# Concept Bottleneck Models

Alberto TONDA, Ph.D. (Senior permanent researcher, DR)

*UMR 518 MIA-PS, INRAE, AgroParisTech, Université Paris-Saclay*
*UAR 3611, Institut des Systèmes Complexes de Paris Île-de-France*

# Outline

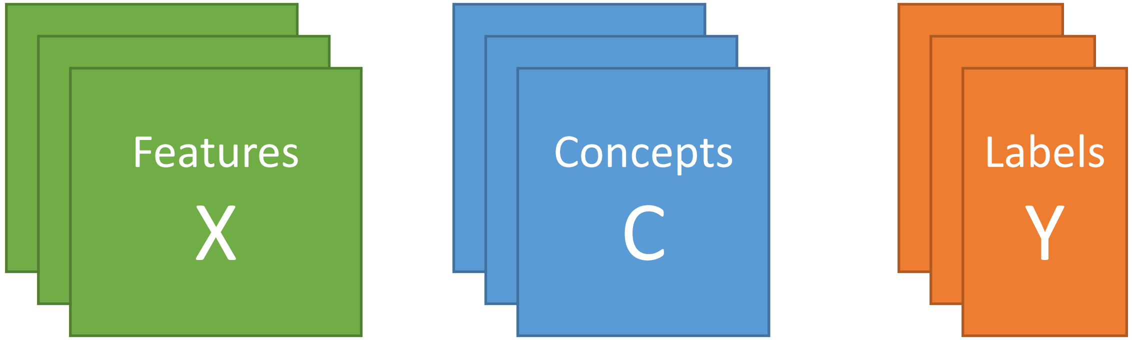
- Neural-symbolic approaches
- Concept bottleneck models
- Concept embedding models
- Deep concept reasoner
- Experimental evaluation
- Conclusions

# Neural-Symbolic approaches

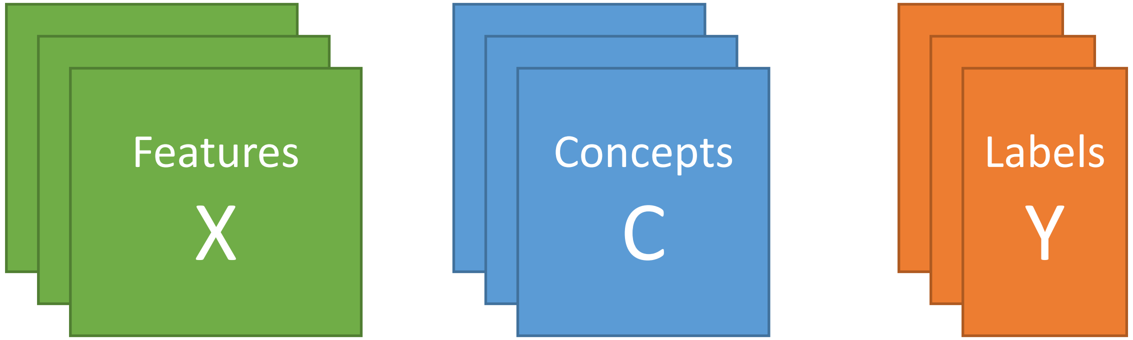
- Neural-Symbolic (NeSy) combines ML/DL and Symbolic AI
  - Even if it is called “neural”, also ML
  - It’s a field, lots of different methods
  - Idea of combining **efficacy** (ML) with **interpretability** (symbolic AI)
  - Gaining in popularity as DL is showing limits (maybe)

- *What* to do is clear, *how* to do it, is difficult
  - ML/DL and Symbolic AI don’t mix very well
  - Maybe embeddings could help bridge **features** and **symbols**

# Concept bottleneck models



# Concept bottleneck models

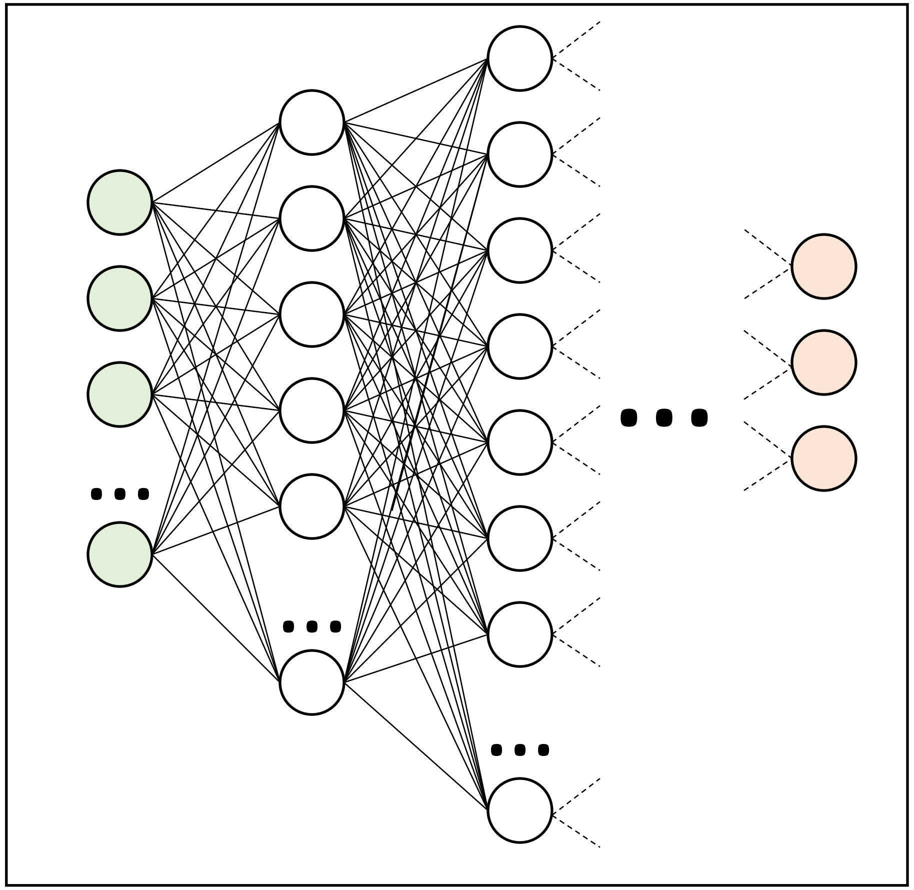



Laysan Albatross
Rusty Blackbird
Fish Crow
Brewer Blackbird
Shiny Cowbird
...

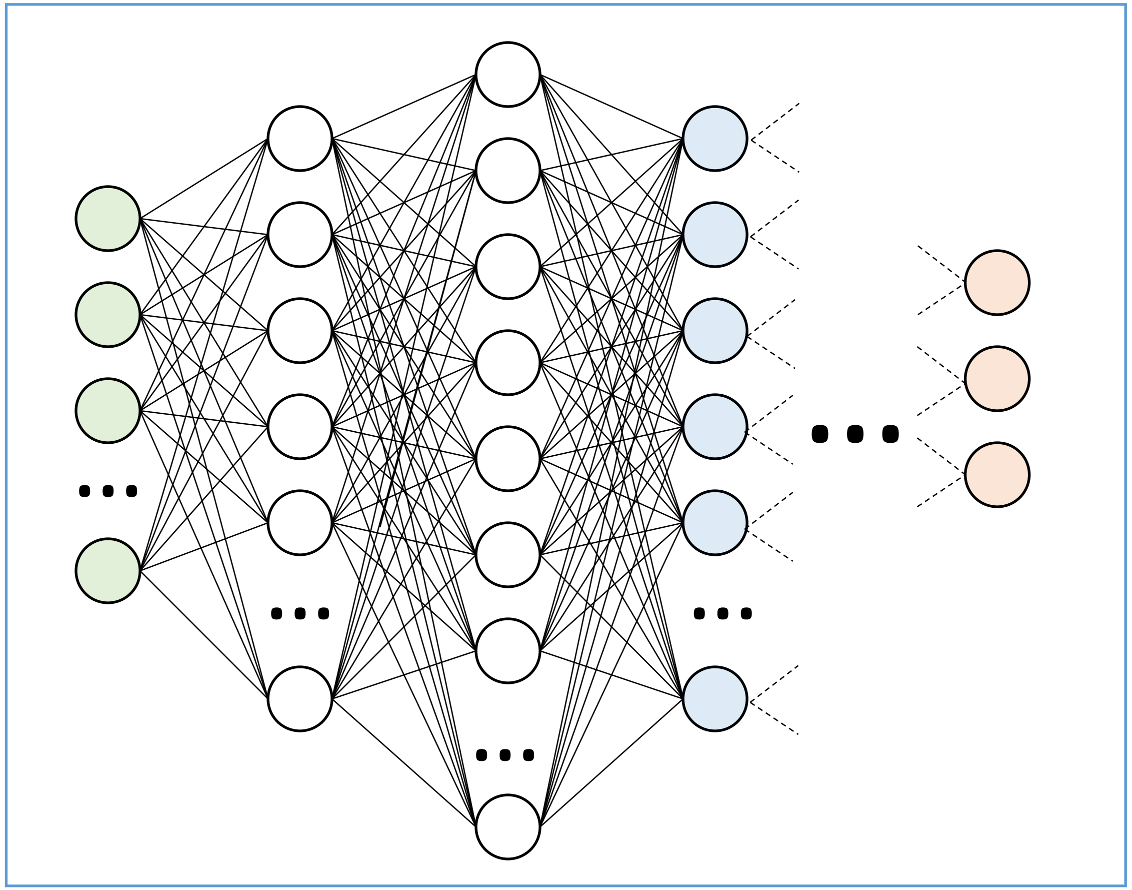
White_Wings: yes/no
Curved_Beak: yes/no
Yellow_Beak: yes/no
Black_Body: yes/no
...

# Concept bottleneck models

Classic Artificial Neural Network/Deep Learning



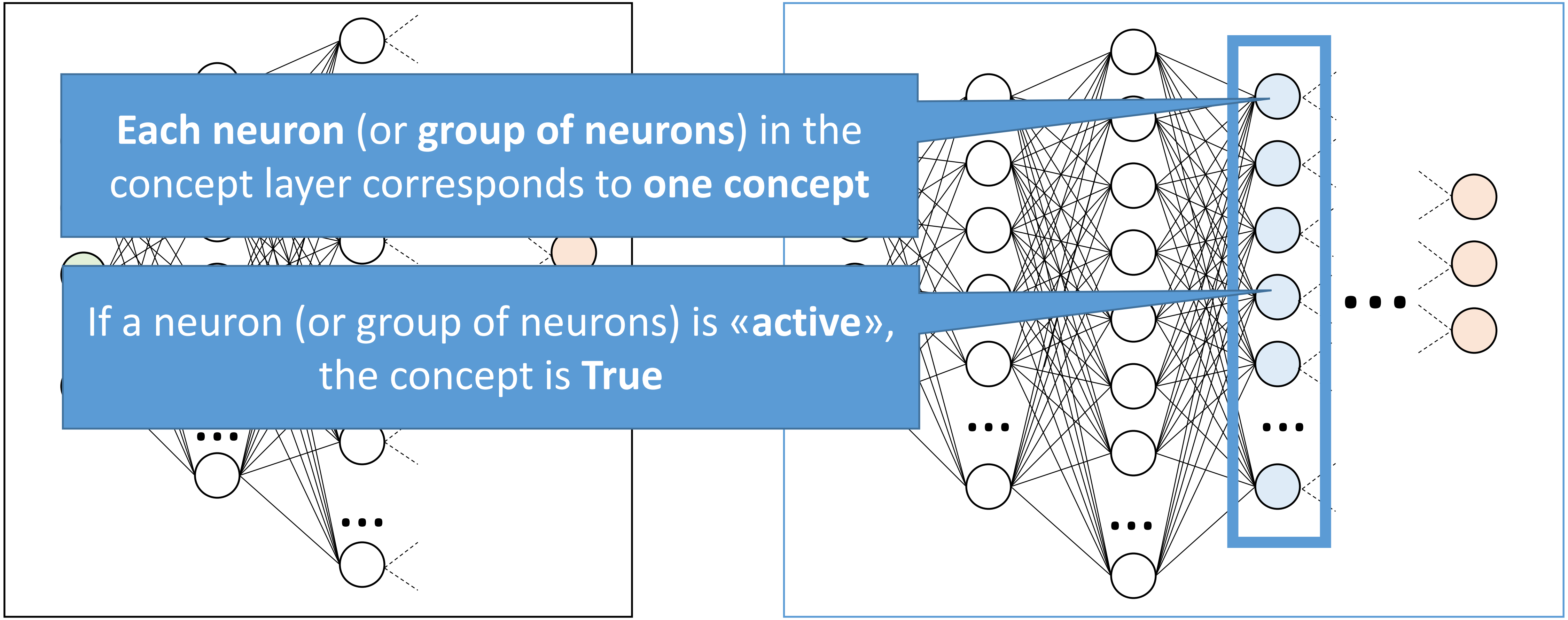
ANN/DL model with a concept bottleneck



# Concept bottleneck models

Classic Artificial Neural Network/Deep Learning

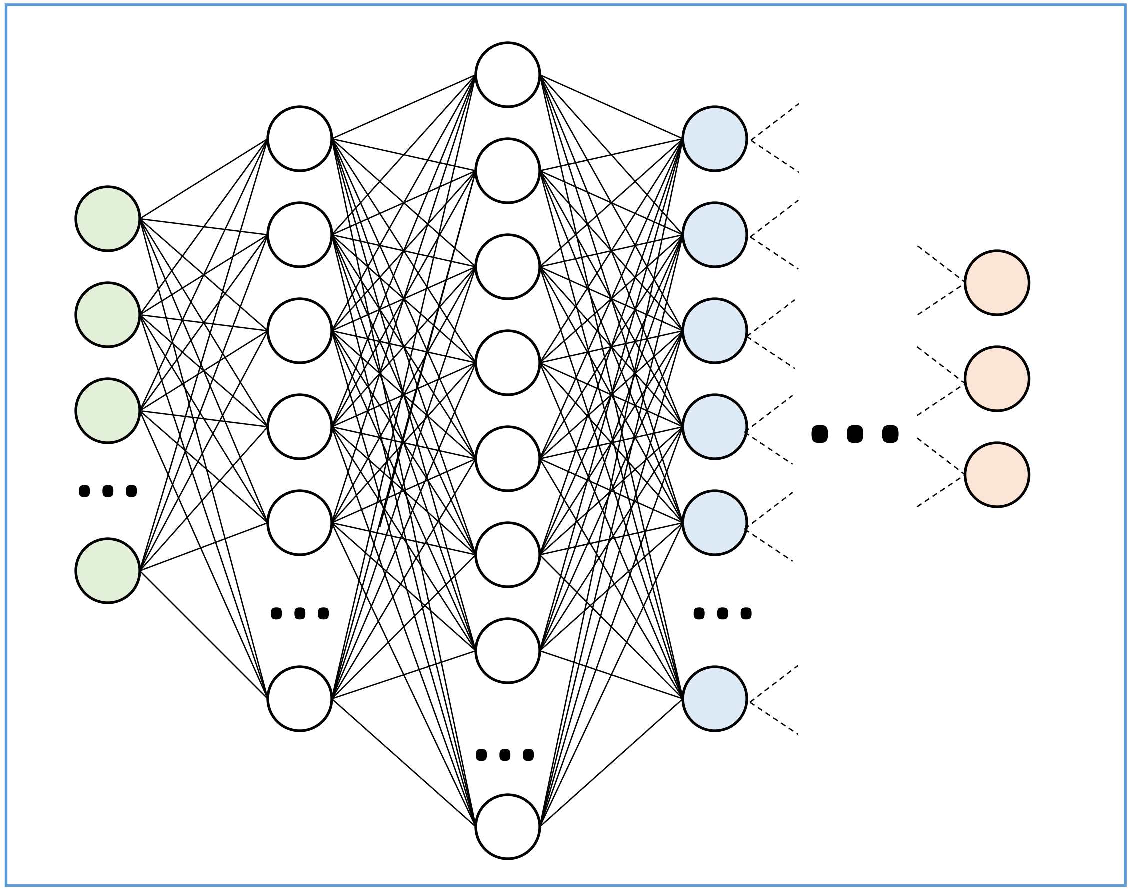
ANN/DL model with a concept bottleneck



# Concept bottleneck models

- Concepts for **explanations**
- «I think this is a Laysan Albatross because I detected white wings (White_Wings=True), black body (Black_Body=True), the beak is not curved (Curved_Beak=False) ...»

