UiT The Arctic University of Norway

Explainable machine learning

From scalars to vectors

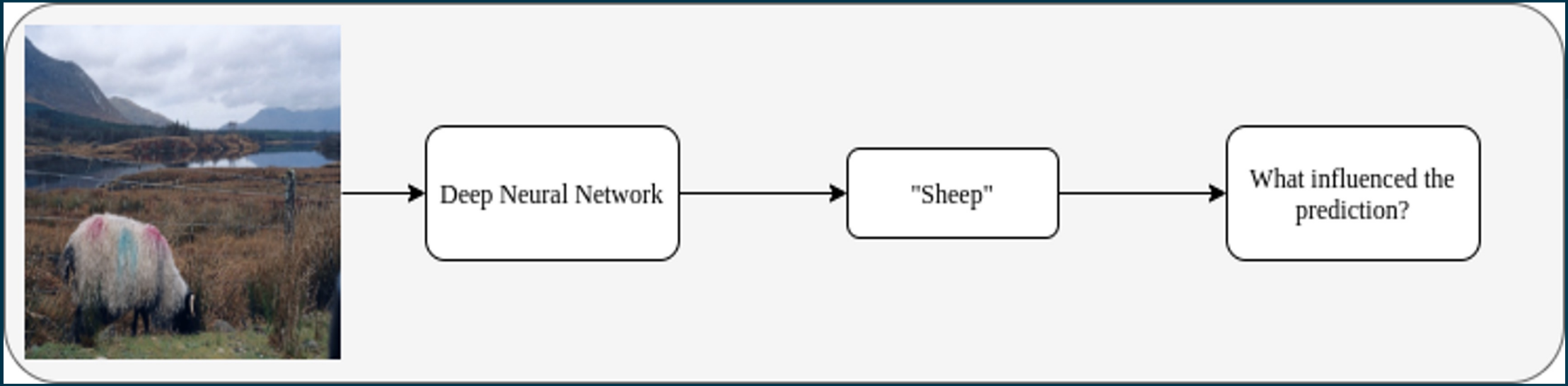
Kristoffer Knutsen Wickstrøm
UiT Machine Learning Group and Visual Intelligence

Schedule

- First lecture – Introduction to explainable artificial intelligence (XAI)
  - Why do we need explainability?
  - How do we get explainability?
  - Challenges in XAI
- Second lecture – XAI in representation learning
  - How to explain vectorial representations of data?
  - Why are standard XAI techniques not suitable.
  - Representation learning explainability with RELAX

What is explainability?

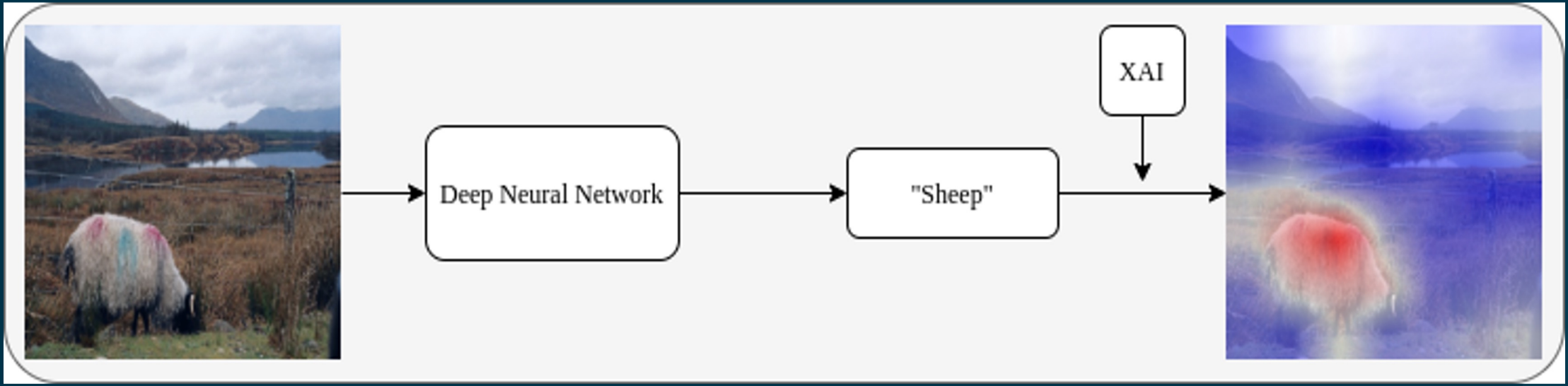
- A tool for answering the question “why?”[1]



[1]*A. Holzinger et al., “Explainable AI methods - a brief overview”. xxAI - Beyond Explainable AI, 2022*

What is explainability?

