CAPE: CAM as a Probabilistic Ensemble for Enhanced DNN Interpretation

Townim Faisal Chowdhury[1], Kewen Liao[2], Vu Minh Hieu Phan[1], Minh-Son To[3], Yutong Xie[1], Kevin Hung[4], David Ross[4], Anton van den Hengel[1], Johan W. Verjans[1], Zhibin Liao[1†]

[1]Australian Institute for Machine Learning, University of Adelaide, Australia, [2]Australian Catholic University, Australia,
[3]Flinders University, Australia, [4]SA Pathology, Central Adelaide Local Health Network, Australia

Abstract

Deep Neural Networks (DNNs) are widely used for visual classification tasks, but their complex computation process and black-box nature hinder decision transparency and interpretability. Class activation maps (CAMs) and recent variants provide ways to visually explain the DNN decision-making process by displaying 'attention' heatmaps of the DNNs. Nevertheless, the CAM explanation only offers relative attention information, that is, on an attention heatmap, we can interpret which image region is more or less important than the others. However, these regions cannot be meaningfully compared across classes, and the contribution of each region to the model's class prediction is not revealed. To address these challenges that ultimately lead to better DNN Interpretation, in this paper, we propose CAPE, a novel reformulation of CAM that provides a unified and probabilistically meaningful assessment of the contributions of image regions. We quantitatively and qualitatively compare CAPE with state-of-the-art CAM methods on CUB and ImageNet benchmark datasets to demonstrate enhanced interpretability. We also test on a cytology imaging dataset depicting a challenging Chronic Myelomonocytic Leukemia (CMML) diagnosis problem. Code is available at: **https://github.com/AIML-MED/CAPE**.

1. Introduction

Deep neural networks (DNNs), despite achieving superior performance on various tasks such as computer vision and natural language processing, are known to be black boxes [23] that lack the ability to explain their decision-making process. The black-box nature is commonly regarded as a result of the complex model structure characterized by stacked computation layers, involving non-linear functions and many model parameters. Explainable DNN decisions are crucial to many life-critical scenarios [26] such as AI-powered autonomous driving and medical diagnostics. Taking the example of healthcare applications [2], decision transparency is critical for doctors to understand and trust AI analysis, and to use AI to make insightful and accurate diagnoses or decisions.

DNN interpretability is an emerging and actively studied research field. For visual classification tasks, a common type of DNN interpretability analysis is to explain DNN outputs via finding and displaying model attention on the input image, *i.e.*, identifying which image regions the model focused on during the decision-making process. This type of visual explanation can be achieved via methods of gradient-based attention visualization [25], perturbation-based input manipulation [6, 21], and class activation map (CAM)-based visualization [11, 24]. In particular, CAM is an inherent intermediate step of DNN prediction which represents the actual region relevance produced by the network. CAM stands out due to its efficient feedforward process, yet its attention values can not directly explain and compose model outcomes. Specifically, CAM values are class-wise relative probabilities. They only represent the relative region importance compared to the highest attention value within each class map. Thus, CAM values provide a limited explanation within the context of one target class. This means that they are incomparable between classes, and cannot explain image-level predictions. Take the CAM visualization in Fig. 1 as an example, CAM assigns similar attention values to two dog breed classes Siberian Husky and Alaskan Malamute. Differencing the two CAM maps between the breeds fails to yield meaningful comparisons.

The limited analytical capability of current CAM-based approaches hinders their use in many downstream applications. For example, fine-grained classification analysis requires the model's ability to discriminate regions between closely related concepts. In addition, for tasks such as weakly supervised semantic segmentation, CAM thresholding is employed to initialize a segmentation method [13] but the threshold choice is often arbitrary without any semantic meaning attached.

In this paper, we reformulate CAM as a Probabilistic

†Corresponding author.

