Concept-based Explainable AI

Explainable and Trustworthy AI

Gabriele Ciravegna

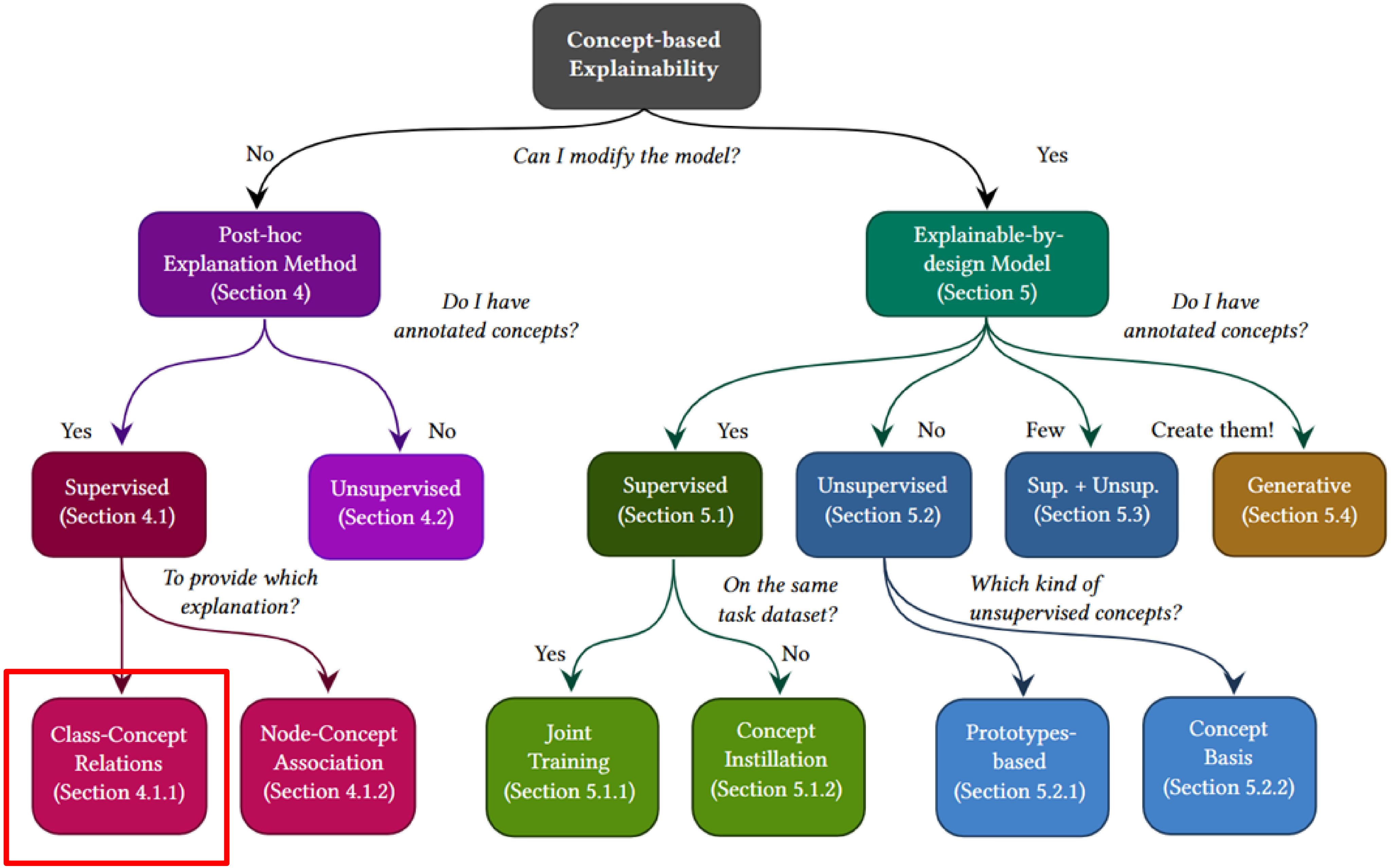
OUTLINE

1. Motivation
2. Concept-based eXplainable AI (C-XAI)
3. Testing with Concept Activation Vectors (T-CAV)
4. Concept Bottleneck Models (CBM)
5. Concept Embedding Models (CEM)

C-XAI PART I

C-XAI PART II

3. Testing with Concept Activation Vectors (T-CAV)



Concept-based Explainability
Can I modify the model?
No
Yes
Post-hoc Explanation Method (Section 4)
Explainable-by-design Model (Section 5)
Do I have annotated concepts?
Yes
No
Supervised (Section 4.1)
Unsupervised (Section 4.2)
To provide which explanation?
Class-Concept Relations (Section 4.1.1)
Node-Concept Association (Section 4.1.2)
Do I have annotated concepts?
Yes
No
Few
Create them!
Supervised (Section 5.1)
Unsupervised (Section 5.2)
Sup. + Unsup. (Section 5.3)
Generative (Section 5.4)
On the same task dataset?
Yes
No
Joint Training (Section 5.1.1)
Concept Instillation (Section 5.1.2)
Which kind of unsupervised concepts?
Prototypes-based (Section 5.2.1)
Concept Basis (Section 5.2.2)

Example: Post-training explanation

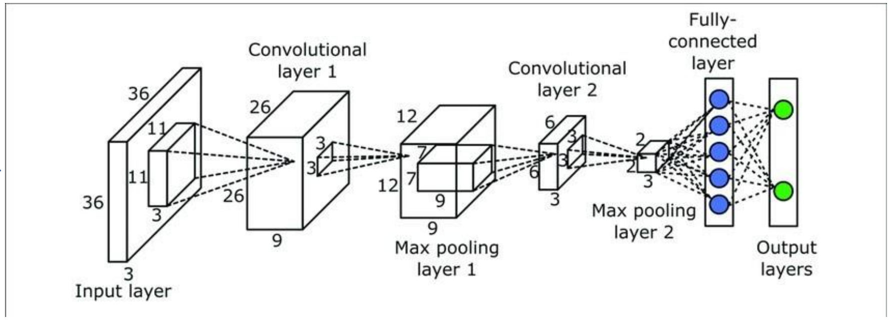
- To use machine learning responsibly, we need to ensure that
  - Our **values are aligned**
  - Our **knowledge is reflected**

- Standard XAI Solutions
  - **Interpretable** ML model (e.g. linear model)
    - Simple but we significantly lose the performance
  - **Post-training** explanation
    - E.g. Perturbation-based/sensitivity analysis-based methods
    - May be difficult to trust for standard users

Example: Post-training explanation

Given Image
(Cash-machine)

Trained ML model
(e.g. GoogleNet)



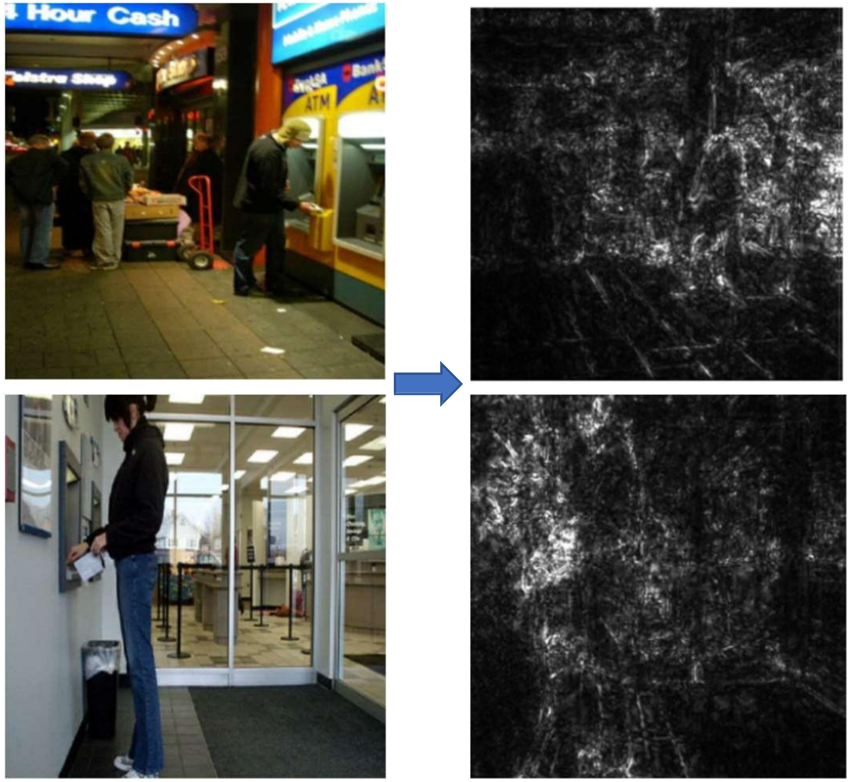
Predicted as
Cash-machine

$$p(z)$$

- **Why was this a cash machine?**

Problem Objective

Given Image | **Corresponding Saliency Map**



Prediction: Cash-machine

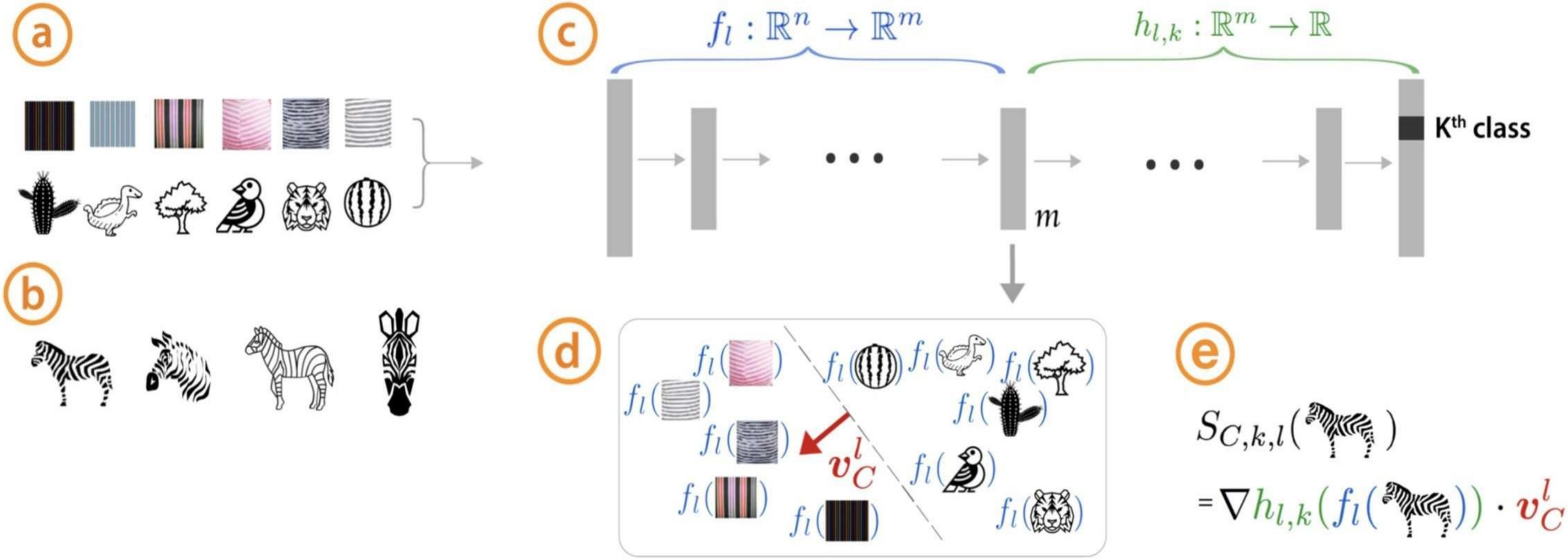
Prediction: Sliding door

- Did the **'human'** concept matters?
- Did the **'paper'** concept matters?
- Did the **'ATM'** or **'Cash'** concept matters?

