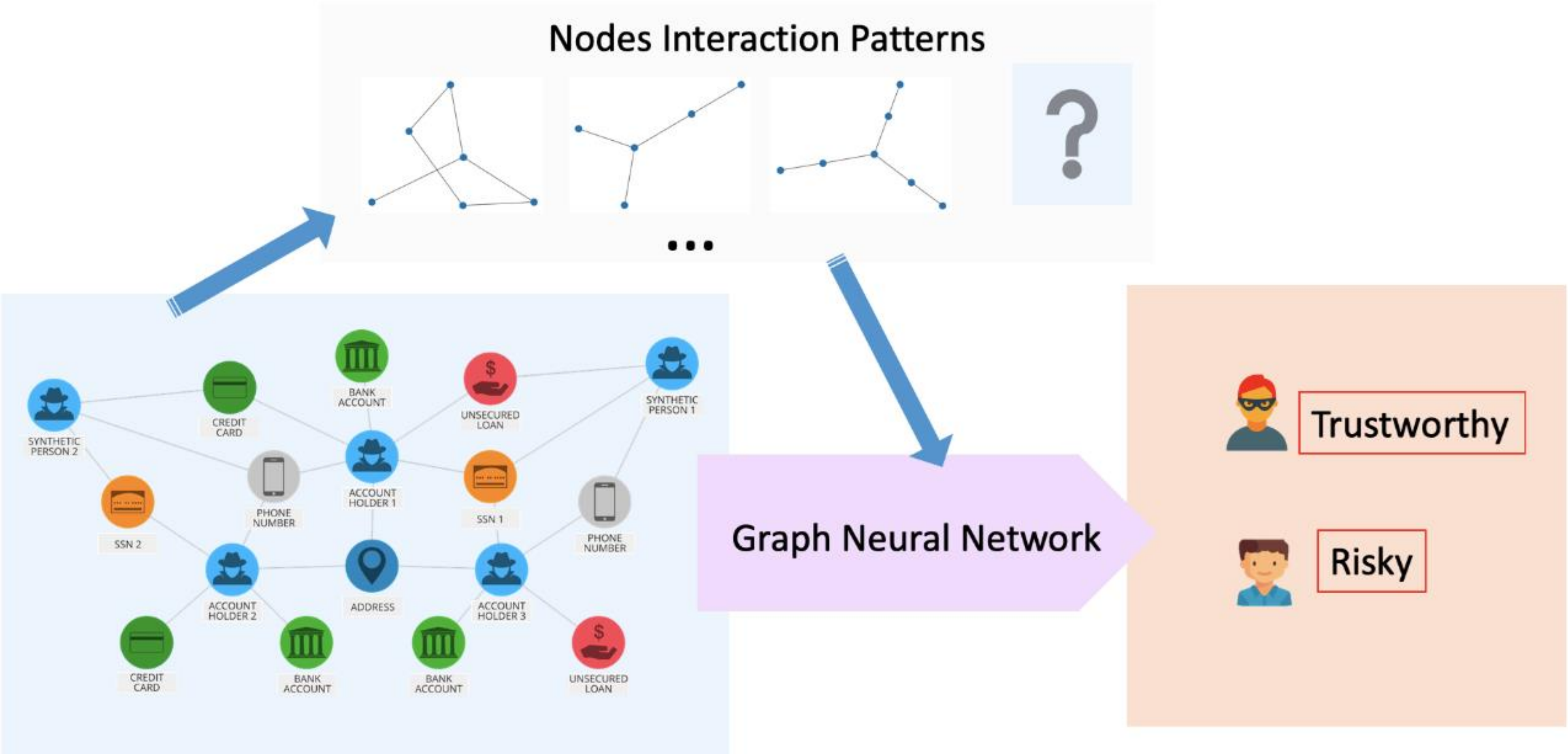
Outline of Today's Lecture

1. Explainability and its Problem Settings

2. **GNNExplainer**
   The first and very commonly used GNN explainability method
   Reference: GNNExplainer (NeurIPS 2019)

3. Explainability Evaluation

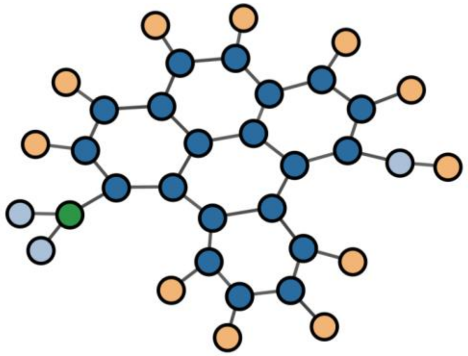
Example: Financial markets as graphs



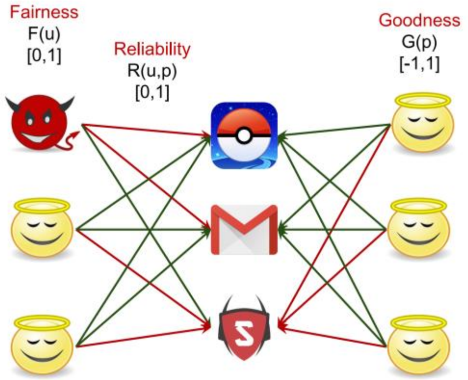
GNN Explainability Use Cases

- **Questions after training GNNs (post-hoc setting):**
  - Why is an item recommended to a user?
  - Why is the molecule mutagenic?
  - Why is the user classified as fraudulent?

Recommender System



Mutagenic Molecule

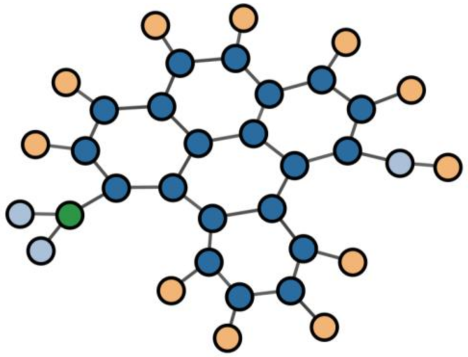


Fraudulent Detection

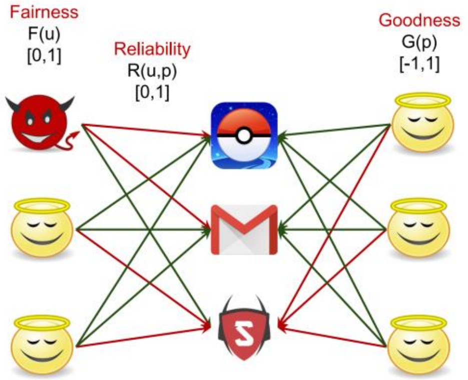
Explainability: Motivation (2)

- **Example questions after training GNNs:**
  - Why is an item recommended to a user? → Explain link prediction
  - Why is the molecule mutagenic? → Explain graph classification
  - Why is the user classified as fraudulent? → Explain node classification

Recommender System



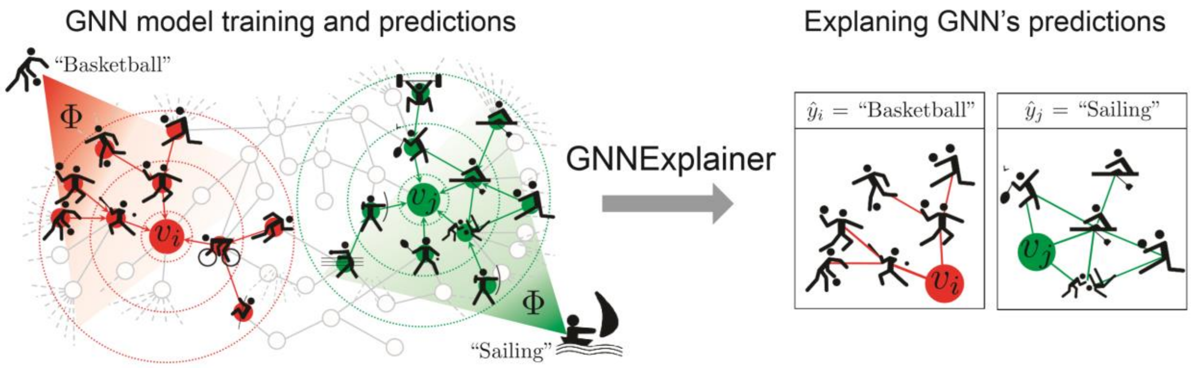
Mutagenic Molecule



Fraudulent Detection

GNNExplainer Pipeline

- **Training time:**
  - Optimize GNN on training graphs
  - Save the trained model
- **Test time:**
  - Explain predictions made by the GNN
  - On unseen instances (nodes, edges, graphs)



