1 Interpretability

Motivation: A single metric is an incomplete description of most real-world tasks.

1. Improve models
2. Justify models
   a. Creators
   b. Operators
   c. Executors
   d. Decision
   e. Auditors
   f. Data Subjects

   Stakeholders of a AI System
3. Discover Insights

1.1 What is Interpretability?

Definition: Interpretability is:

1. Degree to which a human can understand the cause of a decision
2. Where a user can correctly and efficiently predict the method's results.
3. Science of understanding AI models from the inside out.

1.1.1 Mechanistic Interpretability

Definition: Reverse engineering the algorithm of a NN.

1.2 Types of Interpretability

Summary:

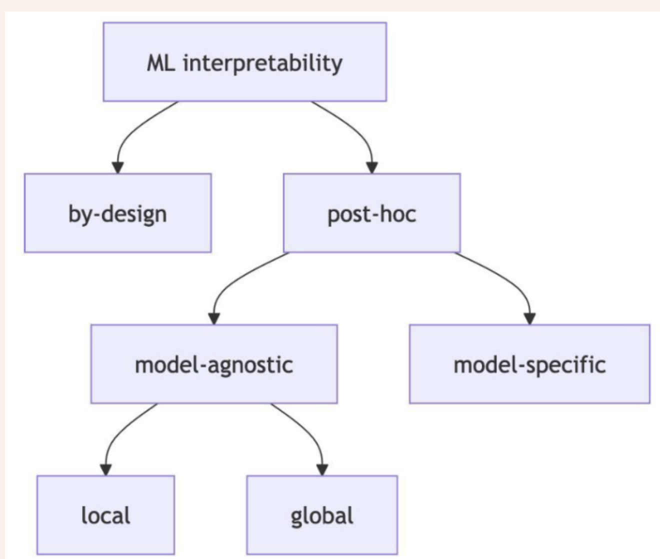

Figure 1

- **ML Interpretability:**
  - **By-design:** Interpretability built directly into the model (e.g., decision trees, linear models).
  - **Post-hoc:** Interpretability techniques are applied after the model is trained, without altering the model.
    * **Model-agnostic:** Interpretation methods that can be applied to any model.
      · **Global:** Provides an overall understanding of the model's behavior across the entire dataset.
      · **Local:** Explains the model's prediction for a specific input instance.

* **Model-specific:** Interpretation methods that are tailored to specific models.

1.3 Attribution

Motivation: One tool in the interpretability toolkit

Definition: Attribution techniques assign ranked importance values to parts of the input that relate to the output.

1.3.1 Issues

Summary:

Issue	Description
Spurious Correlations	Correlations learned by a model that appear predictive in the training data, but do not reflect true causal relationships in the real world.
Dataset Biases	Systematic distortions in the training data that misrepresent the underlying population or task, leading to unfair or inaccurate model predictions.
Imperfect Model	Models do not have perfect accuracy so attributions are likely to not be perfectly accurate

1.4 Examples

1.4.1 Interpretability in LLMs

Example:

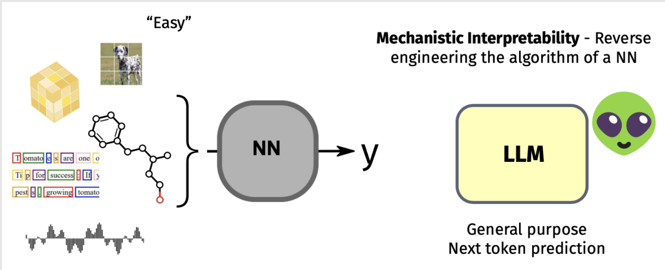

Figure 2

- **LS:** Inputs feed into a neural network (NN), which then outputs some prediction. Interpretability is easier here.
- **LLM:** Interpretability is very hard for next token prediction so need to use mechanistic interpretability.

