Crossing Cross-Entropy:

The Power of Provably Faithful Interpretability

J. Setpal

October 25, 2024

What is Interpretability?

What is Interpretability?

Interpretability within Machine Learning is the **degree** to which we can understand the **cause** of a decision, and use it to consistently <u>predict the model's prediction</u>.

What is Interpretability?

Interpretability within Machine Learning is the **degree** to which we can understand the **cause** of a decision, and use it to consistently predict the model's prediction.

Traditionally, interpretability & performance is seen as a trade-off.[a]

What is Interpretability?

Interpretability within Machine Learning is the **degree** to which we can understand the **cause** of a decision, and use it to consistently predict the model's prediction.

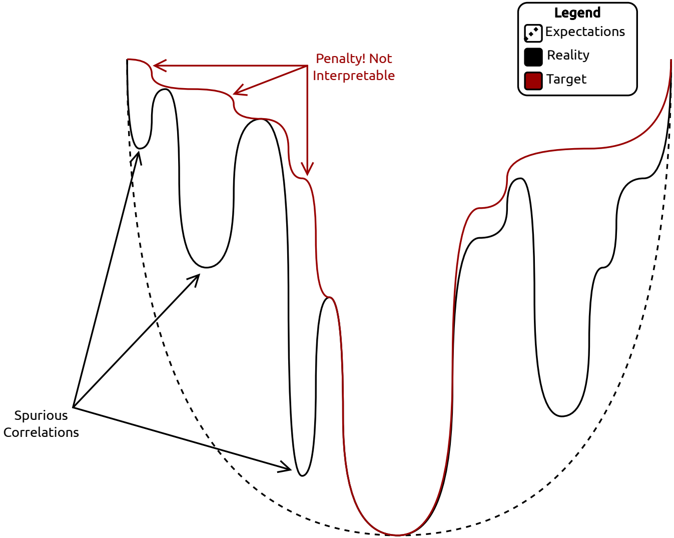
Traditionally, interpretability & performance is seen as a trade-off.[a]

Our work demonstrates a deep intersect between these two *seemingly* orthogonal research foci.

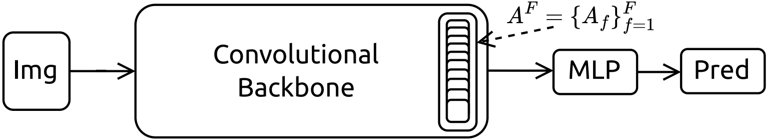
[a] Dziugaite, Ben-David, Roy. [Arxiv 2020]

Overarching Motivation

Goal: Constrain learning to interpretable "sanity checks".



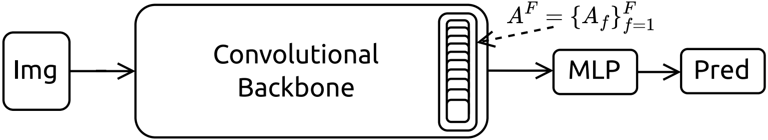
HiResCAMs are a provably faithful interpretability technique:



$$\tilde{\mathcal{A}}_c^{\text{HiResCAM}} = \sum_{f=1}^{F} \frac{\partial \hat{y}_c}{\partial A_f} \odot A_f \tag{1}$$

Contrastive Activation Maps (1/2)

HiResCAMs are a provably faithful interpretability technique:



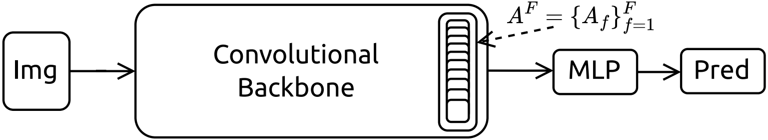
$$\tilde{\mathcal{A}}_c^{\text{HiResCAM}} = \sum_{f=1}^{F} \frac{\partial \hat{y}_c}{\partial A_f} \odot A_f \tag{1}$$