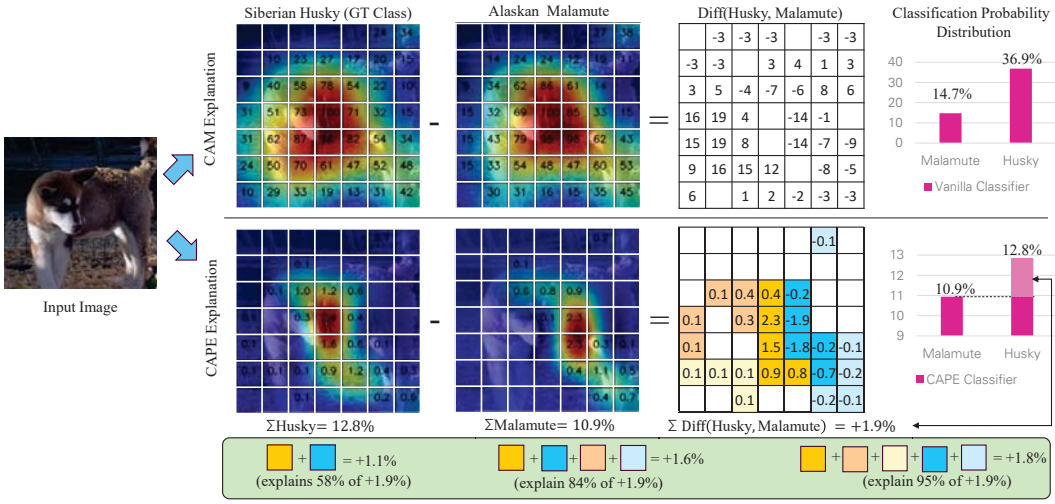

Figure 1. The comparison between CAM and the proposed CAPE explanation methods for a fine-grained class difference analysis example between Siberian Husky (Husky) and Alaskan Malamute (Malamute) classes on ImageNet. We overlay the explanation values before up-sampling on top of the produced heatmaps. CAM explanation is class independent which highlights similar regions for similar object classes, making the explanation maps incomparable. Instead, CAPE-produced explanation values (before up-sampling and min-max normalization) are probability values for each spatial location (image region) and class combination. We color code the top-5, next-5 (top-6 to top-10), *etc*., for the positive values (*i.e*., more Husky) and the negative values (*i.e*., more Malamute) on the Diff graph. The green box shows an example analysis of the $+1.9\%$ class difference by summing the color-coded regions and demonstrating to what levels they explain the class difference.

Ensemble, and name it CAPE. Diverging from the current CAM methods, CAPE's activation map seizes the probabilistic and absolute contributions of each image region toward class predictions while enabling meaningful comparisons between classes. As illustrated in Fig. 1, CAPE enforces a direct composition relationship between the overall model prediction and image region contributions. Our main contributions are summarized as follows:

- We propose a novel CAPE method to explicitly capture the relationship between the model's attention map and the decision process. For each class, the summation of the image region-wise attention values in CAPE is identical to the image-level prediction, providing a basis for the analytical understanding of the model attention.
- CAPE inference is efficient, introducing nearly zero extra model parameters and only takes a feed-forward inference to generate the explanation. By reformulating the softmax activation function, CAPE only adds a single trainable scalar, *i.e*., the Softmax temperature variable.
- We discover that CAPE explanation maps tend to highlight class discriminative regions whereas CAM explanation maps are independent for each class that also highlight class mutual regions. Hence, we further propose an alternative class mutual region inclusive CAPE explanation, namely the μ-CAPE (μ denotes 'mutual'), which restores the attention of CAPE on class mutual regions, achieving enhanced performance on commonly evaluated CAM interpretability metrics.

2. Related Work

In this section, we cover closely related works in interpretable machine learning and softmax-based aggregation.

Interpretable Machine Learning. For DNNs, the most common interpretation approaches are via saliency/heatmap types of visual explanation using model attention. The heatmap visualization methods can be loosely grouped into three categories: gradient-based attention visualization [25], class activation maps (CAMs) base visualization [11, 24], and perturbation-based [6, 21] input manipulation. Among them, the CAM method has gained significant research interest due to its ability to produce intuitive and high-quality visual attention [14]. CAM employs linear weighting of backbone-produced feature maps by using classification layer weights to produce a heatmap for each class category. The heatmap can correspond to class-wise salient regions of the input image. Based on how the CAM's weights are computed, recent works can be categorized into gradient-based and score-based methods. Gradient-based CAM methods [3, 10, 17, 24] use the gradients of a target class with respective to the activation maps as a CAM's weights to combine feature maps from the backbone. On the other hand, score-based methods such as Score-CAM [29] weights CAMs using a score computed by the increase of prediction confidence before and after masking the input image with initial CAM-produced attention. A more recent method, FD-CAM [14], leverages

gradient-based weights and score-based weights to obtain the CAM's weightings, benefiting from both schemes. The model-agnostic methods treat models as black boxes that can often be interpreted by input perturbation. LIME [21] and SHAP [16] are two typical model-agnostic methods to explain DNNs via input perturbation. They require additional sampling processes and fitting separate explainer models to approximate the original model's inference process, thereby consuming more computations.