Ying, Rex, et al. "[Gnnexplainer: Generating explanations for graph neural networks](#)." *NeurIPS 2019*

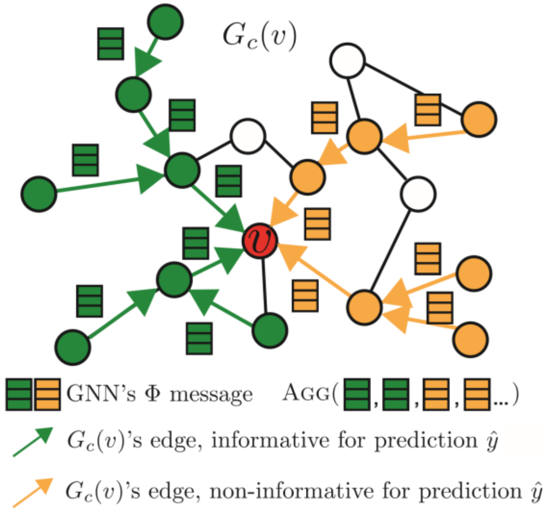
Challenges

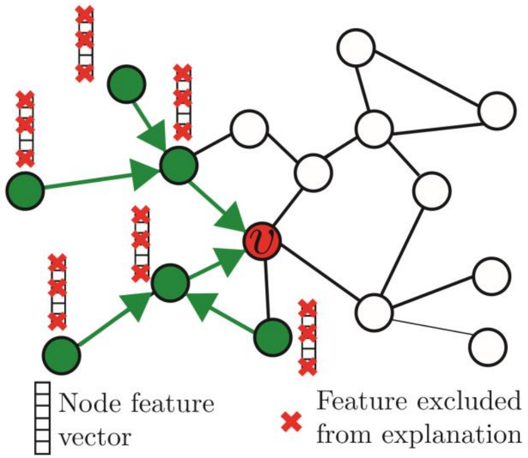
- **Explain predictions for multiple tasks**
  - Node classification
  - Graph classification
  - Link prediction
- **Model agnostic (*post-hoc*)**
  - Need to be applied to a variety of GNN models: GCN, GraphSAGE, GAT etc.
- Predictions on graphs are induced by a **complex combination** of **nodes**, **edges** between them, and even **motifs** / **subgraph** structures.
- Unlike in CV, gradient is a less reliable signal on real-world graphs due to the **discrete** nature of edges
  - In many cases (counterfactual explanation, model-level explanations), gradients cannot be used at all

Ying, Rex, et al. "Gnnexplainer: Generating explanations for graph neural networks." *NeurIPS 2019*

How to explain a GNN

- Consider the general message-passing framework
- The importance of node features





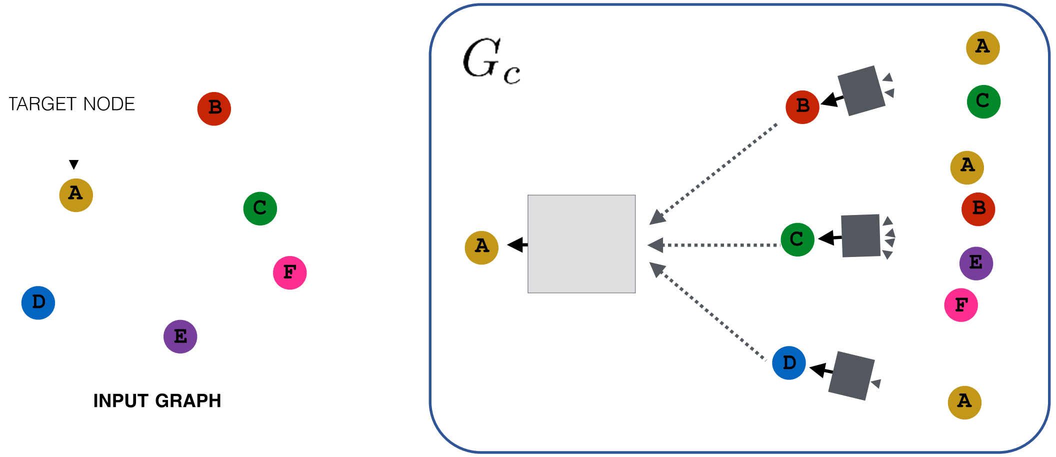
Structural explanation **Feature explanation**

- GNNExplainer explain both aspects simultaneously

Ying, Rex, et al. "Gnnexplainer: Generating explanations for graph neural networks." *NeurIPS 2019*

GNNExplainer Input

- Without loss of generality, consider node classification task:



Suppose GNN predicts label $\hat{y}$ for node v

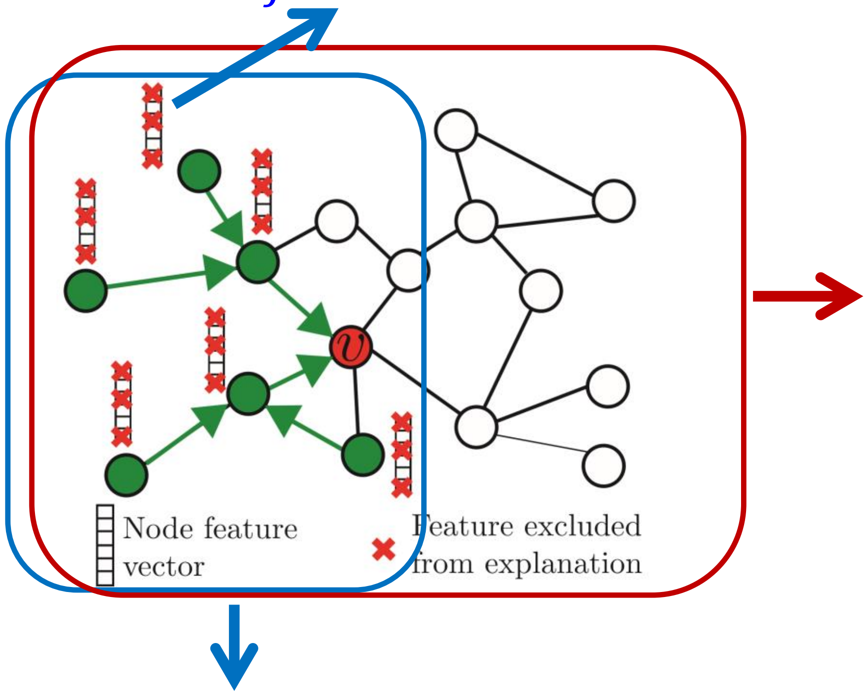
- Input computation graph: $G_c(v)$
- Adjacency matrix of G_c : $A_c(v) \in \{0,1\}^{n \times n}$
- Node Feature: $X_c(v) = \{x_j | v_j \in G_c(v)\}$

Ying, Rex, et al. "Gnnexplainer: Generating explanations for graph neural networks." *NeurIPS 2019*

GNNExplainer Output

- GNN model ϕ learns $P_\phi(Y \mid A_c(v), X_c(v))$
- Y denotes predicted label of v
- **GNNExplainer** outputs (A_S, X_S^F)
- Graph G_S with adjacency matrix A_S is a subgraph of graph with adjacency matrix $A_c(v)$ (omit v)
- $X_S^F = \{x_j^F \mid v_j \in G_S\}$ are features for G_S
- Mask F masks out unimportant dimensions

x_j^F (unimportant dimensions masked)



Computation Graph $G_c(v)$

Explanation $G_S(v)$

Ying, Rex, et al. "Gnnexplainer: Generating explanations for graph neural networks." *NeurIPS 2019*

