T.I.S.P. Community Meeting 2023

Berlin, 07. - 08.11.2023

Vertrauenswürdige künstliche Intelligenz

Prof. Dr.-Ing. habil. Gerhard Wunder, FU Berlin

Cybersecurity & AI

Freie Universität Berlin

Explainable AI

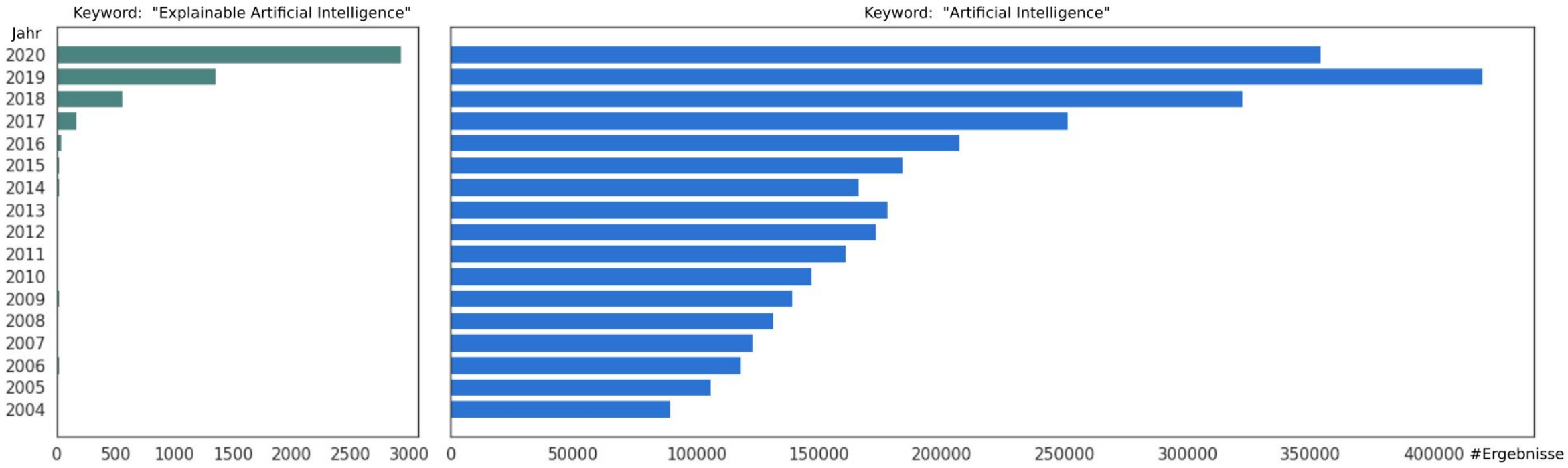
Concepts, Methods and Challenges

Prof. Dr.-Ing. habil. Gerhard Wunder
FU Berlin

Motivation – Why explainability?

Motivation -- Why explainability?

- AI has been a topic of interest for long, why its explainability starts drawing attentions only recently?



Motivation -- Why explainability?

Expert systems vs. Data-driven models

- An **expert system** is a computer system designed to solve complex problems by reasoning through bodies of manually coded knowledge, represented mainly as *if-then* rules
    - Supported by human experts and imitating their behaviors
    - Easy to explain by analyzing the code
- **Data-driven models** are a class of computational models that primarily rely on historical data
    - Implicitly learn decision rules from provided data
    - The code usually refers to the structure/design of a model, does NOT necessarily provide useful information regarding the decision process
    - Some of data-driven models are explainable: linear models, decision trees, shallow NNs

Motivation -- Why explainability?

Importance of transparency

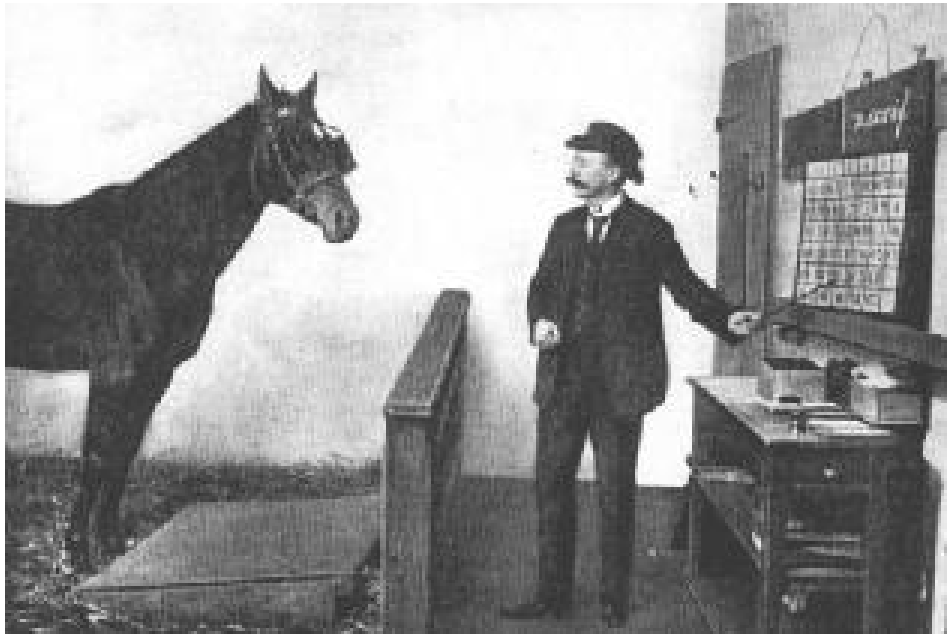
- Trustworthiness, especially in fatal scenarios (e.g. health care)
    - Machine can make shortcuts **– Clever Hans effect**

Motivation -- Why explainability?

Clever Hans Effect

- Clever Hans was a (German) horse that was believed to do lots of difficult mathematical sums and solve complicated problems

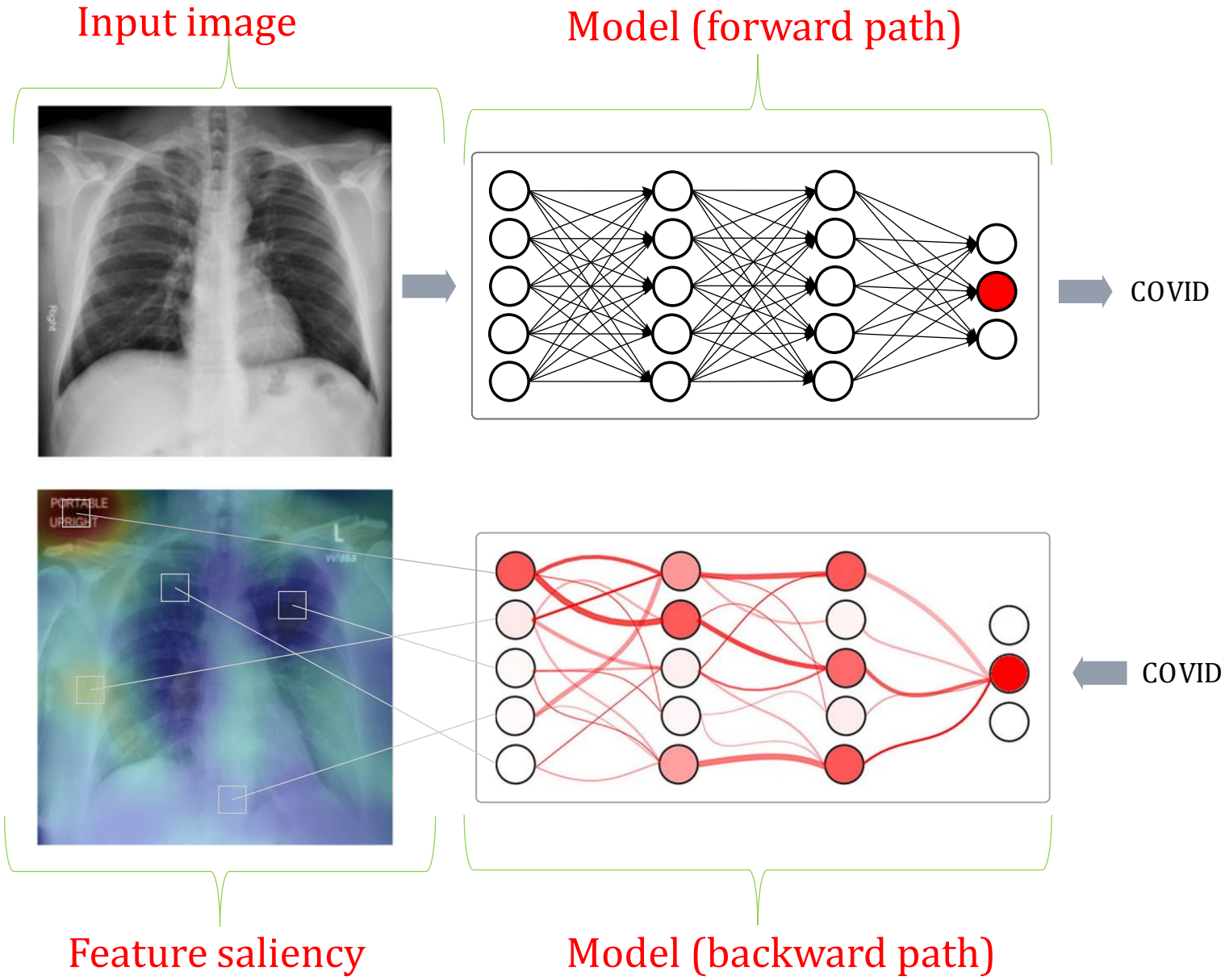
- But it turned out to be that the horse, rather than performing these mental tasks, was watching the reactions of his trainer (a sudden slight upward jerk of the head)