TCAV objective:

Quantitatively measure how important are **"user- chosen concepts"**

TCAV: Overview



TCAV components

a) A dataset annotated with both **examples of concepts** and **random images**

b) The dataset with the **original classes**

c) The **model** to explain

d) The Concept Activation Vectors (CAV)

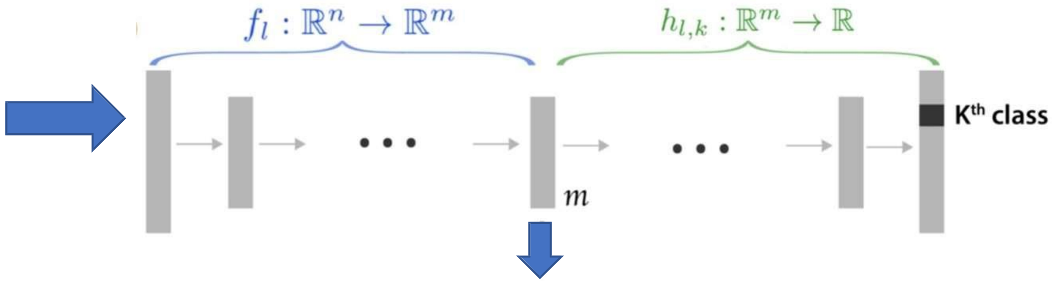
e) The TCAV score showing the **influence** of a concept on a given class

TCAV: (1) How to define CAV?

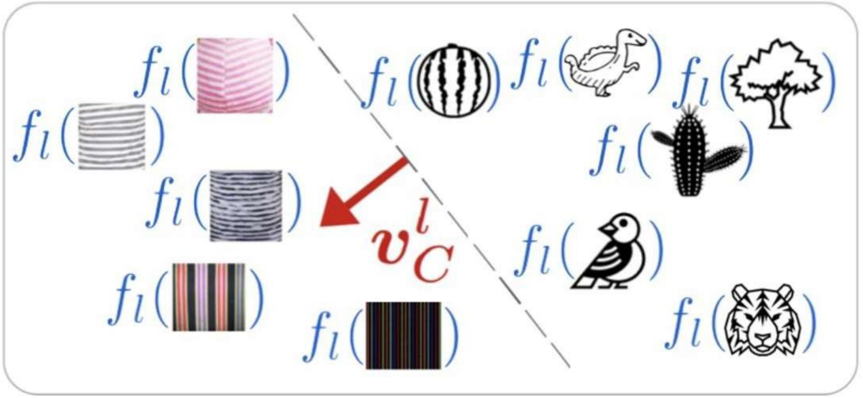
Strip Images

Random Images

Trained ML model



Internal tensors

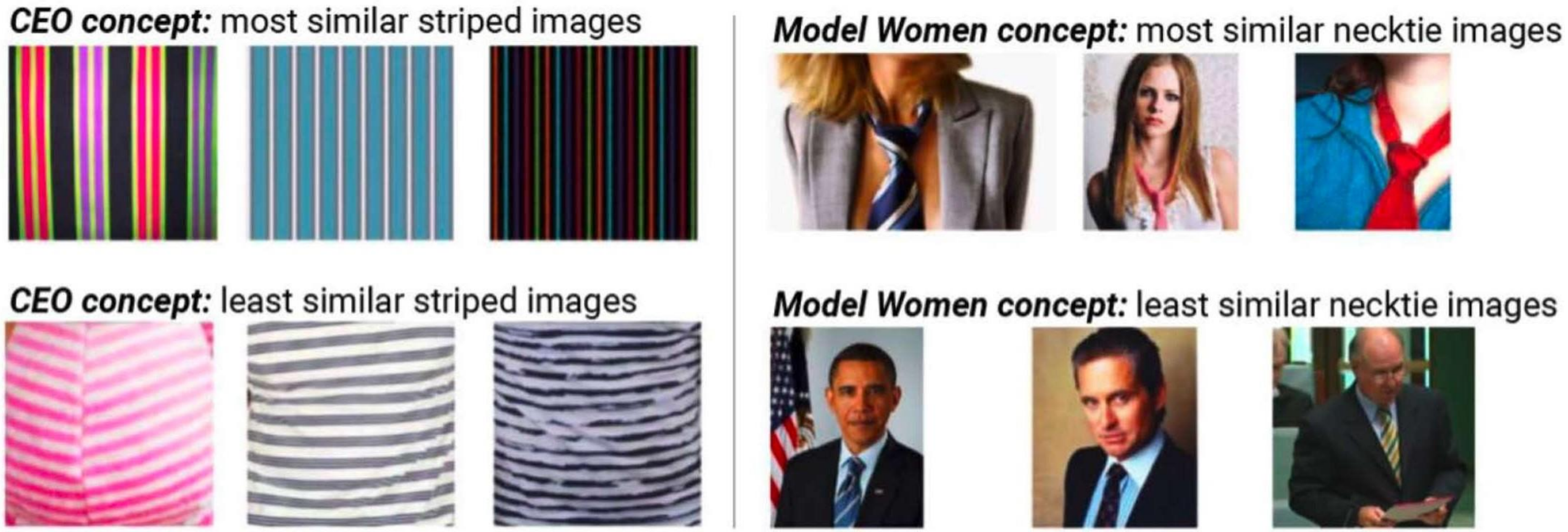


Train a **linear classifier** to separate the projection of the concepts from the random images

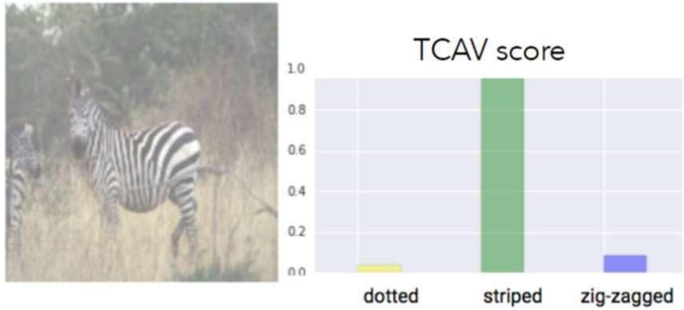
CAV (v_C^l) is the vector **orthogonal** to the decision boundary

Sorting Images with CAVs

- Given a set of images (e.g., belonging to the same class)
- Compute the cosine similarity between
  - the latent representation of an image $f_l(x)$
  - the CAV v_C^l of the selected concept



TCAV: (2) How to compute TCAV scores?



zebra-ness $\longrightarrow$ $\partial p(z)$

striped CAV $\longrightarrow$ $\partial \boldsymbol{v}_C^l$

$$\frac{\partial p(z)}{\partial \boldsymbol{v}_C^l} = S_{C,k,l}(\boldsymbol{x})$$

$$\begin{aligned} S_{C,k,l}(\boldsymbol{x}) &= \lim_{\epsilon \to 0} \frac{h_{l,k}(f_l(\boldsymbol{x}) + \epsilon \boldsymbol{v}_C^l) - h_{l,k}(f_l(\boldsymbol{x}))}{\epsilon} \\ &= \nabla h_{l,k}(f_l(\boldsymbol{x})) \cdot \boldsymbol{v}_C^l, \end{aligned} \tag{1}$$

Directional derivative with CAV:

- $S_{C,k,l}(x) > 0$: *positive influence*
- $S_{C,k,l}(x) < 0$: *negative influence*

$S_{C,k,l}($  $)$

$S_{C,k,l}($  $)$

$S_{C,k,l}($  $)$

$S_{C,k,l}($  $)$

$$\boxed{\mathrm{TCAV}_{\mathrm{Q}_{C,k,l}} = \frac{|\{\boldsymbol{x} \in X_k : S_{C,k,l}(\boldsymbol{x}) > 0\}|}{|X_k|}}$$

The TCAV score is the number of class samples having a positive directional derivative w.r.t. the CAV

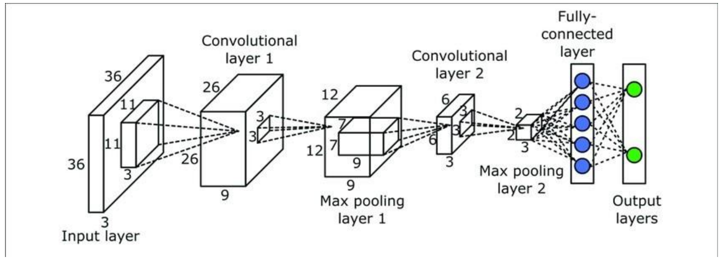
TCAV score characteristcs

- $TCAV_{C,k,l} \in [0, 1]$
  - $TCAV_{C,k,l} > 0.5$: *positive* influence $TCAV_{C,k,l} < 0.5$: *negative* influence
  - Of concept C
  - Over class k
  - Computed in layer l

TCAV Example 1 (Zebra)

Given Image
(Zebra)

Trained ML model
(e.g. GoogleNet)



Predicted as
Zebra

$p(z)$

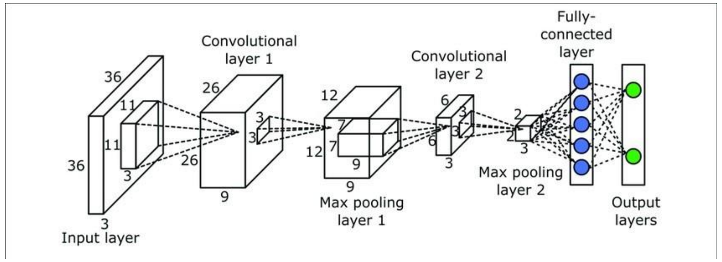
Was **Stripe concept** important to this **zebra** image classifier?