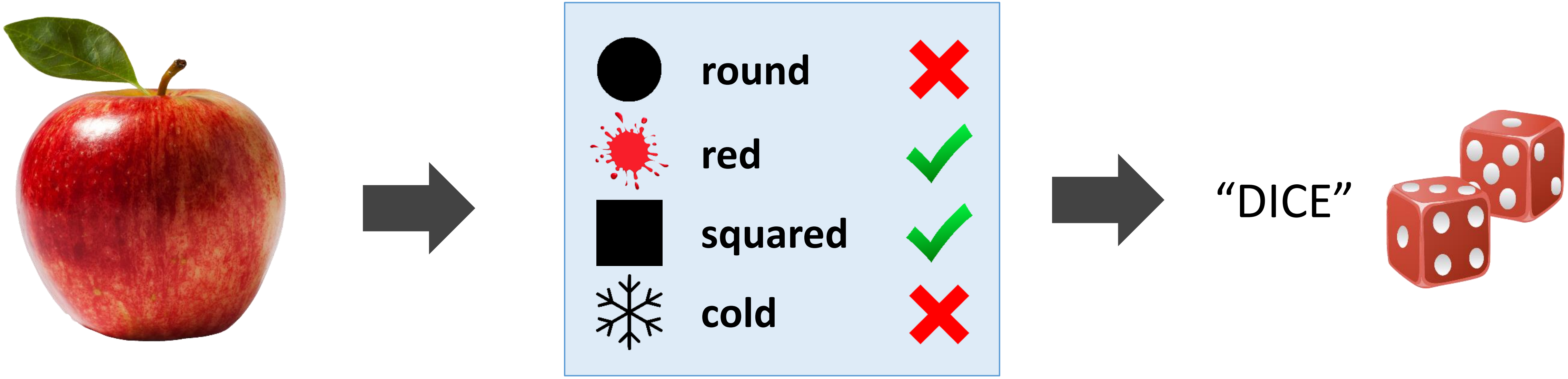
ANN/DL model with a concept bottleneck



# Concept bottleneck models

- Concepts for **interventions**

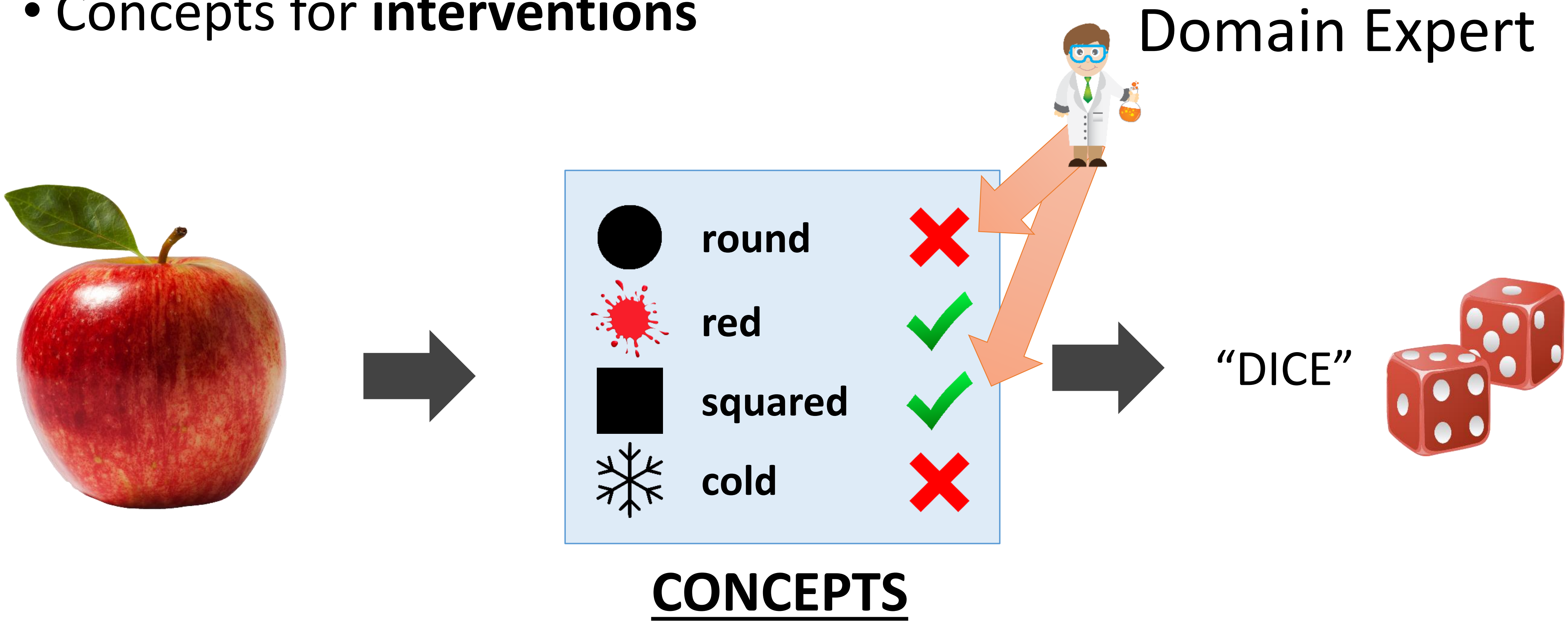
Domain Expert




CONCEPTS

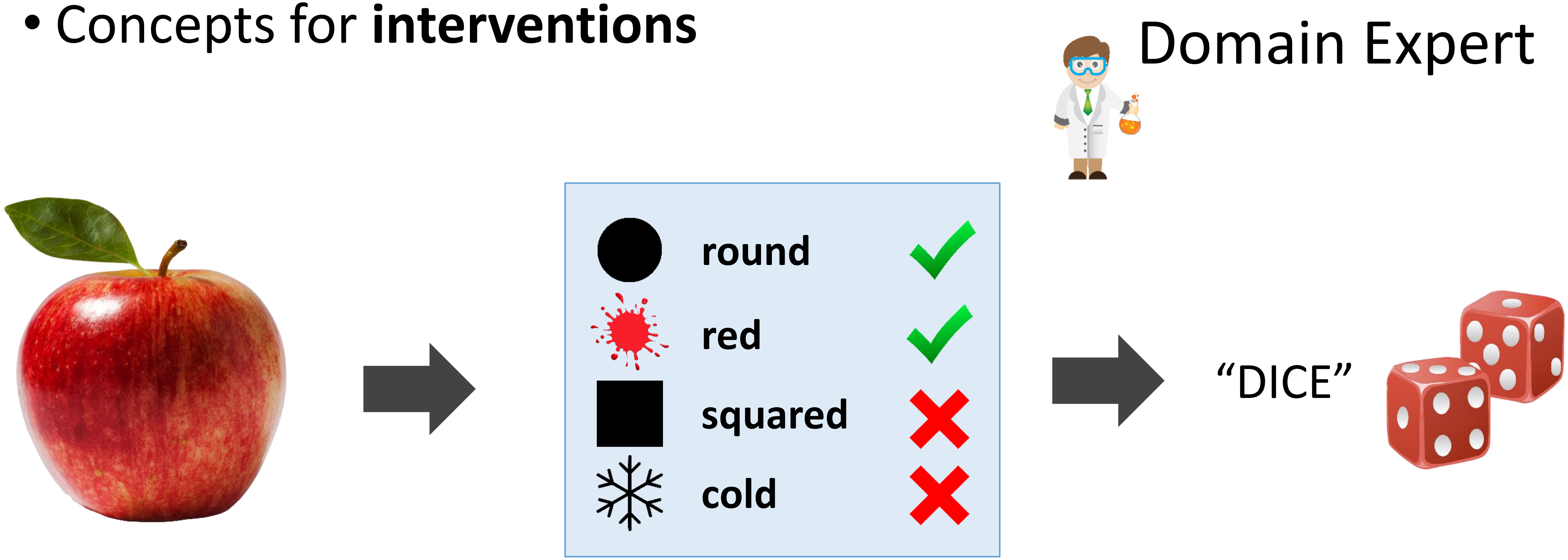
# Concept bottleneck models

- Concepts for **interventions**



# Concept bottleneck models

- Concepts for **interventions**




**CONCEPTS**

# Concept bottleneck models

- Concepts for **interventions**

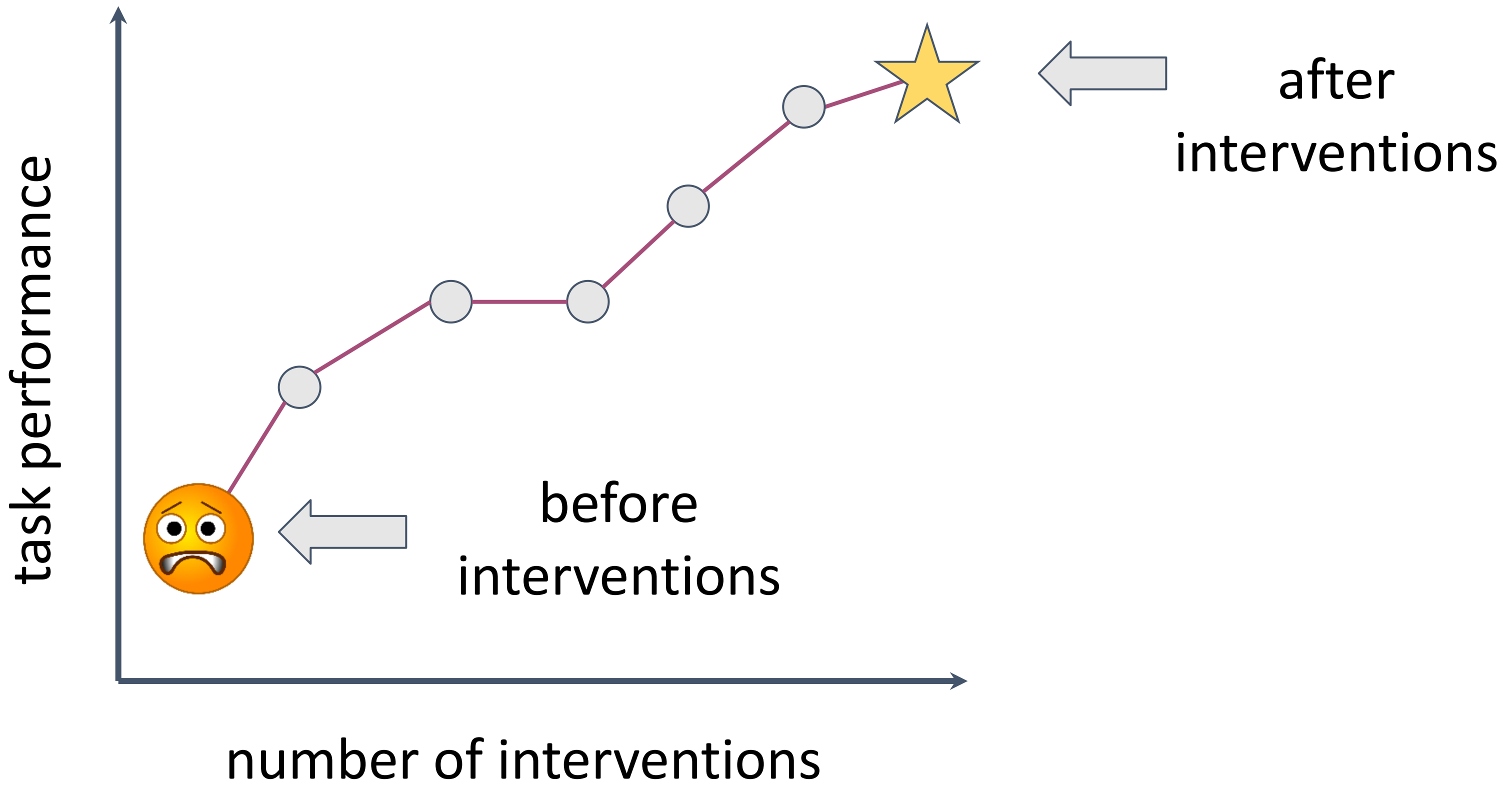
Domain Expert

| | | |
|---|---|---|
| ● | **round** | ✔ |
| | **red** | ✔ |
| ■ | **squared** | ✘ |
| ❄ | **cold** | ✘ |

**CONCEPTS**

"APPLE" ✔

# Concept bottleneck models



# Concept bottleneck models

- How to train concept bottleneck models in practice?
  - Modify the loss function to take into account concepts
  - Force neurons to be «active» when the concept is True
  - In training, access to «real» values of concepts; in test no need

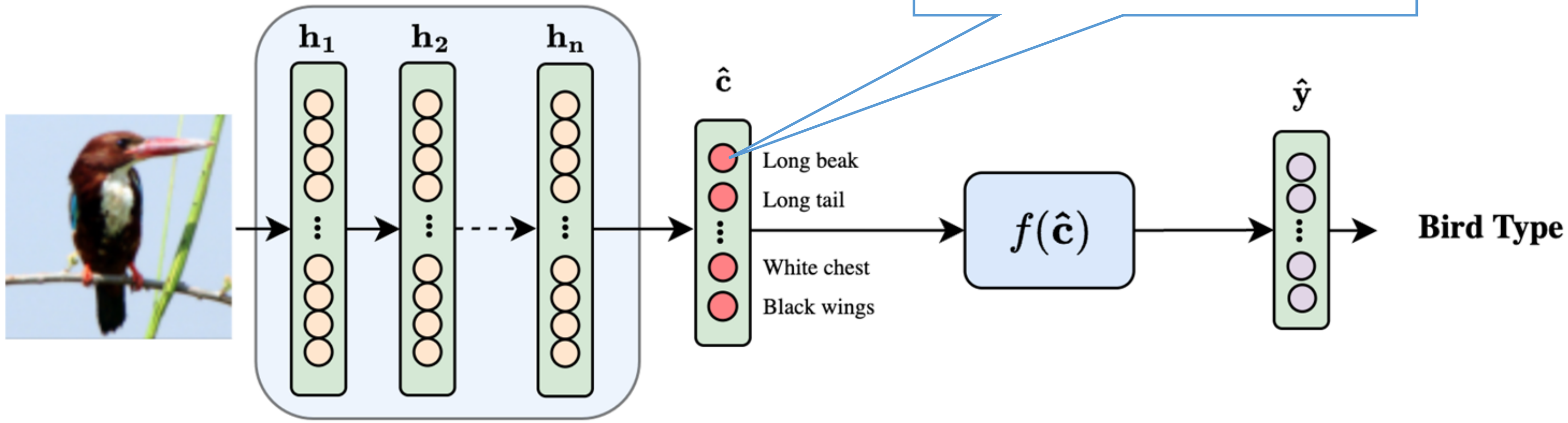
$$\mathcal{L} \triangleq \mathbb{E}_{(\mathbf{x},y,\mathbf{c})}\left[\mathcal{L}_{\text{task}}\Big(y, f\big(g(\mathbf{x})\big)\Big) + \alpha\mathcal{L}_{\text{CrossEntr}}\Big(\mathbf{c}, \hat{\mathbf{p}}(\mathbf{x})\Big)\right]$$

- Open questions and issues
  - Concept bottleneck **hinders predictive performance** (Y)
  - What does it mean to have an **«active» concept**? > 0.5?