Softmax-based Aggregation. The softmax function gives soft-weighted assignments of member contribution and has the nice property of summing to 1. Gao *et al*. [7] proposed a softmax-based local importance-based pooling method to down-sample spatial features in receptive fields. The attention mechanism [1] is another example of softmax-based feature aggregation which has been the core component of the modern transformer networks [27]. In capsule networks [22], softmax is used in the dynamic routing algorithm which can be viewed as a form of parallel attention mechanism to connect capsule layers. Our proposed interpretation method also utilizes softmax functions to construct probabilistically comparable attention, overcoming CAM's analytical limitation.

3. Methodology of Model Interpretation

3.1. Class Activation Maps (CAMs)

Let $\mathbf{x}$ be a single image and $y \in \mathcal{C}$ be the corresponding label, where $\mathcal{C}$ denotes the label set of the dataset. A function f produces a feature tensor from $\mathbf{x}$, *i.e.*, $\mathbf{F} = f(\mathbf{x}; \theta_f)$, where $\mathbf{F} \in \mathbb{R}^{H \times W \times K}$, H and W denote the spatial dimensions and K represents the number of channels. A typical deep learning classification model utilizes a sequence of a global average pooling layer and a fully-connected layer with a softmax activation function (referred as a vanilla classification layer) to produce the likelihood probability distribution $p(\mathcal{C}|\mathbf{x}, \theta)$ (denote as $\mathbf{p}$) from $\mathbf{F}$, which can be written as:

$$\mathbf{p} = \mathrm{softmax}_c\left(\mathbf{W}^\intercal \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{F}_{ij} + \mathbf{b}\right), \quad (1)$$

where $\theta = \{\theta_f, \mathbf{W} \in \mathbb{R}^{K \times |\mathcal{C}|}, \mathbf{b} \in \mathbb{R}^{|\mathcal{C}|}\}$ denotes the set of trainable parameters. The class activation map $\mathbf{M}_c$ for class $c \in \mathcal{C}$ is obtained by aggregating the activation maps $\mathbf{F}_k$ weighted by their class weights $\mathbf{W}_{kc}$, *i.e.*, $\mathbf{M}_c = \sum_{k=1}^{K} \mathbf{W}_{kc} \mathbf{F}_k$ where $\mathbf{M} \in \mathbb{R}^{H \times W \times |\mathcal{C}|}$. CAM is commonly used as a heatmap type of visualization. $\mathbf{M}_{ijc}$ indicates the importance of the activation of an image region at position (i, j) toward class c. For simplicity, we refer an image region as a *pixel* and a 3D indexed element (like $\mathbf{M}_{ijc}$) as a *voxel* hereafter. The common approach [12] to produce the explanation (or attention) map $\mathbf{E}$ of the classification model is to apply the rectifier transformation, up-sampling, and normalization in sequence:

$$\mathbf{E}_c^{\mathrm{CAM}} = \phi\big(\max(\mathbf{M}_c, 0)\big), \quad (2)$$

where $\phi(.)$ denotes a sequential process of up-sampling and min-max normalization operations.

As shown in the top row of Fig. 1, the normalized CAM explanation map $\mathbf{E}_{ijc}^{\mathrm{CAM}} \in [0, 100\%]$ are not comparable across classes. Note that the comparability could be restored if the min-max normalization uses the global maximum value of the entire $\mathbf{E}^{\mathrm{CAM}}$ but even if this is applied, $\mathbf{E}^{\mathrm{CAM}}$ values only explain the relative importance between voxels but not the absolute importance/contribution toward the model outcome. This raises an important research question of *whether CAM methods can show how much each image region actually contributes to the DNN decision*.