TCAV tells that **Stripe has a positive importance** for the classification of **zebras**

TCAV Example 2 (Doctor)

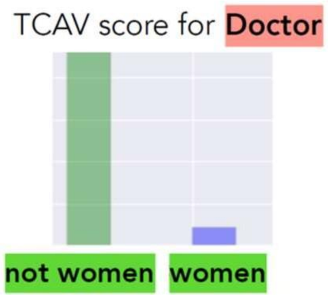
Given Image
(Doctors)

Trained ML model
(e.g. GoogleNet)



Predicted as
Doctor
$p(z)$

Was **Woman concept** important to this **doctor** image classifier?

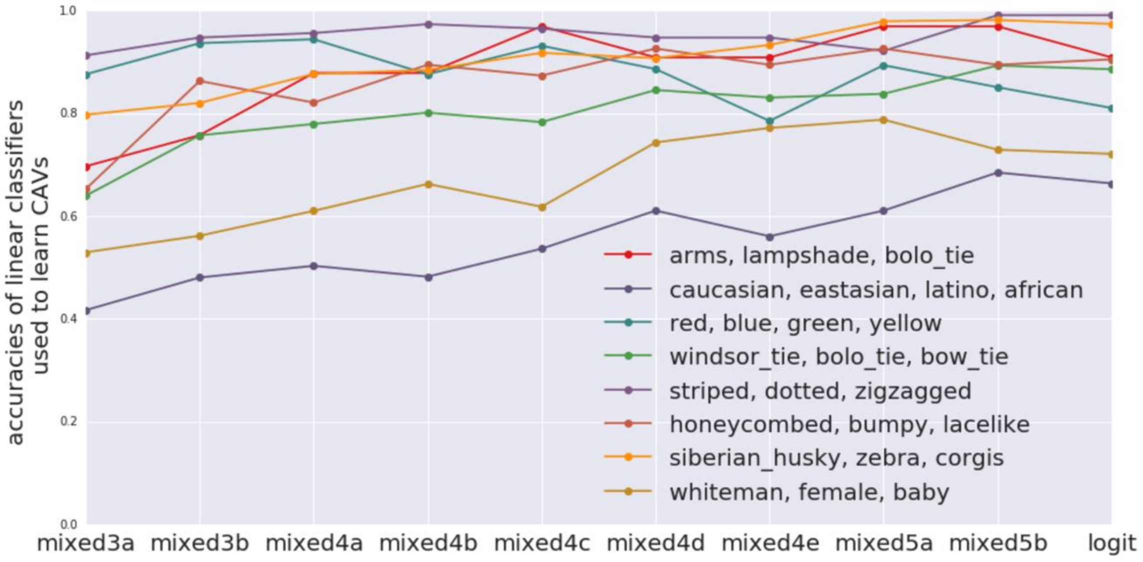


TCAV tells that **Woman has a negative importance** for the classification of doctors

BIAS IDENTIFICATION!

When and where can concept be learnt?

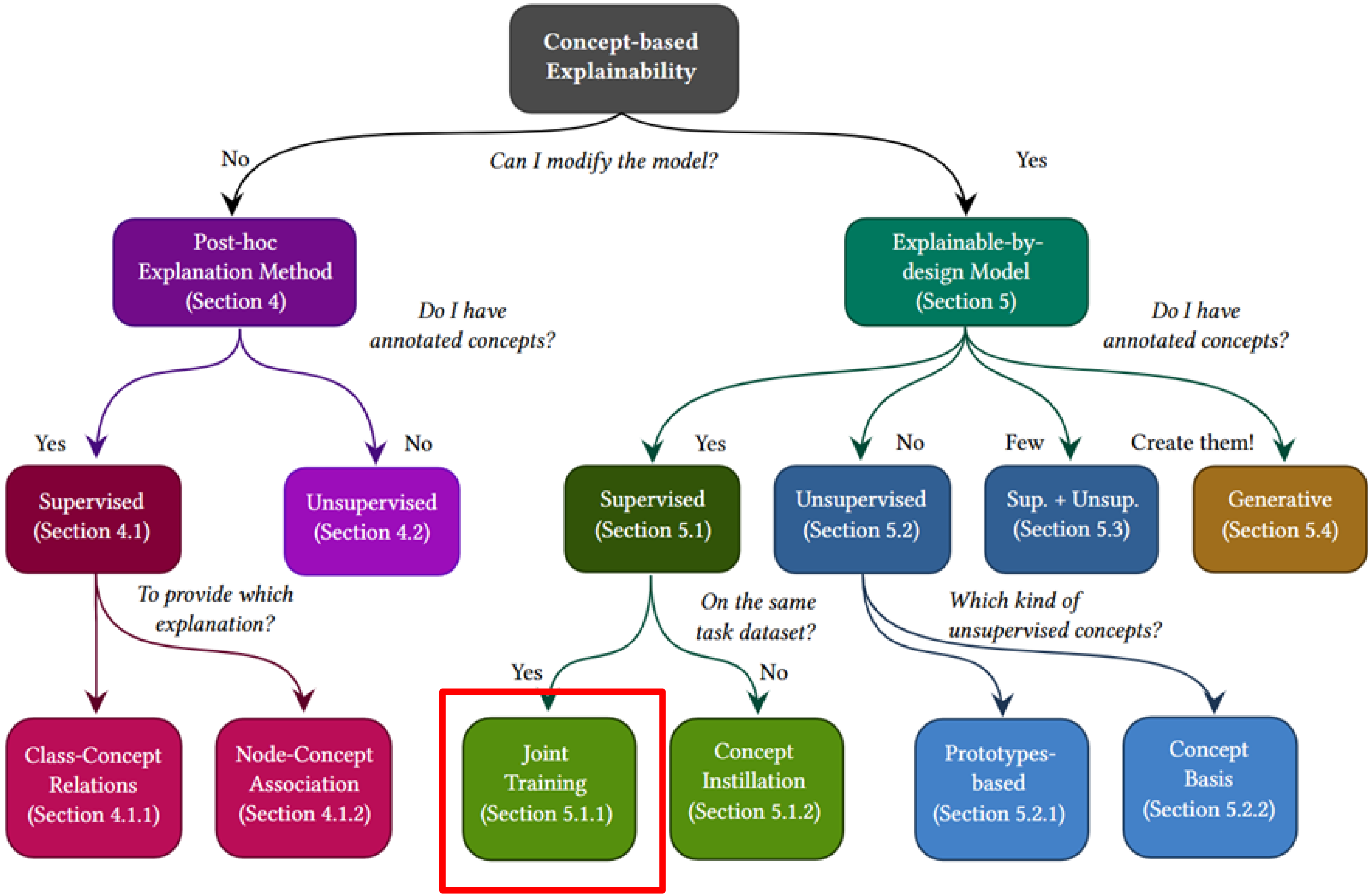
- Accuracy of the «linear probe»
  - *High* implies the network **has automatically learnt** a concept
  - *Low* implies the network **does not use** that concept for predicting the final class



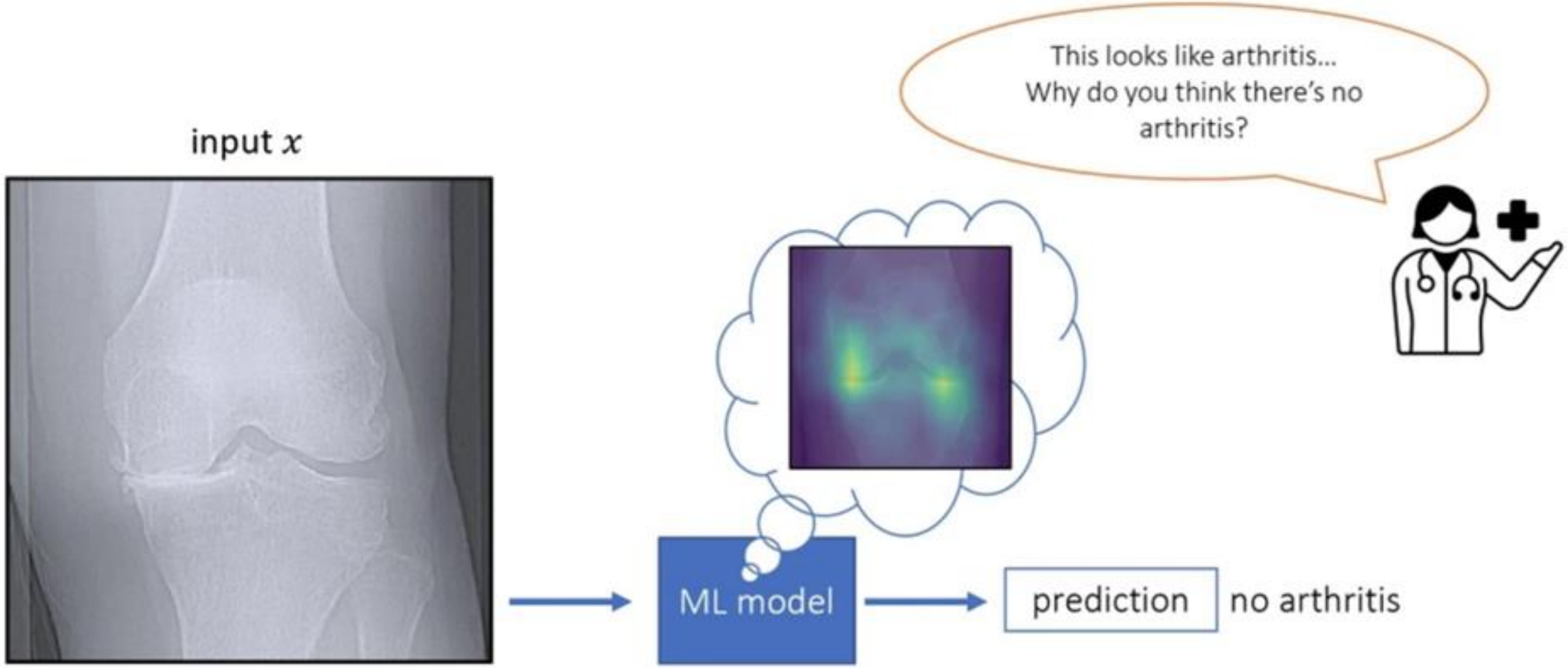
- Simpler concepts have high accuracy throughout the NN