Explain by Mutual Information

- **Mutual information (MI)**
  - A measure of the mutual correlation between the two random variables.
  - Good explanation should have **high correlation** with model prediction
  - Relation to entropy:

$$MI(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$$

- GNNExplainer **Objective**:
  - Maximize MI between **label** and **explanation**

$$\max_{G_S} MI\big(Y; (A_S, X_S)\big) = H(Y) - H(Y|A = A_S, X = X_S^F)$$

Ying, Rex, et al. "Gnnexplainer: Generating explanations for graph neural networks." *NeurIPS 2019*

Explain by Optimization

- By relation to entropy, the objective is equivalent to minimization of conditional entropy:

$$\max_{A_S} MI\big(Y|(A_S, X_S)\big) = \min_{A_S} H(Y|\underbrace{\boxed{A = A_S,}}_{\text{Subgraph}}\underbrace{\boxed{X = X_S^F}}_{\text{Feature subset}})$$

- Finding A_S that minimizes the conditional entropy is computationally expensive!
  - Issue: Exponentially many possible A_S
- Solution: Treat explanation as a distribution of "**plausible explanations**", instead of a single graph
  - Optimize the expected explanation
  - Benefit 1: captures multiple possible explanations for the same node
  - Benefit 2: turns discrete optimization to continuous

GNNExplainer Model

- Continuous relaxation
  - Optimize the expected adjacency matrix A_S

$$\min_{\mathcal{A}} \boxed{\mathbb{E}_{A_S \sim \mathcal{A}}} H(Y|A = A_S, X = X_S)$$

expectation of explanations

  - View $\mathbb{E}_{A_S \sim \mathcal{A}}$ as an adjacency matrix where entries are continuous
- Approximation

continuous

$$\min_{\mathcal{A}} H(Y|A = \boxed{\mathbb{E}_{\mathcal{A}}[A_S]}, X = X_S)$$

- Optimize the expectation by masking

Element-wise multiply

- Use $A_C \odot \mathrm{Mask}$ to represent $\mathbb{E}_{\mathcal{A}}[A_S]$
- If $\mathrm{Mask}_{\mathrm{ij}}$ close to 1, keep edge (i, j); if close to 0, drop edge (i, j).

GNNExplainer Model

- Let $\text{Mask} = \sigma(\boldsymbol{M})$ be the adjacency mask
  - Continuous relaxation: $\sigma(M) \in \mathbb{R}$ instead of binary
  - **Sigmoid** function σ squashes $\boldsymbol{M}$ into $[0, 1]$
  - Masking: Element-wise multiply $\sigma(\boldsymbol{M})$ by A_c

- Objective:

$$\min_{M} -H(P_{\phi}(Y = y | G = A_C \odot \sigma(M), X = X_S))$$

GNNExplainer Model

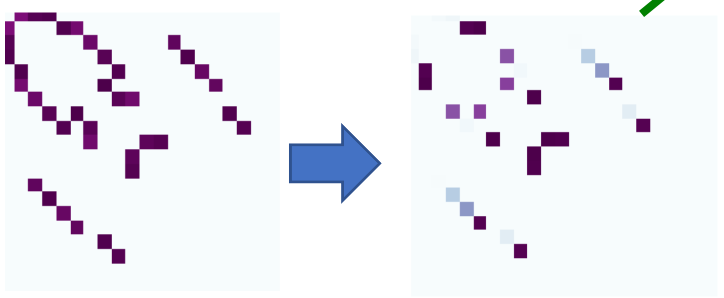
- Optimize M :

$$\min_{M} -H(\boxed{P_{\phi}(Y = y | A = A_c \odot \sigma(M), X = X_S)})$$

Prediction probability distribution by the GNN with parameters ϕ

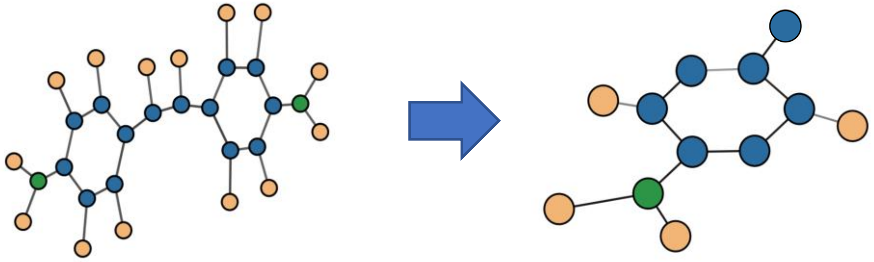
- $A_c \odot \sigma(M)$ is the **relaxed adjacency matrix**
  - Entries are real-values in $[0, 1]$, instead of being binary
- Threshold $A_c \odot \sigma(M)$ to get G_S. Example:

$A_c \odot \sigma(M)$ **Relaxed adjacency matrix**



Before training After training M

Graph G_S after thresholding



Before training After training M

Feature Explanation

- Similarly select features by optimizing for feature mask F
  $X_S^F = \{x_j^F | v_j \in G_S\}, \quad x_j^F = [x_{j,t_1}, \ldots, x_{j,t_k}]$

  For the selected dimensions, $\sigma(F_{t_i}) \to 1$

- Problem: Zero value could be important!
- Solution: Measure feature importance by how much drop in model confidence when features are replaced with explainability **baselines**.
- Concept: explainability baseline is the "null model" of a feature, such as the mean of the marginal distribution of each feature.

Regularization Constraints

- **Optimize feature and adjacency masks jointly with regularization**
- **Concise explanation**
  - Mask size: $\mathrm{Sum}\big(\sigma(M)\big)$
  - Feature size: $\mathrm{Sum}(\sigma(F))$
- **Final Objective**

M, F are learnable Parameters when explaining $G_c(v)$

$$\min_M -H\big(P_\phi(Y = y|G = A_c \odot \sigma(M), X = X_S^F\big) + \underbrace{\lambda_1 \mathrm{Sum}\big(\sigma(M)\big) + \lambda_2 \mathrm{Sum}(\sigma(F))}_{\text{Sum of entries in feature and adjacency masks}}$$

- Threshold $A_c \odot \sigma(M)$ to get the explanation G_S
- The optimization is performed when explaining every instance

GNNExplainer Model

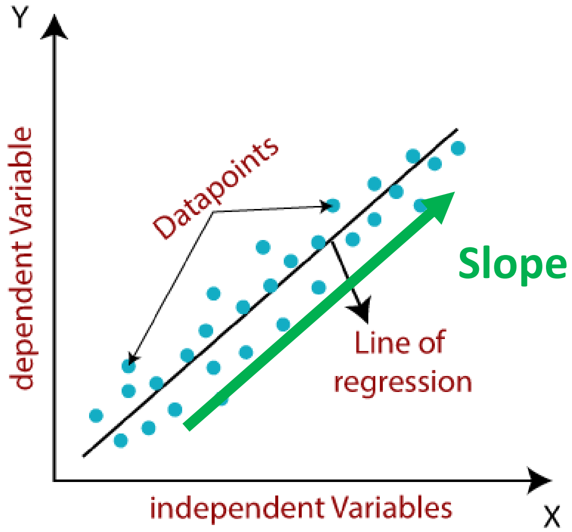
- **Explain different tasks**
  - **Node classification**: optimize mask (M, F) on the **node's neighborhood** (computation graph)
  - **Link prediction**: optimize mask (M, F) on **union of 2 node neighborhoods**
  - **Graph classification**: optimize mask (M, F) on the **entire graph**