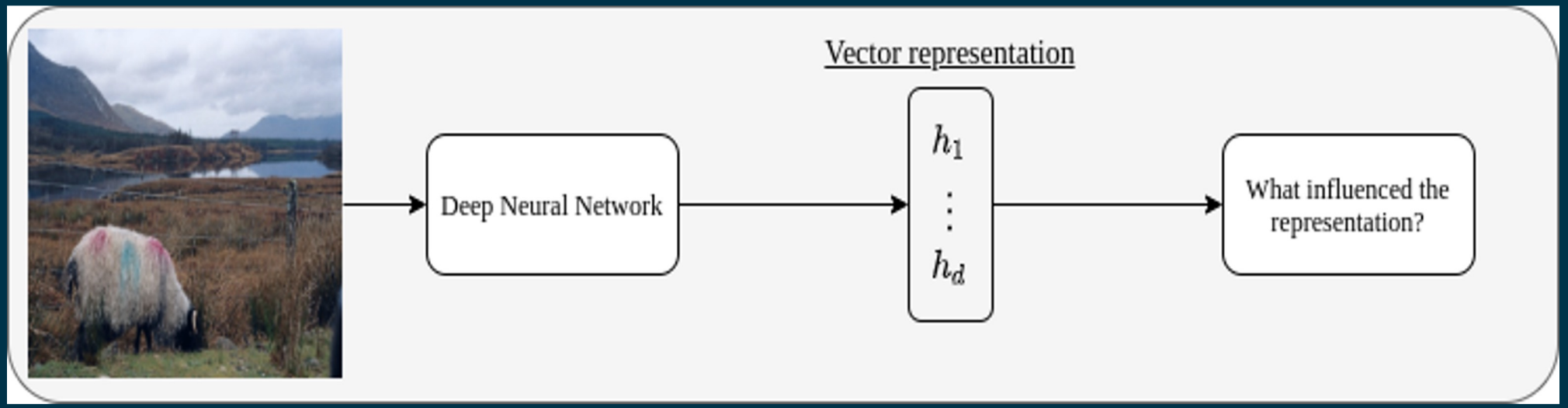
- A tool for answering the question “why?”[1]



[1]*A. Holzinger et al., “Explainable AI methods - a brief overview”. xxAI - Beyond Explainable AI, 2022*

How to go beyond explaining “just” decisions?

- A wide selection of methods exists for explaining predictions/scores
- How to explain representations?



Why can we not use standard explainability methods?

- Current methods often rely on labels and explaining a scalar value

Gradient explanation

$$\mathbf{g} = \frac{dy_c}{d\mathbf{x}}$$

Gradient of vector

$$? = \frac{d\mathbf{z}}{d\mathbf{x}}$$

Why can we not use standard explainability methods?

- Current methods often rely on labels and explaining a scalar value

Gradient explanation

$$\mathbf{g} = \frac{dy_c}{d\mathbf{x}}$$

(C, H, W)

Gradient of vector

$$? = \frac{d\mathbf{z}}{d\mathbf{x}}$$

(d, C, H, W)

Why can we not use standard explainability methods?

- Current methods often rely on labels and explaining a scalar value

Layerwise relevance propagation[1]

$$R_{i\leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j + \epsilon \cdot \text{sign}(z_j)} R_j^{(l+1)}$$

LIME training procedure[2]

The recipe for training local surrogate models:

- Select your instance of interest for which you want to have an explanation of its black box prediction.
- Perturb your dataset and get the black box predictions for these new points.
- Weight the new samples according to their proximity to the instance of interest.
- Train a weighted, interpretable model on the dataset with the variations.
- Explain the prediction by interpreting the local model.

[1]S. Bach et al., "On pixelwise explanations for non-linear classifier decisions by layer-wise relevance propagation". PLOS ONE, 2015
[2]M. Ribeiro et al., "Why should I trust you?: explaining the predictions of any classifier". KDD, 2016

Why can we not use standard explainability methods?

- Current methods often rely on labels and explaining a scalar value

Layerwise relevance propagation[1]

$$R_{i\leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j + \epsilon \cdot \text{sign}(z_j)} R_j^{(l+1)}$$

LIME training procedure[2]

The recipe for training local surrogate models:

- Select your instance of interest for which you want to have an explanation of its black box prediction.
- Perturb your dataset and get the black box predictions for these new points.
- Weight the new samples according to their proximity to the instance of interest.
- Train a weighted, interpretable model on the dataset with the variations.
- Explain the prediction by interpreting the local model.