- High-level concepts can be detected better at higher layers

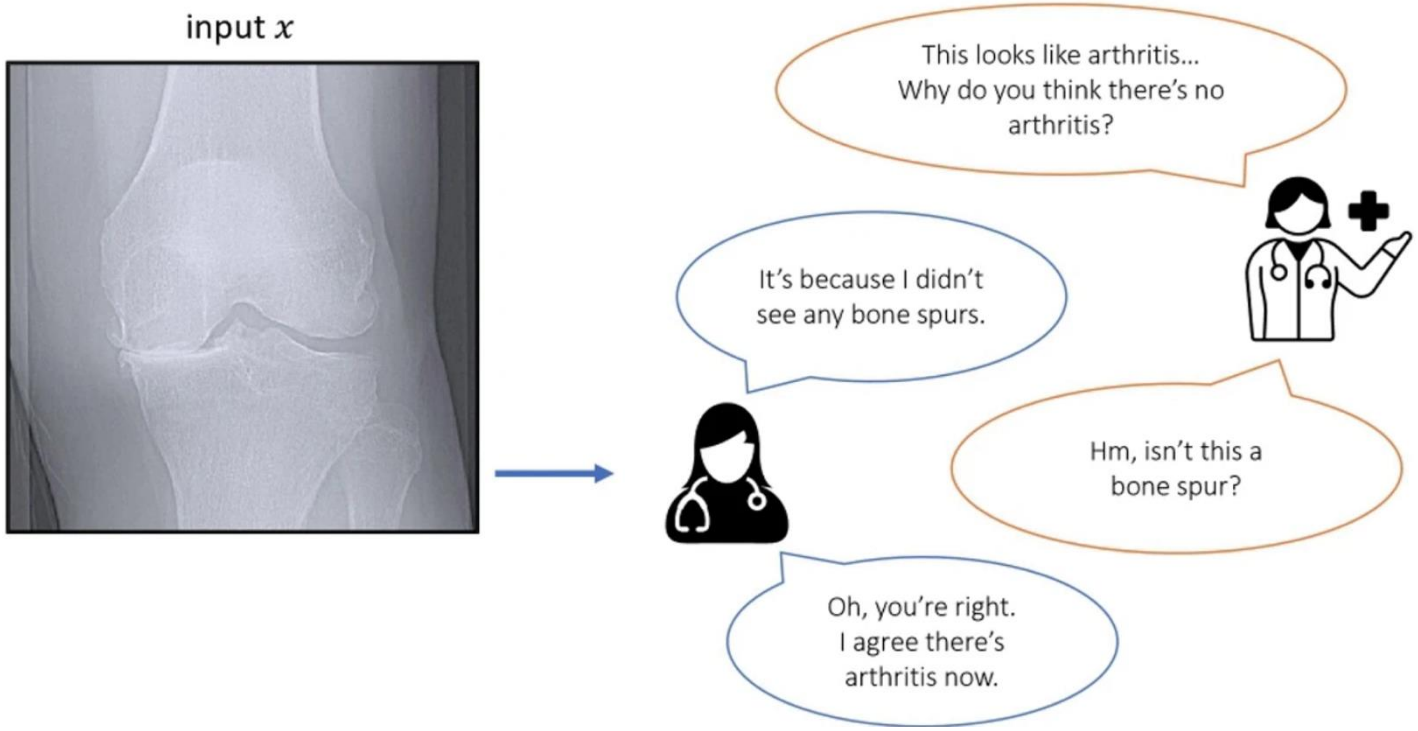
2. Concept Bottleneck Models (CBMs)


Concept-based Explainability
Can I modify the model?
No
Yes
Post-hoc Explanation Method (Section 4)
Explainable-by-design Model (Section 5)
Do I have annotated concepts?
Yes
No
Supervised (Section 4.1)
Unsupervised (Section 4.2)
Do I have annotated concepts?
Yes
No
Few
Create them!
Supervised (Section 5.1)
Unsupervised (Section 5.2)
Sup. + Unsup. (Section 5.3)
Generative (Section 5.4)
To provide which explanation?
Class-Concept Relations (Section 4.1.1)
Node-Concept Association (Section 4.1.2)
On the same task dataset?
Yes
No
Joint Training (Section 5.1.1)
Concept Instillation (Section 5.1.2)
Which kind of unsupervised concepts?
Prototypes-based (Section 5.2.1)
Concept Basis (Section 5.2.2)

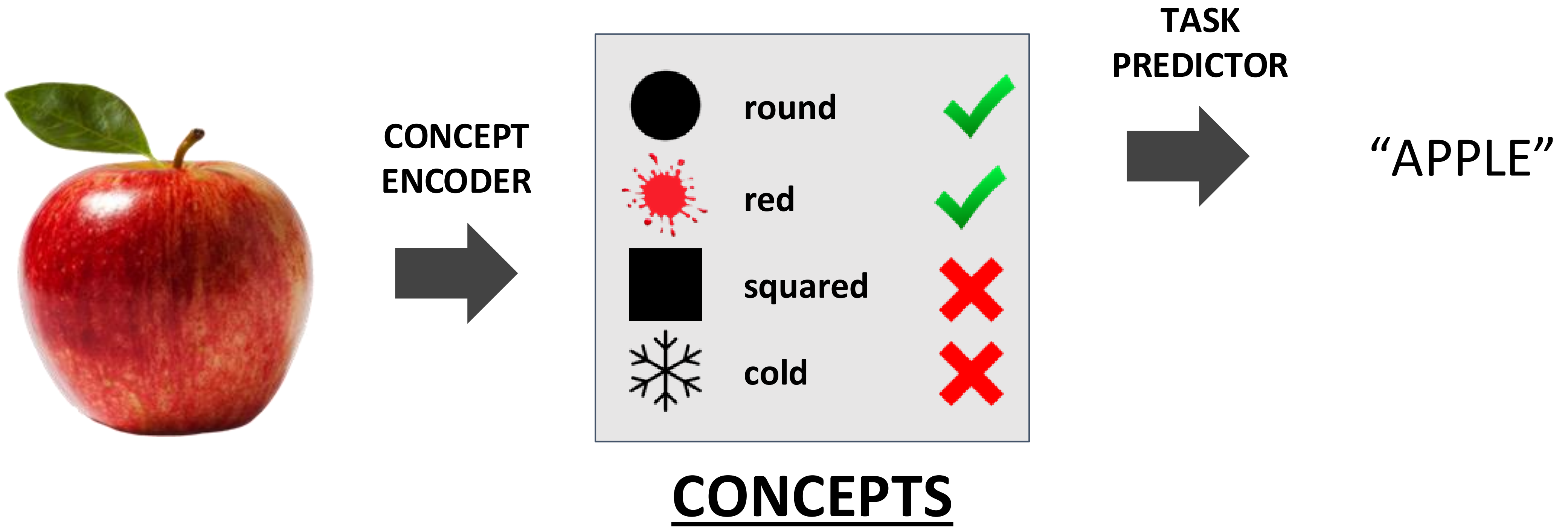
End-2-End models are difficult to interact with



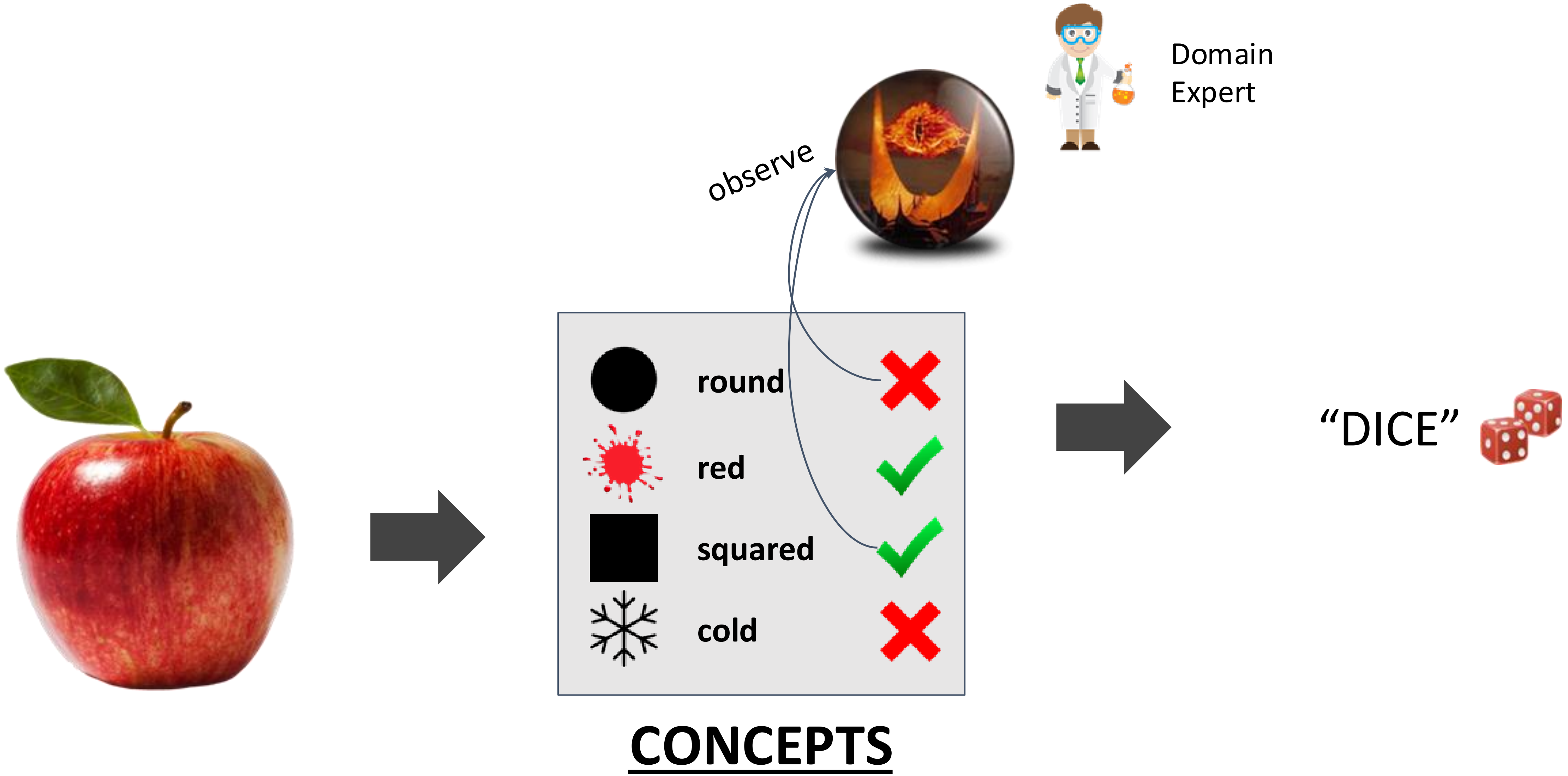
Ideal: Interact through high-level concepts