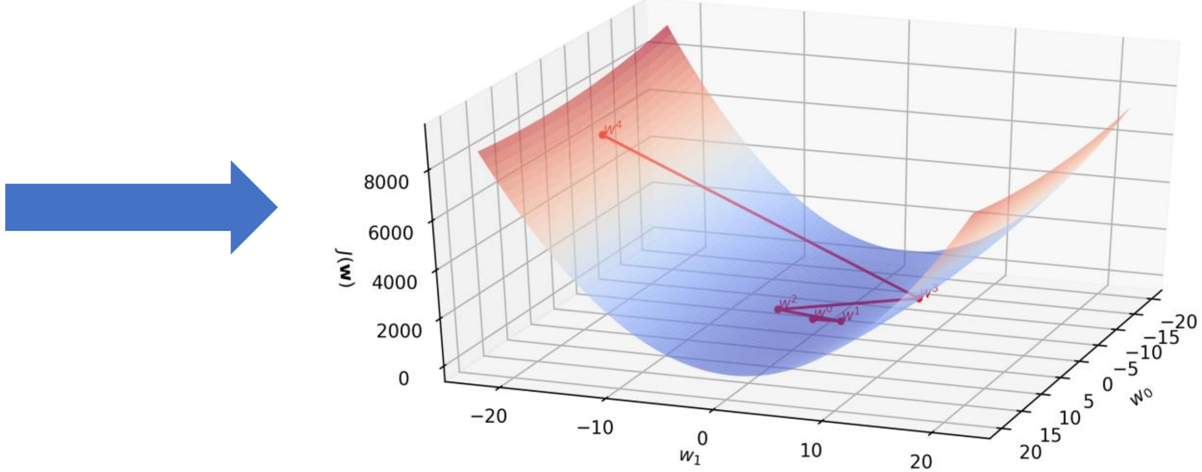
- **Can adapt to different architectures**
  - Graph Attention Networks
  - Gated Graph Sequence
  - Graph Networks
  - GraphSAGE
  - …

We replace P_ϕ with the respective architecture

Experiments: Alternative Approaches (1)

- **GNN saliency map based on gradients** of output score with respect to inputs

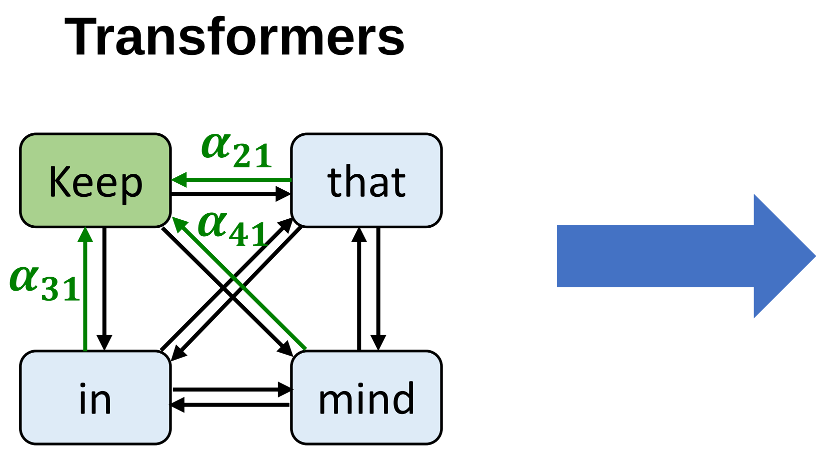


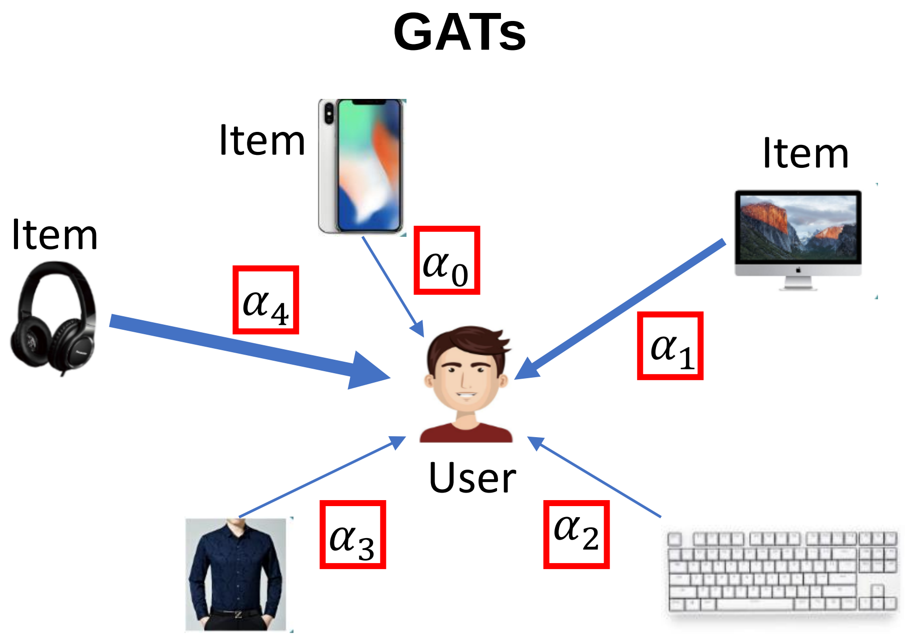


- Gradient is a **local approximation** of the slope
- We compute gradient of objective with respect to the **edges and features**

Experiments: Alternative Approaches (2)

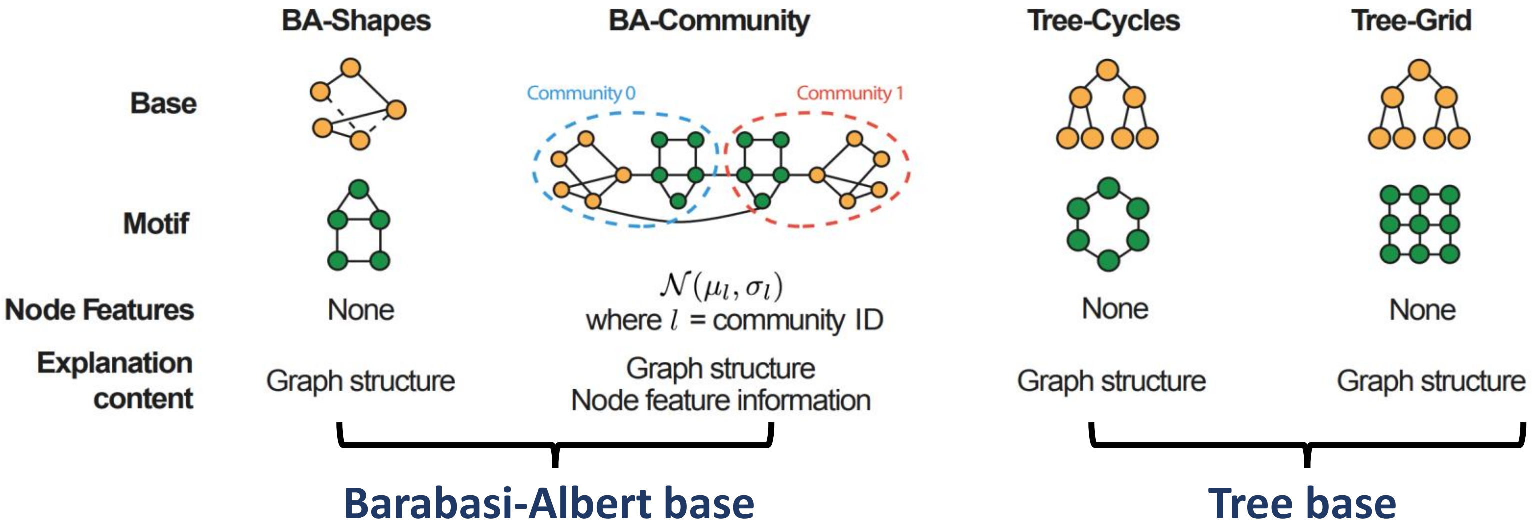
- **Attention** values based on Graph Attention Networks (GAT)
  - Edge importance indicated by average attention weights across layers for each edge
  - Attention-based importance is available for edges





Experiments: Datasets (1)

- Synthetic task: is a node part of a given motif?
  - 100 Motifs are randomly attached to nodes in base graphs (500 nodes)
  - Node classification (structural roles)



Experiments: Datasets (2)