[1] *S. Bach et al., "On pixelwise explanations for non-linear classifier decisions by layer-wise relevance propagation". PLOS ONE, 2015*
[2] *M. Ribeiro et al., "Why should I trust you?: explaining the predictions of any classifier". KDD, 2016*

How to explain representations of data?

- A new direction in explainability research
- Two approaches:
  - Adapt current methods to handle vectors
  - Design new methods specifically for the task of explaining vectors

Adapting existing methods for vectors

- Idea: explain each component and aggregate[1]
- Problem: many elements of representation can be uninformative

Gradient explanation

$$\mathbf{g} = \frac{dy_c}{d\mathbf{x}}$$

Gradient of vector

$$? = \frac{d\mathbf{z}}{d\mathbf{x}}$$

Gradient explanation of vector

$$\mathbf{g} = \frac{1}{D}\sum_{d=1}^{D} \nabla f(\mathbf{X})_d,$$

[1] *J. Crabbé et al., "Label-free explainability for unsupervised models". ICML, 2022*

Adapting existing methods for vectors

- Idea: explain each component and aggregate[1]
- Problem: many elements of representation can be uninformative

Gradient explanation

$$\mathbf{g} = \frac{dy_c}{d\mathbf{x}}$$

(C, H, W)

Gradient of vector

$$? = \frac{d\mathbf{z}}{d\mathbf{x}}$$

(d, C, H, W)

Gradient explanation of vector

$$\mathbf{g} = \frac{1}{D}\sum_{d=1}^{D} \nabla f(\mathbf{X})_d,$$

(C, H, W)

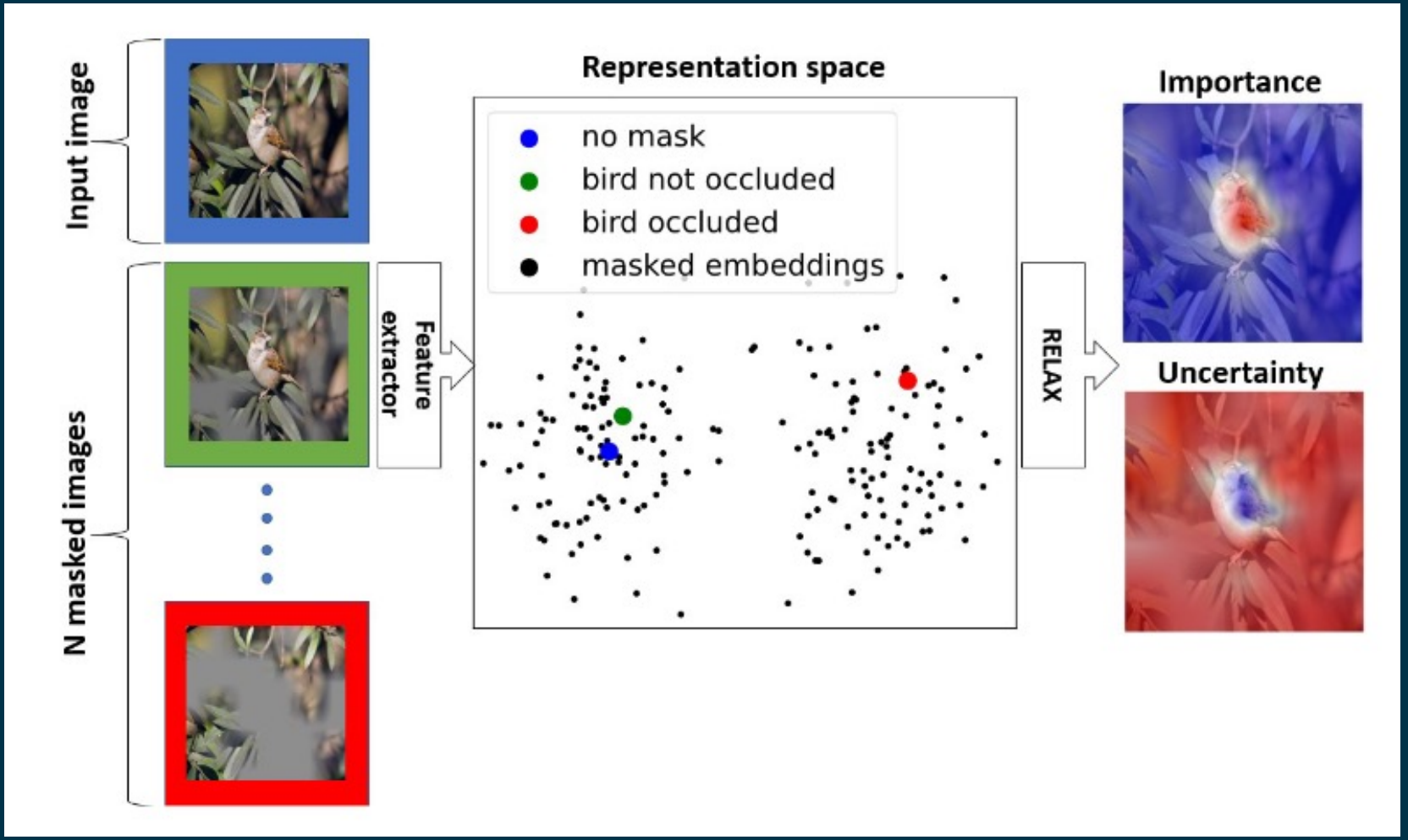
[1] *J. Crabbé et al., "Label-free explainability for unsupervised models". ICML, 2022*