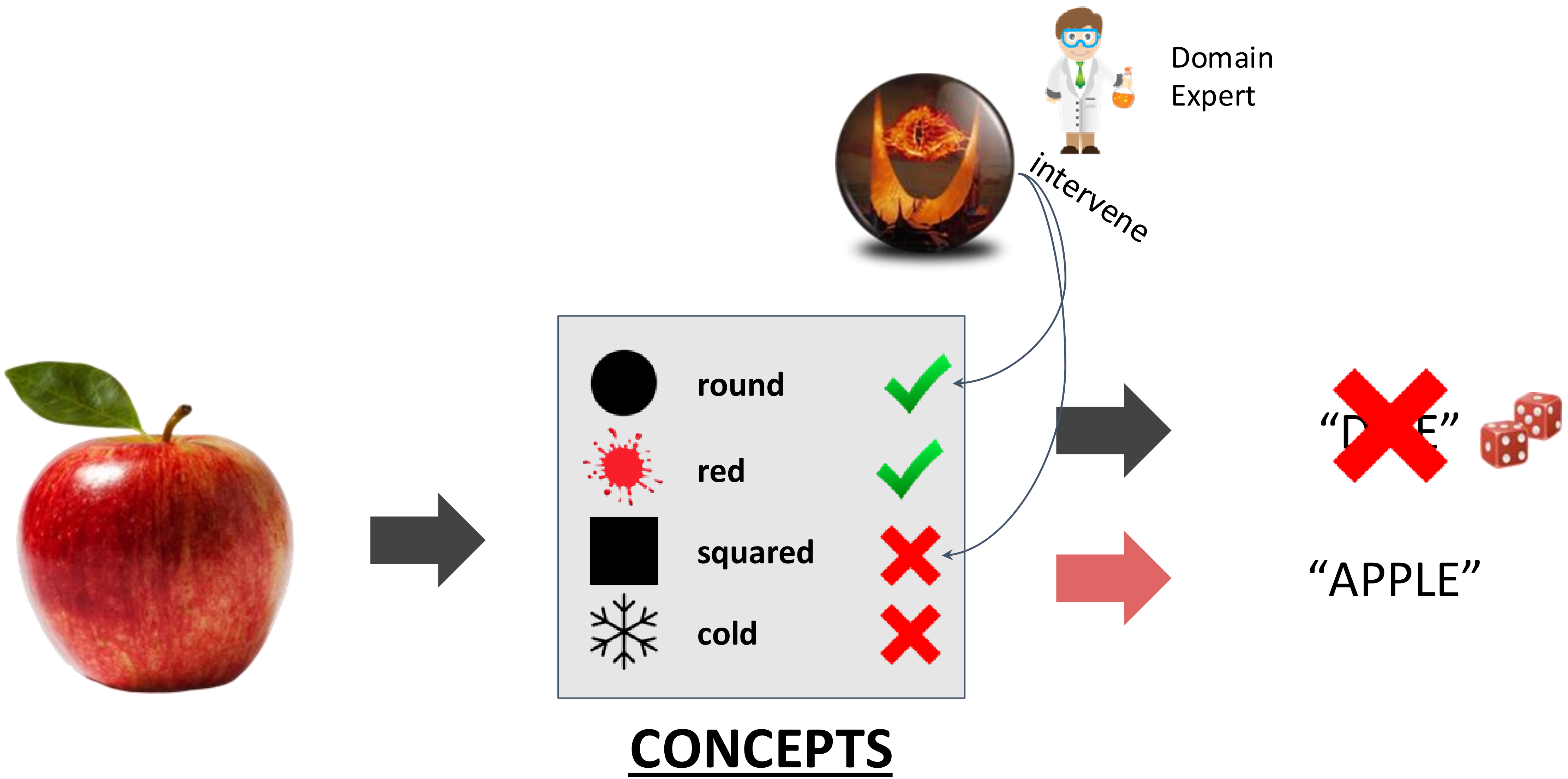
CBMs Explicitly Represents Concepts



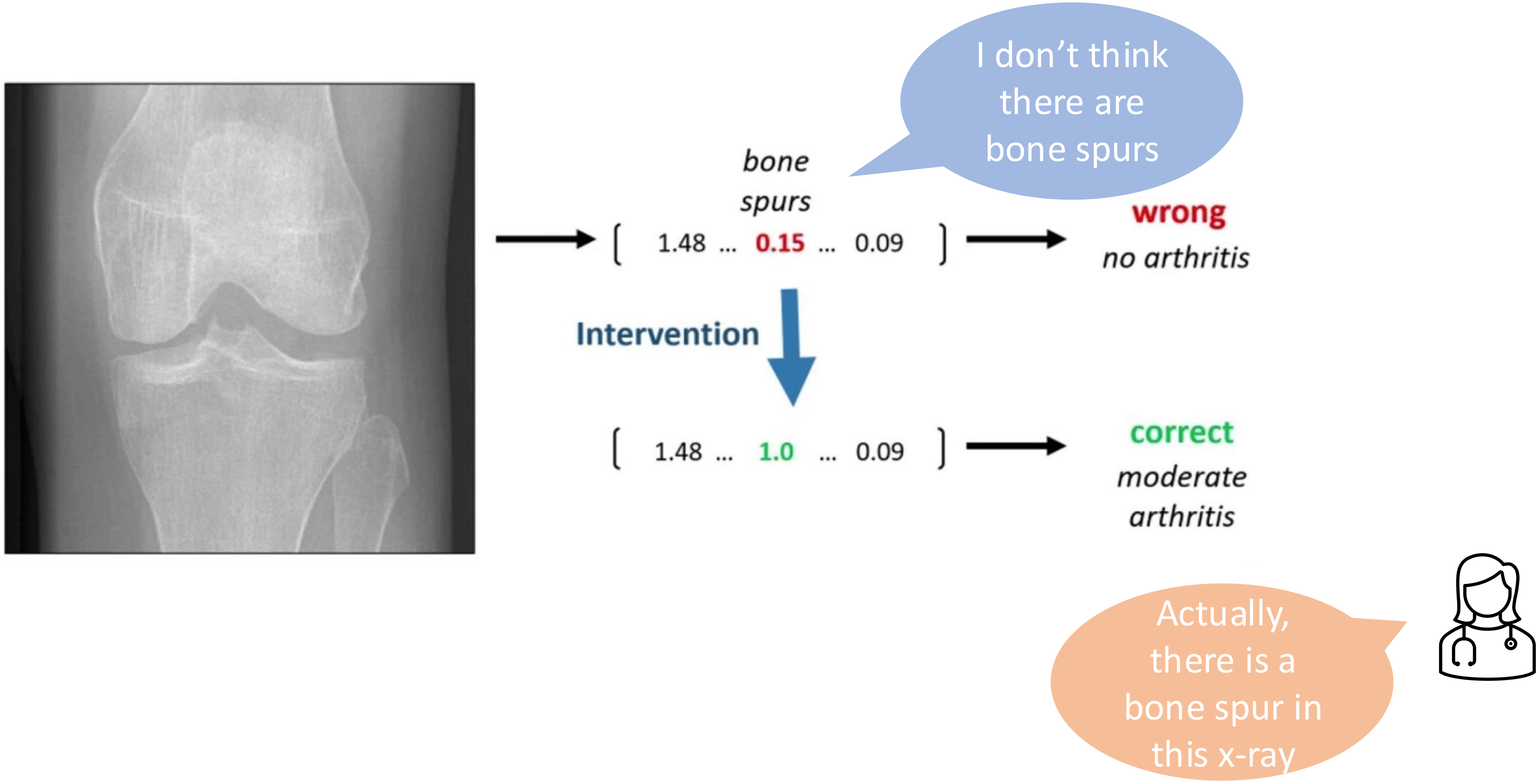
CBMs Allows Interactions!



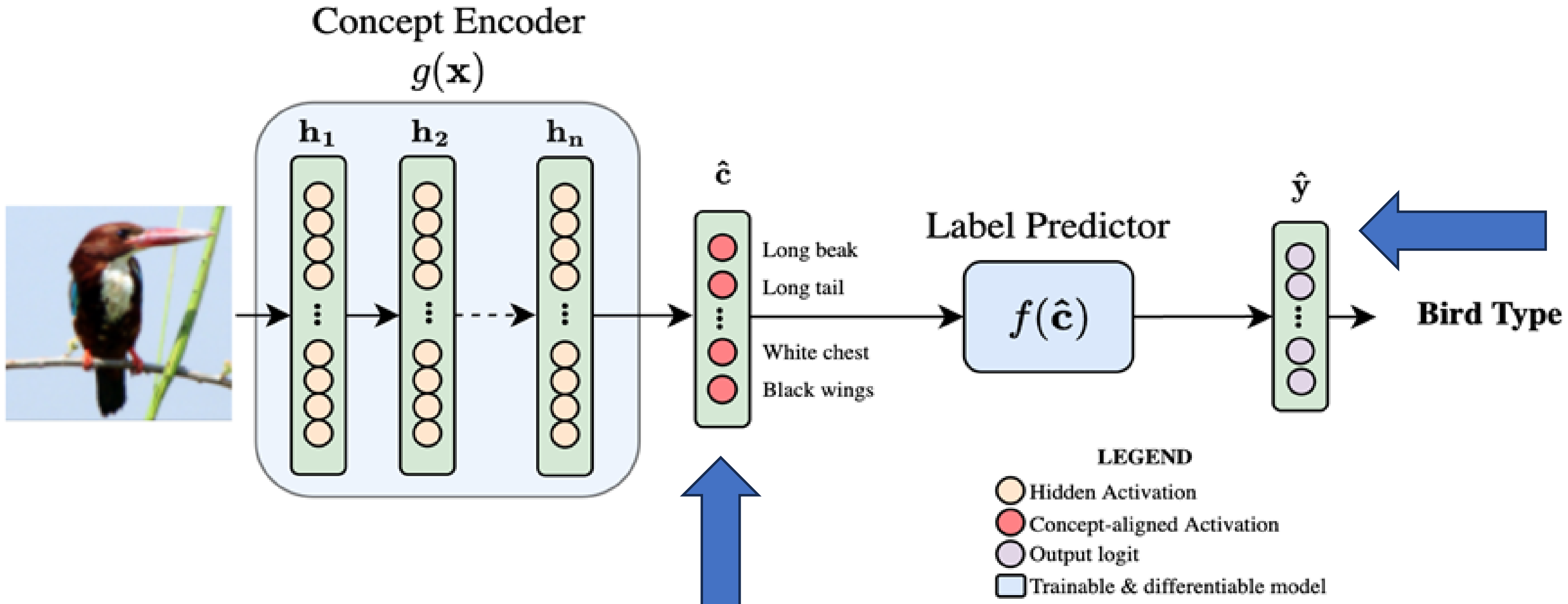
CBMs Allows Interactions!