# Concept bottleneck models

- Boolean bottleneck

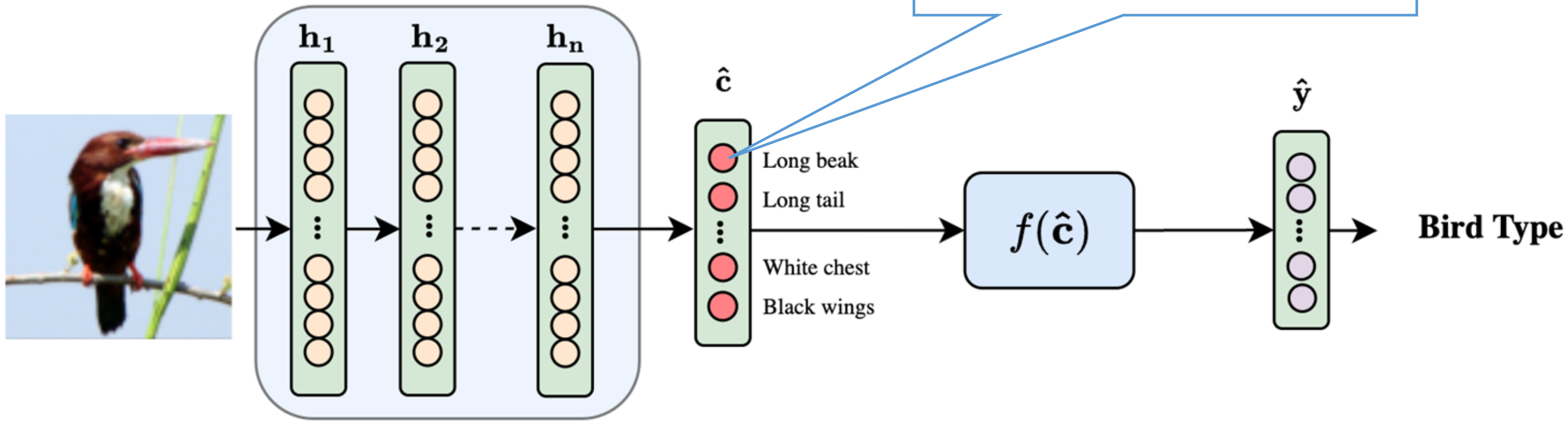
$$s(x) \triangleq \mathbb{1}_{x \geq 0.5}$$