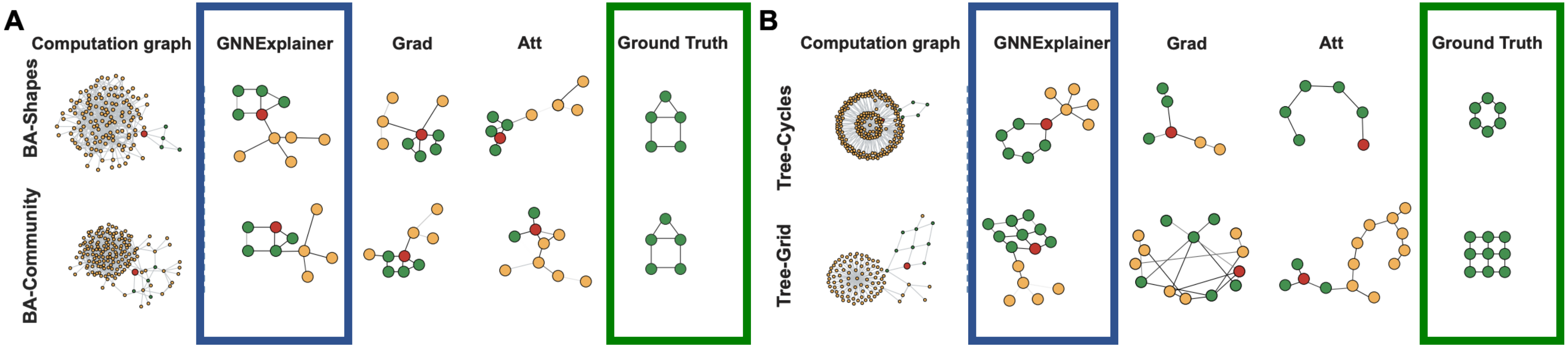
- Real-world tasks
  - **Social networks** (Reddit-binary dataset)
    - Reddit community prediction
  - **Chemistry** (Mutagenic molecule dataset)
    - Chemical property prediction
  - Graph classification

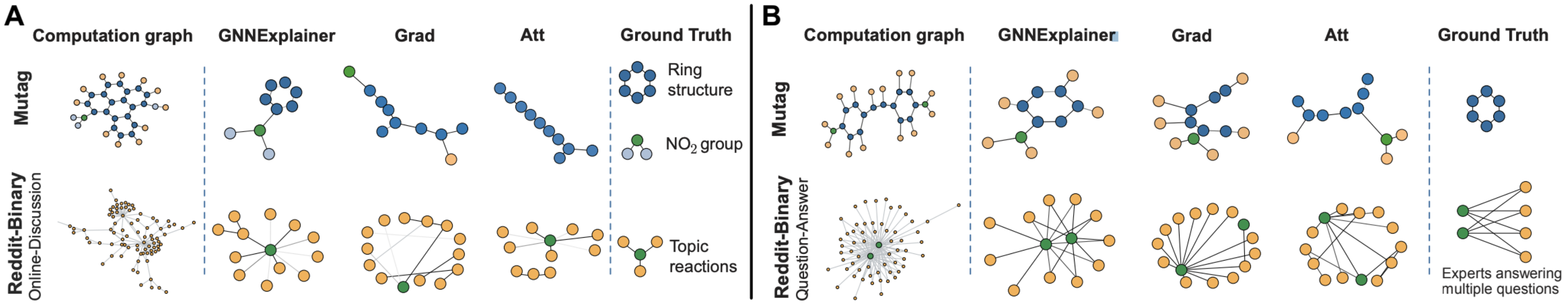
Results: Quantiative Analysis

- Node classification with ground-truth
- Measures accuracy of explanation with respect to ground-truth

	BA-House	BA-Comm	Tree-Cycle	Tree-Grid
Grad	88.2	73.9	82.4	61.2
Att	81.5	75.0	90.5	66.7
GNN-Explainer	**92.5**	**83.6**	**94.8**	**87.5**

Results: Qualitative Analysis





Outline of Today's Lecture

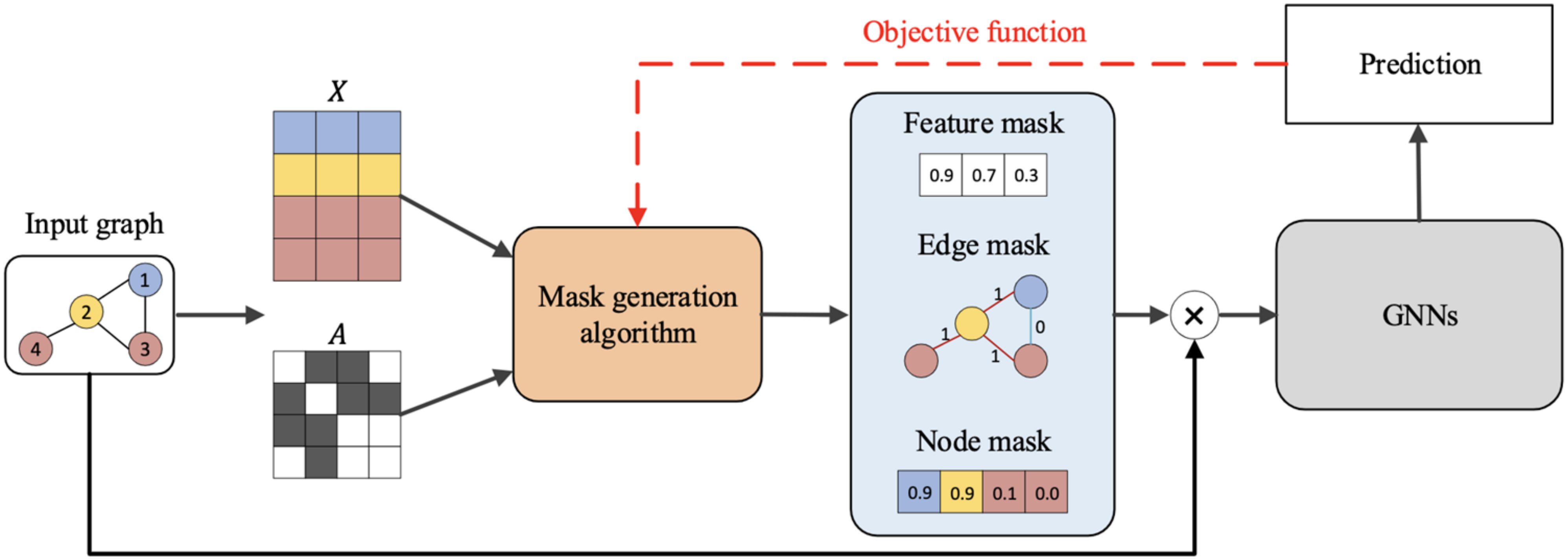
1. Explainability and its Problem Settings