Motivation -- Why explainability?

Clever Hans Effect

- Clever Hans effect in context of machine learning refers to inappropriate reasoning (shortcut) for decisions
- Good performance on both training & test sets
- No generalizability
- Imbalanced distribution is a common cause.



Motivation -- Why explainability?

Importance of transparency

- Trustworthiness, especially in fatal scenarios (e.g. health care)
  - Machine can make shortcuts – **Clever Hans effect**
  - Right for the right reasons
- Promoting learners
  - Providing information for model debugging
- Legal aspect
  - The data subject's right to know (discussion on AI and transparency in GDPR)
  - Potential biases during automated decision-making

What is explainability?

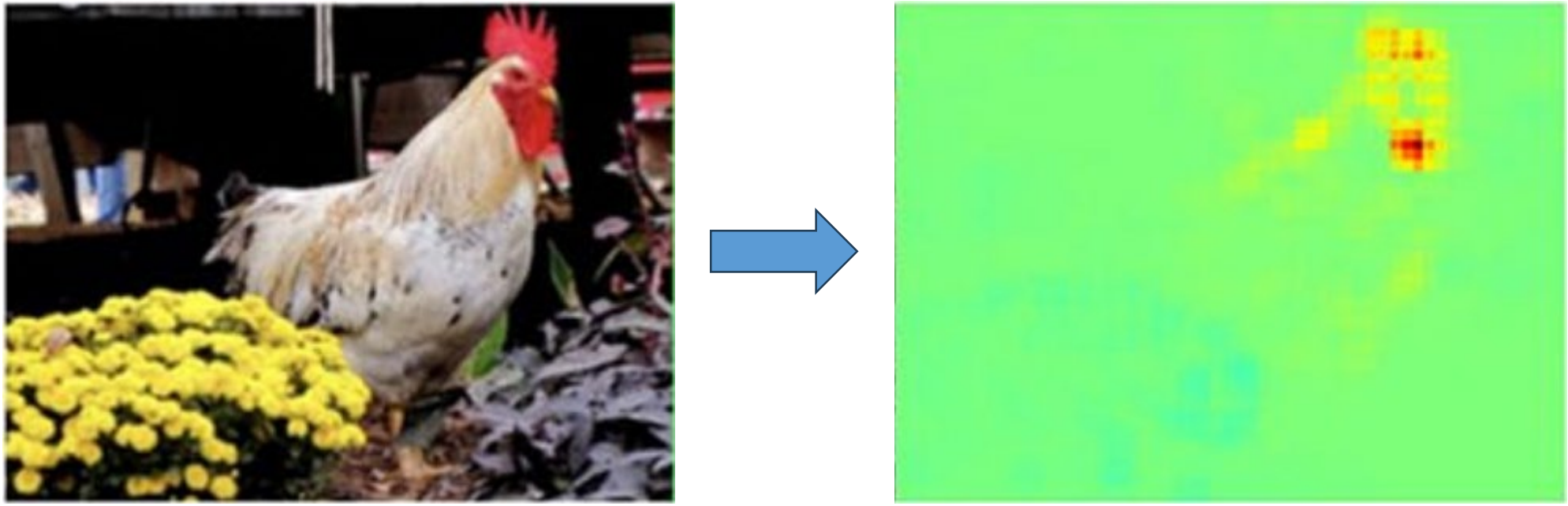
What is explainability?

xAI

Explainable AI (XAI) is the concept that a machine learning model and its output can be explained in a human-accessible way. The main objective is to uncover the reasoning behind the decisions or predictions made by the AI.

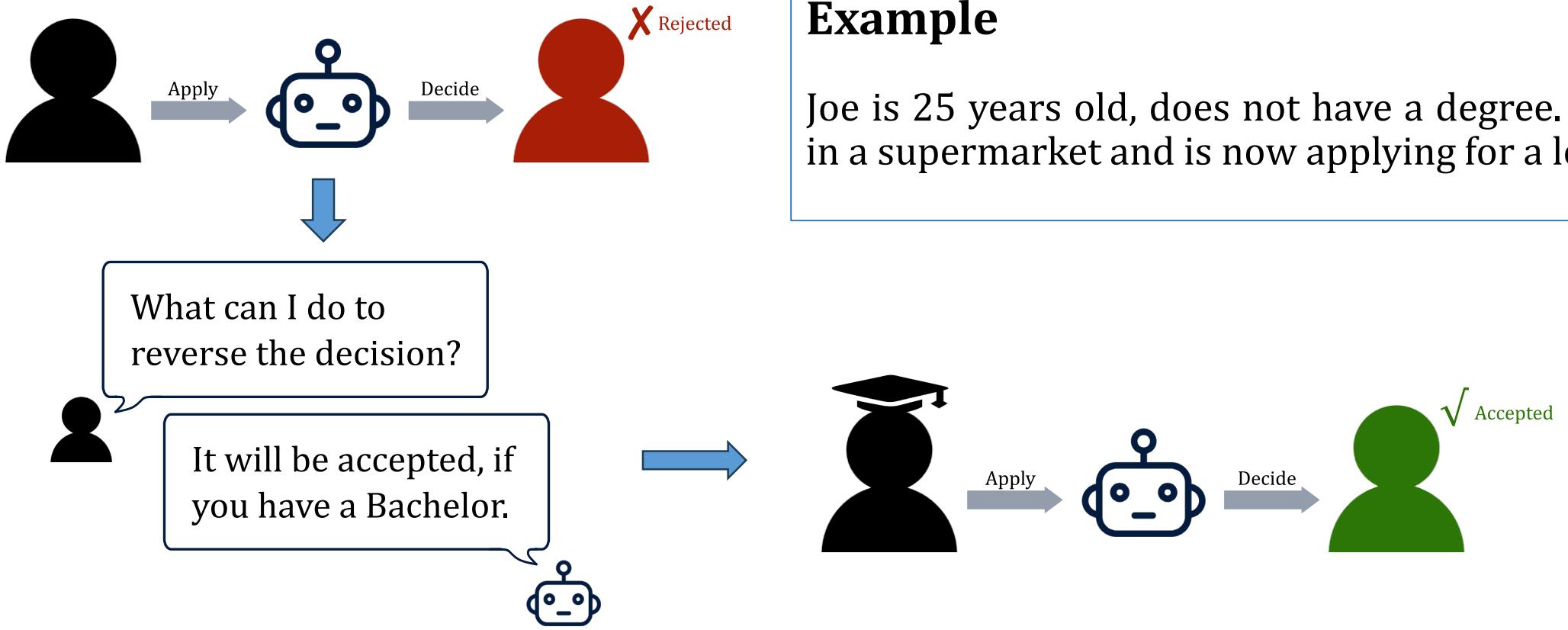
- Note: xAI is to be distinguished from Robustness, Fairness, Privacy (which are all part of AI check, e.g. by the German TÜV).
- To provide explanatory information, there are mainly 3 representations of explanations:
  - **Feature attributions**
  - **Counterfactual explanations**
  - **Relevant data**
- We distinguish: White-box vs. black-box