As the original CAM formulation ignores the bias term but the bias is involved in model outcome computation, we first restore the bias term by defining shifted CAM maps as $\mathbf{M}' = \mathbf{M} + \mathbf{b}$, and then we define:

$$p(\mathcal{C}|\mathbf{M}'_{ij}, \mathbf{x}, \theta) = \mathrm{softmax}_c(\mathbf{M}'_{ij}), \quad (3)$$

to represent the probability distribution of $\mathcal{C}$ at the pixel location (i, j). Then, a naive way to compute image level prediction $\mathbf{p}$ is to aggregate all pixel probability distributions by averaging:

$$\hat{\mathbf{p}} = \sum_{i=1}^{H} \sum_{j=1}^{W} \frac{p(\mathcal{C}|\mathbf{M}'_{ij}, \mathbf{x}, \theta)}{H \times W}. \quad (4)$$

Even though $\sum_c \hat{\mathbf{p}} = 1$ appears to satisfy the law of total probability, the model prediction $\mathbf{p}$ and the composed prediction $\hat{\mathbf{p}}$ are not identical, *i.e.*:

$$\mathrm{softmax}_c\Big(\frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathbf{M}'_{ij}\Big) \neq \sum_{i=1}^{H} \sum_{j=1}^{W} \frac{\mathrm{softmax}_c(\mathbf{M}'_{ij})}{H \times W}, \quad (5)$$

because the softmax function is neither additive (*i.e.*, $f(x + y) = f(x) + f(y)$), nor homogeneous (*i.e.*, $f(\alpha x) = \alpha f(x)$). Therefore, $p(c|\mathbf{M}'_{ij}, \mathbf{x}, \theta) = \frac{\mathrm{softmax}_c(\mathbf{M}'_{ij})}{H \times W}$ is not the true representation of the *voxel contribution to the overall decision* $\mathbf{p}$, and the exact compositional contribution of each voxel to the overall decision $\mathbf{p}$ is intractable.

3.2. CAM as a Probabilistic Ensemble (CAPE)

Since the voxel contributions to $\mathbf{p}$ are intractable, we propose to consider $\hat{\mathbf{p}}$ as the model's classification outcome for CAPE. This allows us to build the relationship between the voxel contributions and the model prediction outcome as a probabilistic ensemble of voxel contributions:

$$\hat{\mathbf{p}} = \sum_{i=1}^{H} \sum_{j=1}^{W} p(\mathcal{C}|\mathbf{M}'_{ij}, \mathbf{x}, \theta) p(\mathbf{M}'_{ij}|\mathbf{x}, \theta). \quad (6)$$

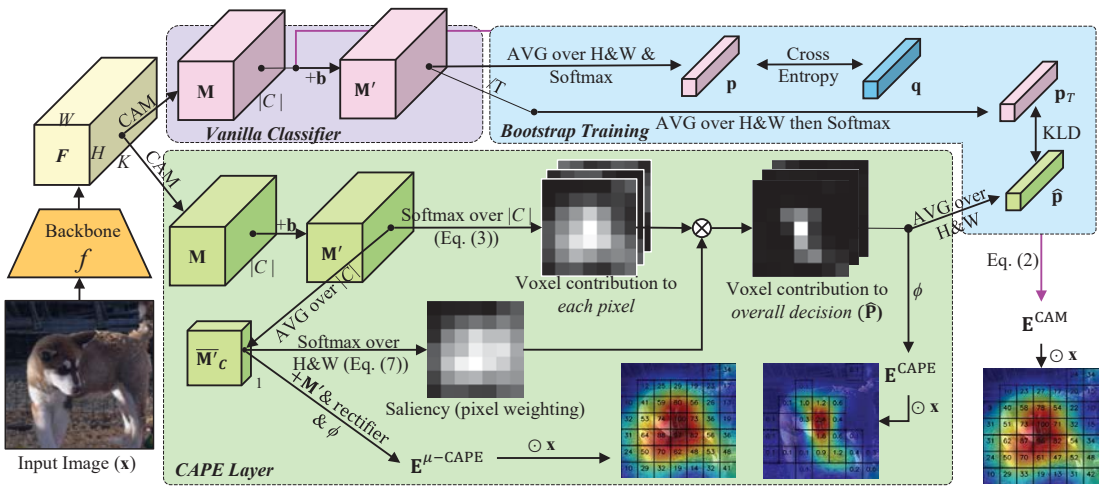

Figure 2. The overview of the proposed CAPE classification layer with bootstrap training. AVG stands for averaging.