# Concept bottleneck models

- Fuzzy bottleneck

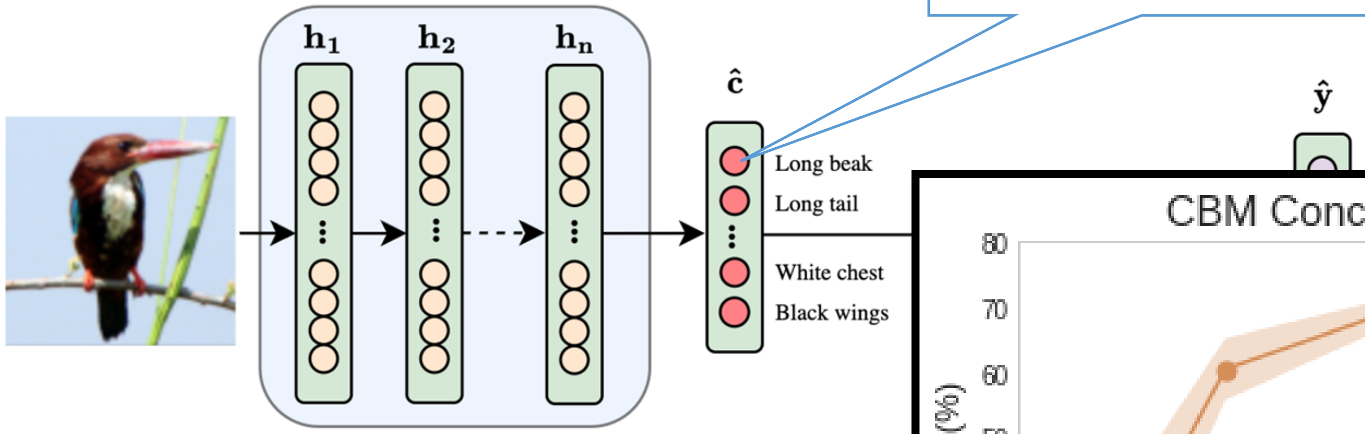
$$s(x) \triangleq 1/(1+e^{-x})$$

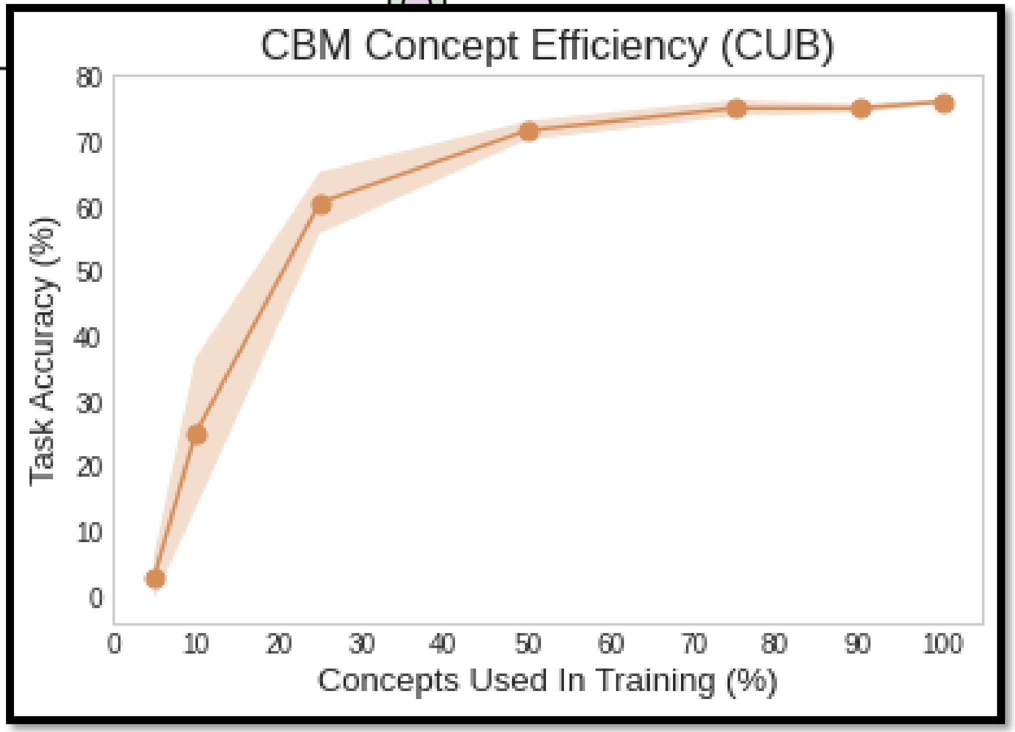


# Concept bottleneck models

- Fuzzy bottleneck

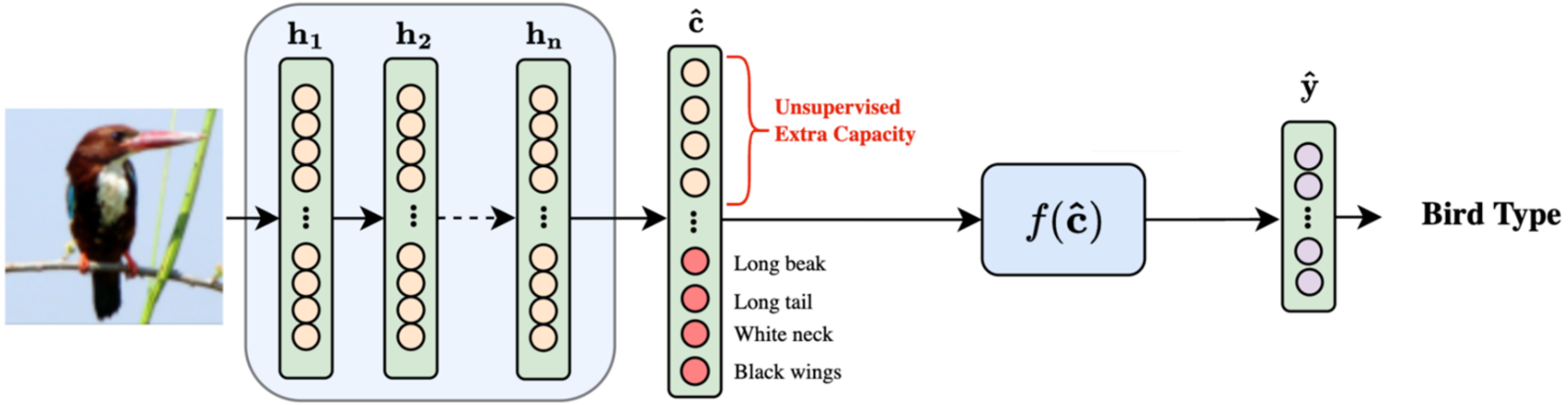
$$s(x) \triangleq 1/(1+e^{-x})$$



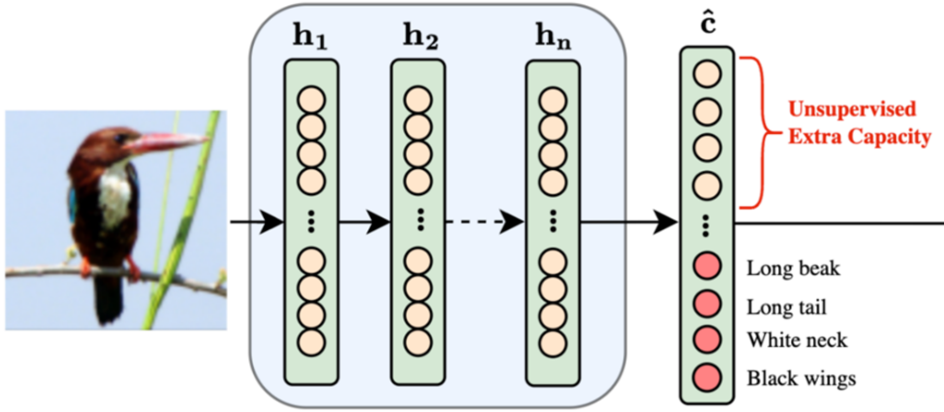
# Concept bottleneck models

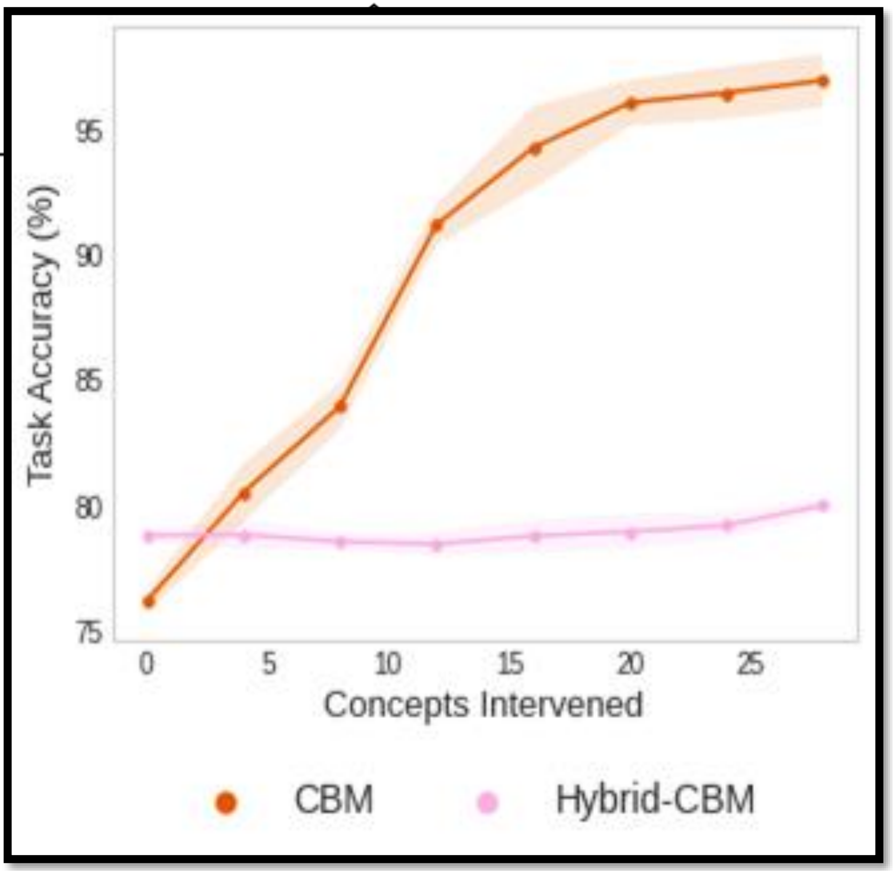
- Hybrid bottleneck



# Concept bottleneck models

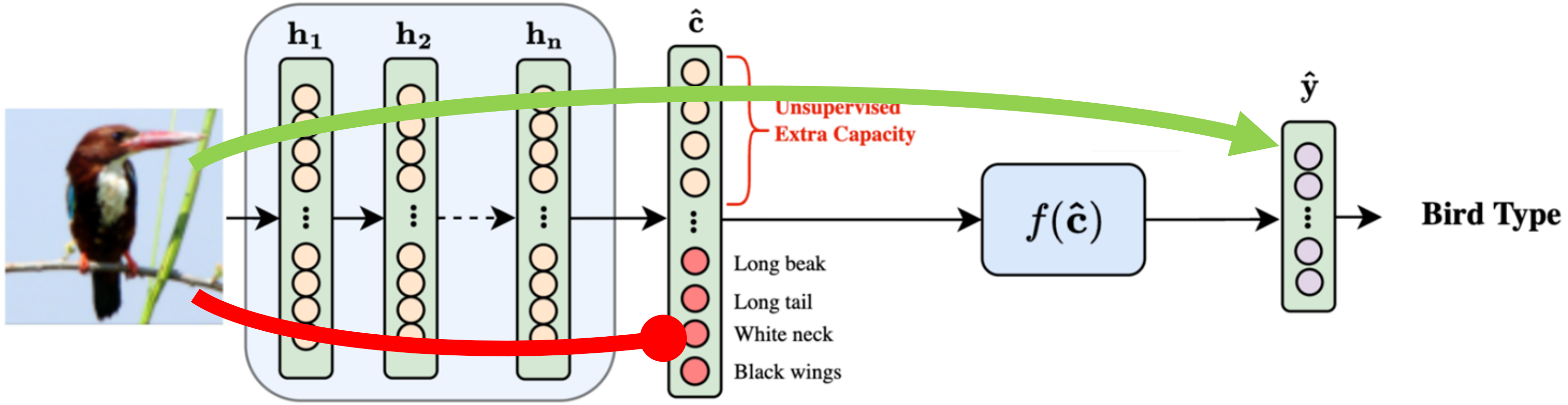
- Hybrid bottleneck



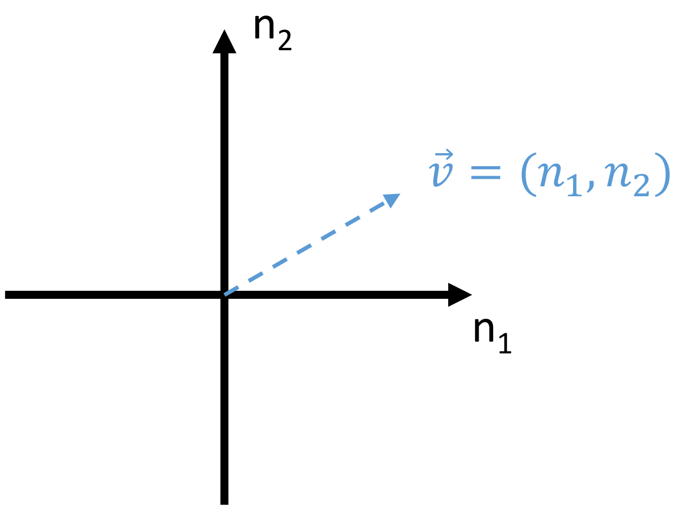
# Concept bottleneck models

- Hybrid bottleneck



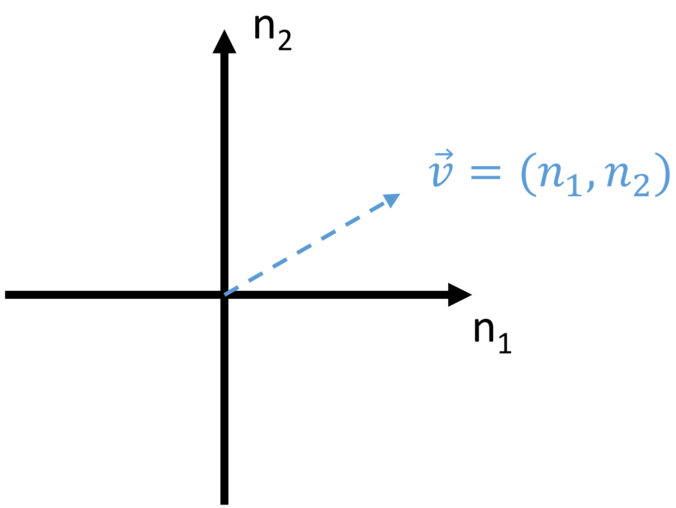
# Concept embedding models

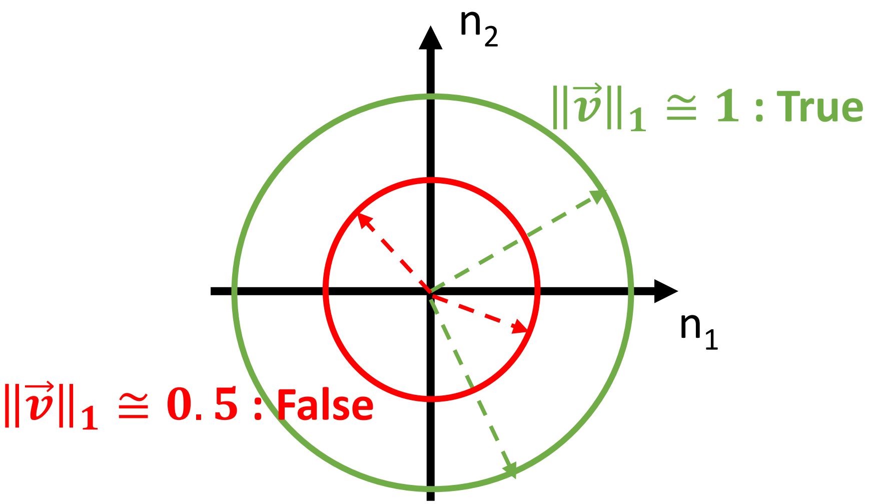
- Instead of associating **one** concept to a **single** neuron...
  - One concept is associated with a group of neurons
  - Amount of neurons is user-defined (hyperparameter)
- Example: norm (module) of a vector in $k$ dimensions



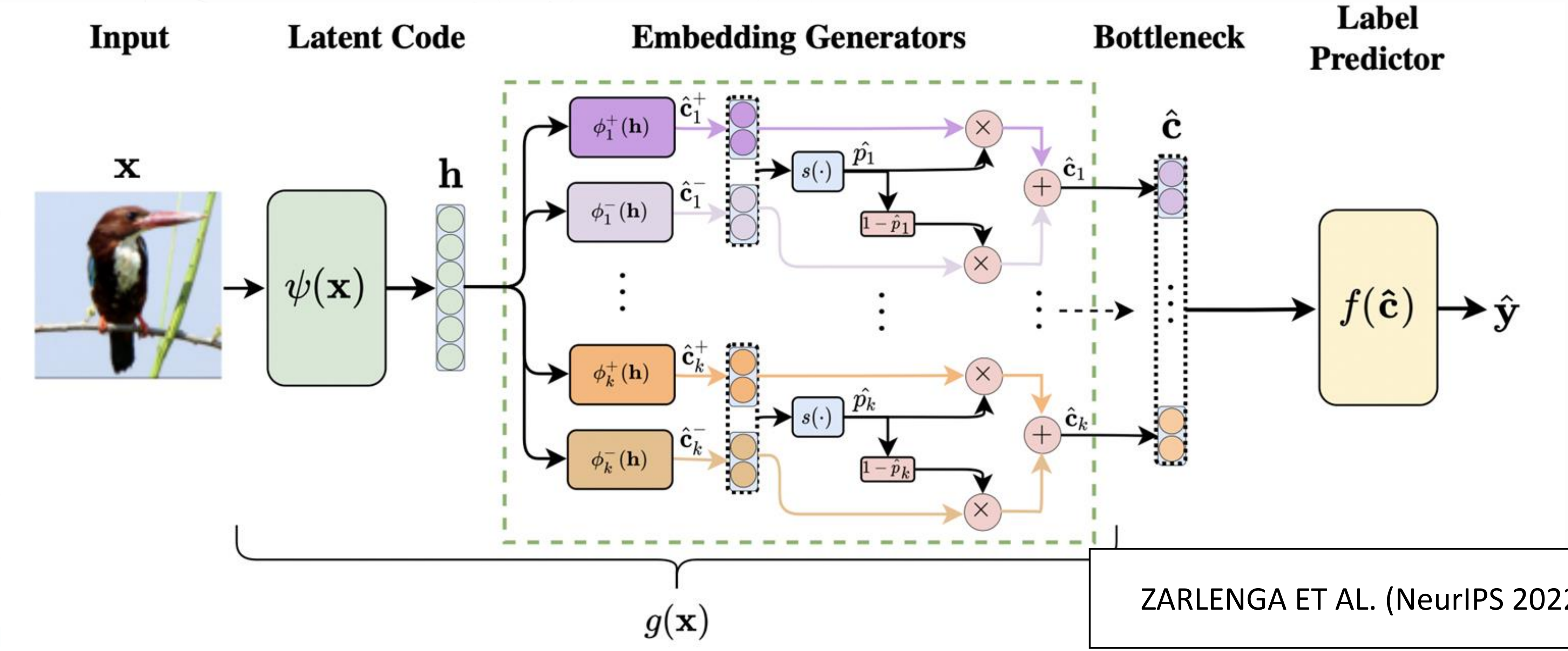
# Concept embedding models

- Instead of associating one concept to a single neuron...
  - One concept is associated with a group of neurons
  - Amount of neurons is user-defined
- Example: norm (module) of a vector in $k$ dimensions



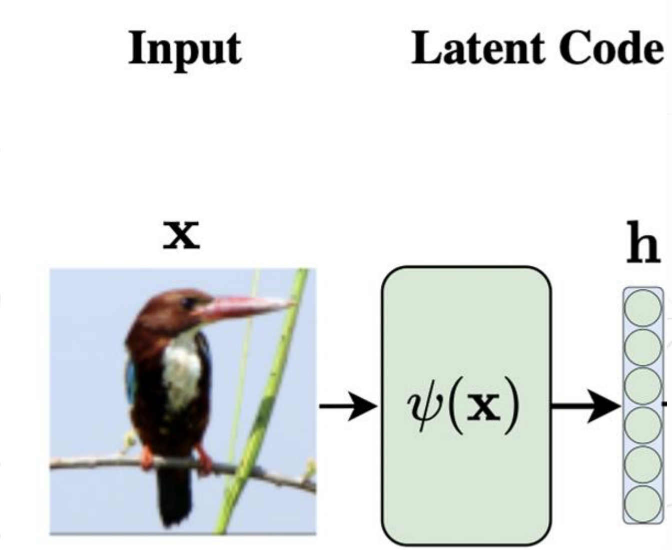



# Solution: Concept Embedding Models

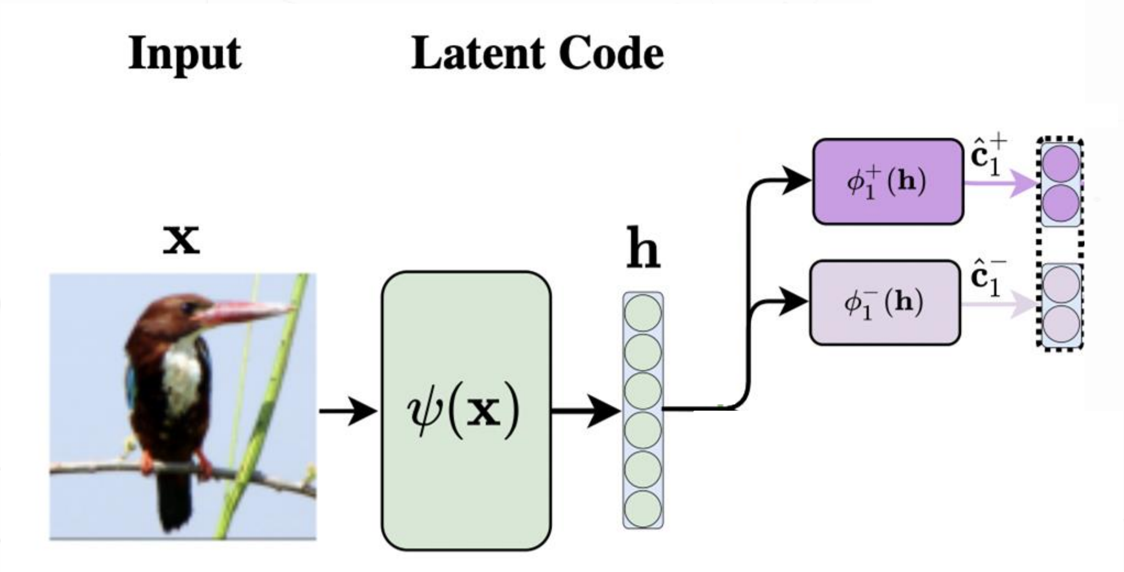



ZARLENGA ET AL. (NeurIPS 2022)