Provably faithful because:

$$\hat{y}_c = \sum_{d_1,d_2}^{D_1,D_2} \tilde{\mathcal{A}}_{c,d_1,d_2}^{\text{HiResCAM}} + b_c \tag{2}$$

Contrastive Activation Maps (1/2)

HiResCAMs are a provably faithful interpretability technique:



$$\tilde{\mathcal{A}}_c^{\text{HiResCAM}} = \sum_{f=1}^{F} \frac{\partial \hat{y}_c}{\partial A_f} \odot A_f \tag{1}$$

Provably faithful because:

$$\hat{y}_c = \sum_{d_1,d_2}^{D_1,D_2} \tilde{\mathcal{A}}_{c,d_1,d_2}^{\text{HiResCAM}} + b_c \tag{2}$$

However, softmax-activated multi-class classification relies on **inter-class logit differences!!!**, while HiResCAMs only re-construct *absolute values*.

Contrastive Activation Maps (2/2)

To recover logit differences, we define **ContrastiveCAMs**:

$$\tilde{\mathcal{A}}^{\text{contrastive}}_{(c_t, c_{t'})} := \left\{ \tilde{\mathcal{A}}^{\text{HiResCAM}}_{c_t} - \tilde{\mathcal{A}}^{\text{HiResCAM}}_{c_{t'}} \right\}^{|c|-1}_{c_{t'} \in c \setminus c_t} \tag{3}$$

[1] with subtle changes to the architecture

To recover logit differences, we define **ContrastiveCAMs**:

$$\tilde{\mathcal{A}}^{\text{contrastive}}_{(c_t, c_{t'})} := \left\{ \tilde{\mathcal{A}}^{\text{HiResCAM}}_{c_t} - \tilde{\mathcal{A}}^{\text{HiResCAM}}_{c_{t'}} \right\}^{|c|-1}_{c_{t'} \in c \setminus c_t} \tag{3}$$

Next, we can now define an objective equivalent to cross-entropy:[1]

$$\max_{\theta} \sum_{d_1, d_2}^{D_1, D_2} \tilde{\mathcal{A}}^{\text{contrastive}}_{(c, c'), d_1, d_2} \; \forall c' \in \mathbb{Z}_+(|c| - 1) \tag{4}$$

[1] with subtle changes to the architecture

Contrastive Activation Maps (2/2)

To recover logit differences, we define **ContrastiveCAMs**:

$$\tilde{\mathcal{A}}^{\text{contrastive}}_{(c_t, c_{t'})} := \left\{ \tilde{\mathcal{A}}^{\text{HiResCAM}}_{c_t} - \tilde{\mathcal{A}}^{\text{HiResCAM}}_{c_{t'}} \right\}^{|c|-1}_{c_{t'} \in c \setminus c_t} \tag{3}$$

Next, we can now define an objective equivalent to cross-entropy:[1]

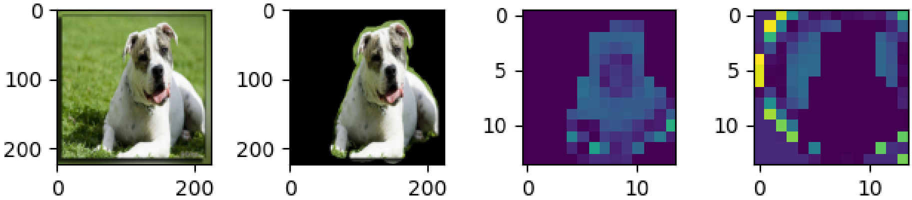
$$\max_{\theta} \sum_{d_1, d_2}^{D_1, D_2} \tilde{\mathcal{A}}^{\text{contrastive}}_{(c, c'), d_1, d_2} \; \forall c' \in \mathbb{Z}_+(|c| - 1) \tag{4}$$

With one key difference: **we've preserved spatial information**.