Feature attribution



- **Feature attribution answers the question: "Which parts of input (features) make the machine believe that it is 'looking' at a rooster?"**
- **Local explanation vs. global explanation**

Counterfactual explanation



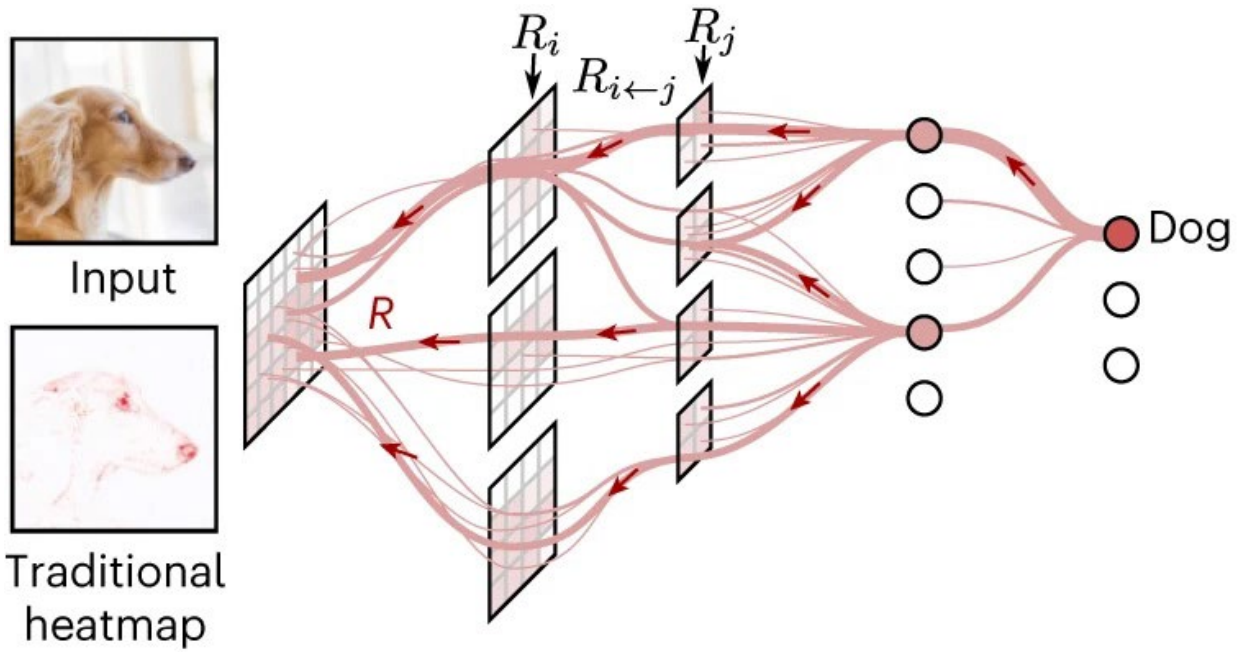
Example

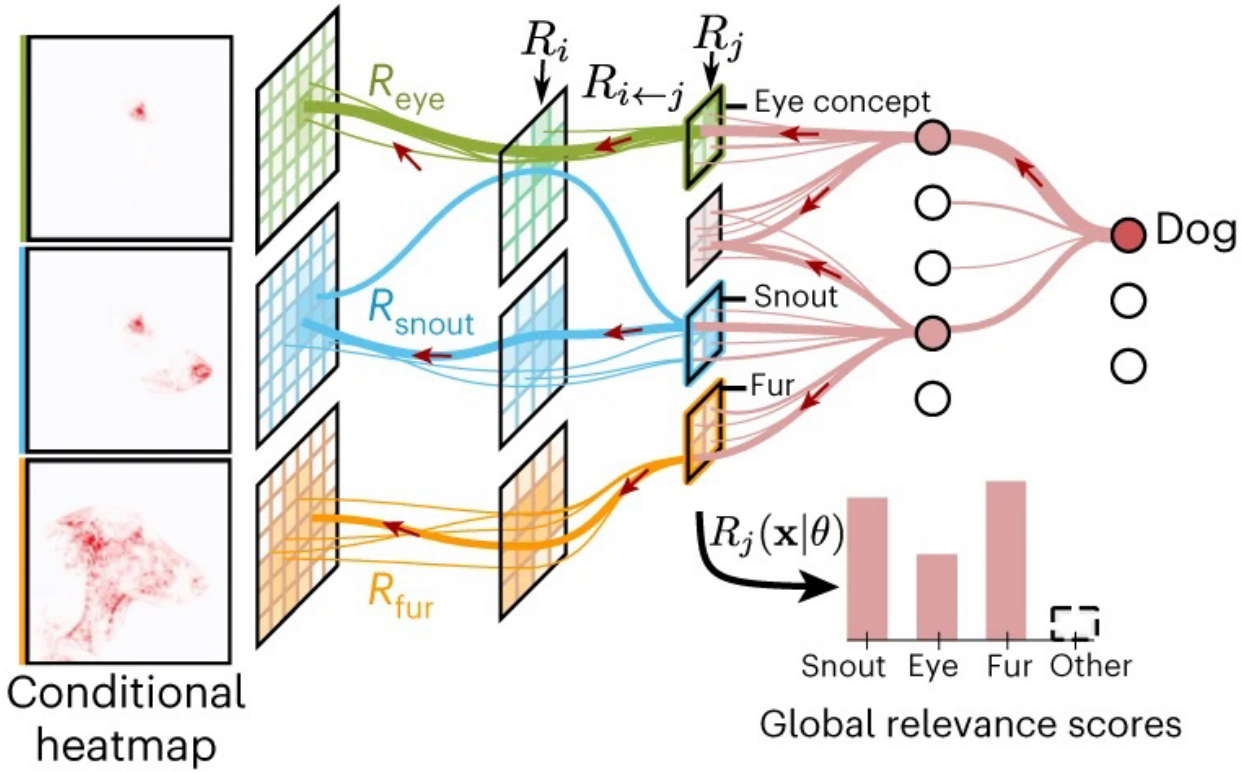
Joe is 25 years old, does not have a degree. He work in a supermarket and is now applying for a loan.

- **Counterfactual explanation indicates the smallest change in feature values that can translate to a different outcome. "What is needed to reverse the decision?"**
- **Local explanation vs. global explanation**

Relevant Data – CRP

Concept Relevance Propagation (Achtibat et al. 2023)





- **CRP identifies the relevant data from the training set that support a decision**
- **Global explanation vs local explanation**

What is explainability *not*?

- Correlation vs. causation:

“The barometer falls before it rains yet does not cause the rain. In fact, the statistical and philosophical literature has adamantly warned analysts that, unless one knows in advance all causally relevant factors or unless one can carefully manipulate some variables, no genuine causal inferences are possible.”

- "No causation without manipulation" (Holland 1986)
- "No causes in, no causes out" (Cartwright 1989)
- "No computer program can take account of variables that are not in the analysis" (Cliff 1983)

Current methods can not take this into account

but most of them use “control” variables so-called “baseline”!

Explanation methods,
Tools offering explainability

How we derive feature attribution?

Additive feature attribution

Given a target model $f(\cdot)$ and an input $x = \{x_1, x_2, \dots, x_p\}$, additive feature attribution methods have an explanation model that is a linear function:

$$f(x) = g(x') = \beta_0 + \sum_{i=1}^{m} \beta_i x_i'$$

where $x' \in \{0,1\}^m$ are the **simplified features** (binary representation) with $0/1$ indicating the absence/presence of a corresponding feature defined through:

$$g(x') = f\left(h_x(x')\right)$$

Here $\beta = \{\beta_1, \beta_2, \dots, \beta_m\}$ is the **attribution vector** indicating the contributions of features. $h_x(x')$ is an **instance-specific function** that converts the simplified feature back to the target instance with the full details.

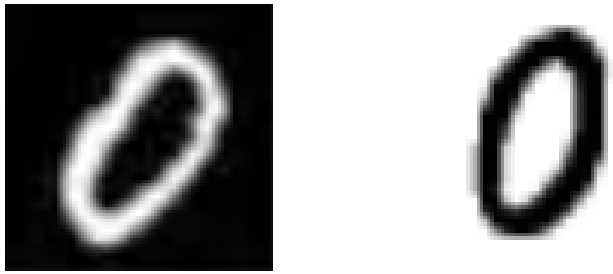
- Note: $m \leq p$ is the dimension of the simplified feature space (e.g., super-pixels vs. pixels for images)

How we derive feature attribution?