1.4.2 Attribution for Scientific Discovery: Olfaction

Example:

1. **Overview:** Identifying mechanisms and patterns is at the heart of formulating a scientific hypothesis.
   - **Olfaction:** Sense of smell from chemicals
2. **Boelens' Rose Rule:** A chemical compound smells like rose if:
   - **Functional Group:** OH, OR or OCOR
   - **Carbon Chain:** Carbon atoms
   - **F:** Alpha-branched, unsaturated, or aromatic phenyl moiety.

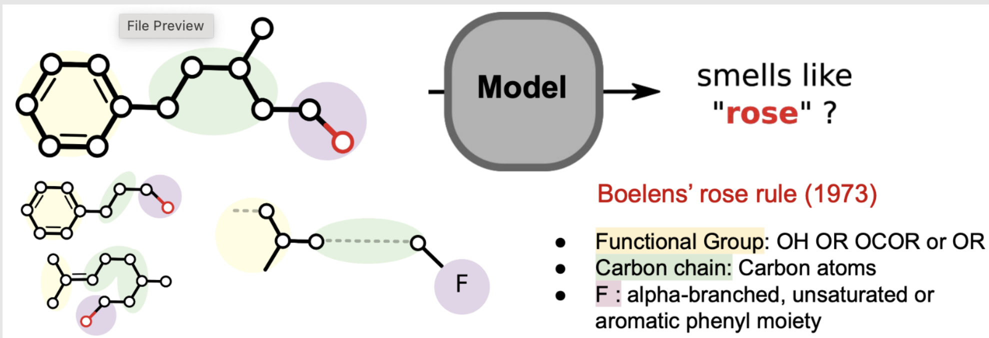

Figure 3

3. **Problem:** Want to build attributions that can explain the rose rule.
4. **Solution:** Easily build attributions with generalized linear models and bag of subgraphs for graphs (in a linear way).

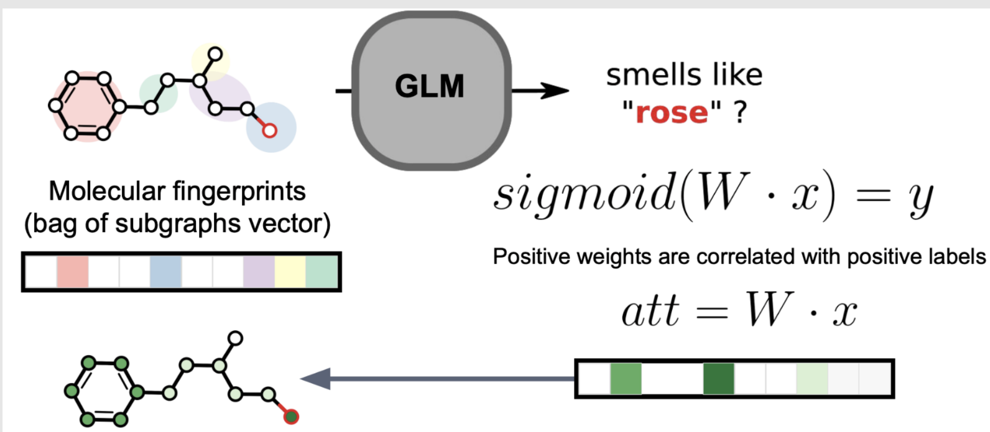

Figure 4

- **Molecular Fingerprints:** Molecular structures (graphs) are converted into vectors.
  - **Bag of Subgraphs:** Each dimension encodes the presence of a subgraph.
- **Model:** A GLM processes these fingerprint vectors to output a prediction (smell like rose or not).
  - W: Learned weights
  - x: Molecular fingerprint vector
- **Attribution:** $att = W \cdot x$ provides a linear attribution score per subgraph, indicating its contribution to the prediction.
- **Interpretation:** Positive weights in W correlate with subgraphs associated with positive labels (e.g., rose scent).