RELAX: Representation Learning Explainability[1]

- Perturbation-based explainability method for representations.
- Inspired by RISE-framework.[2]
- Compare non-perturbed representation with perturbed representation.

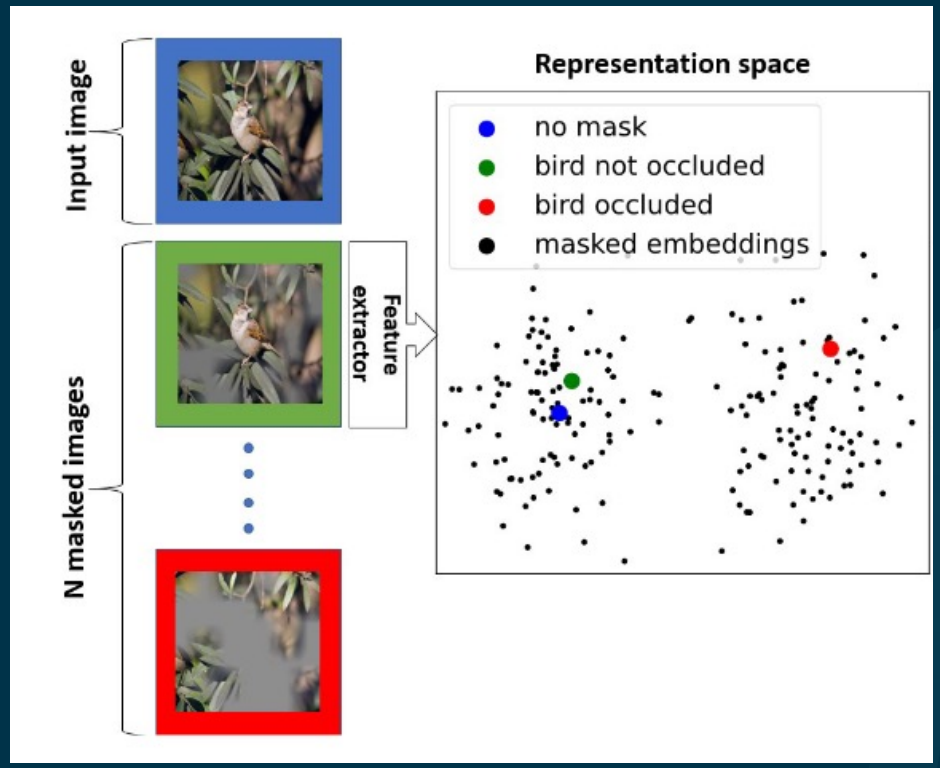
[1] *K. Wickstrøm et al., "RELAX: representation learning explainability". IJCV, 2023*
[2] *V. Petsiuk et al., "RISE: Randomized Input Sampling for Explanation of Black-box Models", BMVC, 2018.*

RELAX: Representation Learning Explainability[1]



Explaining representations with RELAX

- h and $\bar{h}$ should be similar if non-informative parts are masked out.
- Opposite for informative parts.
- Define importance as:
- $R_{ij} = \mathrm{E}_M\left[s\left(h, \bar{h}\right) M_{ij}\right]$
- Mask continuous in (0,1)



Explaining representations with RELAX

- $R_{ij} = \mathrm{E}_M\left[s\left(h, \bar{h}\right) M_{ij}\right]$
- Not computationally feasible to integrate over entire support
- Approximate by sampling:
- $R_{ij} = \frac{1}{N} \sum_{n=1}^{N} s\left(h, \bar{h}_n\right) M_{ij}(n)$
- s: cosine similarity
  - No hyperparameters
  - Often used in self-supervised frameworks

What is RELAX measuring?