Importance of Concept Intervention



Concept bottleneck models architecture



Task loss:
$L_y(f(c_i); y_i)$

Concept loss: $L_c(g(x_i); c_i)$

Different training strategy

- Indipendent: $\hat{f} = \arg min_{\mathrm{f}} \sum_i L_y(f(c_i), y_i)$

  $\hat{g} = \arg \min_{g} \sum_i L_c(g(x_i), c_i)$

  f is trained using the truth concepts

- Sequential: $\hat{f} = \arg min_f \sum_i L_y\left(f\left(g(x_i)\right), y_i\right)$

  g is trained first as above, then freezed

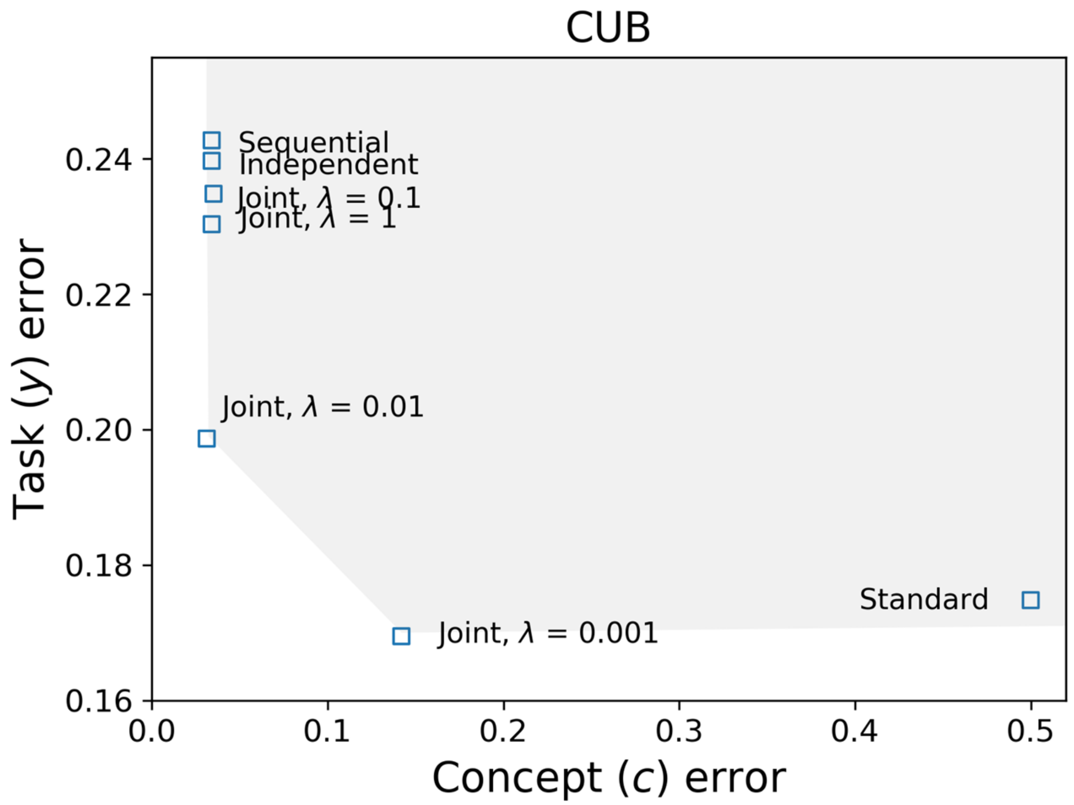
- Joint: $\hat{f}, \hat{g} = \arg min_{\mathrm{f}} \sum_i L_y(f(c_i), y_i) + \lambda \arg min_g \sum_i L_c(g(x_i), c_i)$

  f,g trained together for some λ > 0

- Standard: $\hat{f}, \hat{g} = \arg min_{\mathrm{f}} \sum_i L_y(f(c_i), y_i)$

  It ignores the concepts loss

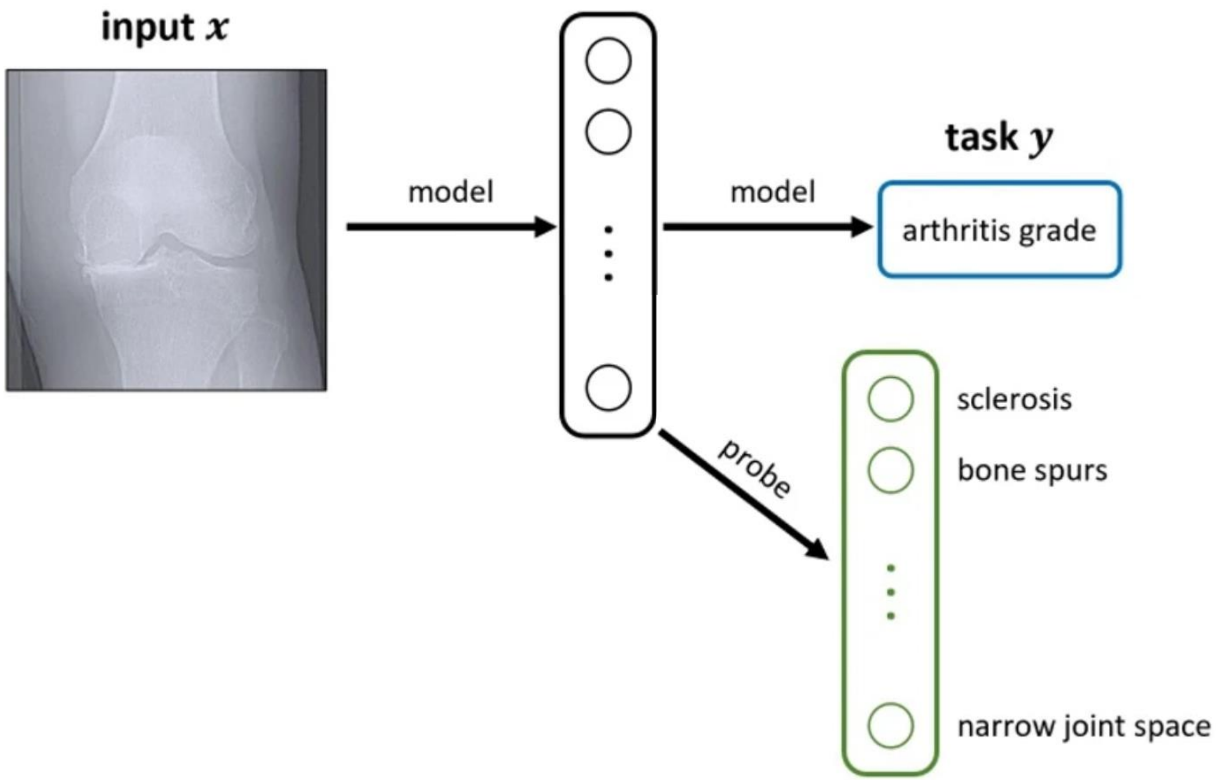
Different interpretability/performance trade-offs



- **Sequential** and **indipendent** are the more «trustworthy» beacause they ensure no concept leakage
- **Joint** strategy provides better task accuracy
  - Different trade-offs according to the λ value
- **Standard** model still has higher accuracy on average

Explictly concept training ensure model learns the concepts

Standard E2E trained model



Method	X-Ray Concept Error (↓)
Independent	0.53
Sequential	0.53
Joint	0.54
TCAV [Probe]	0.68

In a trained model, identifying some concepts may not be possible, because it might not have learnt them automatically

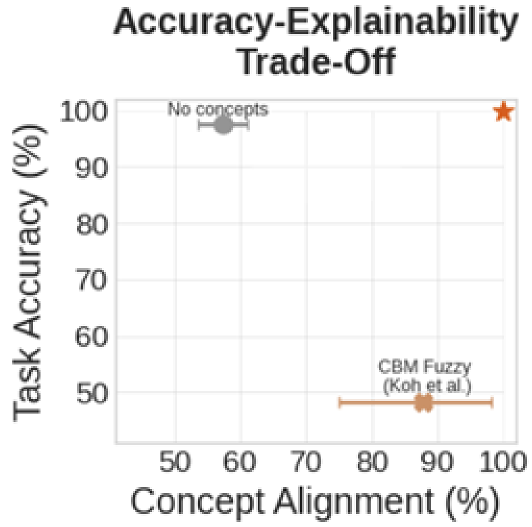
→ Only by explicitly training a model we can ensure it represents all concepts!

CBM Drawbacks

Poor Trade-offs

—

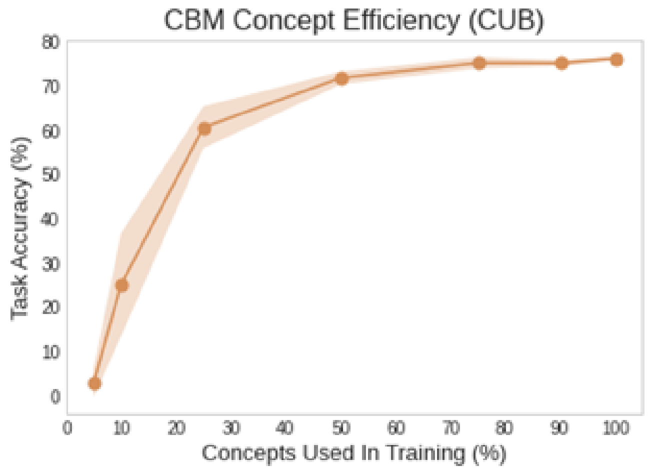
Struggle to compromise between accuracy and explainability