[1] with subtle changes to the architecture

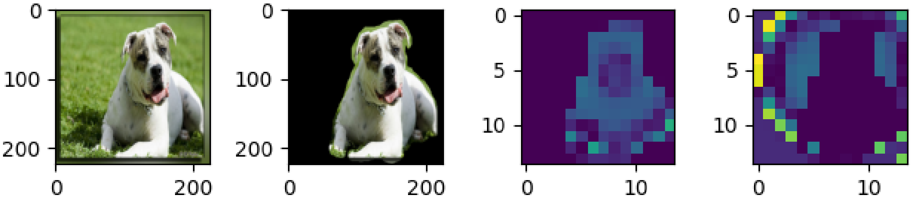
Understanding the Problem

We evaluated models trained using Cross-Entropy Loss using ContrastiveCAMs:



Understanding the Problem

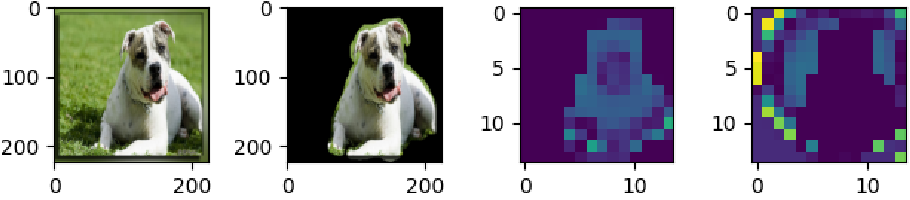
We evaluated models trained using Cross-Entropy Loss using ContrastiveCAMs:



Problem Statement: For image classification tasks, Cross-Entropy motivates learning *spurious correlations*.

Understanding the Problem

We evaluated models trained using Cross-Entropy Loss using ContrastiveCAMs:

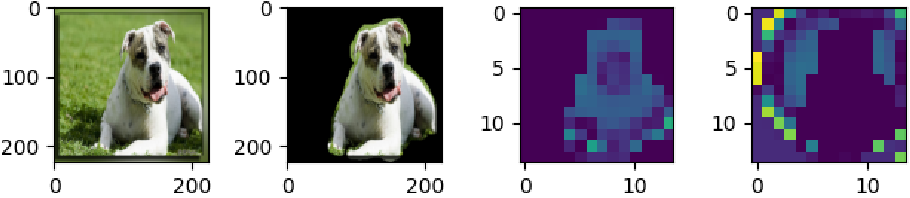


Problem Statement: For image classification tasks, Cross-Entropy motivates learning *spurious correlations*.

Provided the target class contains the largest logit, cross-entropy is happy.

Understanding the Problem

We evaluated models trained using Cross-Entropy Loss using ContrastiveCAMs:



Problem Statement: For image classification tasks, Cross-Entropy motivates learning *spurious correlations*.

Provided the target class contains the largest logit, cross-entropy is happy.

We can use ContrastiveCAMs to optimize our network under a "foreground-only" constraint!

Contrastive Optimization

Cross-Entropy Loss is defined as follows:

$$\mathcal{J}(y, \hat{y}) = -\sum_{c \in C} y_c \log(\sigma_{\text{softmax}}(\hat{y}_c)) \tag{5}$$

Contrastive Optimization

Cross-Entropy Loss is defined as follows:

$$\mathcal{J}(y, \hat{y}) = -\sum_{c \in C} y_c \log(\sigma_{\text{softmax}}(\hat{y}_c)) \tag{5}$$

We derive cross-entropy as function of ContrastiveCAMs, then **penalize the background**:

$$\begin{aligned} &\mathcal{J}(\{\tilde{\mathcal{A}}_{c,i}^{\text{contrastive}}\}_i^{|c|}, h, c) = \\ &-\log\left(\frac{1}{\sum_i \exp\left(-\sum h \odot \tilde{\mathcal{A}}_{(c,i)}^{\text{contrastive}} + \sum \left|(1-h) \odot \tilde{\mathcal{A}}_{(c,i)}^{\text{contrastive}}\right|\right)}\right) \end{aligned} \tag{6}$$

Contrastive Optimization

Cross-Entropy Loss is defined as follows:

$$\mathcal{J}(y, \hat{y}) = -\sum_{c \in C} y_c \log(\sigma_{\text{softmax}}(\hat{y}_c)) \tag{5}$$