- RELAX from a kernel perspective.
- Assume similarity measure is a kernel

$$
\begin{aligned}
R_{ij} &\overset{MC}{\approx} \frac{1}{Np}\sum_{n=1}^{N}\kappa(\mathbf{h},\bar{\mathbf{h}}_n)M_{ij}(n) \\
&= \frac{1}{Np}\sum_{n=1}^{N}\langle\phi(\mathbf{h}),\phi(\bar{\mathbf{h}}_n)\rangle M_{ij}(n) \\
&= \langle\phi(\mathbf{h}),\frac{1}{Np}\sum_{n=1}^{N}\phi(\bar{\mathbf{h}}_n)M_{ij}(n)\rangle.
\end{aligned}
$$

- Related to Parzen density estimation

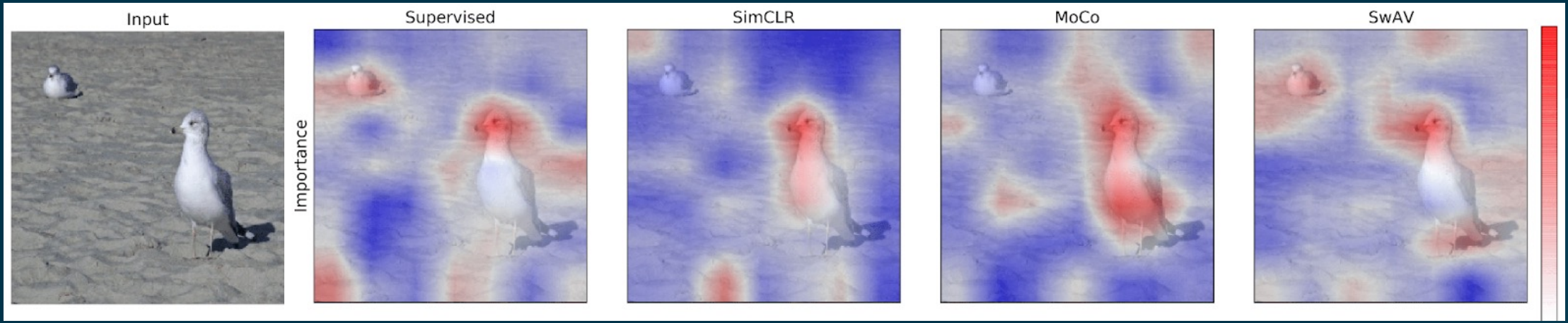
Theorem 3. *Suppose $s(\cdot,\cdot)$ is a valid Parzen window (Theodoridis and Koutroumbas, 2009). Then:*

$$\bar{R}_{ij} \propto p_{ij}(\mathbf{h}), \tag{6}$$

where $p_{ij}(\cdot)$ is a weighted Parzen density estimate (Parzen, 1962) of the density of the masked embeddings:

$$p_{ij}(\cdot) = \frac{1}{\sum_{n'=1}^{N} M_{ij}(n')}\sum_{n=1}^{N} s(\cdot,\bar{\mathbf{h}}_n)M_{ij}(n). \tag{7}$$