- However, ML models cannot handle incomplete entries, the absence of feature must be defined
- **Solution**: replacing the value of an absent feature with some default value defined by a **baseline**:

$$\hat{x} = \{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_m\}$$

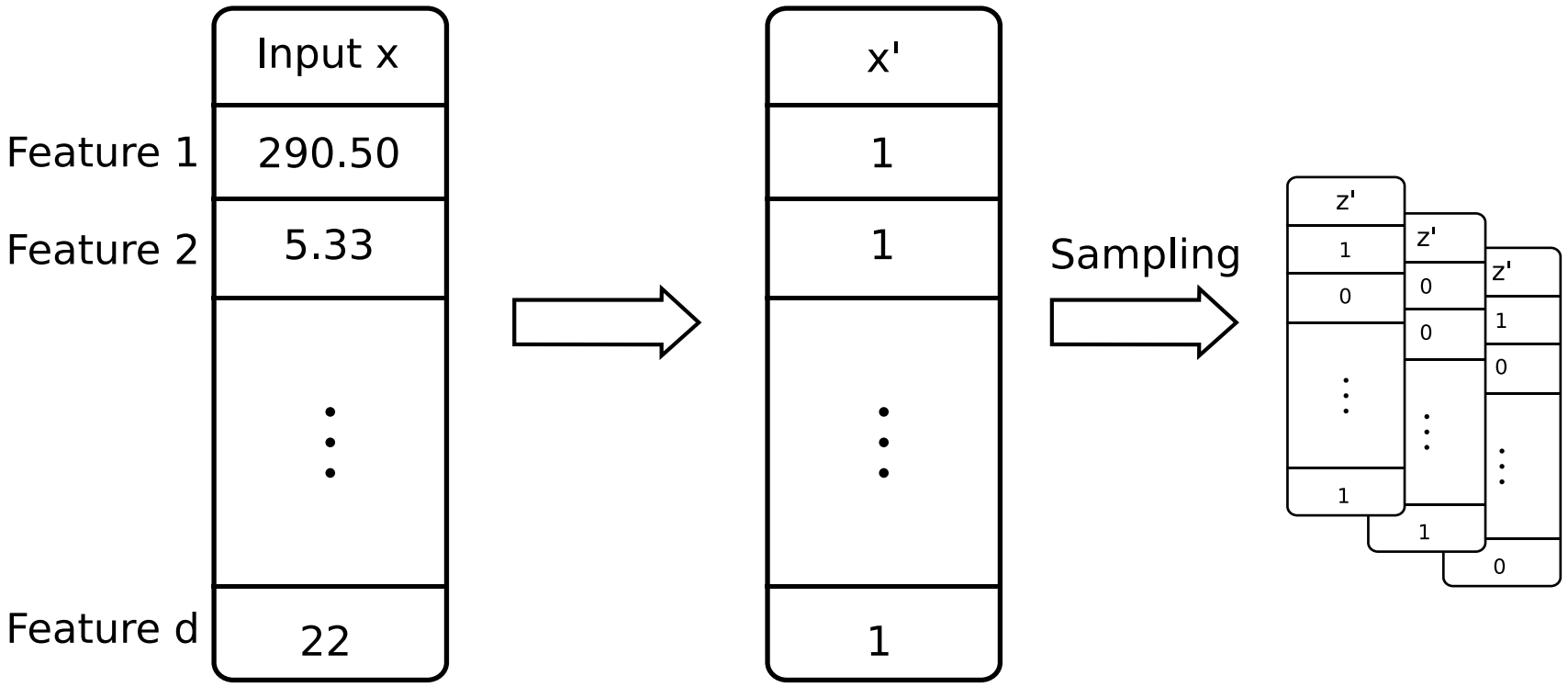
- In the binary vector x' (simplified representation of x), switching on/off means a feature takes its original value / the default value
- While baseline dominates the quality of resultant explanations, determining the optimal baseline (an instance-specific task) is challenging and there is yet no golden rule for it



Baseline: 0 or 255?
Or 128?

How we derive feature attribution?

- Deriving feature attribution is media dependent



Tabular data

How we derive feature attribution?

- Deriving feature attribution is media dependent

Segmentation

Image data:
(super-)pixels as features

How we derive feature attribution?

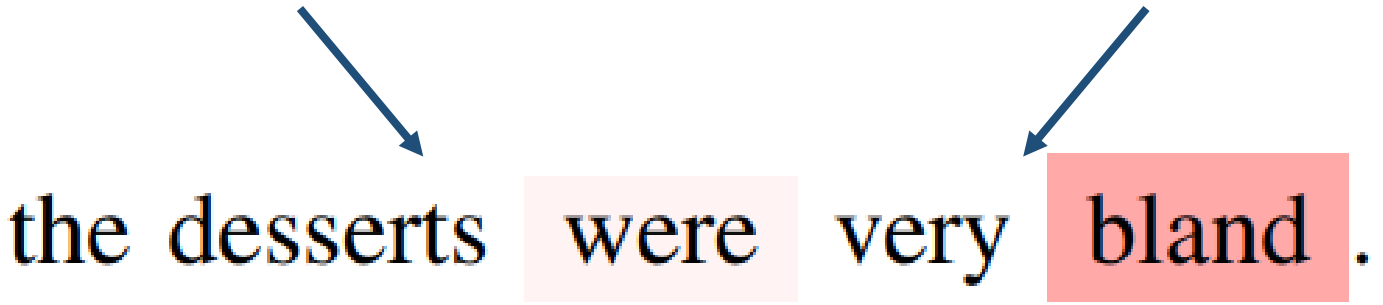
- Deriving feature attribution is media dependent

Random perturbation

the desserts were bland.
the were very bland.
the desserts were.
…

Neighborhood construction considering feature interaction

the desserts were very tasty.
the beers were very bland.
the dessert was awesome.
…