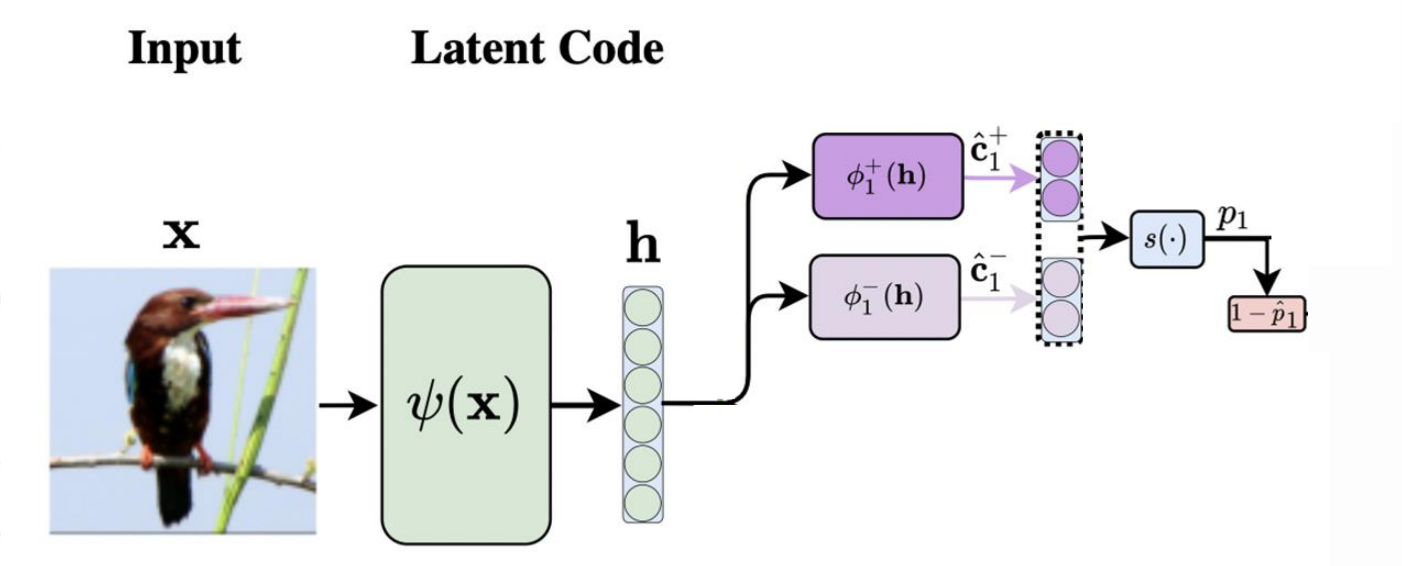
# Solution: Concept Embedding Models



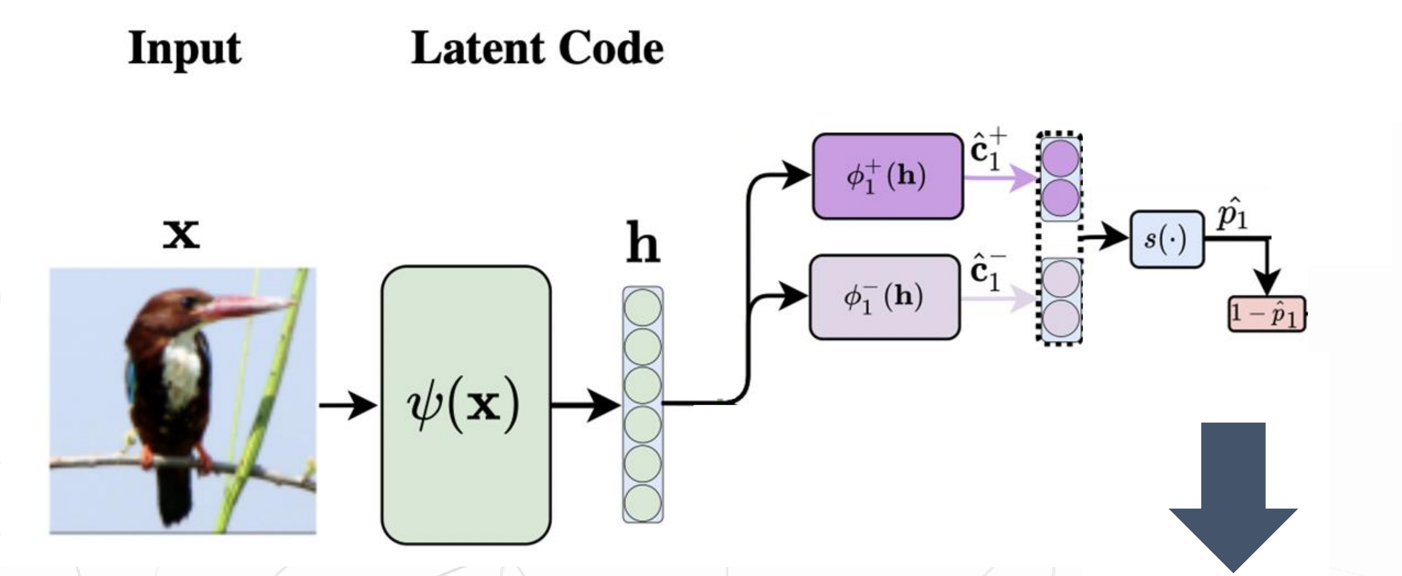
# Solution: Concept Embedding Models



# Solution: Concept Embedding Models

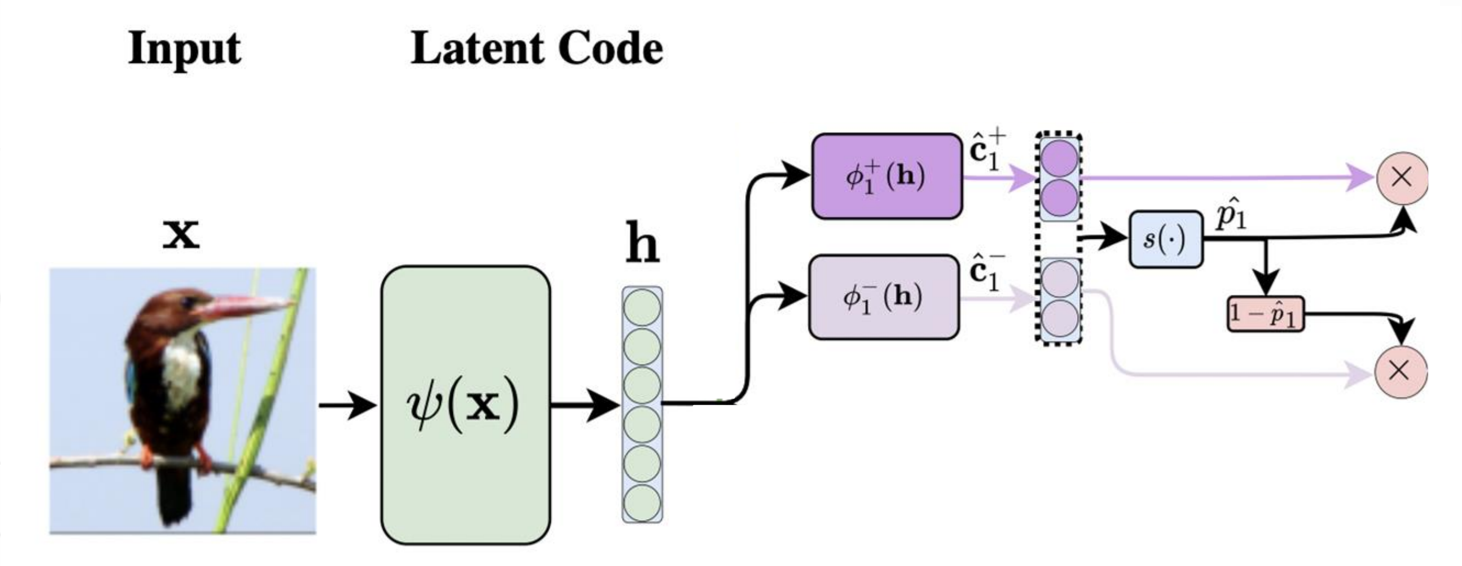


# Solution: Concept Embedding Models

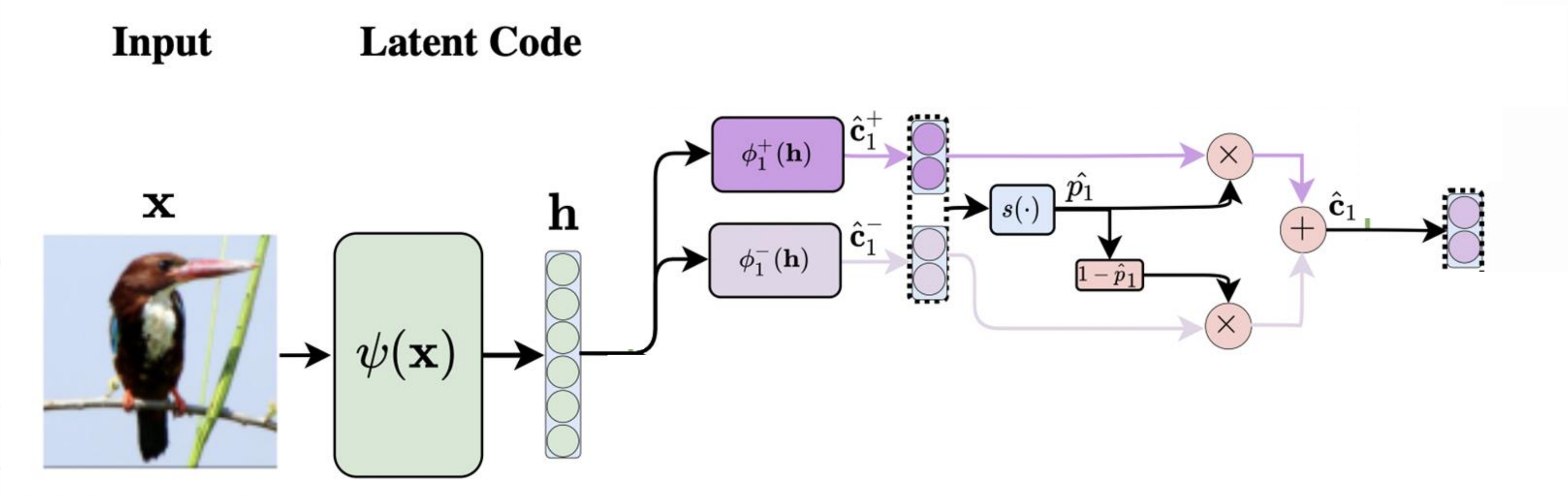



$\mathcal{L}_{\text{concepts}}$

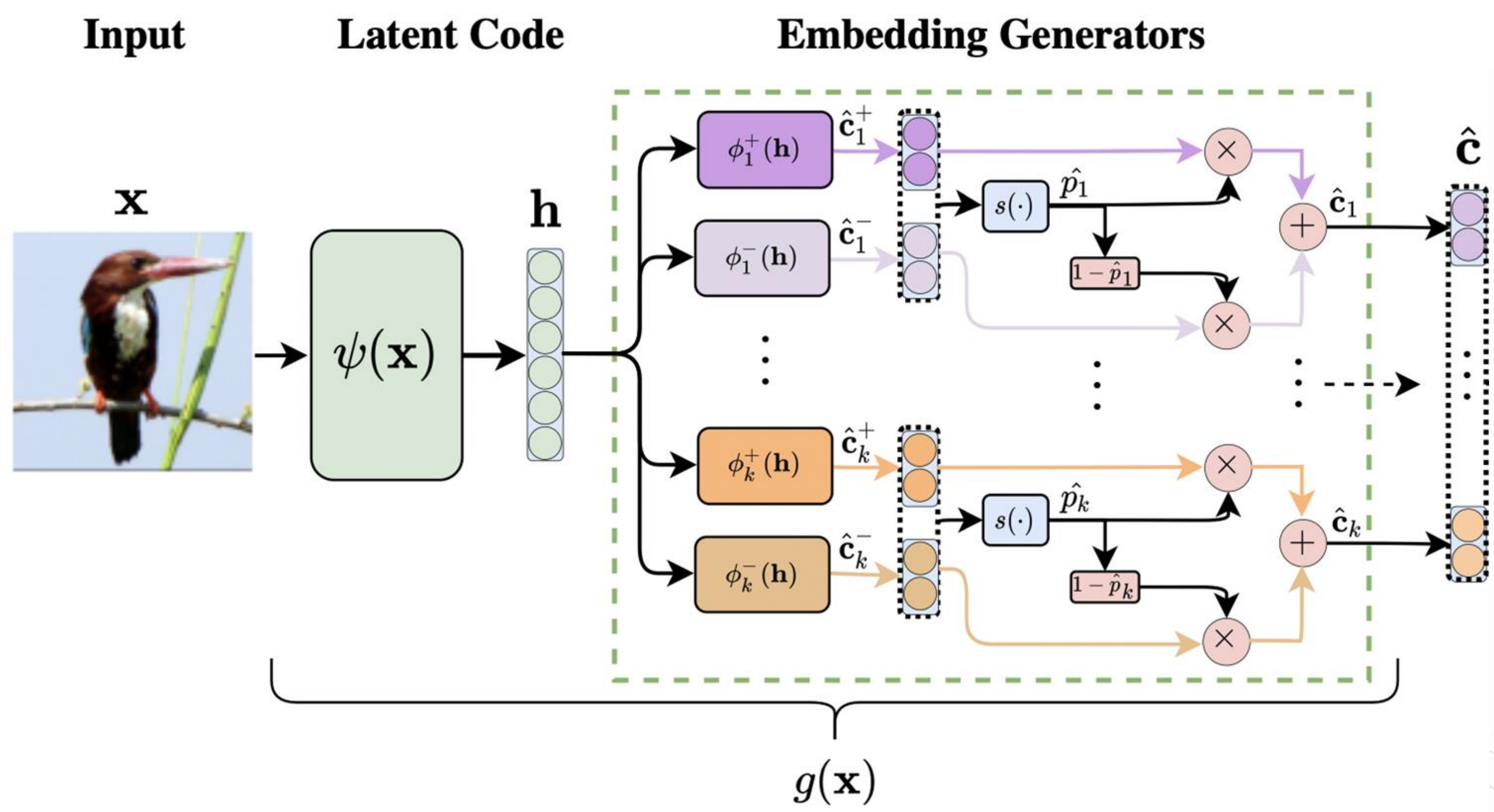
# Solution: Concept Embedding Models



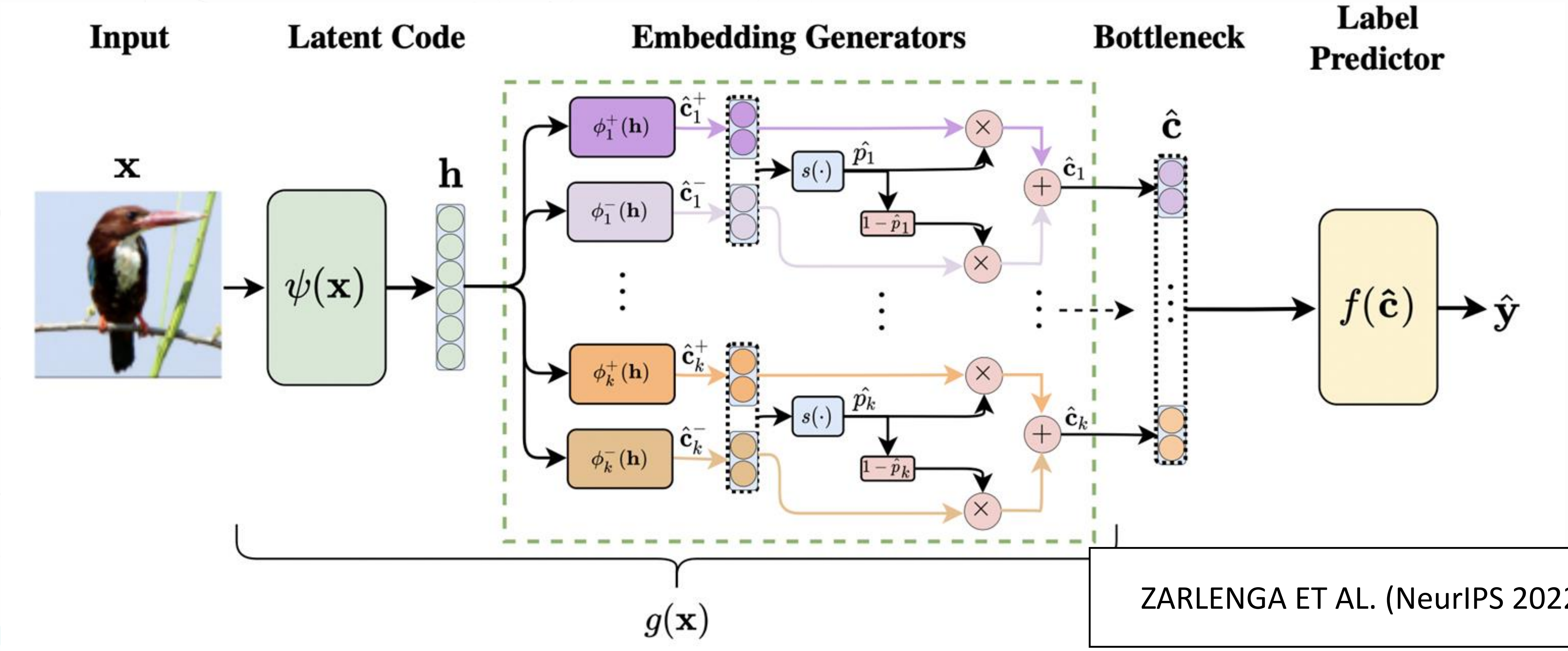
# Solution: Concept Embedding Models



# Solution: Concept Embedding Models

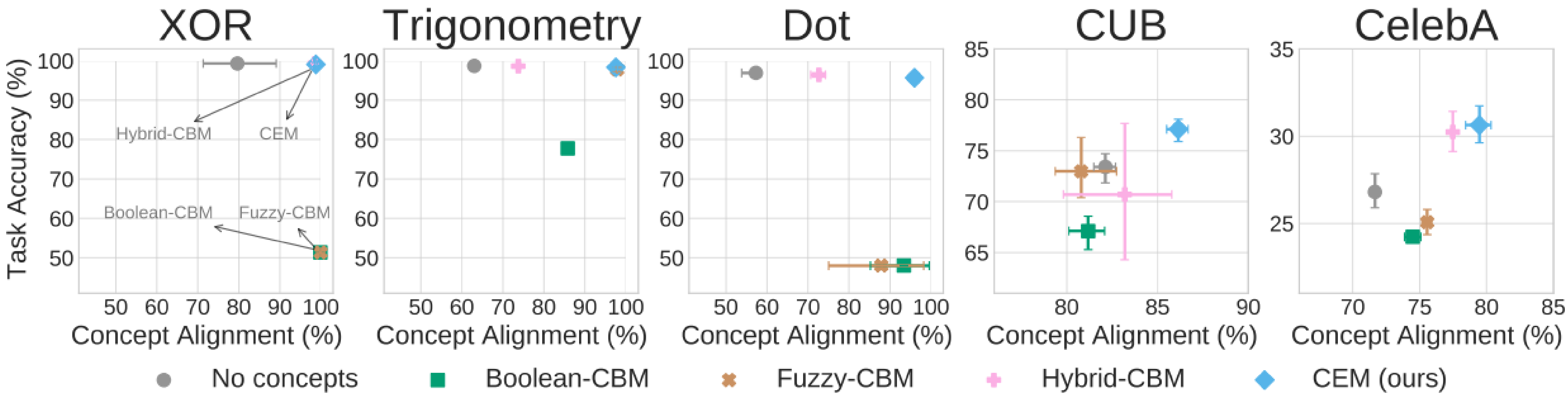


# Solution: Concept Embedding Models



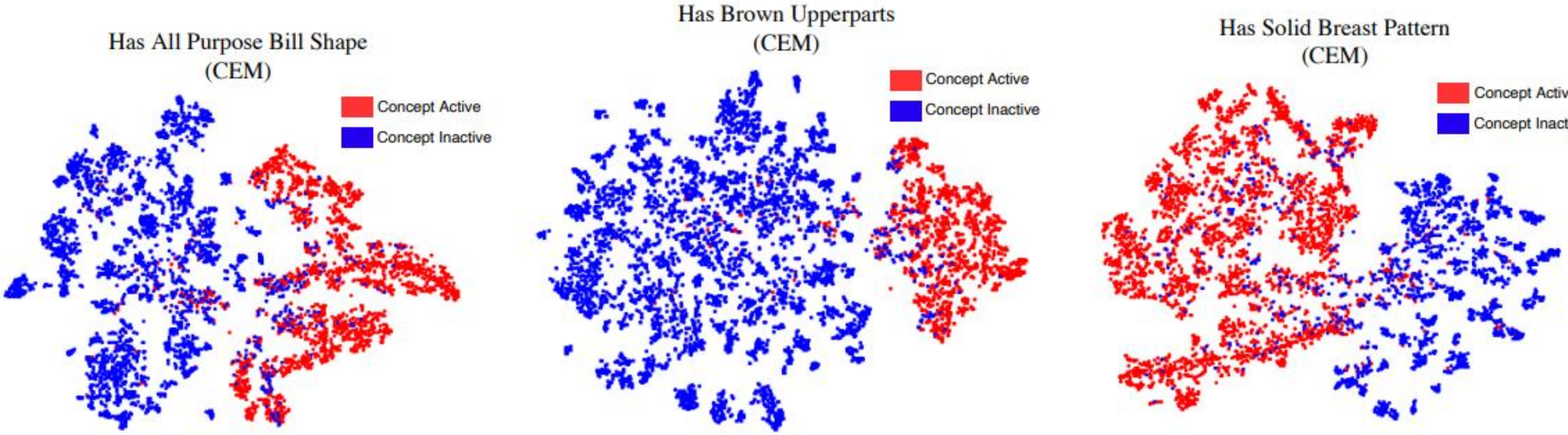

ZARLENGA ET AL. (NeurIPS 2022)