Qualitative results



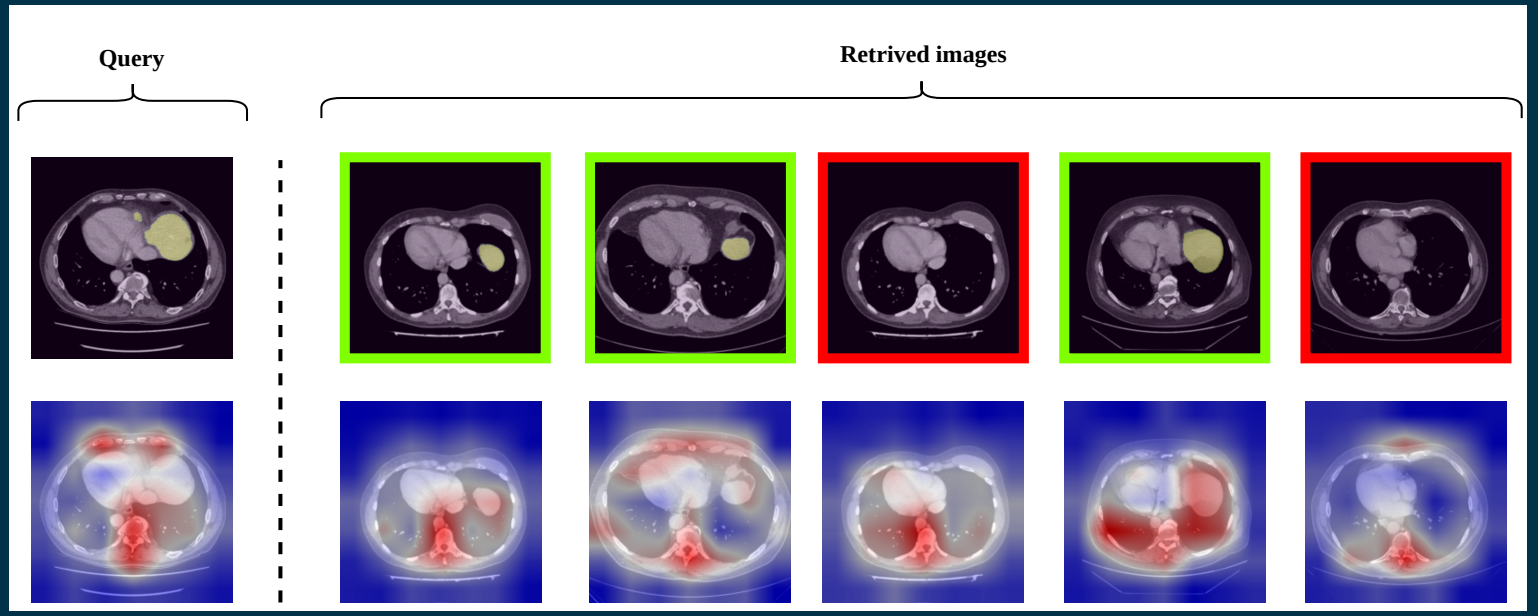
Quantitative results

Scores	Methods	Supervised		SimCLR		SwAV	
		COCO	VOC	COCO	VOC	COCO	VOC
Pointing game	Saliency	67.1 ± 0.0	82.8 ± 0.0	59.9 ± 0.0	75.9 ± 0.0	60.0 ± 0.0	76.3 ± 0.0
	Smooth Saliency	62.8 ± 0.0	79.5 ± 0.0	60.1 ± 0.0	75.9 ± 0.0	59.8 ± 0.0	76.4 ± 0.0
	Guided Saliency	66.6 ± 0.0	82.9 ± 0.0	58.4 ± 0.0	73.3 ± 0.0	59.5 ± 0.0	75.8 ± 0.0
	Integrated Gradients	47.8 ± 0.0	59.1 ± 0.0	32.9 ± 0.0	48.2 ± 0.0	36.5 ± 0.0	51.5 ± 0.0
	Grad CAM	66.8 ± 0.4	78.7 ± 0.5	47.7 ± 0.7	57.0 ± 0.6	48.7 ± 1.0	58.6 ± 0.8
	RELAX	**72.6 ± 0.1**	**86.6 ± 0.2**	**68.7 ± 0.3**	**85.2 ± 0.3**	**67.8 ± 0.2**	**84.7 ± 0.2**
	U-RELAX	72.1 ± 0.3	86.4 ± 0.4	68.6 ± 0.2	85.0 ± 0.5	66.7 ± 0.7	84.1 ± 0.4
Top k	Saliency	62.2 ± 0.0	80.1 ± 0.0	56.5 ± 0.0	71.3 ± 0.0	56.5 ± 0.0	71.4 ± 0.0
	Smooth Saliency	59.2 ± 0.0	74.1 ± 0.0	56.4 ± 0.0	71.1 ± 0.0	56.4 ± 0.0	71.3 ± 0.0
	Guided Saliency	62.2 ± 0.0	80.2 ± 0.0	55.1 ± 0.0	69.0 ± 0.0	56.3 ± 0.0	71.1 ± 0.0
	Integrated Gradients	47.7 ± 0.0	61.0 ± 0.0	35.4 ± 0.0	52.8 ± 0.0	33.2 ± 0.0	49.0 ± 0.0
	Grad CAM	64.0 ± 0.0	78.3 ± 0.0	43.6 ± 0.0	55.3 ± 0.0	43.1 ± 0.1	54.8 ± 0.0
	RELAX	**72.8 ± 0.4**	**86.9 ± 0.1**	**69.0 ± 0.3**	**85.6 ± 0.2**	**68.1 ± 0.4**	**85.1 ± 0.2**
	U-RELAX	72.2 ± 0.4	86.5 ± 0.2	68.8 ± 0.4	85.3 ± 0.1	66.6 ± 0.4	84.2 ± 0.3
Relevance rank	Saliency	46.8 ± 0.0	59.5 ± 0.0	41.2 ± 0.0	53.6 ± 0.0	40.9 ± 0.0	53.4 ± 0.0
	Smooth Saliency	42.6 ± 0.0	54.6 ± 0.0	41.1 ± 0.0	53.4 ± 0.0	40.9 ± 0.0	53.3 ± 0.0
	Guided Saliency	46.8 ± 0.0	59.8 ± 0.0	40.6 ± 0.0	53.0 ± 0.0	40.9 ± 0.0	53.3 ± 0.0
	Integrated Gradients	38.4 ± 0.0	51.9 ± 0.0	31.9 ± 0.0	47.2 ± 0.0	32.3 ± 0.0	48.3 ± 0.0
	Grad CAM	46.0 ± 0.0	60.2 ± 0.0	37.5 ± 0.0	50.7 ± 0.0	37.8 ± 0.0	50.9 ± 0.0
	RELAX	**56.4 ± 0.0**	**70.2 ± 0.1**	**54.2 ± 0.2**	**69.8 ± 0.1**	**52.4 ± 0.1**	**69.1 ± 0.0**
	U-RELAX	52.4 ± 0.0	64.7 ± 0.1	50.7 ± 0.1	63.3 ± 0.1	46.2 ± 0.1	59.5 ± 0.0