The overview of the proposal is depicted in Fig. 2 and described as follows.

3.2.1 Image Region Importance (Saliency)

From Eq. (4), we know the naive representation of $p(\mathbf{M}'_{ij}|\mathbf{x},\theta)$ is $\frac{1}{H\times W}$. However, to make this formulation less rigid, we apply the softmax aggregation to compute the pixel weighting, also using $\mathbf{M}'$:

$$p(\mathbf{M}'_{ij}|\mathbf{x},\theta) = \text{softmax}_{ij}\big(\overline{\mathbf{M}'}_{\mathcal{C}}\big), \tag{7}$$

where the subscripts ij of the softmax function indicate that the softmax normalizes over both spatial dimensions. Here, $\overline{\mathbf{M}'}_{\mathcal{C}} = \frac{1}{|\mathcal{C}|}\sum_{c\in\mathcal{C}}\mathbf{M}'_c$ denotes the average operation over the classes of $\mathbf{M}'$. The rationale behind the usage of $\overline{\mathbf{M}'}_{\mathcal{C}}$ comes from the concept of saliency. When a spatial location processes high activation values in one pixel, it is likely that this pixel contains an object part and therefore should be focused. The average normalization is used to improve the numerical stability instead of using summation. Note that Eq. (7) reuses the values from $\mathbf{M}'_c$ which means that the modification to a neural network is only limited to the softmax function in the output layer without introducing additional network parameters.

3.2.2 CAPE Explanation

We can compute the *exact* decomposition of overall model prediction $\hat{\mathbf{p}}$ to the contribution from each voxel location $\hat{\mathbf{P}}_{ijc}$ by multiplying Eq. (3) and Eq. (7):

$$\hat{\mathbf{P}}_{ijc} = \frac{\exp(\mathbf{M}'_{ijc})}{\sum_{c'\in|\mathcal{C}|}\exp(\mathbf{M}'_{ijc'})}\cdot\frac{\exp(\overline{\mathbf{M}'}_{\mathcal{C}})_{ij}}{\sum_{i'j'}\exp(\overline{\mathbf{M}'}_{\mathcal{C}})_{i'j'}} \tag{8}$$

where $\hat{\mathbf{P}} \in \mathbb{R}^{H\times W\times|\mathcal{C}|}$. To form an explanation map, we perform the same operations in Eq. (2) and propose: $\mathbf{E}^{\text{CAPE}}_c = \phi\big(\hat{\mathbf{P}}_c\big)$. Note $\hat{\mathbf{P}}_{ijc} \in [0,1]$, therefore clipping negative values of $\hat{\mathbf{P}}_c$ is unnecessary.

3.2.3 μ-CAPE Explanation

Although the voxel contributions $\hat{\mathbf{P}}$ are the exact decomposition of the image-level prediction, they do not necessarily produce better quantitative measurement values for the commonly used CAM interpretability evaluation metrics (see CAPE (TS) and (PF) rows in Table 1). We found the reason being $\mathbf{E}^{\text{CAPE}}_c$ creates "sharper" attention than $\mathbf{E}^{\text{CAM}}_c$ and it tends to place high attention on the class discriminative regions of the objects (*e.g.*, what differentiates Husky and Malamute) but suppresses class mutual regions (*e.g.*, what is common between Husky and Malamute such as dog turso). We also found that the reason for the sharp attention lies in the super-linearity of the exponential function used in softmax (and when the rectifier function and min-max normalization are applied), which causes the relative distance between the outputs larger than the corresponding inputs, *i.e.*, $\frac{\exp(x)-\exp(y)}{\exp(x)-0} > \frac{x-y}{x-0}$, for any $x > y > 0$. Therefore, using $\mathbf{E}^{\text{CAPE}}_c \odot \mathbf{x}$ to reclassify the image will take away the decision support from the class mutual regions and cause a large change in the image classification confidence compared to using $\mathbf{E}^{\text{CAM}}_c \odot \mathbf{x}$, resulting lower measurement scores as shown in Table 1. A visual illustration can be found in Fig. 3 between CAM and CAPE (PF) columns.