# Experimental evaluation of CEM

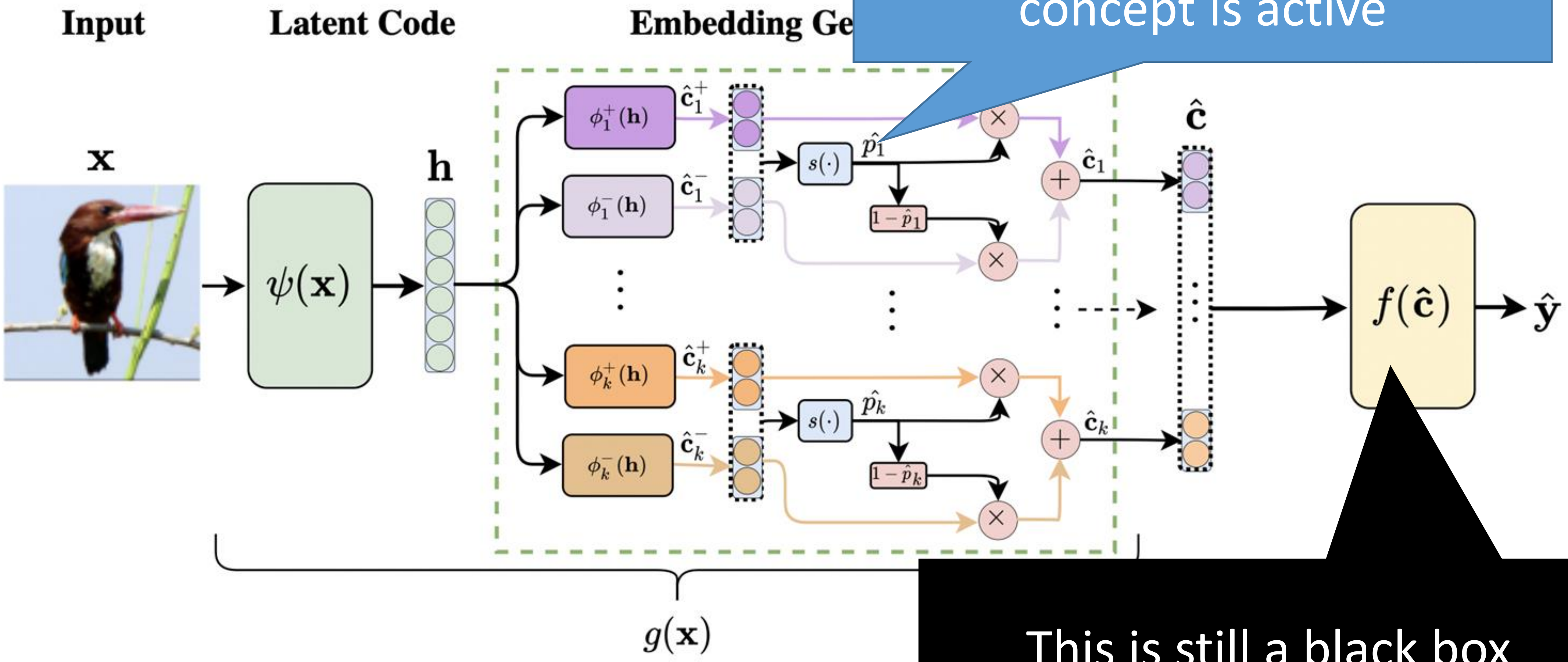


# Concept embeddings

- Plots with t-SNE of the concept embedding



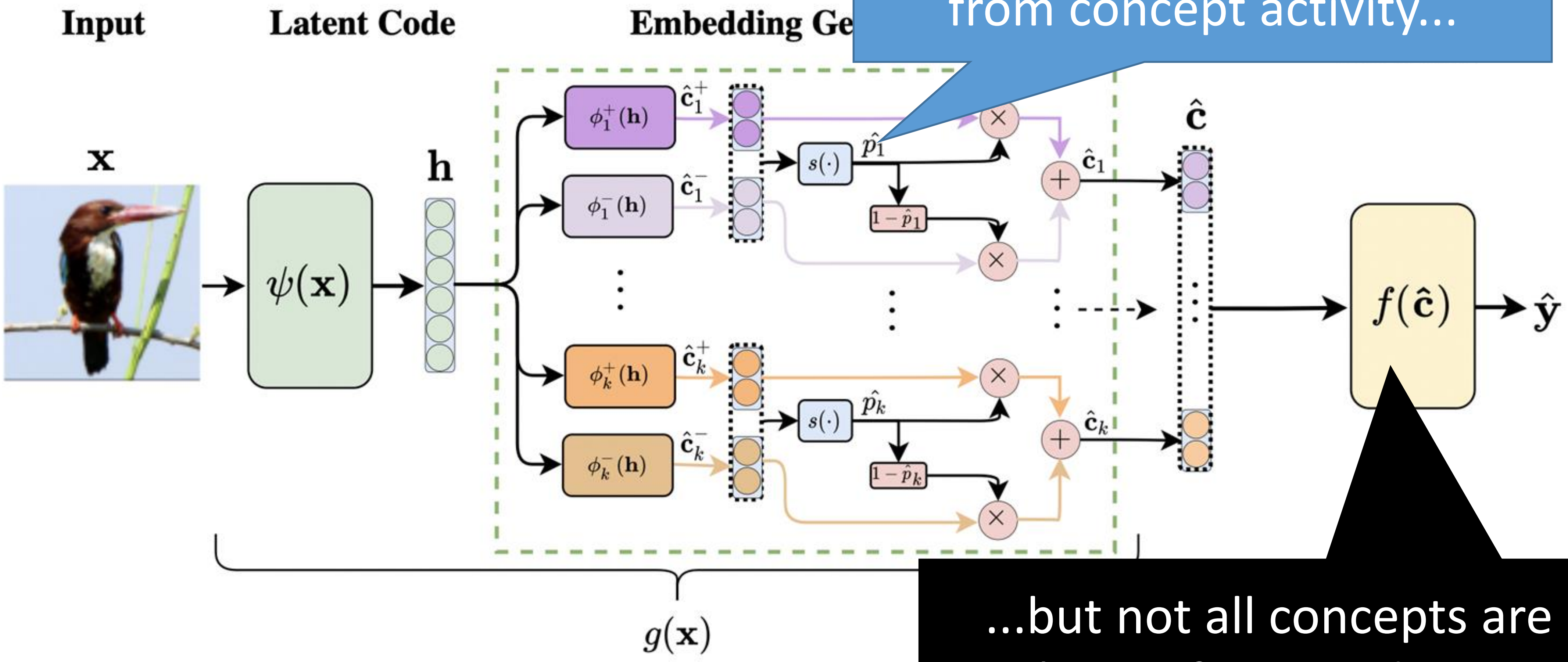
# Concept embedding models: issues




This tells us whether the concept is active

This is still a black box

# Concept embedding models: issues




In principle, extract rules from concept activity...

...but not all concepts are relevant for a prediction!

# Concept embedding models: issues

- CEMs are better than other CBMs
  - Better performance on tasks
  - Better concept alignment
  - Interventions are easy to perform and improve results
- Explainability could be improved
  - It’s possible to extract rules from CEMs, but **lots of concepts**
  - Ideally, rules should be compact (few concepts)
- Can we do better than a CEM?

# Deep concept reasoner

- **Automatically build rules** based on concepts
- For <u>each class</u>, for <u>each concept</u>
  - The concept might be **relevant** or *irrelevant*
  - The concept might be useful if True or False (negated)
- Fuzzy logic rules (differentiable, real-valued operators)
  - Extension of Boolean logic rules
  - t-norm ∧: $[0,1] \times [0,1] \rightarrow [0,1]$; $x \wedge y = x \cdot y$
  - t-norm ∨: $[0,1] \times [0,1] \rightarrow [0,1]$; $x \vee y = x + y - x \cdot y$
  - $\neg x = 1 - x$

# Deep concept reasoner

- Example:
  - Class «banana», concepts: «round», «yellow», «soft»
  - $y_{banana} \leftrightarrow \neg c_{round} \wedge c_{yellow}$
  - “soft” is not relevant, round is negated
- General form of a DCR rule for class *j* and concepts *i*

$$\hat{y}_j \Leftrightarrow \bigwedge_{i:\, r_{ji}=1} l_{ji} \qquad \hat{y}_j \iff \bigwedge_{i=1}^{k} (\neg r_{ji} \vee l_{ji})$$

  - Where *r* (relevance) defines if concept is relevant
  - *l* (literal) if it should appear negated

# Deep concept reasoner

- Example:
  - Class «banana», concepts: «round», «yellow», «soft»
  - $y_{banana} \leftrightarrow \neg c_{round} \wedge c_{yellow}$
  - "soft" is not relevant, round is negated
- General form of a DCR rule for class *j* and concepts *i*

$$\hat{y}_j \Leftrightarrow \bigwedge_{i:\, r_{ji}=1} l_{ji} \qquad\qquad \hat{y}_j \Longleftrightarrow \bigwedge_{i=1}^{k} (\neg r_{ji} \vee l_{ji})$$

  - Where *r* (relevance) defines if concept is relev
  - *l* (literal) if it should appear negated

For IRRELEVANT (r=0) concepts, the whole expression goes to 1

# Deep concept reasoner

- Example: banana

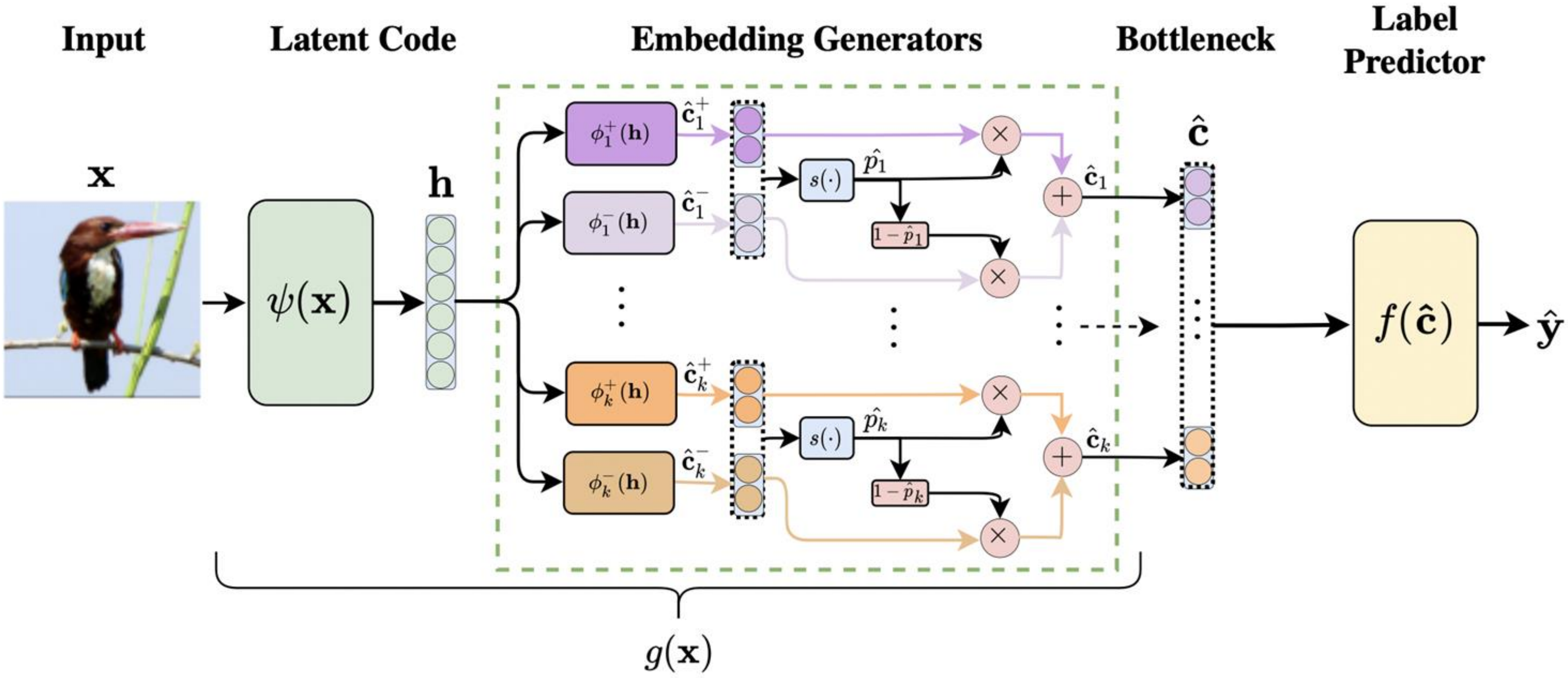
$$y_{banana} \iff (\neg r_{soft} \vee l_{soft}) \wedge (\neg r_{yellow} \vee l_{yellow}) \wedge (\neg r_{round} \vee l_{round})$$

$$y_{banana} \iff (1 \vee l_{soft}) \wedge (0 \vee l_{yellow}) \wedge (0 \vee l_{round})$$

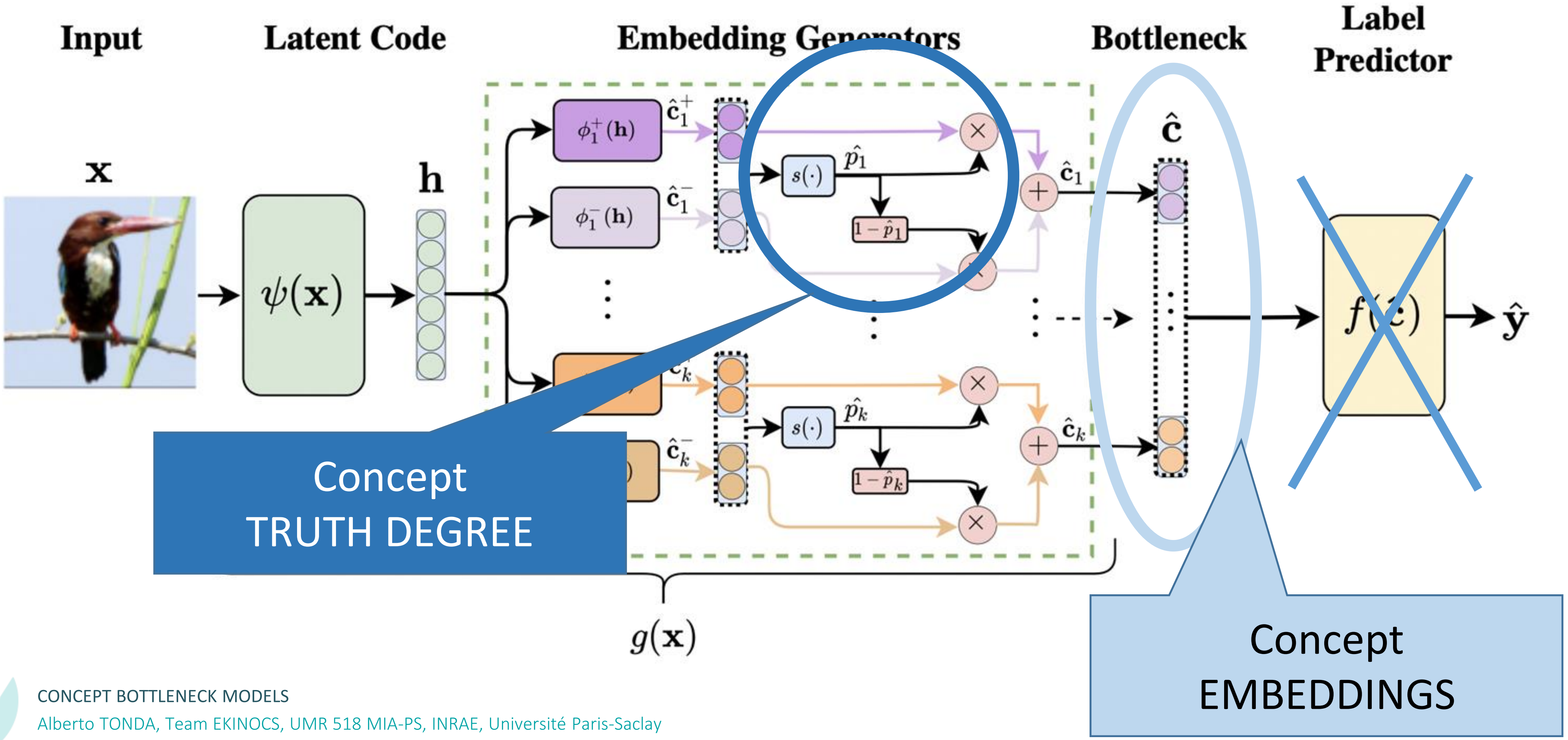
$$y_{banana} \iff \cancel{(1 \vee l_{soft})} \wedge l_{yellow} \wedge l_{round}$$