Higher is better and bold numbers highlight the top performance. Results show that our method improves on the baseline across all scores

Medical example

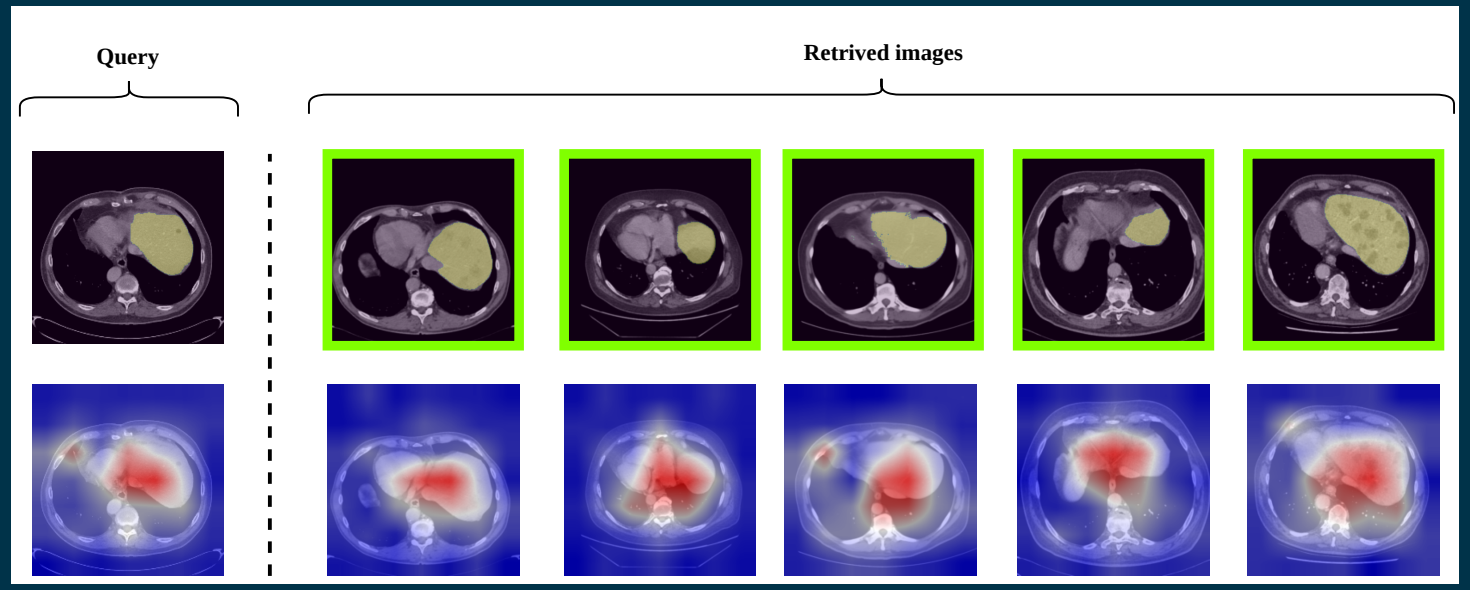
- Feature extractor trained for Imagenet classification.
- Contet-based image retrieval of CT liver images.



Wickstrøm, et.al., “A clinically motivated self-supervised approach for content-based image retrieval of CT liver images”, Computerized Medical Imaging and Graphics, 2023.

Medical example

- Feature extractor trained using self-supervised learning and clinical information.
- Contet-based image retrieval of CT liver images.



Wickstrøm, et.al., “A clinically motivated self-supervised approach for content-based image retrieval of CT liver images”, Computerized Medical Imaging and Graphics, 2023.