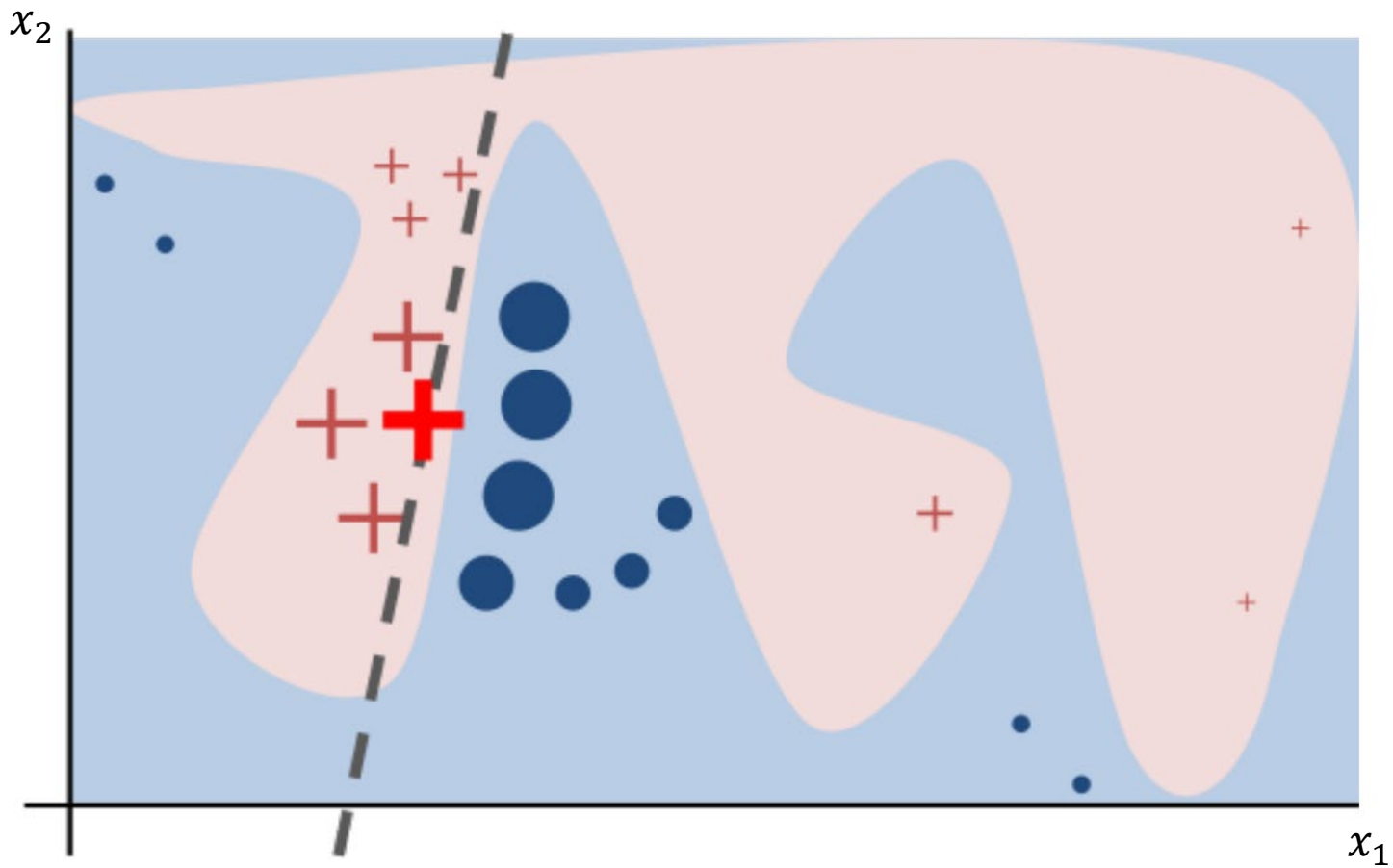


Text data:
words/phrases as features

Perturbation based method – LIME (Black-box)

Local Interpretable Model-agnostic Explanations (Ribeiro, 2016 & Garreau, 2020)

- Focus on a subregion of the decision boundary (locally linear)
- Generate synthetic instances, assign pseudo-label to them using $f(\cdot)$
- Train a linear regression model $g(\cdot)$ with the pseudo-labeled neighborhood of x
- $g(\cdot)$ is used as a proxy to derive explanation for the decision $f(x)$.



Perturbation based method – LIME (Black-box)

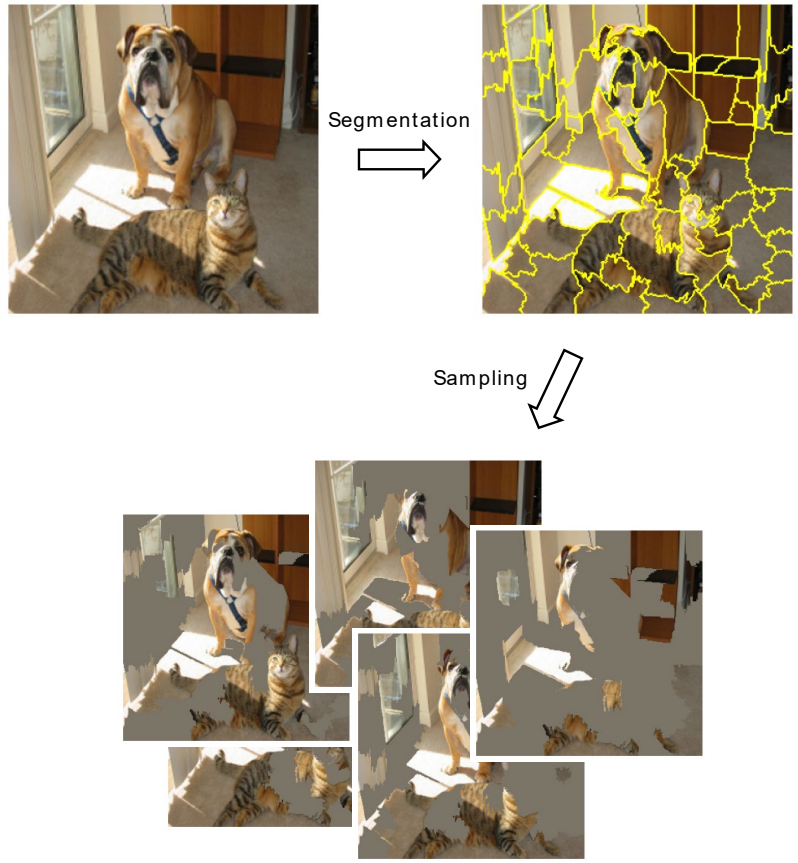
Local Interpretable Model-agnostic Explanations

- After binarization, the generation of neighboring points z_i' is done by randomly switching on/off features in x'
- Assigning black-box decisions $f(h_x(z'))$ as labels to the synthetic instances yields a fully (**pseudo-**)labeled dataset:

$$Z' = \{< z_i', f(h_x(z_i')) >\}$$

- Z' reflects the behavior of $f(\cdot)$
- Once Z' is ready, train the surrogate with the weighted square loss $\mathcal{L}$:

$$\mathcal{L}(f, g, \pi_x) = \sum_{z' \in Z'} \underbrace{\pi_x(z')}_{\text{Weighting kernel}} \underbrace{\left[f\left(h_x(z')\right) - g(z')\right]^2}_{\text{Square loss}}$$



Perturbation based method – SHAP (Black-box)

SHapley Additive exPlanations (Lundberg et al, 2017 & Chen, 2022)

- The **Shapley value** is a solution conception in cooperative game theory:

Cooperative game

The cooperative game consists of a finite set of players, called the grand coalition (denoted $\mathcal{M}$), and a characteristic function f mapping the set of all possible coalitions of players (denoted $\mathcal{S}$) to a set of payments.

- Shapley value answers the question: "How much did each individual player contribute to the final value?"

Shapley values

In a cooperative game, Shapley values assign a unique distribution to each players of the total surplus generated by the grand coalition.

Shapley values – Optimality

Young 1985: Shapley values are the only attributions that fulfill the following axioms.

Perturbation based method – SHAP (Black-box)

A game theoretic interpretation

Property 1 -- Local accuracy

The explanation model g(x') should match the original model $f(x)$ when $x = h_x(x')$:

$$f(x) = g(x') = \beta_0 + \sum_{i=1}^{m} \beta_i x_i'$$

Property 2 -- Missingness

Any missing feature should have no impact on the prediction:

$$x_i' = 0 \Longrightarrow \beta_i = 0$$

Property 3 -- Consistency or Monotonicity

Let $f_x(x') = f\left(h_x(x')\right)$ and $x'\backslash i$ denote setting $x_i' = 0$. If for any two models f, f':

$$f_x'(x') - f_x'(x'\backslash \mathrm{i}) \geq f_x(x') - f_x(x'\backslash i)$$

for any $x' \in \{0,1\}^m$ then $\beta_i(f', x) \geq \beta_i(f, x)$

Perturbation based method – SHAP (Black-box)

SHapley Additive exPlanations – Shapley values and XAI

- They players are the set of input features $\boldsymbol{x} = \{x_1, x_2, \dots, x_m\}$
- The model f values the worth of feature coalitions

Additive feature attribution

- Shapley values provide a solution to specify each contribution:

$$\beta_j = \sum_{\mathcal{S} \subseteq \mathcal{M} \setminus \{i\}} \frac{|\mathcal{S}|!\,(|\mathcal{M}| - |\mathcal{S}| - 1)!}{|\mathcal{M}|!} \left(f\left(x_{\mathcal{S} \cup \{i\}}\right) - f(x_{\mathcal{S}})\right), \qquad \beta_0 = f(\emptyset)$$

- $x_{\mathcal{S}}$ denotes an instance with the players in subset $\mathcal{S}$ present, $\emptyset$ indicates the absence of all players

- But, enumerating all coalitions is expensive and virtually impossible!