$$y_{banana} \iff c_{yellow} \wedge \neg c_{round}$$

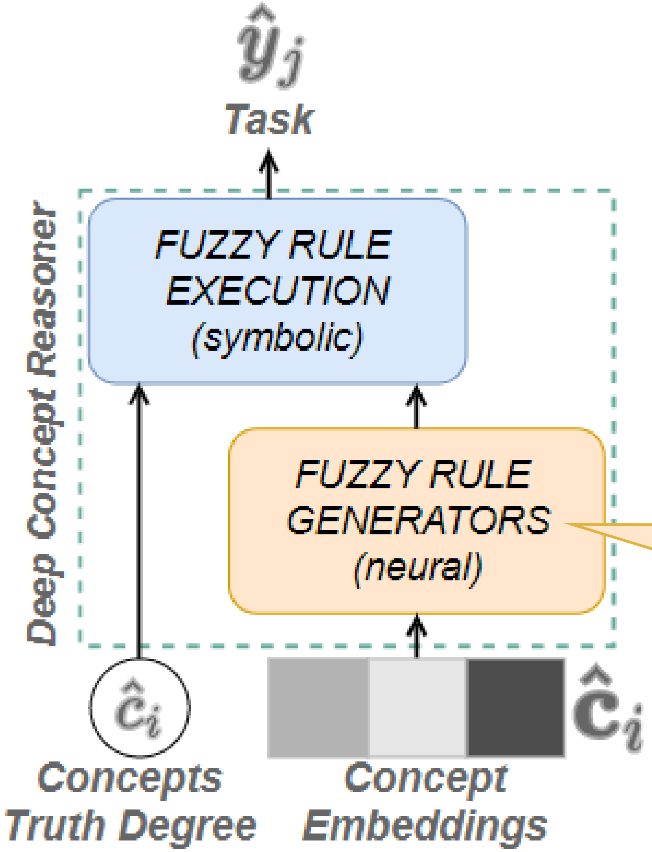
# Deep concept reasoner



# Deep concept reasoner



# Deep concept reasoner



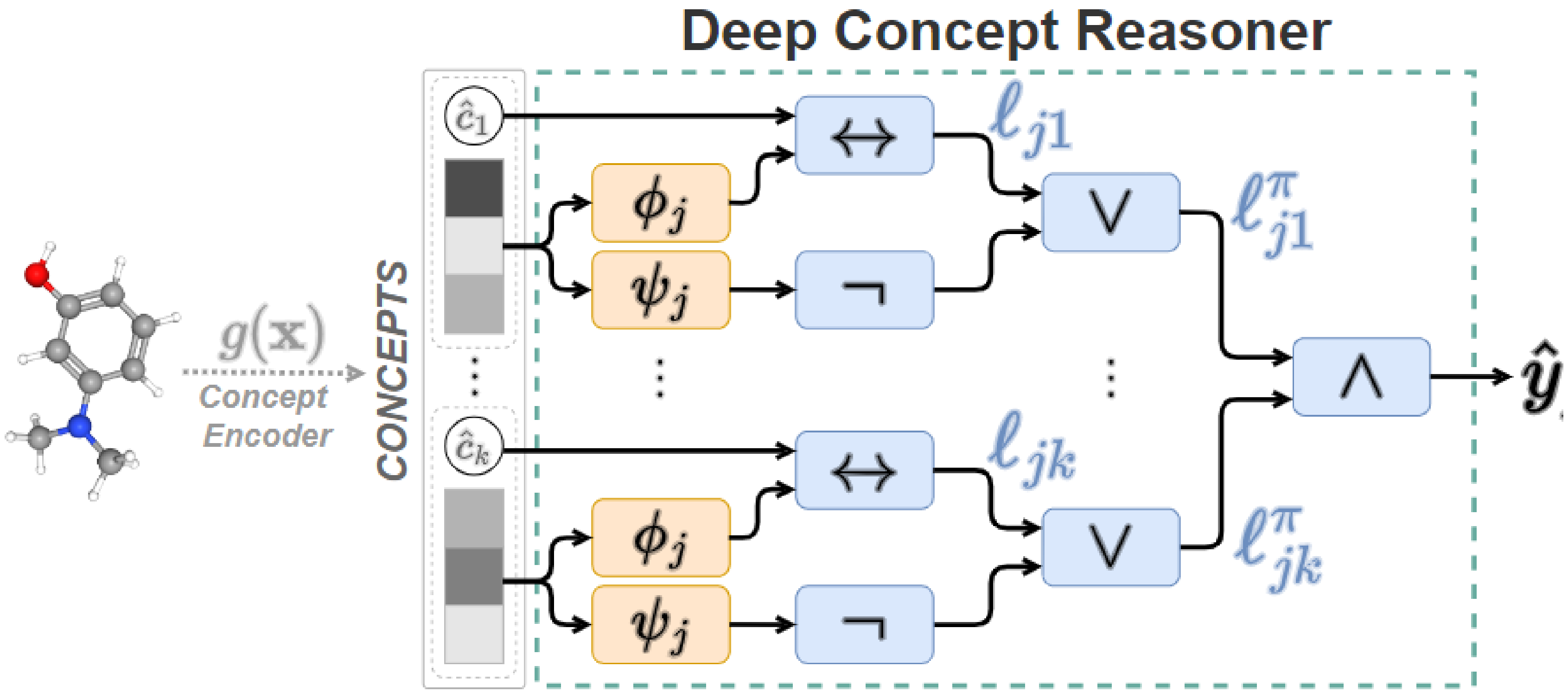

$$y_{banana} \iff (\neg r_{soft} \vee l_{soft}) \wedge (\neg r_{yellow} \vee l_{yellow}) \wedge (\neg r_{round} \vee l_{round})$$

$$y_{banana} \iff (1 \vee l_{soft}) \wedge (0 \vee l_{yellow}) \wedge (0 \vee l_{round})$$

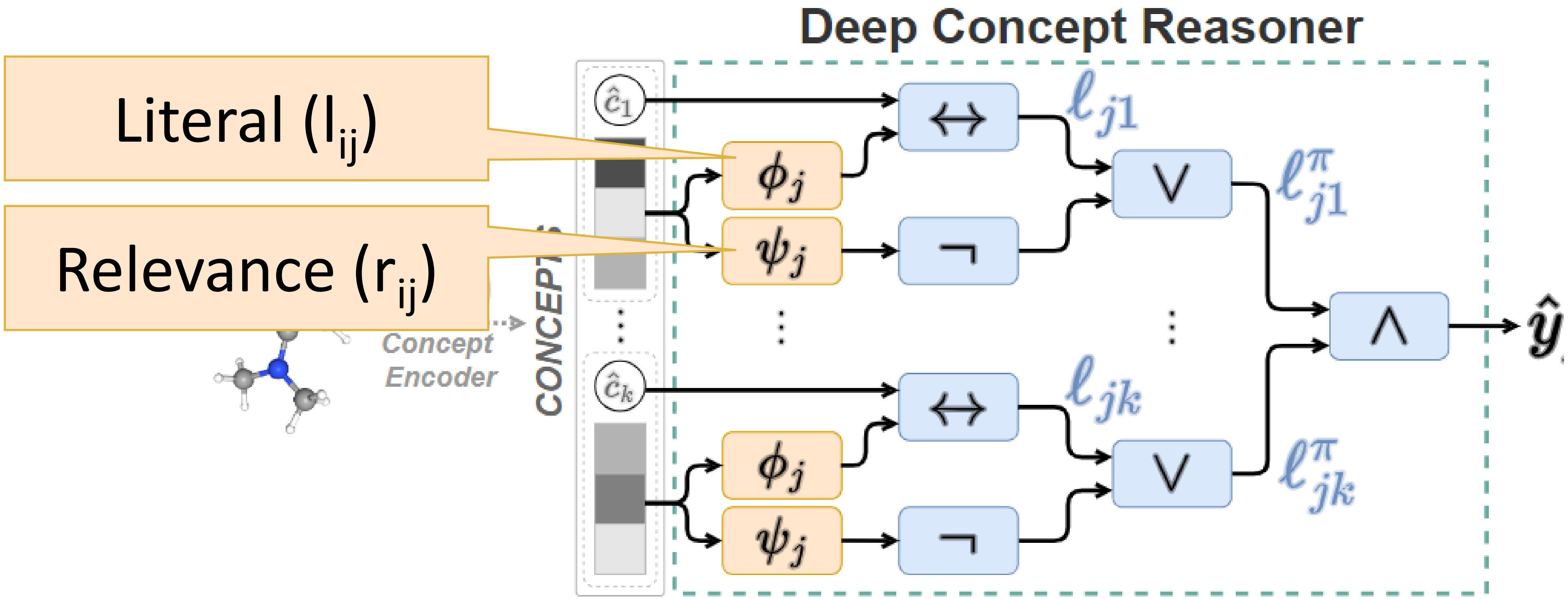
$$y_{banana} \iff \cancel{(1 \vee l_{soft})} \wedge l_{yellow} \wedge l_{round}$$

$$y_{banana} \iff c_{yellow} \wedge \neg c_{round}$$

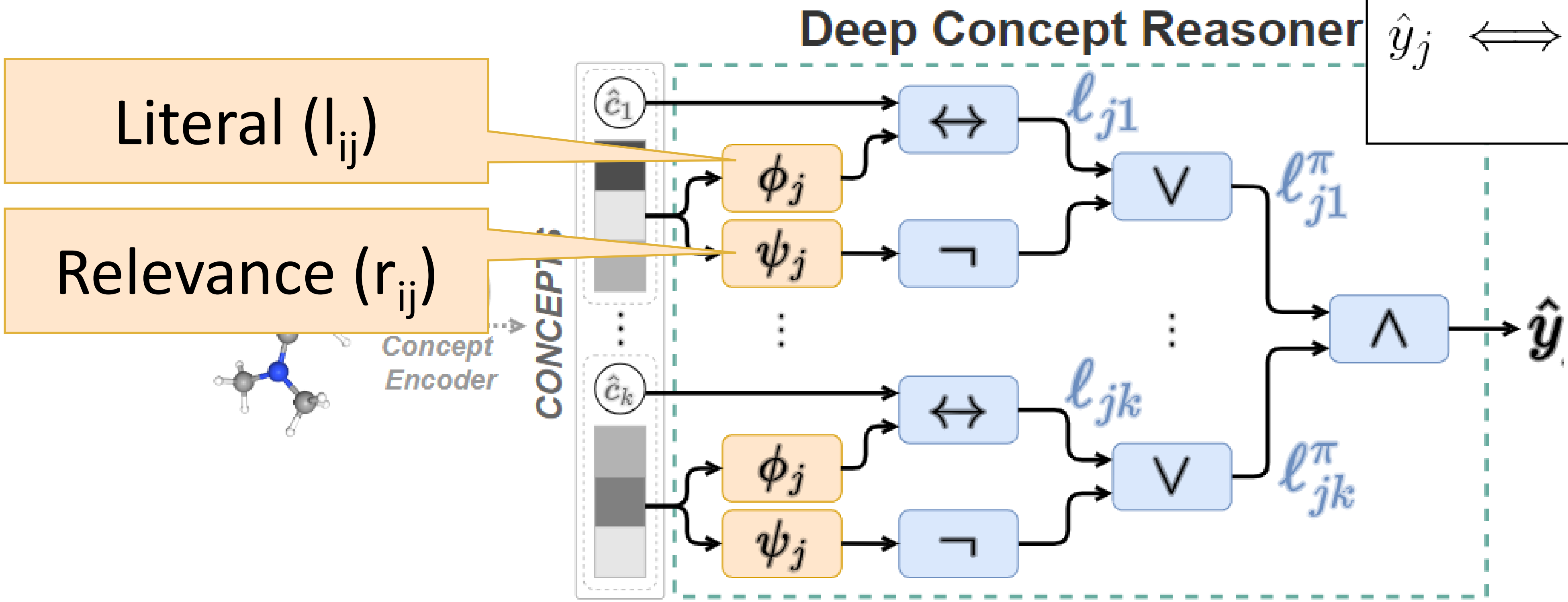
# Deep concept reasoner



# Deep concept reasoner

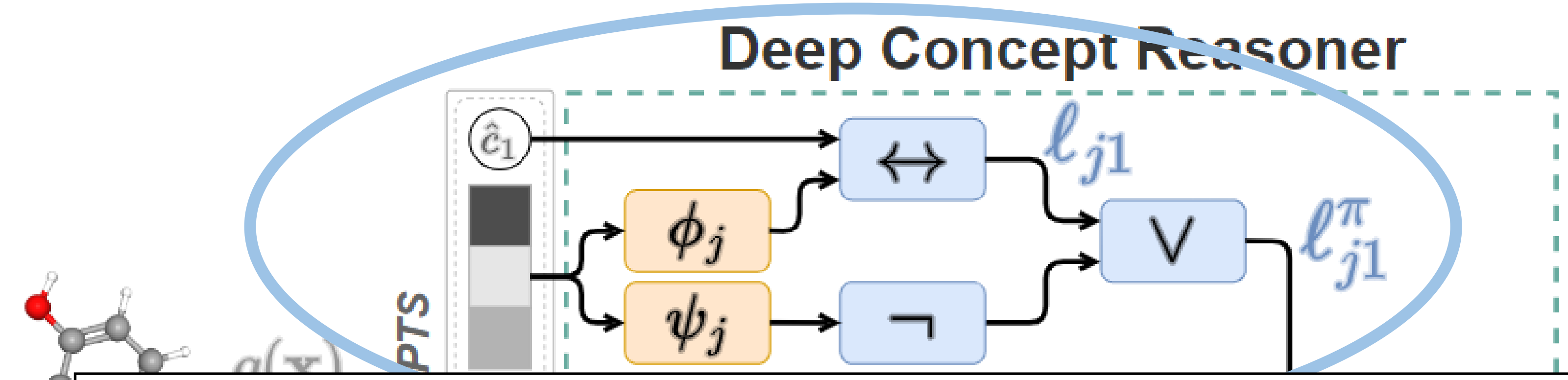


# Deep concept reasoner




$$\hat{y}_j \iff \bigwedge_{i=1}^{k} (\neg r_{ji} \vee l_{ji})$$

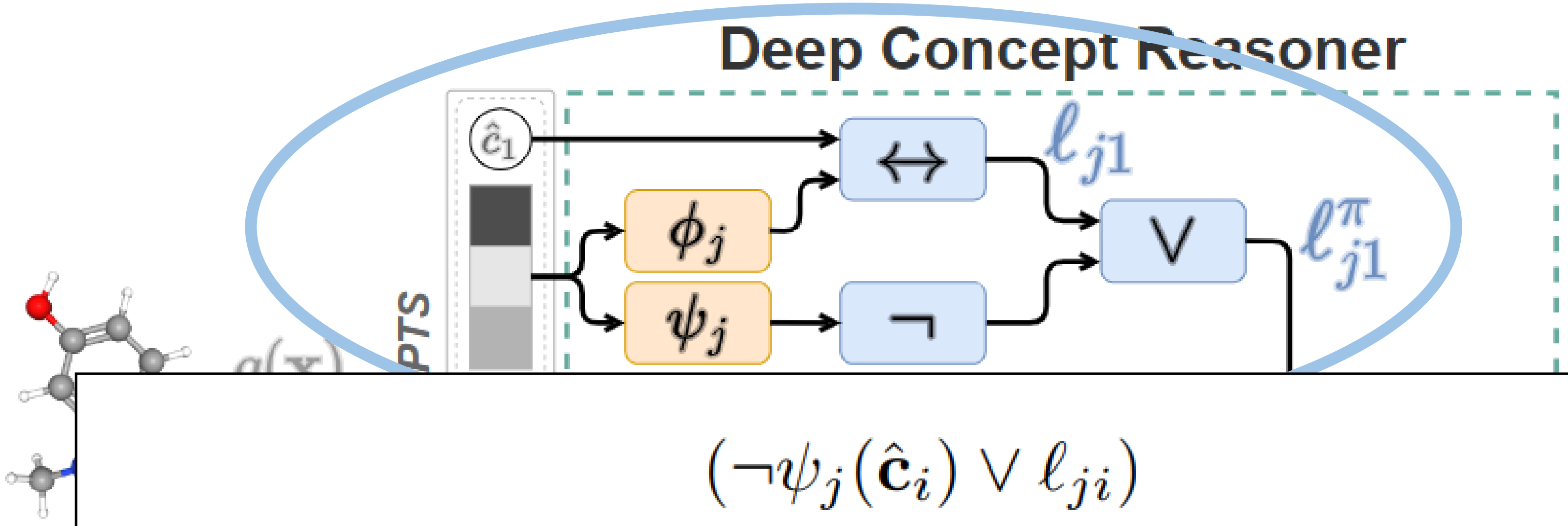
# Deep concept reasoner




$$\ell_{ji} = (\phi_j(\hat{\mathbf{c}}_i) \Leftrightarrow \hat{c}_i)$$

$$\begin{aligned}
\ell_{ji} \quad &= \phi_j(\hat{\mathbf{c}}_i) \Leftrightarrow \hat{c}_i = (\phi_j(\hat{\mathbf{c}}_i) \Rightarrow \hat{c}_i) \wedge (\hat{c}_i \Rightarrow \phi_j(\hat{\mathbf{c}}_i)) = \\
&= (\neg\phi_j(\hat{\mathbf{c}}_i) \vee \hat{c}_i) \wedge (\neg\hat{c}_i \vee \phi_j(\hat{\mathbf{c}}_i)) = \\
&= \min\{\max\{1 - \phi_j(\hat{\mathbf{c}}_i), \hat{c}_i\}, \max\{1 - \hat{c}_i, \phi(\hat{\mathbf{c}}_i)\}\}
\end{aligned}$$