2. GNNExplainer

3. Explainability Evaluation
   GNN Explainability Taxonomy and Evaluation
   Reference: GraphFramEx (LoG 2022)

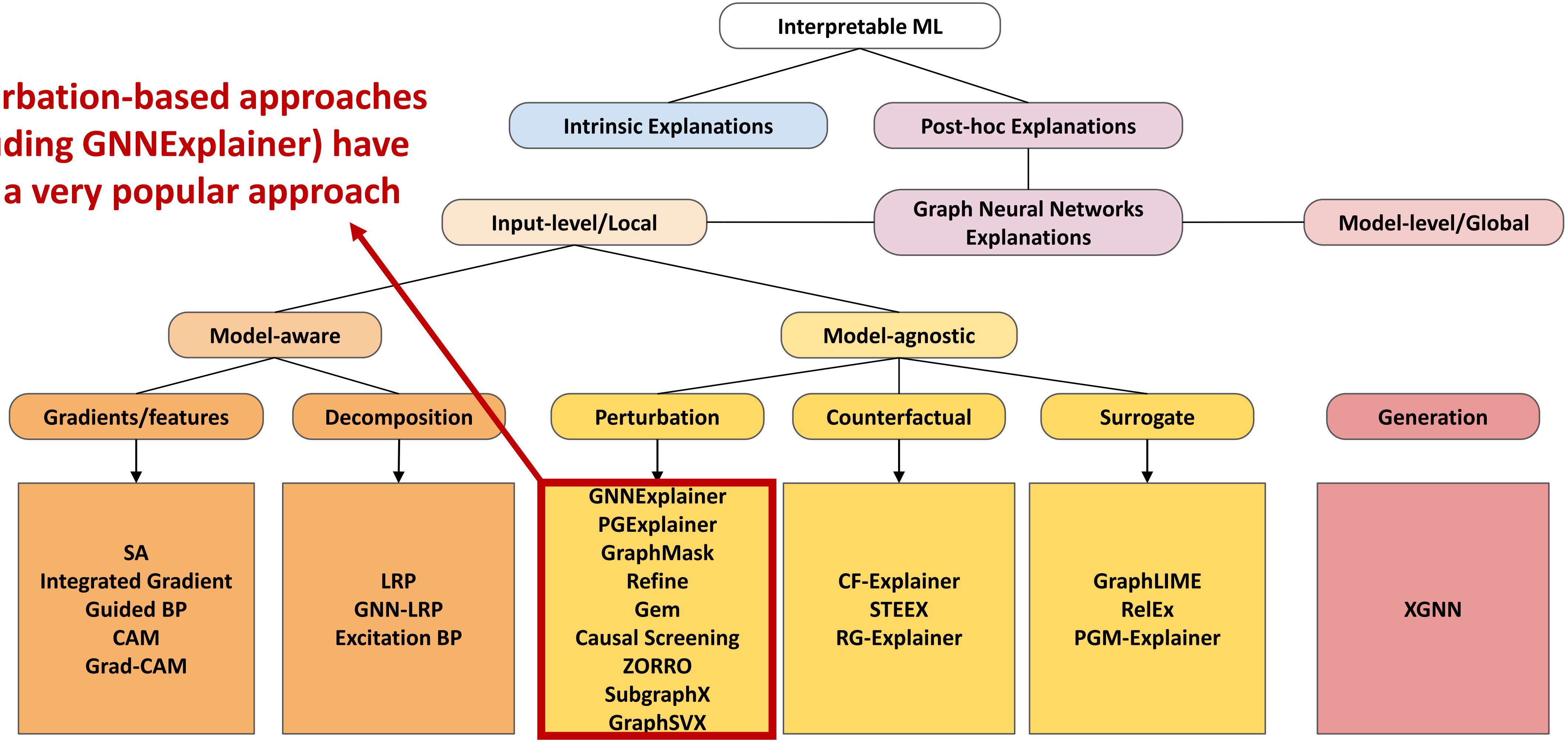
GNN Post-hoc Explanation Pipeline

- **Goal recap**: identify important subgraph structures and node features (masks)



Taxonomy of GNN Explainability Methods

Perturbation-based approaches (including GNNExplainer) have been a very popular approach



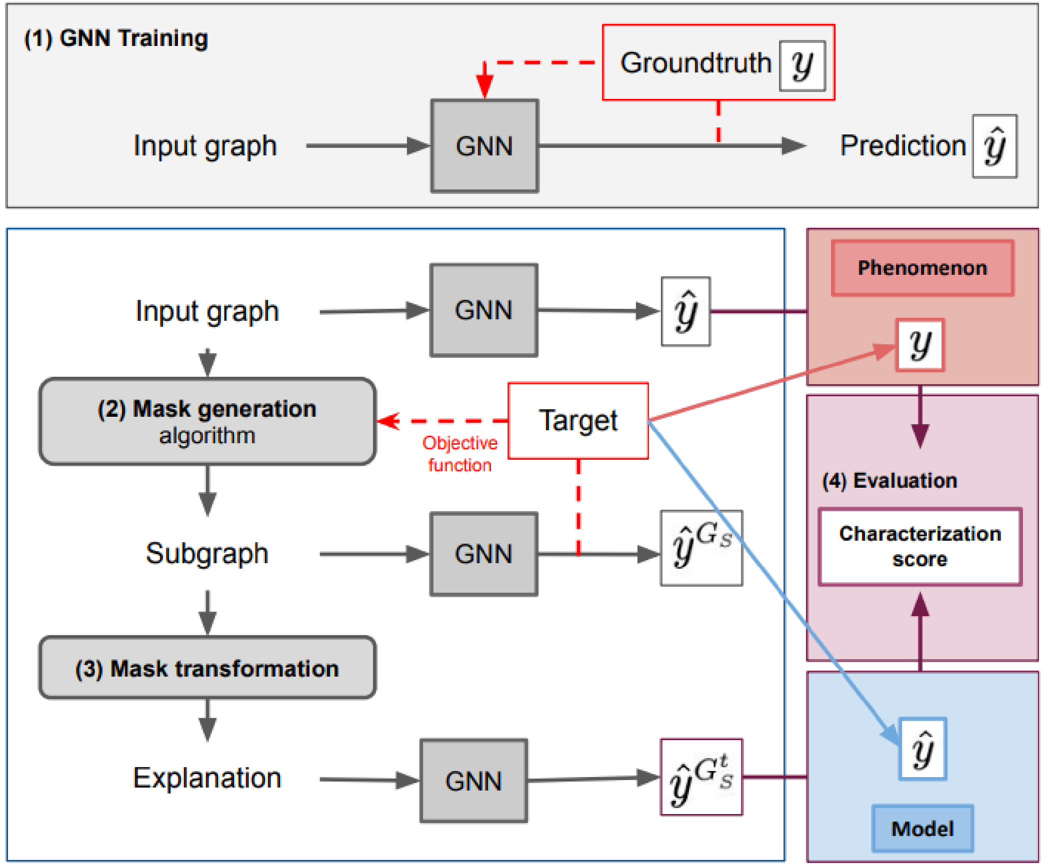
Amara, Kenza, et al. "GraphFramEx: Towards Systematic Evaluation of Explainability Methods for Graph Neural Networks.", LoG 2022

Explainability Method Evaluation

- **Challenge: groundtruth might not always be available**
- Evaluation is **multi-dimensional**

- **Goal** (phenomenon *vs.* model)
- **Masking** strategy
- **Type** (sufficiency vs. necessity)

- **GraphFramEx**
  Benchmarks and evaluation criteria for graph explainability



Explanation Goal

- **Phenomenon** Explanation
  - Explain the underlying reasons for the ground truth phenomenon

- **Model** Explanation
  - Explain why model makes a particular prediction

- We will explain the **fidelity** metric in both cases:

Explanation Goal: Fidelity Metric

- Define 2 fidelity metrics: fid_+ and fid_- to capture different aspects of **explanation quality**
- The formula of fidelity depends on the goal:
  - **Goal 1**: explain **phenomenon** of the data
  - **Goal 2**: explain what has the **model** learned

Goal

Phenomenon

prediction

$$fid_+ = \frac{1}{N}\sum_{i=1}^{N}\left|\mathbb{1}(\hat{y}_i = y_i) - \mathbb{1}(\boxed{\hat{y}_i^{G_{C\setminus S}}} = \boxed{y_i})\right|$$

groundtruth

$$fid_- = \frac{1}{N}\sum_{i=1}^{N}\left|\mathbb{1}(\hat{y}_i = y_i) - \mathbb{1}(\hat{y}_i^{G_S} = y_i)\right|$$

Model

$$fid_+ = 1 - \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}(\hat{y}_i^{G_{C\setminus S}} = \hat{y}_i)$$

$$fid_- = 1 - \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}(\hat{y}_i^{G_S} = \hat{y}_i)$$

Fidelity Metric Details

- **Characteristics of a good explanation**
- fid_+: removal important subgraph will result in dramatic decrease of the confidence
- fid_-: Using only the important subgraph will result in similar confidence

Phenomenon

$$fid_+ = \frac{1}{N}\sum_{i=1}^{N}\left|\mathbb{1}(\hat{y}_i = y_i) - \boxed{\mathbb{1}(\hat{y}_i^{G_{C\setminus S}} = y_i)}\right|$$

Removal of important subgraph

$$fid_- = \frac{1}{N}\sum_{i=1}^{N}\left|\mathbb{1}(\hat{y}_i = y_i) - \boxed{\mathbb{1}(\hat{y}_i^{G_S} = y_i)}\right|$$

Keeping only the important subgraph

Original prediction probability / confidence

Explanation Evaluation Criteria

- Notably, the explanation evaluation criteria are **multi-dimensional**
- **Explanation quality**
  - High fidelity / characterization scores
  - Sufficiency and necessity aspects (see the previous slide)
- **Explanation stability**
  - Explanations are consistent across random optimization seeds (measure variance)
- **Explanation complexity**
  - The explanation should be concise and easy to understand by human (measure size)