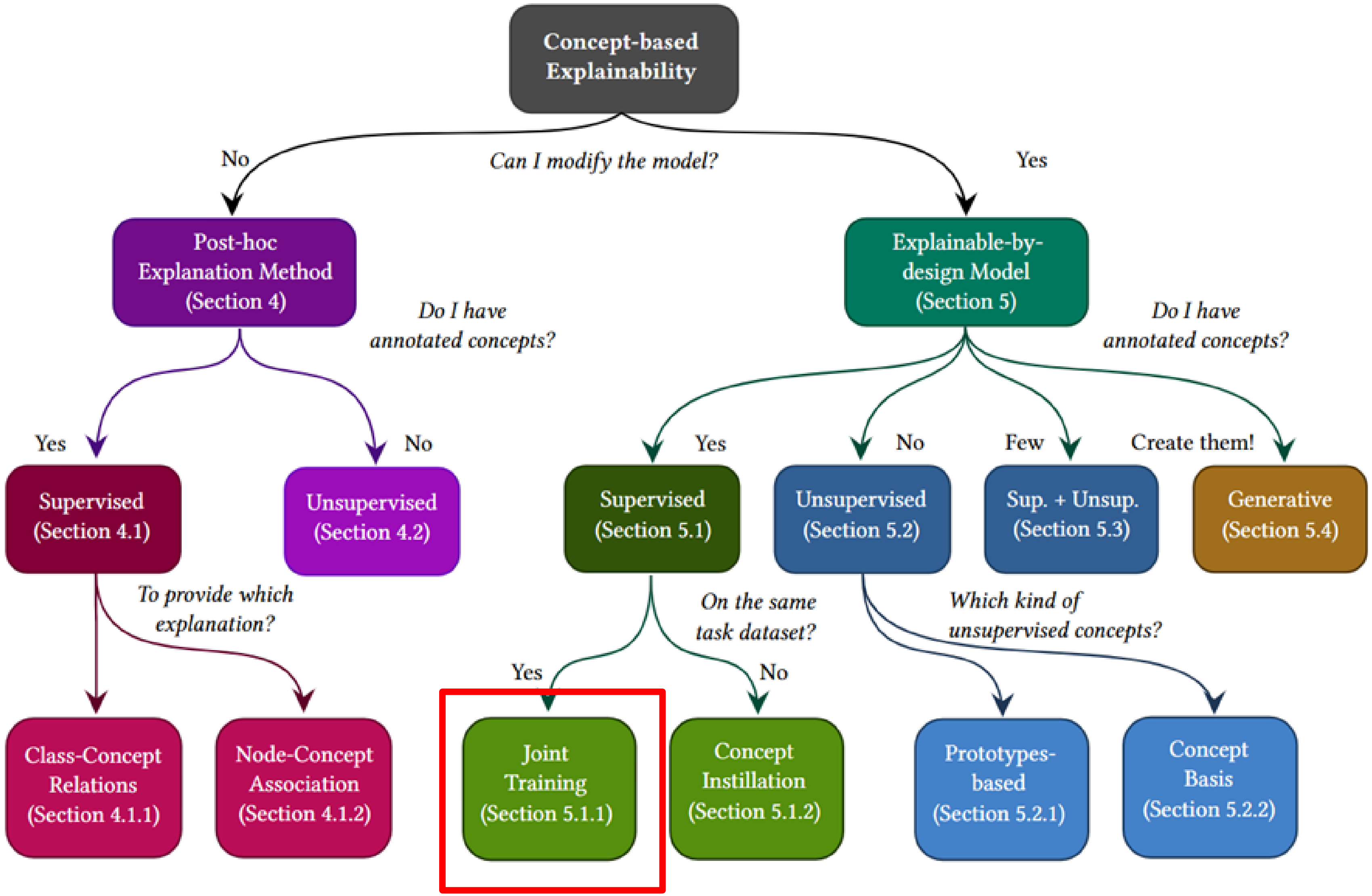
Low Concept Efficiency

—

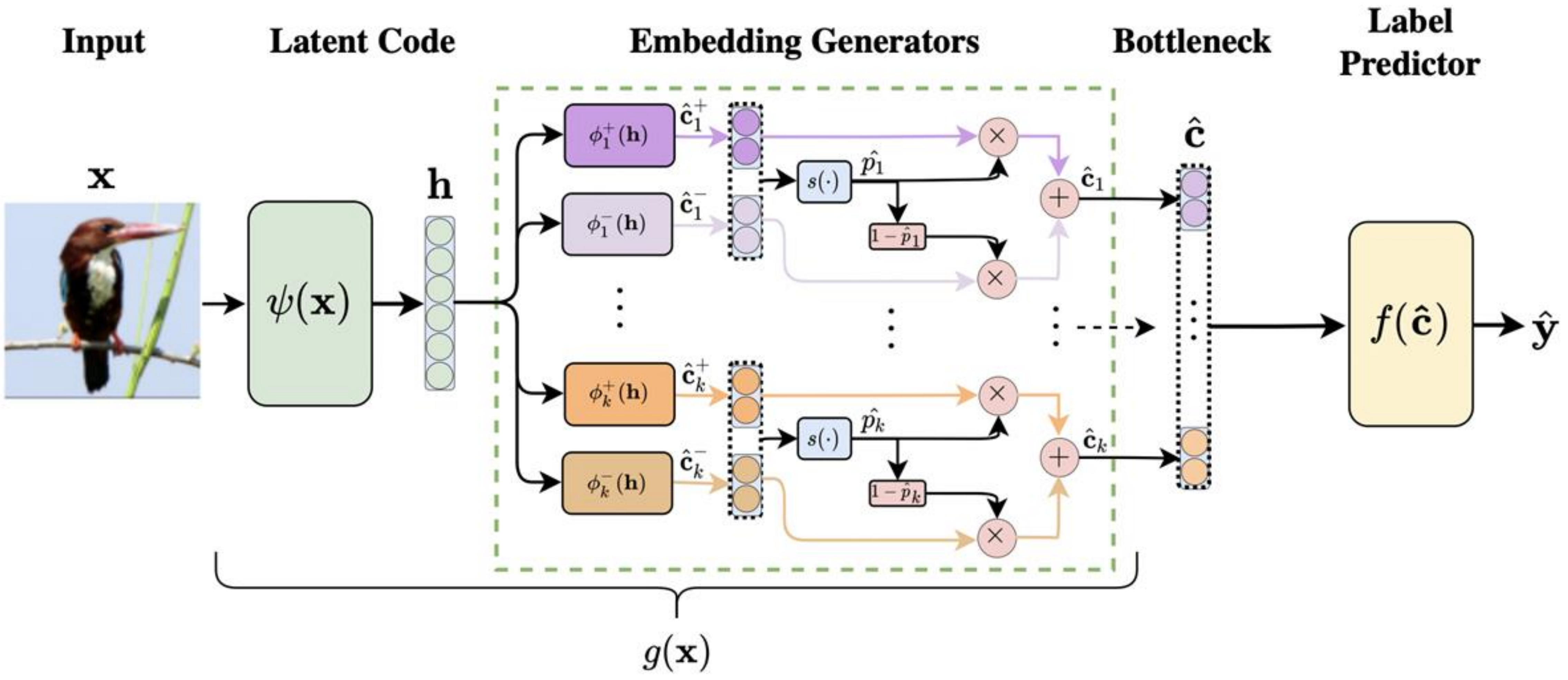
CBMs do not scale in real-world conditions



3. Concept Embedding Models (CEM)


Concept-based Explainability
Can I modify the model?
No
Yes
Post-hoc Explanation Method (Section 4)
Explainable-by-design Model (Section 5)
Do I have annotated concepts?
Do I have annotated concepts?
Yes
No
Yes
No
Few
Create them!
Supervised (Section 4.1)
Unsupervised (Section 4.2)
Supervised (Section 5.1)
Unsupervised (Section 5.2)
Sup. + Unsup. (Section 5.3)
Generative (Section 5.4)
To provide which explanation?
On the same task dataset?
Which kind of unsupervised concepts?
Yes
No
Class-Concept Relations (Section 4.1.1)
Node-Concept Association (Section 4.1.2)
Joint Training (Section 5.1.1)
Concept Instillation (Section 5.1.2)
Prototypes-based (Section 5.2.1)
Concept Basis (Section 5.2.2)

Concept Embedding Models: overview



Concept Embedding workflow

1. $h = \psi(x)$: the latent space of the model

2. $\boldsymbol{c}_{\boldsymbol{i}}^{+} = \phi_i^{+}(x)$: neural model dedicated to represent the i-th **positive** concept embedding

3. $\mathrm{p}_i = s([\boldsymbol{c}_{\boldsymbol{i}}^{+}, \boldsymbol{c}_{\boldsymbol{i}}^{-}])$: the *concept score* (i.e., probability of presence of the ith concept) is a function shared among concepts working on the concatenations of the concept embeddings

4. $\hat{\boldsymbol{c}}_1 = \mathrm{p_i}\mathbf{c}_{\mathbf{i}}^{+} + (1 - \mathrm{p_i})\mathbf{c}_{\mathbf{i}}^{-}$: the *concept embedding* is represented by the weigthed combination of the positive and negative concept embeddings according to its presence

5. $f([\hat{\boldsymbol{c}}_1, \ldots \hat{\boldsymbol{c}}_i, \ldots \hat{\boldsymbol{c}}_k])$: the task predictor works on the concatenation of all the concept embeddings

CEM: A neural-symbolic approach

Neural

—

Concepts are represented with: unsupervised **embeddings**

$$c_i \in \mathbb{R}^k$$

Symbolic (CBM)

—

Concepts are represented with: **supervised** scalars

$$c_i \in [0,1]$$

Neural Symbolic (CEM)

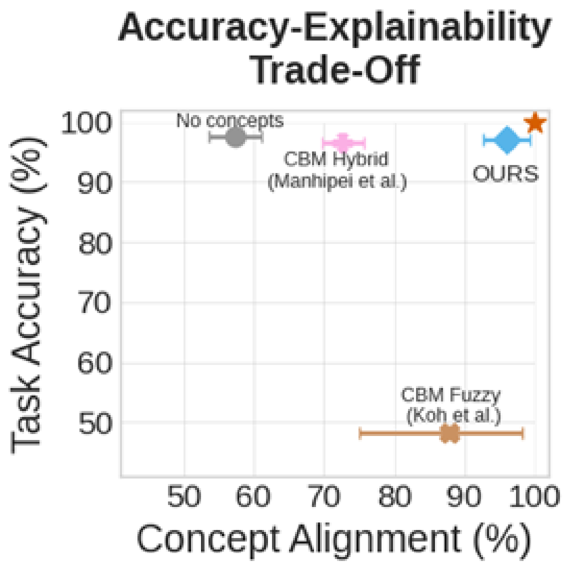
—

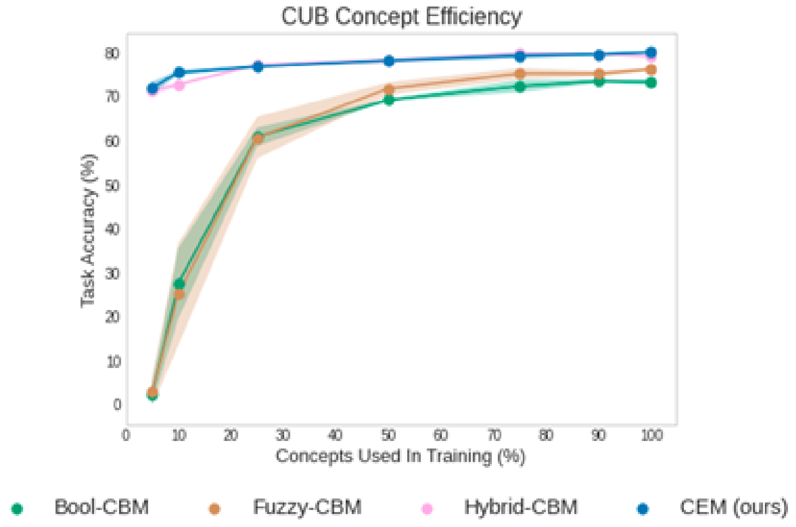
Concepts are represented with: pairs of **supervised embeddings**