Intuitively, to restore the class mutual regions, we would retrieve attention values before the softmax normalization, like $\mathbf{M}_{ijc}$ used in CAM. Therefore, we first transform Eq. (8) to a "single softmax" form, taking the advantage

of $\exp(x)\exp(y) = \exp(x+y)$:

$$\hat{\mathbf{P}}_{ijc} = \frac{\exp\big(\mathbf{M}'_{ijc} + (\overline{\mathbf{M}'}_{\mathcal{C}})_{ij}\big)}{\sum_{c'\in|\mathcal{C}|}\sum_{i'=1}^{H}\sum_{j'=1}^{W}\exp(\mathbf{M}'_{ijc'} + (\overline{\mathbf{M}'}_{\mathcal{C}})_{i'j'})}, \tag{9}$$

where the CAM equivalent term in CAPE is $\mathbf{M}'_{ijc} + (\overline{\mathbf{M}'}_{\mathcal{C}})_{ij}$. We define μ-CAPE explanation as $\mathbf{E}_c^{\mu\text{-CAPE}} = \phi\big(\max(\mathbf{M}'_c + \overline{\mathbf{M}'}_{\mathcal{C}}, 0)\big)$. Note that μ-CAPE restores the class mutual regions but does not maintain the composition relationship to the model outcome.

3.2.4 Bootstrap Training

Finally, our loss function is defined in the form of knowledge distillation [9]:

$$\ell = \alpha \cdot \mathcal{H}(\hat{\mathbf{p}}, \mathbf{q}) + \beta \cdot \mathcal{D}_{\text{KL}}(\hat{\mathbf{p}}_{T'}, \mathbf{p}_T), \tag{10}$$

where $\mathcal{H}(.,.)$ denotes the Cross-Entropy function, $\mathbf{q}$ denotes the classification one-hot label vector, $\text{D}_{\text{KL}}(.,.)$ denotes the Kullback–Leibler divergence (KLD) function. T' denotes the addition of a learnable softmax temperature in Eq. (9) and T denotes the addition of a fixed temperature in Eq. (1), both temperature parameters are omitted in the respective equation for clarity. We propose this form of training using softened $\mathbf{p}_T$ as a mediator because the direct optimization using $\mathcal{H}(\hat{\mathbf{p}}, \mathbf{q})$ is difficult, see the classification results of 'Direct CE' entry in Table 2. Once trained, the vanilla classifier could be removed from the CAPE model to maintain nearly identical model parameters except for the learnable temperature parameter.

We further propose two ways of training the CAPE layer. 1) **training from scratch (TS)** by setting $\alpha = \beta = 1$, and during the training, the backbone model does not receive gradients from the CAPE layer but from the vanilla classification layer. 2) **post-fitting (PF)** CAPE layer to an already trained classifier model (*e.g.*, ImageNet pre-trained models) by setting $\alpha = 0$ and $\beta = 1$. This means that only the CAPE layer is trained, and we initialize the CAPE layer by the vanilla classifier's parameters. Besides, on the ImageNet dataset, we found the optimization is much more complex as KLD needs to match probability distributions in much higher dimensions. To alleviate this optimization difficulty, we propose a selective KLD variation that optimizes $\text{KLD}(\hat{\mathbf{p}}_{T'}, \mathbf{p}_T)$ only if the predicted classes of the CAPE layer and the vanilla classification layer are not the same. Let $\hat{c} = \text{argmax}_{c'}\hat{\mathbf{p}}_{c'}$ present CAPE predicted class and $c = \text{argmax}_{c'}\mathbf{p}_{c'}$ be the vanilla classifier predicted class, the selective KLD-enabled bootstrap loss is then:

$$\ell = \alpha \cdot \mathcal{H}(\mathbf{p}, \mathbf{q}) + \beta\mathbb{1}(\hat{c} \neq c) \cdot \mathcal{D}_{\text{KL}}(\hat{\mathbf{p}}_{T'}, \mathbf{p}_T). \tag{11}$$

The motivation of the design is from the intrinsic prediction discrepancy between CAPE and the vanilla classifier (illustrated in Table 2 in the supplementary material), so it may be unnecessary to match the exact distributions $\hat{\mathbf{p}}_{T'}$ and $\mathbf{p}_T$ for the training samples that $\hat{c} = c$. With the selective KLD loss, we only bootstrap the prediction distribution once $\hat{c} \neq c$, which significantly reduces the optimization difficulty.