# Deep concept reasoner

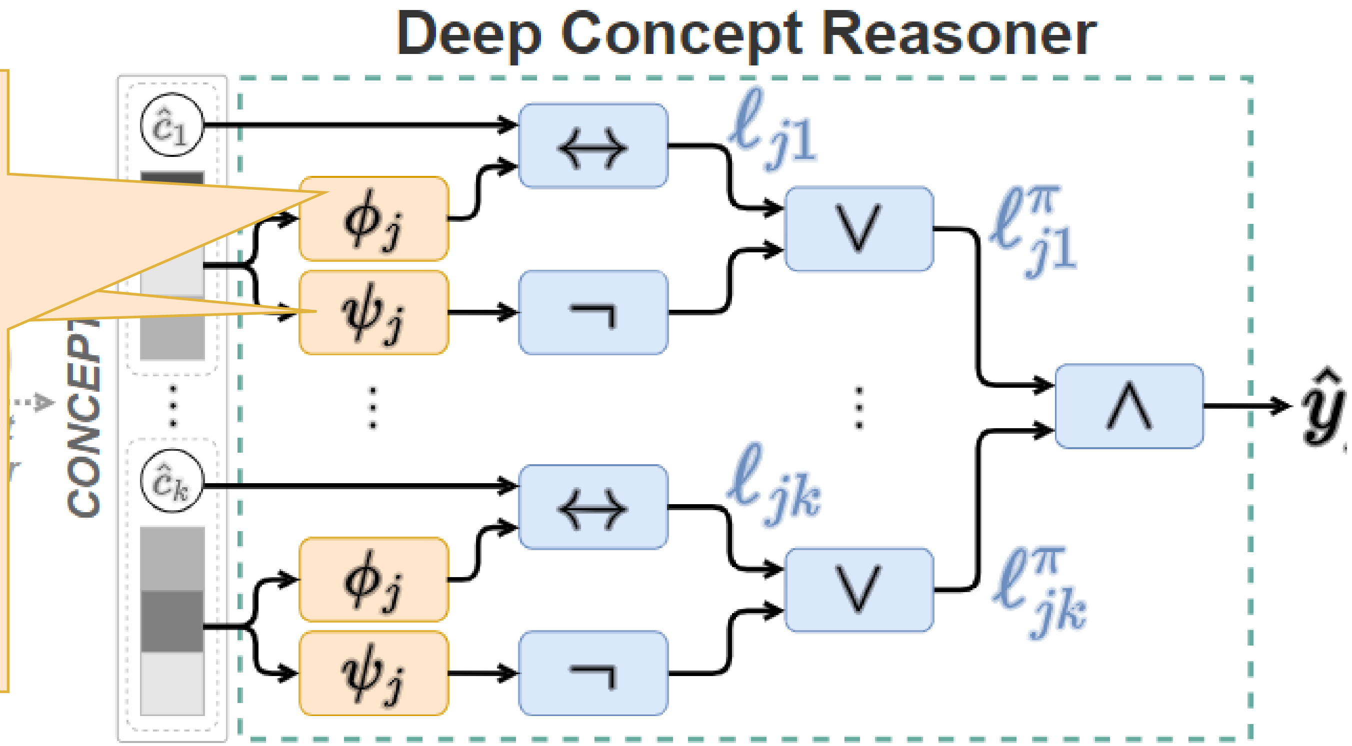



$$(\neg\psi_j(\hat{\mathbf{c}}_i) \vee \ell_{ji})$$

$$\hat{y}_j = \min_{i=1}^{k}\{\max\{1 - \psi_j(\hat{\mathbf{c}}_i), \ell_{ji}\}\}$$

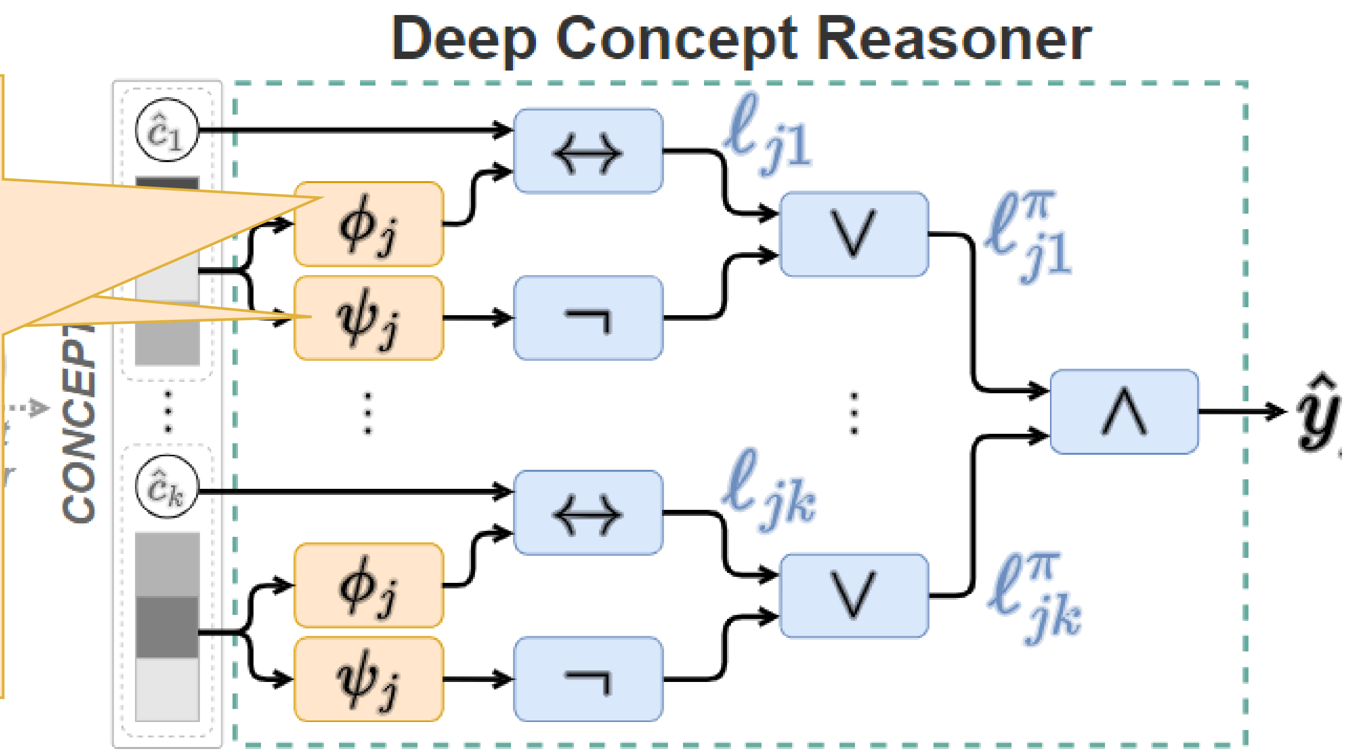
# Deep concept reasoner

They are the same for all concepts of the same class (**training**)



# Deep concept reasoner

In **test**, apply all rules for all classes, and apply softmax



# Deep concept reasoner

- Activation function of $\psi_j$ (relevance) enforces parsimony

$$\gamma_{ji} = \log\left(\frac{\exp(\mathrm{MLP}_j(\hat{\mathbf{c}}_i))}{\sum_{i'=1}^{k}\exp(\mathrm{MLP}_j(\hat{\mathbf{c}}_{i'}))}\right)$$

$$r_{ji} = \psi_j(\hat{\mathbf{c}}_i) = \sigma\left(\gamma_{ji} - \frac{1}{k}\sum_{i'=1}^{k}\gamma_{ji'}\right)$$

How strongly is a concept activated with respect to the others? log-softmax

$$r_{ji} = \sigma\left(\gamma_{ji} - \frac{\tau}{k}\sum_{i'=1}^{k}\gamma_{ji'}\right)$$

# Deep concept reasoner

- Activation function of $\psi_j$ (relevance) enforces parsimony

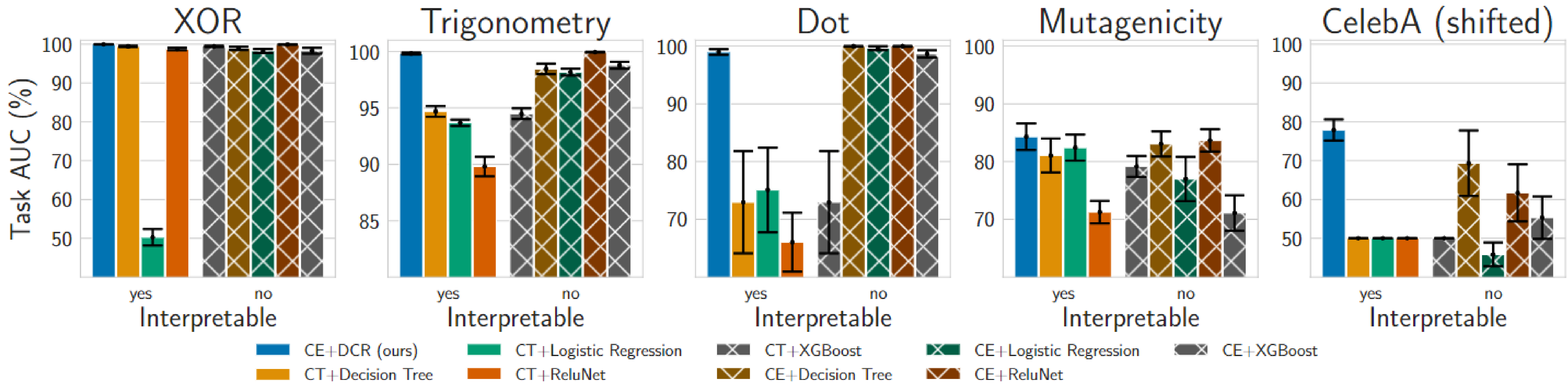
$$\gamma_{ji} = \log\left(\frac{\exp(\mathrm{MLP}_j(\hat{\mathbf{c}}_i))}{\sum_{i'=1}^{k}\exp(\mathrm{MLP}_j(\hat{\mathbf{c}}_{i'}))}\right) \quad (5)$$

$$r_{ji} = \psi_j(\hat{\mathbf{c}}_i) = \sigma\left(\gamma_{ji} - \frac{1}{k}\sum_{i'=1}^{k}\gamma_{ji'}\right) \quad (6)$$

User-defined parameter to regulate number of concepts per rule

$$r_{ji} = \sigma(\gamma_{ji} - \frac{\tau}{k}\sum_{i'=1}^{k}\gamma_{ji'})$$

# Experimental evaluation




*CE* stands for concept embeddings, while *CT* for concept truth degrees.

# Experimental evaluation

| Ground-truth Rule | Predicted Rule | Error (%) |
|---|---|---|
| **XOR** | | |
| $y_0 \leftarrow \neg c_0 \wedge \neg c_1$ | $y_0 \leftarrow \neg c_0 \wedge \neg c_1$ | $0.00 \pm 0.00$ |
| $y_0 \leftarrow c_0 \wedge c_1$ | $y_0 \leftarrow c_0 \wedge c_1$ | $0.00 \pm 0.00$ |
| $y_1 \leftarrow \neg c_0 \wedge c_1$ | $y_1 \leftarrow \neg c_0 \wedge c_1$ | $0.02 \pm 0.02$ |
| $y_1 \leftarrow c_0 \wedge \neg c_1$ | $y_1 \leftarrow c_0 \wedge \neg c_1$ | $0.01 \pm 0.01$ |
| **Trigonometry** | | |
| $y_0 \leftarrow \neg c_0 \wedge \neg c_1 \wedge \neg c_2$ | $y_0 \leftarrow \neg c_0 \wedge \neg c_1 \wedge \neg c_2$ | $0.00 \pm 0.00$ |
| $y_1 \leftarrow c_0 \wedge c_1 \wedge c_2$ | $y_1 \leftarrow c_0 \wedge c_1 \wedge c_2$ | $0.00 \pm 0.00$ |
| **MNIST-Addition** | | |
| $y_{18} \leftarrow c'_9 \wedge c''_9$ | $y_{18} \leftarrow c'_9 \wedge c''_9$ | $0.00 \pm 0.00$ |
| $y_{17} \leftarrow c'_9 \wedge c''_8$ | $y_{17} \leftarrow c'_9 \wedge c''_8$ | $0.00 \pm 0.00$ |
| $y_{17} \leftarrow c'_8 \wedge c''_9$ | $y_{17} \leftarrow c'_8 \wedge c''_9$ | $0.00 \pm 0.00$ |

# Conclusions

- DCR is good for local explanations
- You can try for global explanations
  - Aggregate local explanations for samples of the same class
  - “Booleanize” local fuzzy rules and join them with “or”

$$\hat{y}_j^C = \bigvee_{\mathbf{x} \in \mathcal{X}_{\text{train}}} \hat{y}_j(\mathbf{x})$$

- Obvious limitation: concept-based datasets

# Questions?

## Bibliography

- Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., & Liang, P. (2020, November). *Concept bottleneck models*. In International conference on machine learning (pp. 5338-5348). PMLR.
- Espinosa Zarlenga, M., Barbiero, P., Ciravegna, G., Marra, G., Giannini, F., Diligenti, M., ... & Jamnik, M. (2022). *Concept embedding models: Beyond the accuracy-explainability trade-off*. Advances in Neural Information Processing Systems, 35, 21400-21413. [**CEM**]
- Barbiero, P., Ciravegna, G., Giannini, F., Zarlenga, M. E., Magister, L. C., Tonda, A., ... & Marra, G. (2023, July). *Interpretable neural-symbolic concept reasoning*. In International Conference on Machine Learning (pp. 1801-1825). PMLR. [**DCR**]