5. **Spurious Correlation Issue:** Statistical patterns in our dataset can affect the weights (and explanations) of our model

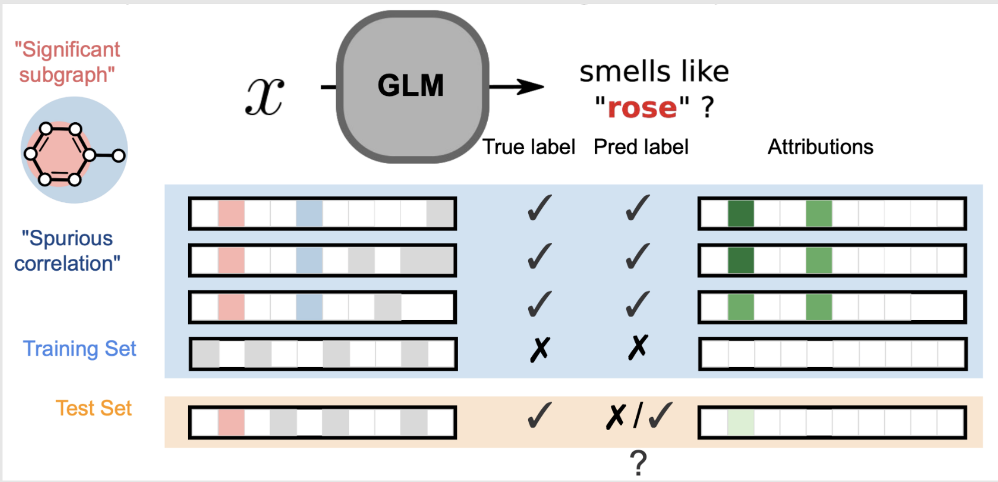

Figure 5

- **Spurious Correlation vs. Significant Subgraph:** Distinguishes between features that are
  - **Actual Correlation:** Significant subgraphs
  - **Spurious Correlation:** Coincidentally correlated with the target label in the training data.
- **Training Set:** GLM learns to associate both real and spurious features with the label "rose" during training (i.e. red subgraph and blue spurious correlation).
- **Test Set:** If the spurious correlations are not present in the test data (i.e. blue not present), the model may:
  - fail to predict the correct label.
  - provide misleading or weak attributions.
  - still succeed due to overlapping structure but without reliable attribution.

6. **Imperfect Model:**

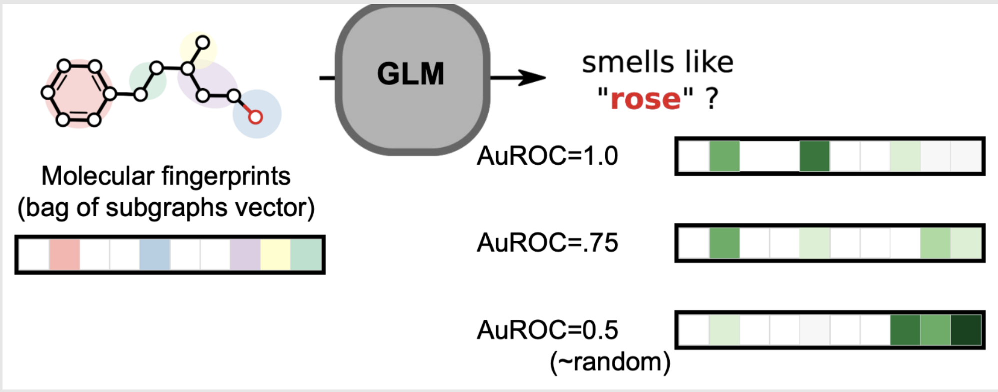

Figure 6

- **Purpose:** If a model does not perform perfectly, its attributions—used for interpretability—may also be unreliable or misleading.
- **Interpretation:**
  - **AuROC = 1.0**, while the model achieves perfect accuracy, the attributions may still reflect spurious or dataset-specific patterns rather than causal substructures.
  - **AuROC = 0.75**, the model makes occasional errors, and the attributions become weaker and less focused.
  - **AuROC = 0.5**, the model performs no better than random guessing, and the attributions are essentially meaningless or noise.
- **Solution:** Use an MLP, but lose access to interpretable weights.