4. Experiments

The proposed CAPE method can be viewed as a replacement for the softmax activation function in the classification module, therefore it is applicable to both DNNs using global average pooling, which can be found in both CNN and Transformer families. Therefore, we choose ResNet-50 [8] and Swin Transformer V2-B [15] as our test beds, please refer to the Section "Experiments on Swin Transformer model" in the supplementary material for the results of Swin Transformer model. All experiments were conducted on a single Nvidia RTX A6000 GPU (48G video memory) using PyTorch [19].

4.1. Datasets and Implementation Details

We benchmark on two public datasets: 1) CUB200-2011 [28]; 2) ImageNet ILSVRC2012 [5]. We also evaluate a cytology image dataset depicting a difficult Chronic myelomonocytic leukemia (CMML) diagnostic problem.

CUB comprises a total of 200 distinct bird species, accompanied by 5,995 training images and 5,794 test images. The input size is 448×448 and the produced CAM has 14×14 spatial size using both ResNet50 and Swin Transformer V2-B model.

ImageNet consists of a total of 1000 object categories with a collection of 1,281,167 images for training and 50,000 images for validation. We follow the convention in the literature [12, 14] by randomly selecting 2000 validation images for interpretability evaluation. The input size is 224×224 and the CAM size is 7×7.

CMML dataset contains 3,899 single-cell (Monocyte, a type of white blood cell) images from 171 individuals, who were annotated as 'Normal' or having 'CMML'. Each individual can have a different amount of monocyte images. We report the average results of 5-fold cross-validation on the CMML dataset. The input size is 352×352 and the derived CAM size is 11×11. Additional information on the CMML dataset and motivation for comparing CAMs on CMML can be found in the Section "CMML Dataset Details" in the supplementary material.

Training Settings. We train all CAPE configurations with SGD optimizer and use temperature $T = 2$. We set $1e{-}4$ as the initial learning rate for post-fitting (PF) CAPE models and trained them for 30 epochs, except for ImageNet, which is 5 epochs. For the training-from-scratch (TS) CAPE models, we uniformly set $1e{-}3$ as the initial learning rate for all three datasets. We employ step decay