Types of Explanations

- **Sufficiency**
  - An explanation is sufficient if it leads by its own to the initial prediction of the model explanation. $(fid_{-} \to 0)$
- **Necessity**
  - An explanation is necessary if the model prediction changes when removing it from the initial graph. $(fid_{+} \to 1)$
- Use the **Characterization** score to summarize the explanation quality

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

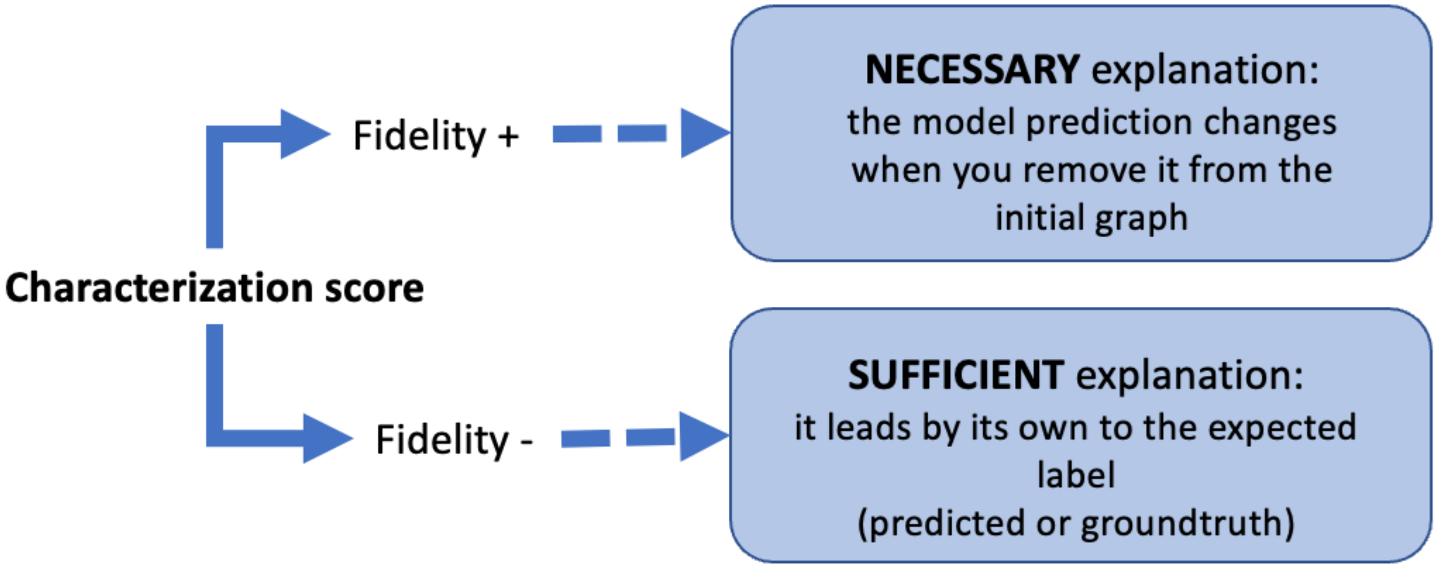
Where w_{+} and w_{-} are the weights of both fidelity metrics (commonly set $w_{+} = w_{-} = 1$)