Warning: Perfect models does not mean perfect attributions as it depends on data, splits, etc.

1.4.3 Spurious Correlation: Wolf vs. Dog

Example:

(a) Husky classified as wolf (b) Explanation

Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

	Before	After
Trusted the bad model	10 out of 27	3 out of 27
Snow as a potential feature	12 out of 27	25 out of 27

Table 2: "Husky vs Wolf" experiment results.

Figure 7

- **Spurious Correlation:** Predicts wolf not because of the characteristics of the wolf, but because of the snow.

2 Attribution Tools

2.1 General Trick: Gradients as Importance

Definition: Use the gradient as a proxy for importance:

$$\text{att} \approx \frac{dy}{dx} \cdot x$$

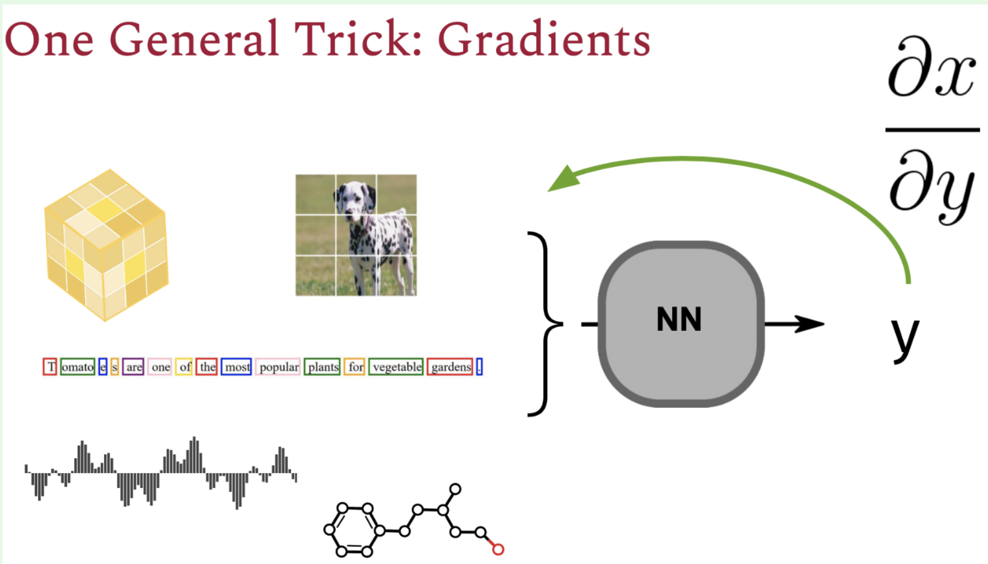

Figure 8

Warning: The shape of the gradient is the same as the shape of the input.

- $\frac{d\text{image}}{dy} \rightarrow$ image-shaped gradient
- $\frac{d\text{graph}}{dy} \rightarrow$ graph-shaped gradient