Summary RELAX

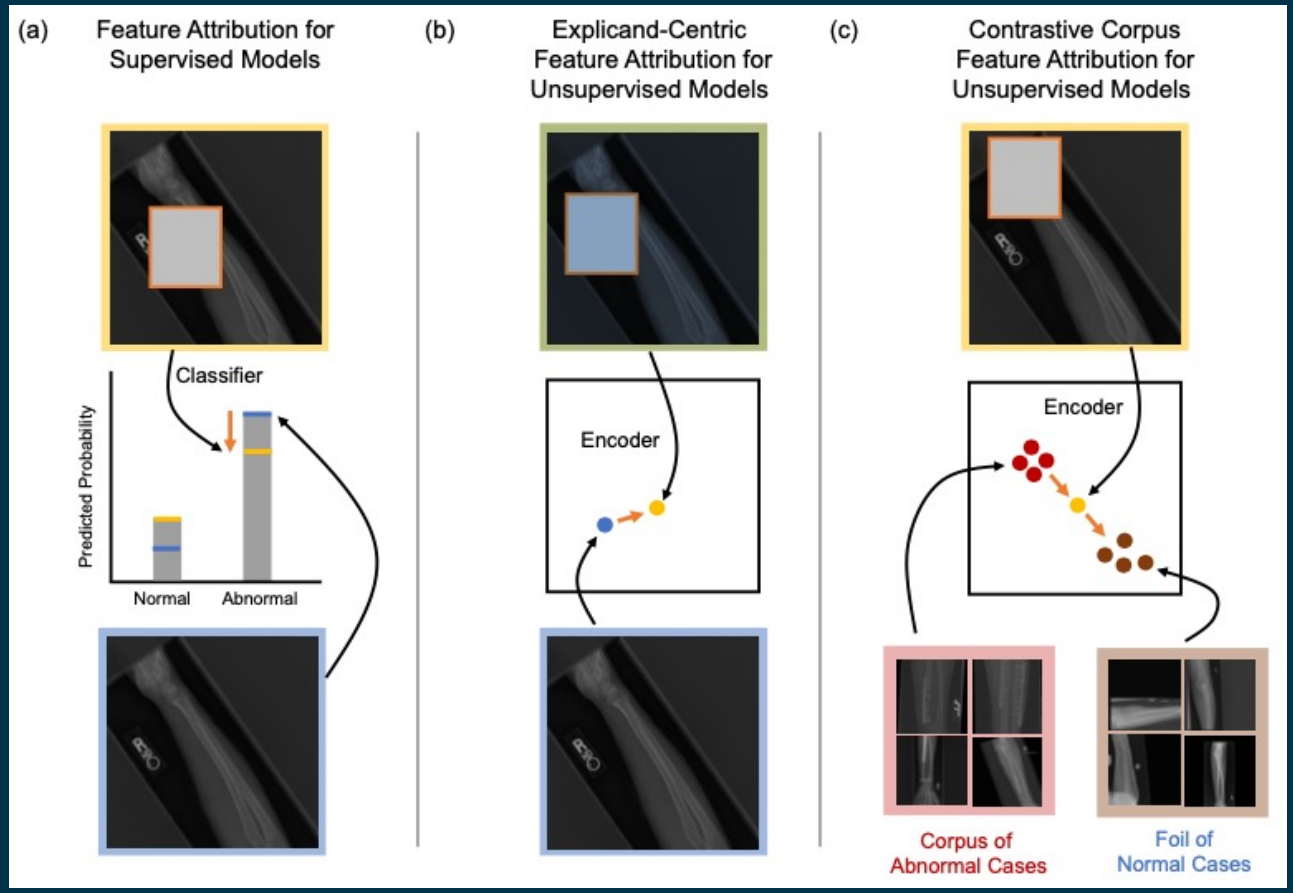
- A framework for explaining representations.
- When should you consider using RELAX?
  - Output is vector.
  - No labels available.
- Use case:
  - Retrieval
  - Dimensionality reduction
  - Debugging / model development
  - ++

How to explain representations in relation to other representations?

- RELAX and label-free explainability:
  - Explain the representation of a single image.
- How to explain a representation in relation to others?[3]

[1]C. LIn et al., “Contrastive corpus attribution for explaining representations”. ICLR, 2023

COCOA: Contrastive Corpus Attributions



[1]Lin et.al. *Contrastive Corpus Attribution for Explaining Representations*, ICLR 2023.

Picking the corpus and foil set

- Corpus set:
  - Randomly sampled images from training set.
- Foil set:
  - Randomly selected images from same class as image being explained.
  - Same image.
- Limitations:
  - Might need label information.

COCOA experiments

Summary of representation learning explainability

- A new direction within explainability.
- Useful to understand what feature extractors trained using self-supervised learning are encoding.

Exercise slide

- Find exercises and notebook at:
  - https://github.com/Wickstrom/ssl-summer-school-dtu