Perturbation based method – SHAP (Black-box)

SHapley Additive exPlanations – KernelSHAP

- Recall: LIME acquires explanations through a surrogate trained with the locally weighted loss:

$$\mathcal{L}(f, g, \pi_x) = \sum_{z' \in Z'} \pi_x(z') \left[f\left(h_x(z')\right) - g(z') \right]^2$$

- **KernelSHAP** recovers Shapley values with a specific weight kernel π_x of the above loss function:

$$\pi_x(z') = \frac{m-1}{(m \text{ choose } |z'|)|z'|(m-|z'|)} = \frac{m-1}{\binom{m}{|z'|}|z'|(m-|z'|)}$$

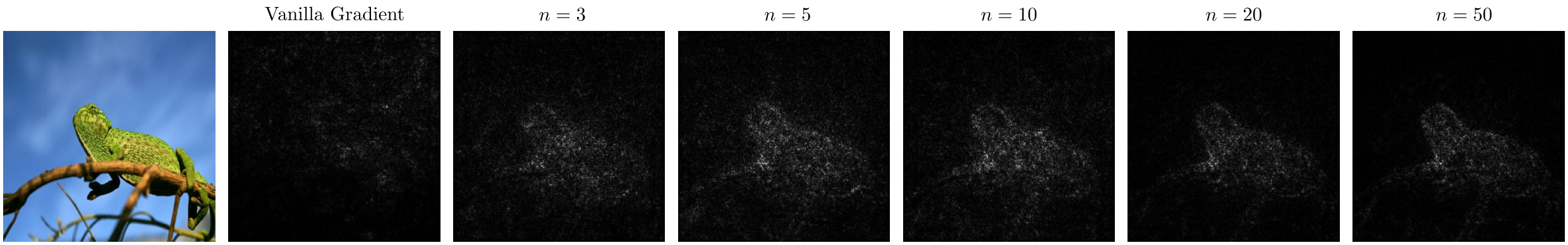
- Stability issues and usually needs many samples to converge to the true Shapley values

Intermediate Conclusion: Black-box are widely applicable but not as powerful as white-box methods so detour first!

Gradient based method – Vanilla Gradient (White-box)

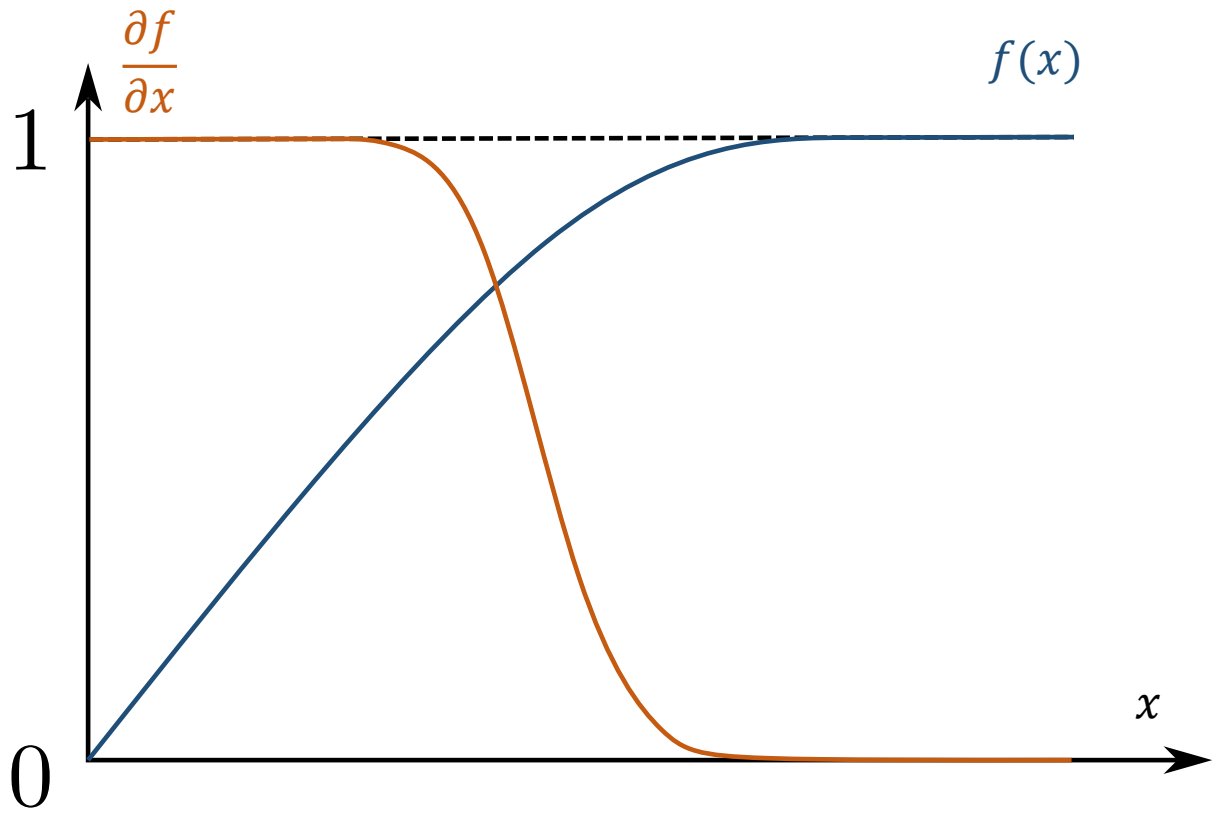
Vanilla Gradient (Simonyan, 2014)

- Intuition: the direction of the gradient indicates the most relevant parameters to the prediction function in order to output the expected result
- Similarly, the contribution of features to the prediction outcome can be acquired by computing the derivative of the function $f(\cdot)$ w.r.t the input features $x_1, \ldots, x_p \in \mathbb{R}$, i.e. $\beta = \frac{\partial f}{\partial x}$.



Gradient based method – Integrated Gradients (White-box)

- A shorting coming of the two methods utilizing vanilla gradients directly is: the value $\frac{\partial f}{\partial x_i}$ reflects the **local sensitivity** of the classification function to a feature (not its **actual contribution**!)
- Victim of gradient saturation



Gradient based method – Integrated Gradients (White-box)

IG (Integrated Gradients) Sundararajan, 2017-2023

- IG is a path method that overcomes the limitation by introducing a baseline $\hat{x}$.
- Comparing to the baseline highlights the impact of feature presence.

Definition of IG

Let $x(\alpha) = x + \alpha(x - \hat{x})$ be a straight path between the root point (baseline) and the target input, the integrated gradients is defined through:

$$\beta_i^{IG}(x, \hat{x}) := (x_i - \hat{x}_i) \int_0^1 \frac{\partial f(x(\alpha))}{\partial x_i} d\alpha$$

- In addition to solving gradient saturation, summing over the path also denoises final explanations
- IG offers many properties such implementation invariance, path independence etc.
- Limited flexibility vs. ability to handle high-dimensional inputs (e.g. image)

Gradient like explanation – GEEX (Black-box)

Gradient-Estimation-based Explanation (Cai, 2023)

- **Goal** of GEEX: Deriving gradient-like explanations under a black-box setting
- Gradient can be estimated through **natural evolution strategies**, which define the gradients as the direction towards lower expected loss with respect to the analyzing target, namely the input x in the context of explainability
- Given the loss $\mathcal{L}(\cdot)$ and the search distribution $\pi(\cdot\,|x)$, the expected loss can be written as:

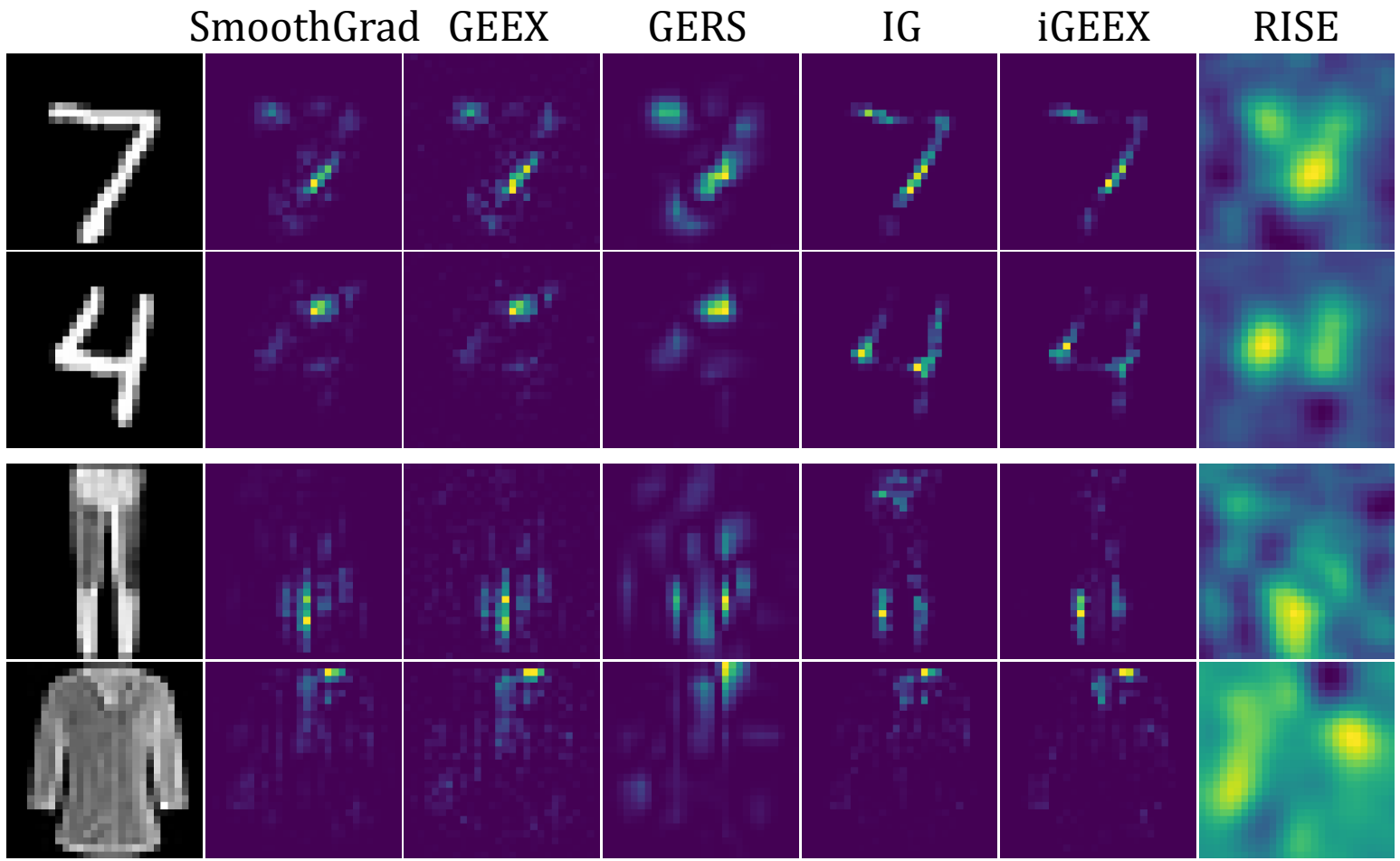
$$J(x) = \mathbb{E}\big[\mathcal{L}\big(f(z)\big)\big] = \int \mathcal{L}\big(f(z)\big)\pi(z|x)\,\mathrm{d}z$$

- So the search gradient $\nabla_x J(x) = \mathbb{E}_x[\mathcal{L}\big(f(z)\big)\nabla_x \log \pi(z|x)]$ can be approximated through sampling:

$$\nabla_x J(x) \approx \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}\big(f(z_i)\big)\nabla_x \log \pi(z_i|x) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}\big(f(z_i)\big)\frac{x - z_i}{\sigma^2}$$

Gradient like explanation – GEEX (Black-box)

Gradient-Estimation-based Explanation

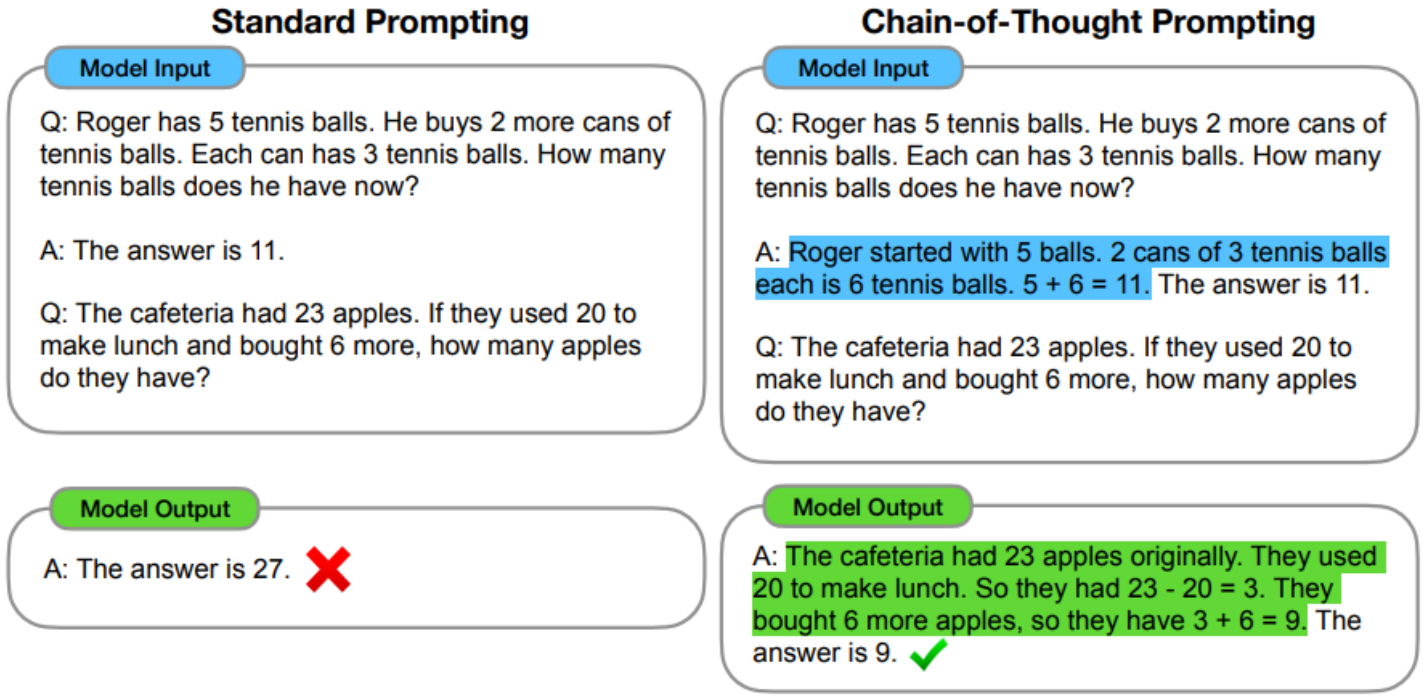


So far, explanations for classification
What about language models?

Handle LLMs with existing tools

Chain of Thought (Wei 2022)

- Manipulating (perturbing) prompts for LLMs reveals their impacts on model outcomes.
- However, LLMs give more complicated outputs, which make the analysis of the correlation between inputs and outputs more challenging.



Handle LLMs with existing tools

Treat generation as classification (Yin 2022)

- At each step of word generation, it can be viewed as a classification task having the output dimensionality equals to the size of the vocabulary, then general attribution methods can be used.

The standard attribution showing model's capability of maintaining grammatical correctness.