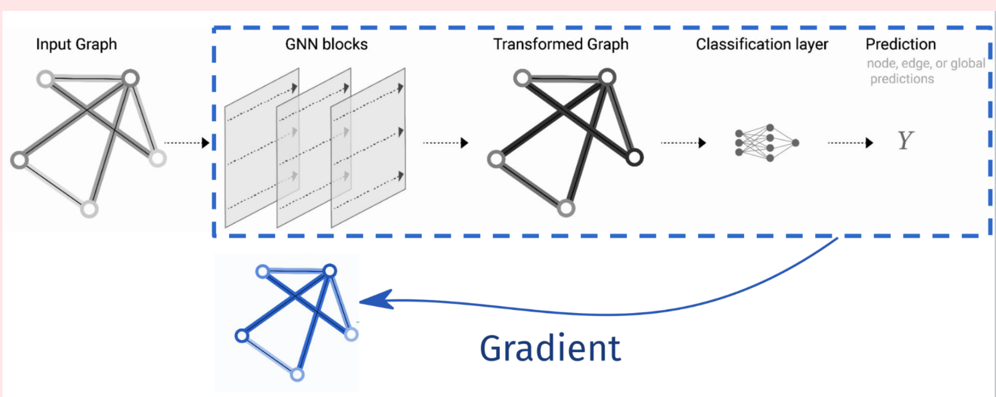

Figure 9

2.2 Attribution Methods

Summary:

Method	**Description**
Class Activation Mapping (CAM - Gradient-less)	$L_{\text{CAM}}^c = \sum_k w_k^c A^k$

- **What?** Produces a heatmap highlighting image regions important for a classification decision
- **How?** Weighting the feature maps of the final convolutional layer by the output layer weights.
  - w_k^c: weight of the k-th feature map for class c
  - A^k: k-th feature map (image-shaped)
  - L_{CAM}^c: class activation map for class c

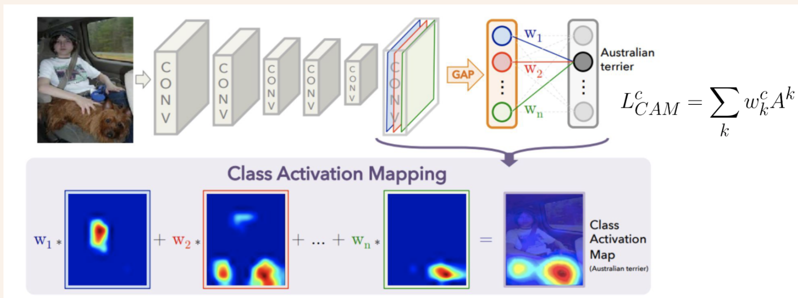

Figure 10

Gradient × Input	Grad × Input $= \mathbf{x} \odot \nabla_{\mathbf{x}} F(\mathbf{x})$

- **What?** Feature importance via gradient magnitude

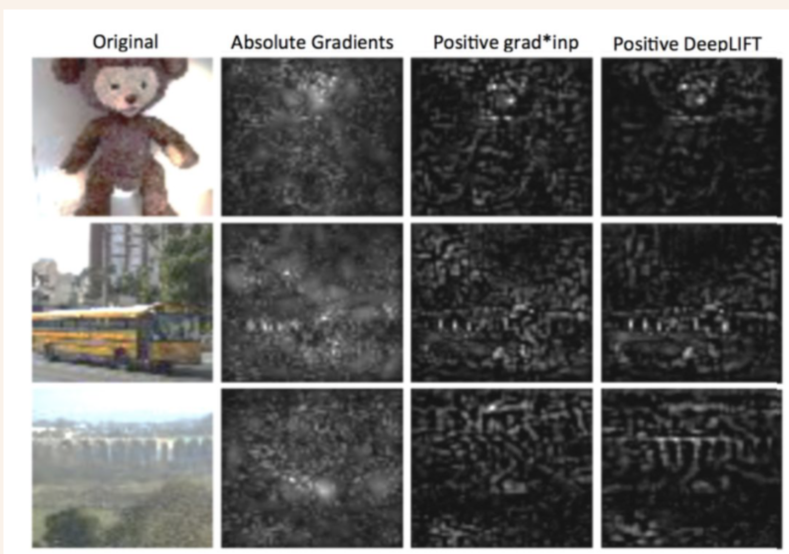

Figure 11

Integrated Gradients

-

GradCAM

-

Smooth Grad

-

2.3 Examples

3 Interpretability Stories

3.1 Examples