We derive cross-entropy as function of ContrastiveCAMs, then **penalize the background**:

$$\begin{aligned} &\mathcal{J}(\{\tilde{\mathcal{A}}_{c,i}^{\text{contrastive}}\}_i^{|c|}, h, c) = \\ &-\log\left(\frac{1}{\sum_i \exp\left(-\sum h \odot \tilde{\mathcal{A}}_{(c,i)}^{\text{contrastive}} + \sum \left|(1-h) \odot \tilde{\mathcal{A}}_{(c,i)}^{\text{contrastive}}\right|\right)}\right) \end{aligned} \tag{6}$$

The model learns to:

1. Use *only* the foreground to base it's prediction.
2. Treat the **background as noise**, and <u>learn invariance to it</u>.

Results (so far) (1/2)

In-distribution fine-grained image classification on Oxford-IIIT Pets:

Method	Valid CE Loss	Train Acc	Valid Acc
Cross-Entropy	3.605	5.1%	5.2%
Interpretable (Ours)	**3.159**	**96.9%**	**51.5%**

Results (so far) (1/2)

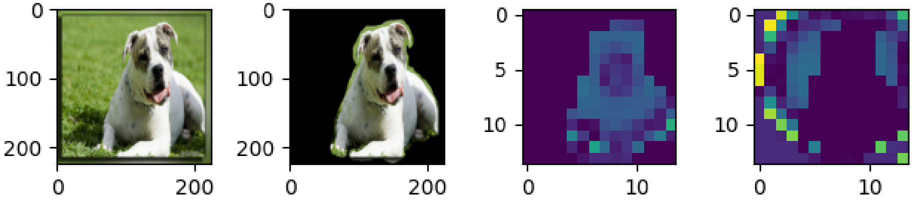
In-distribution fine-grained image classification on Oxford-IIIT Pets:

Method	Valid CE Loss	Train Acc	Valid Acc
Cross-Entropy	3.605	5.1%	5.2%
Interpretable (Ours)	**3.159**	**96.9%**	**51.5%**

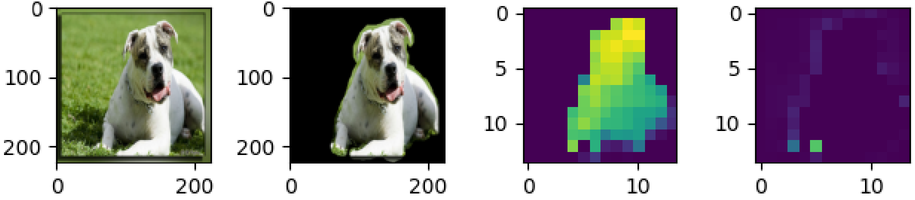
Out-of-Distribution generalization performance on Dogs v/s Cats dataset:

Method	Accuracy
Cross-Entropy	77.0%
Interpretable (Ours)	**83.4%**

Before:



After:



Next Steps

We're targeting the following next steps:

1. Exploring a level deeper: unpacking $\sum_{f=1}^{F} A_f$.
2. Identifying the cause of the generalization gap in multiclass setting.
3. Evaluating adversarial robustness.
4. Mechanistic Interpretability study (circuit identification).
5. Evaluating the approach at scale, using ImageNet-S.

Next Steps

We're targeting the following next steps:

1. Exploring a level deeper: unpacking $\sum_{f=1}^{F} A_f$.
2. Identifying the cause of the generalization gap in multiclass setting.
3. Evaluating adversarial robustness.
4. Mechanistic Interpretability study (circuit identification).
5. Evaluating the approach at scale, using ImageNet-S.

Long-Term Objective: Build proof-backed approaches to optimization that learn **intrinsically interpretable neural networks**.

Thank you!

Have an awesome rest of your day!

Slides: `https://cs.purdue.edu/homes/jsetpal/slides/cont-opt.pdf`
Code: `https://dagshub.com/jinensetpal/contrastive-optimization`

Homepage: `https://jinen.setpal.net/`