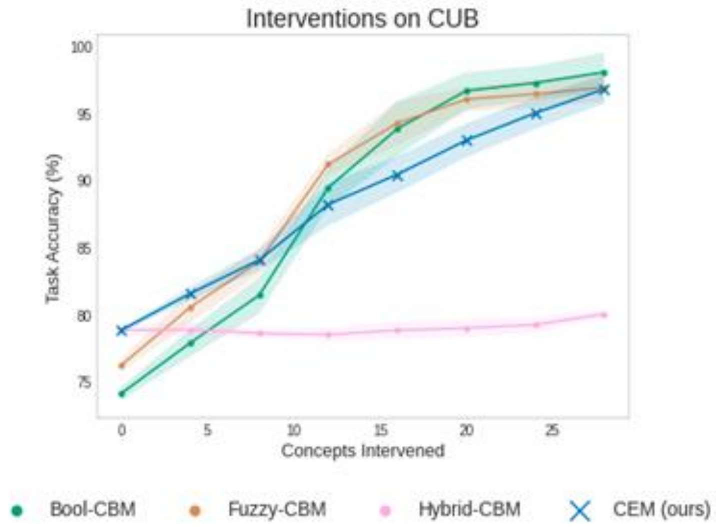
$$c_i \in \mathbb{R}^k$$

$$c_i = agg(c_i^+, c_i^-)$$

CEM Advanatages







Beyond Trade-offs

—

CEMs overcome the current accuracy-explainability trade-off

High Concept Efficiency

—

CEMs scale to real-world conditions where concept supervisions are scarce

Effective interventions

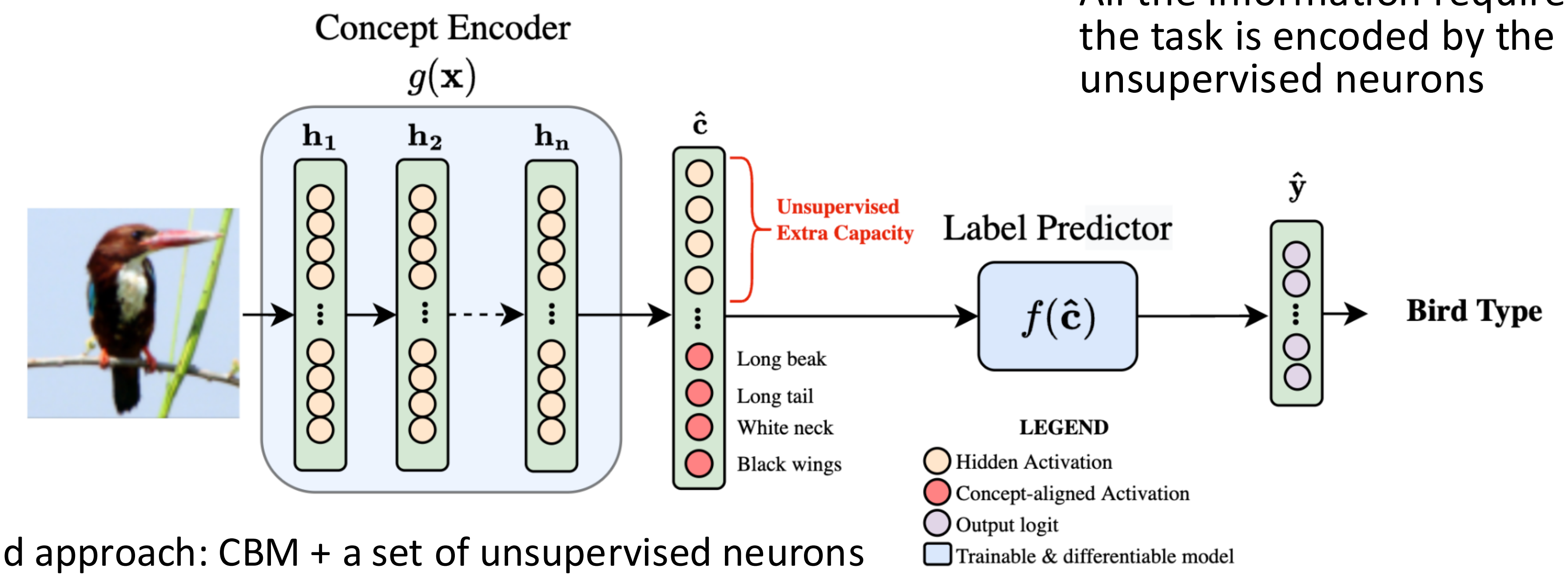
—

CEMs are responsive to concept interventions

CEM vs Hybrid approach

- PROS:
  - Retain high accuracy
  - Has high concept efficiency like CEM

- CONS:
  - Prevent any effect of concept intervention
    - Changing the predicted scores has no effect on the task prediction
  - All the information required to predict the task is encoded by the unsupervised neurons

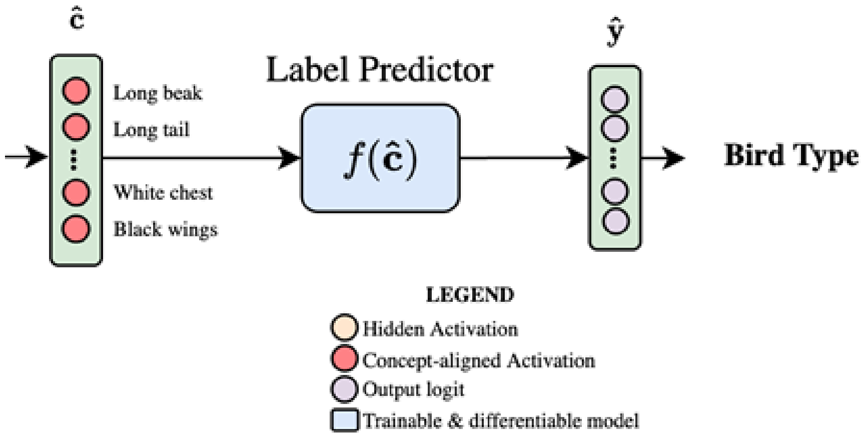


Hybrid approach: CBM + a set of unsupervised neurons

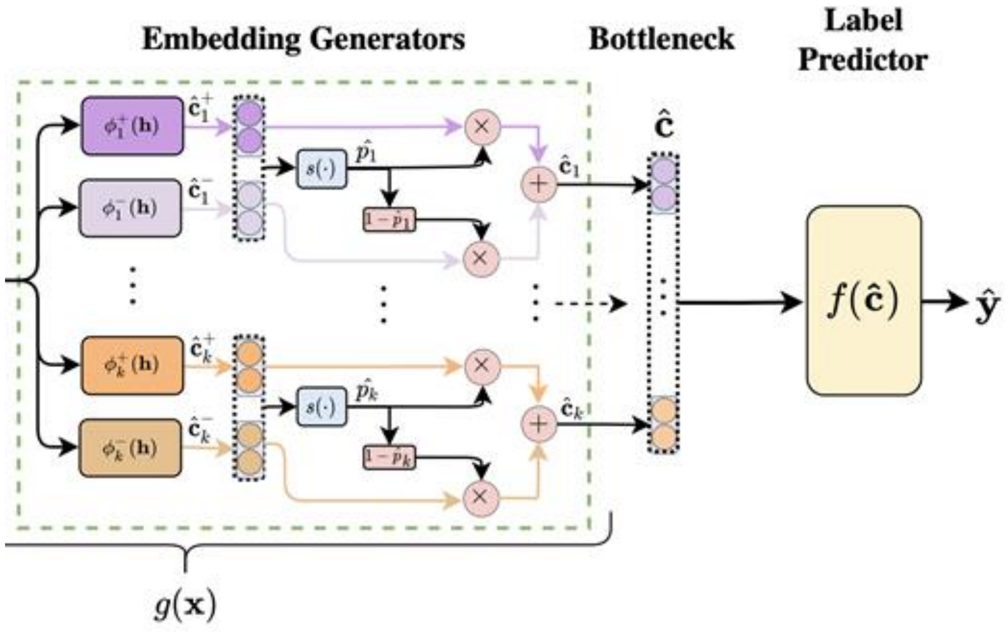
Have we lost something?

Interpretability

CBM: Interpretable

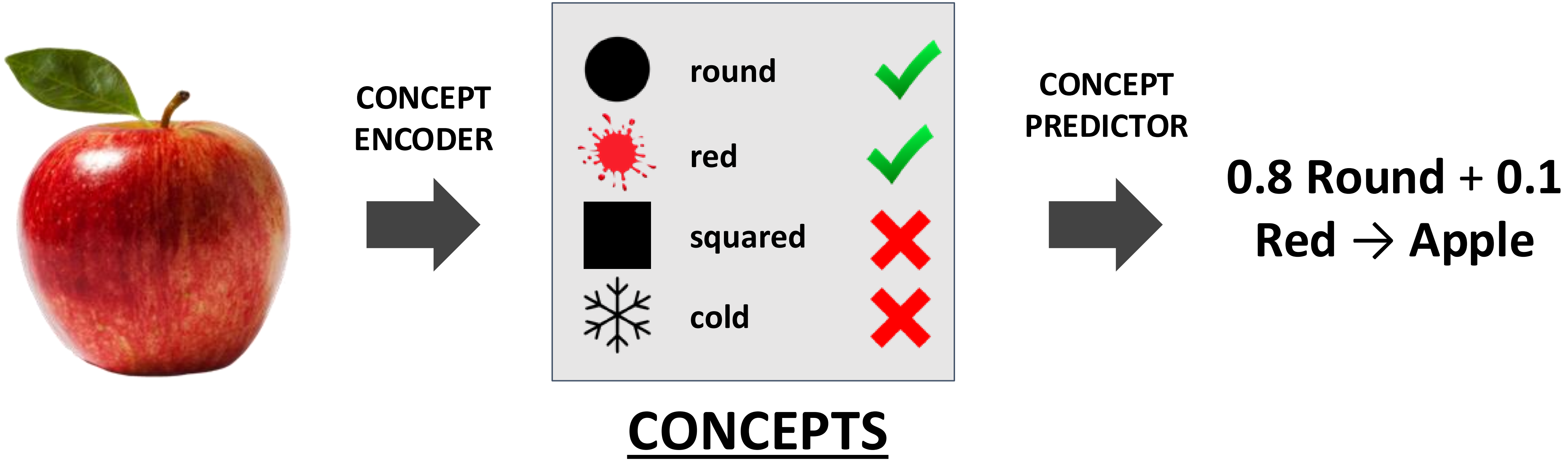


CEM: NON-Interpretable



$$\hat{\mathbf{c}}_{\text{yellow}} = [2.3, 0.3, -3.5, \dots]^T$$

Can we create an Interpretable Model over Concept Embeddings?



Come tomorrow (Friday 24/04) to the Project Presentation!

- You will form groups of about 4 people
- We will provide 8-10 different projects among which you will have to choose

- The remaining of the lecture we will do:
  - A laboratory on C-XAI
  - A guided laboratory on XAI for Text Data with Prof. Eliana Pastor