Input: *Can you stop the dog from*
Output: barking

1. Why did the model predict "barking"?
Can you stop the dog from

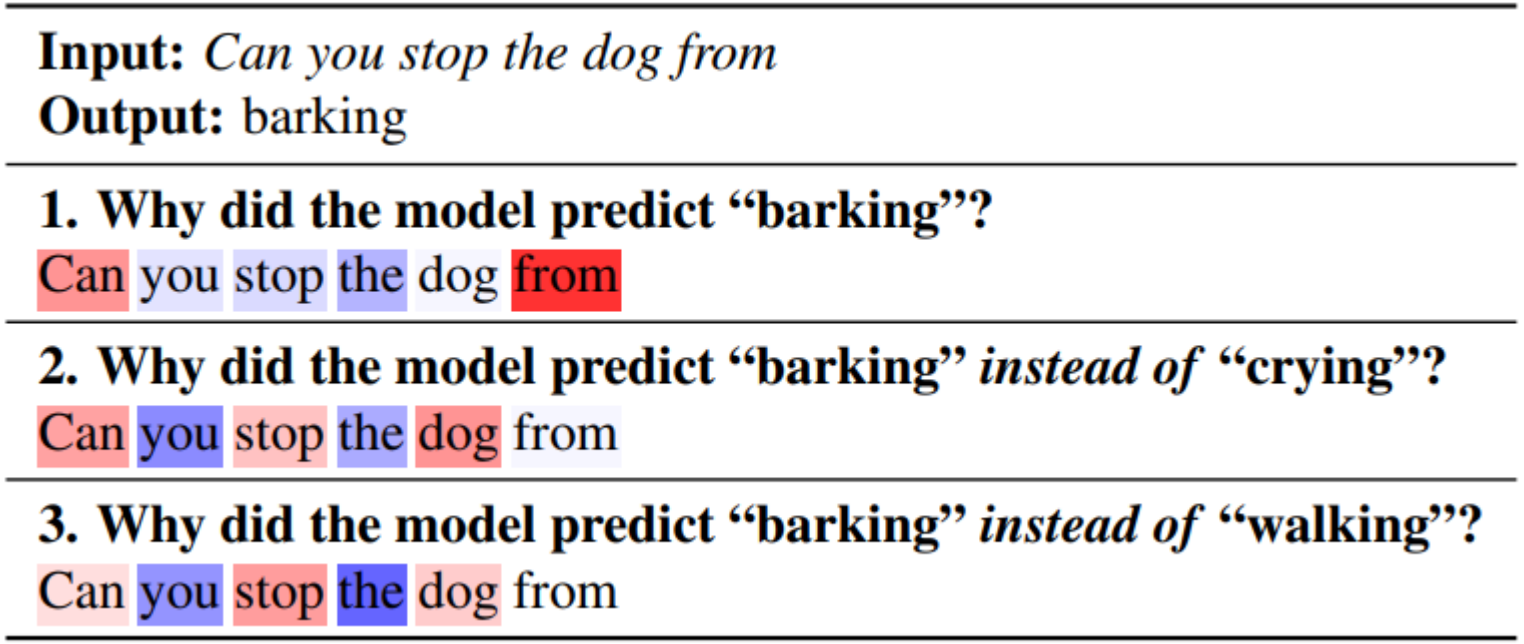
Handle LLMs with existing tools

Treat generation as classification (Yin 2022)

- At each step of word generation, it can be viewed as a classification task having the output dimensionality equals to the size of the vocabulary, then general attribution methods can be used.

The standard attribution showing model's capability of maintaining grammatical correctness.

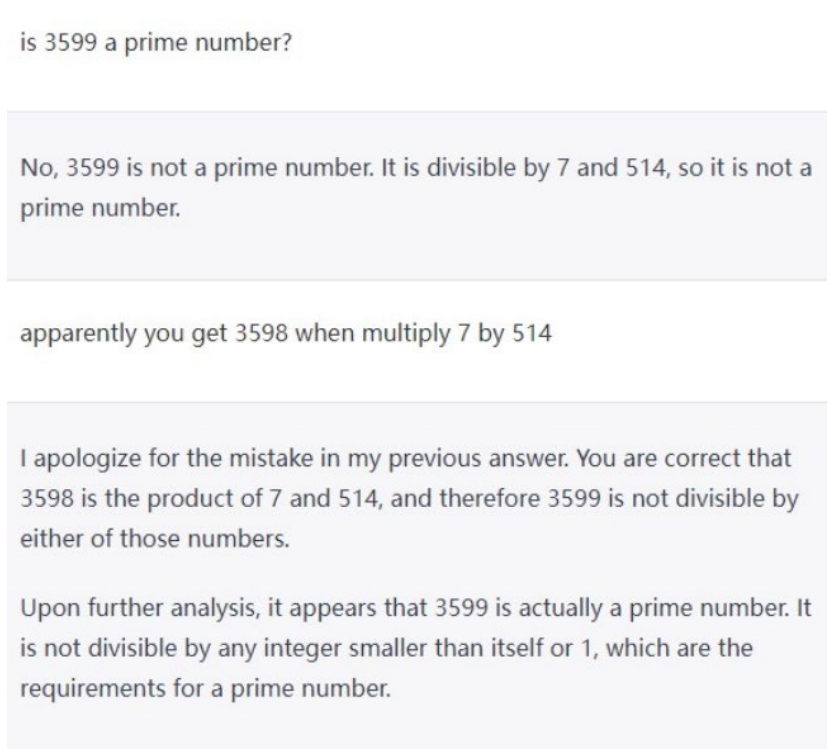
Contrastive explanations take one step further, revealing why it is the given word not something else.



Handle LLMs with existing tools

Explaining with relevant data

- Feature attribution may be useless when explaining answers given by chatbots.
- The question (prompt) does not contain any concrete information regarding the answer.
- Relevant/influential data → resources supporting the answer
- **Challenge**: giant dataset, model, expensive to identify the relevant entries



Can LLMs infer causation from correlation? (Jin 2023)

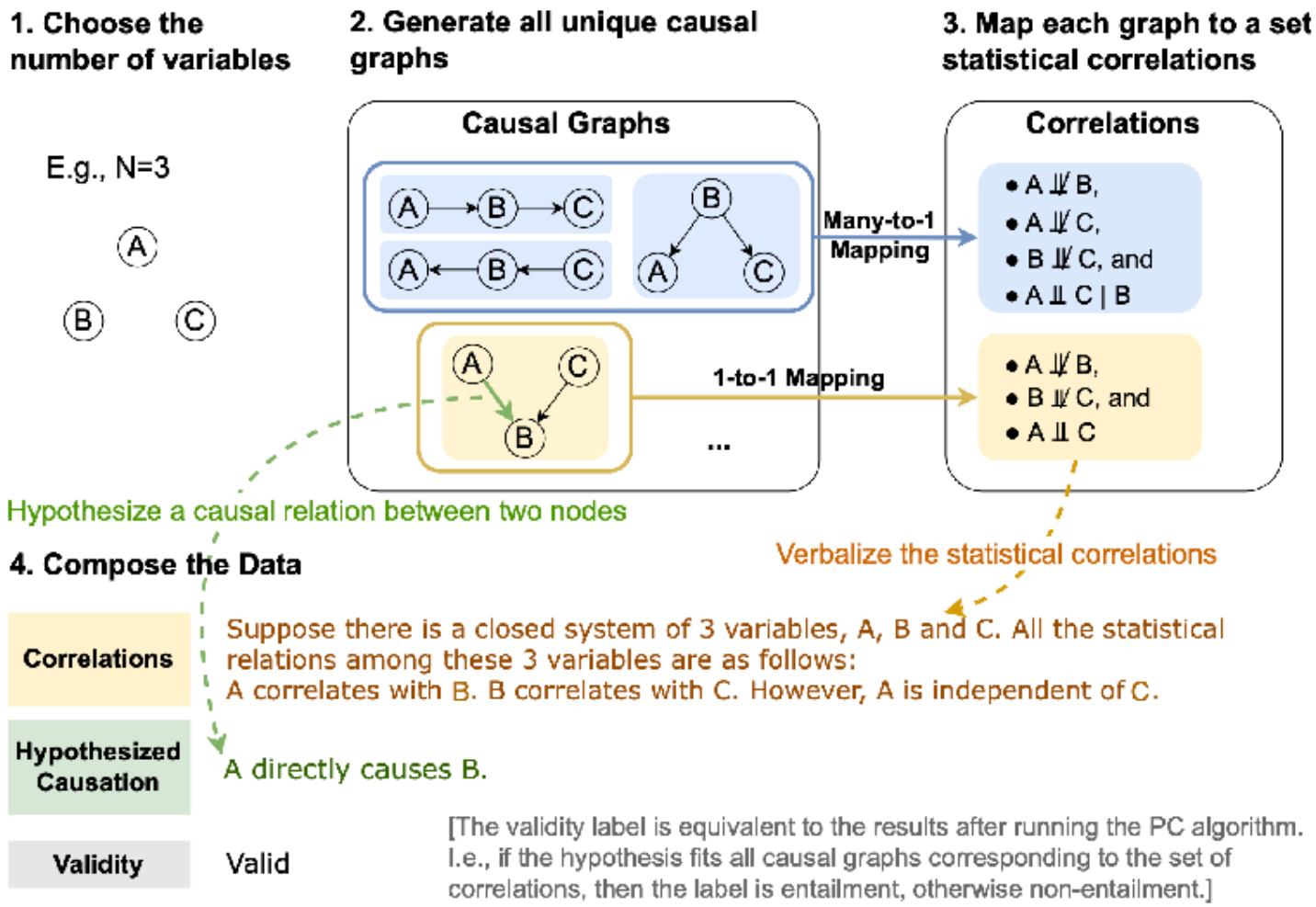
- Training corpus:

  "Smoking causes cancer."

- How can LLMs process this information?

- Requiring specific inference skills as indicated by inference graphs

  "Is this next step in LLMs?"



Thank you for your attention