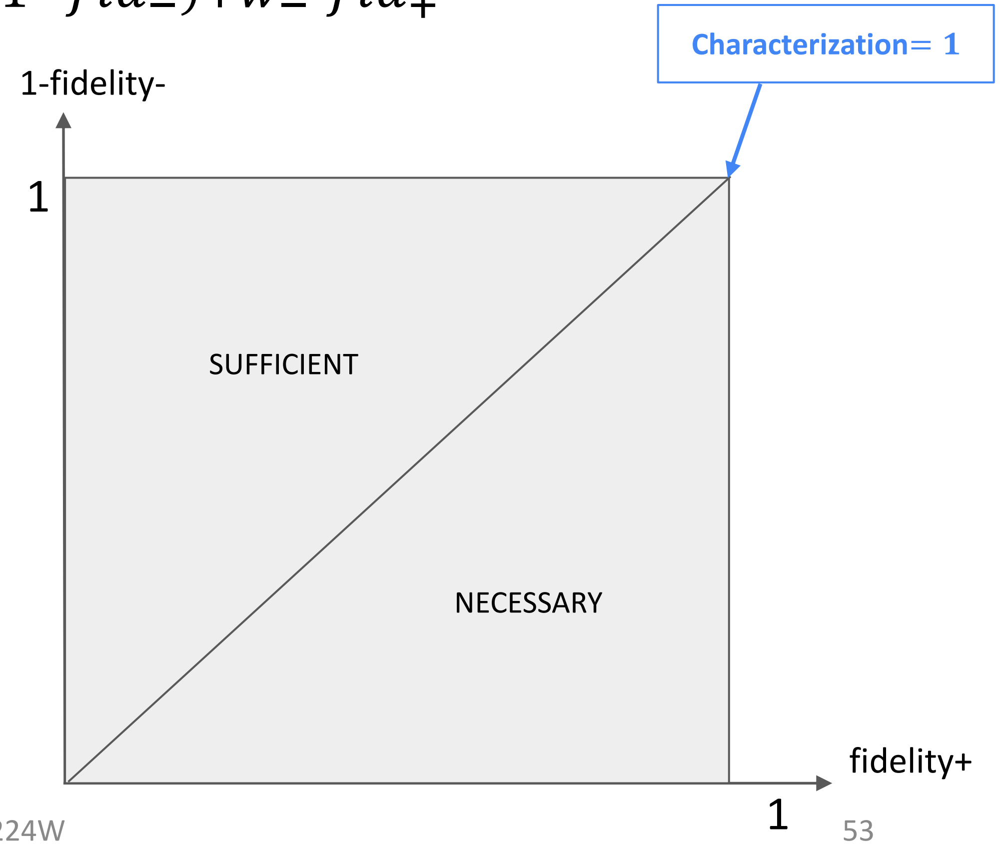
Characterization Score

- **Characterization** score to summarize the explanation quality

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

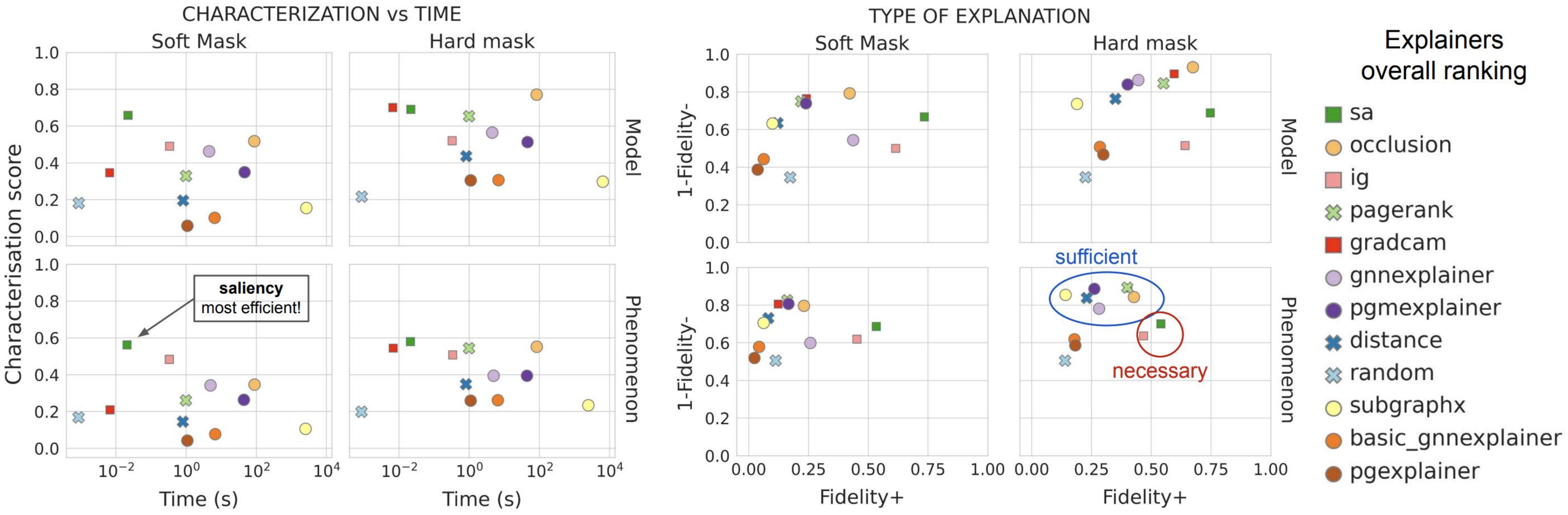
- Necessary AND sufficient





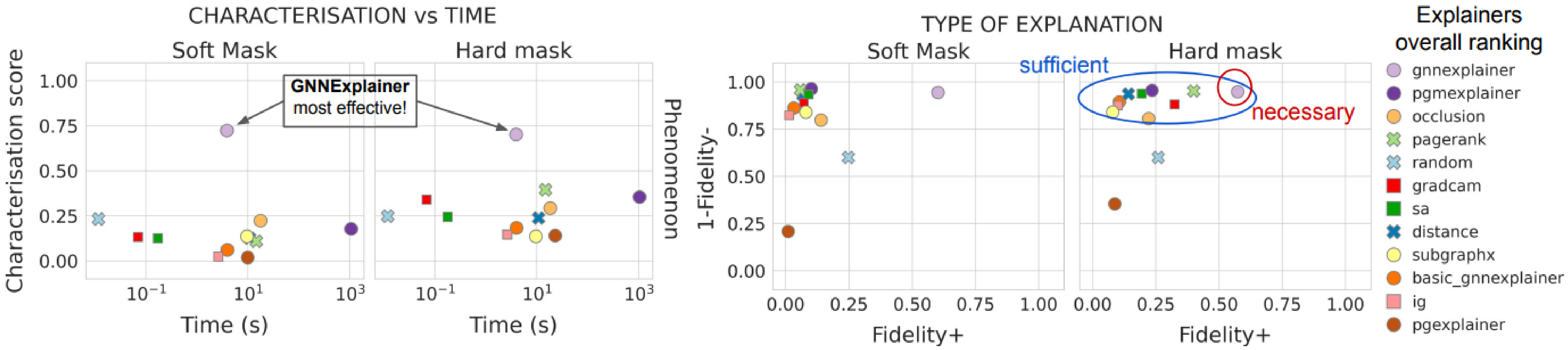
Results: Explain Efficiency vs. Characterization Score

- Multi-dimensional performance comparison of explainability methods
- Explanations have $k = 10$ edges



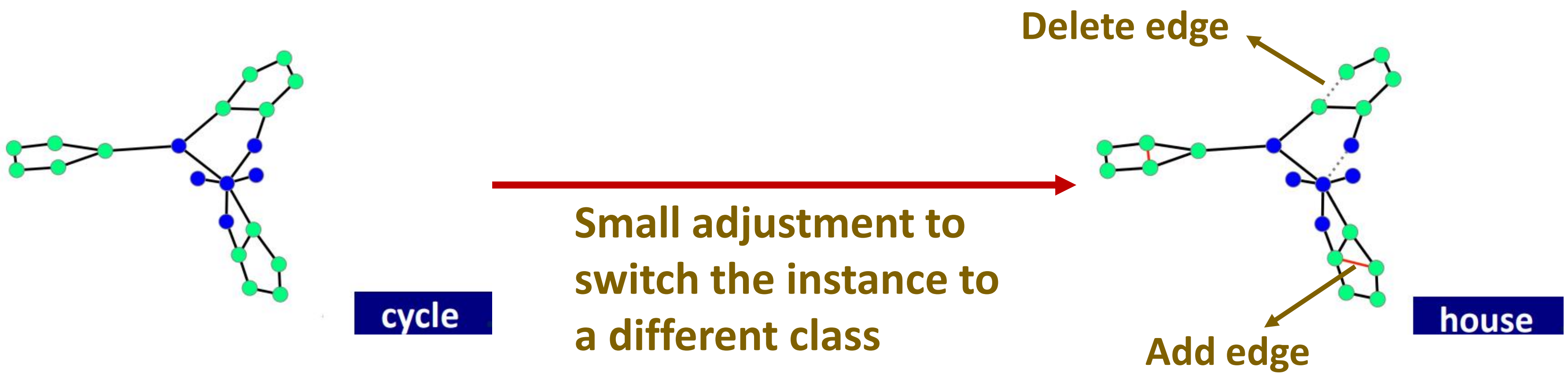
Explainability on Large-scale Real-world Graphs

- The conclusion can be very different depending on datasets and tasks
- Experiment on the **e-commerce graph** at eBay
- GNNExplainer achieves the highest metric in both **necessity and sufficiency** aspects



Other Types of Explanations (1)

- **Counterfactual explanations**: what makes an instance belonging to a **different** class (than the predicted / ground-truth class)?

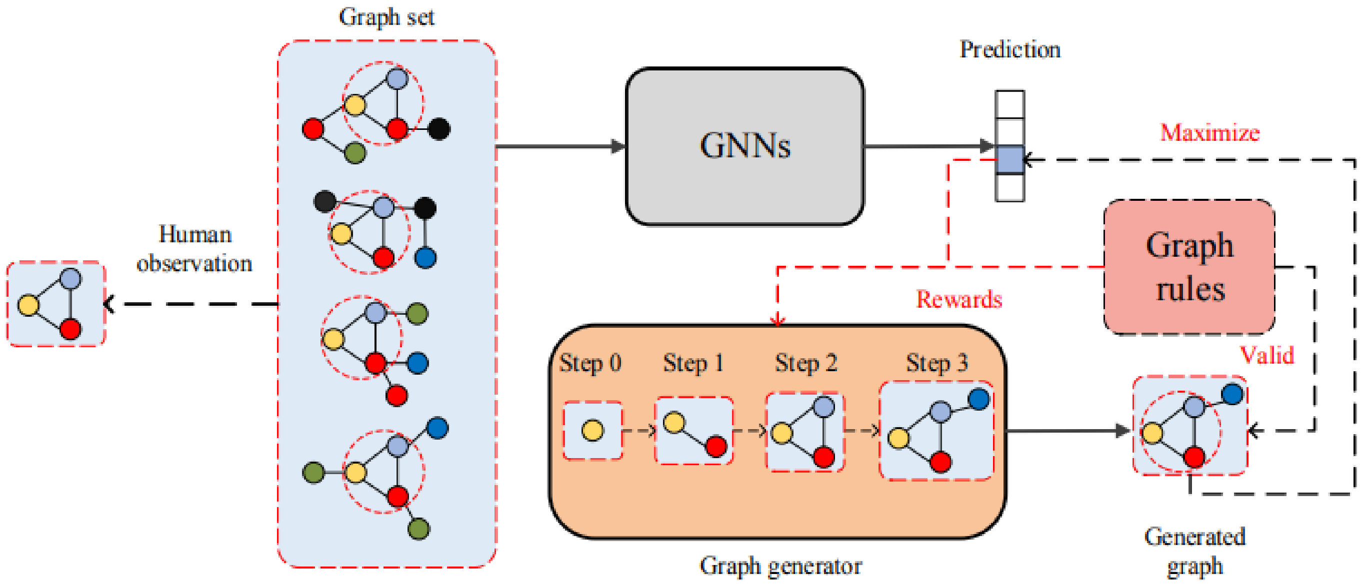


Example method: [CF-GNNExplainer]

- Useful in understanding distinctions between classes
- For example, real-world applications often wants to know what does it take to convert a user from "**inactive / churn**" class to "**active / premium**" class

Other Types of Explanations (2)

- **Model-level explanations**: what are the general characteristics of **ALL** instances belonging to a certain class?



Example method: [XGNN]

- Useful in extracting general insights for all instances of a class

Summary of the Lecture

- **Trustworthy GNN**
  - Robustness, explainability, privacy, fairness, accountability, efficiency and environmental well-being,…
- **GNNExplainer**
  - Perturbation-based approach
  - Optimize for masks that indicate important substructure and node features
- **Explainability evaluation of GNN**
  - Explainability evaluation is **multi-dimensional** in nature
  - Fidelity and characterization scores
  - Other types of explanations: counterfactual, model-level explanations