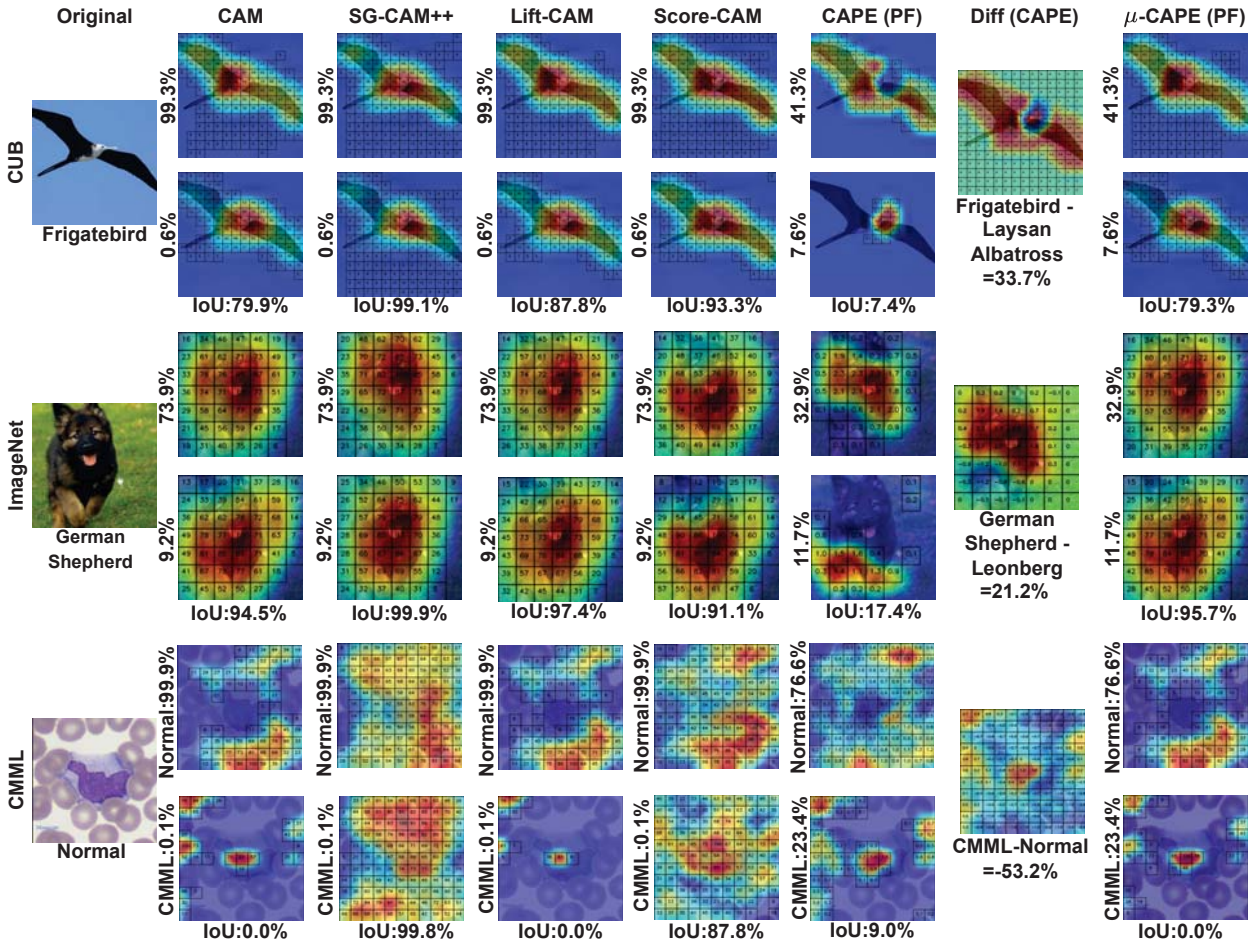

Figure 3. Qualitative visualisation using ResNet-50. Each dataset has two rows for the top-2 predicted classes' explanation maps. Class confidence scores are on the left side of each explanation map. We select CAM, Smooth Grad-CAM++, Lift-CAM, and Score-CAM to represent different visualization ways for the same vanilla classification layer. We show CAPE and μ-CAPE (PF) explanations for the proposed CAPE model, full comparisons are in Fig. 2 to 4 in the supplementary material. "SG-CAM++" denotes Smooth Grad-CAM++.

with a 0.1 decay rate per 30 epochs and a weight decay of $1e{-}4$ for CUB (200 total epochs) and ImageNet (90 total epochs). For the CMML dataset, we use a linear decay with a weight decay of $5e{-}4$ and 100 training epochs by which we reduce the learning rate to $1/100$ of the initial value. These hyperparameter settings of learning rate and number of epochs follow common settings in the literature. The temperature value was validated on a validation set reserved as a random proportion of the training set.

4.2. Qualitative Analysis

We compare to eight state-of-the-art DNN-based CAM interpretation methods, including activation-based CAM methods: CAM [31], Layer-CAM [10], Score-CAM [29], LIFT-CAM [12], FD-CAM [14]), and gradient-based CAM methods: Grad-CAM [24], Grad-CAM++ [3], Smooth Grad-CAM++ [17]).

The qualitative analysis is visualized in Fig. 3 for CUB, ImageNet, and CMML datasets using the ResNet50 model. We compare CAM, Smooth Grad-CAM++, LIFT-CAM, and Score-CAM with our proposed CAPE and μ-CAPE explanations (PF-trained models). For each compared method except CAPE, we plot explanation heatmaps for the top-2 predicted classes in the background and overlay their pre-upsampling attention values in the foreground. Some image region boxes are omitted from the drawing to avoid cluttering and it is based on whether the region's attention value exceeds the 5% threshold of maximum attention values. For the compared CAM methods this 5% threshold is not meaningful. However, for CAPE visualization, and using the ImageNet "German shepherd" dog example, the threshold translates to a minimum probability where below the probability is considered as noise, *i.e.*, $5\% \times 2.9\% \approx 0.145\%$ (2.9% is the largest attention probability), and the kept re-

Method	CUB						ImageNet						CMML					
	AD ↓	IC ↑	ADD ↑	ADCC ↑	mIoU ↓	BC ↑	AD ↓	IC ↑	ADD ↑	ADCC ↑	mIoU ↓	BC ↑	AD ↓	IC ↑	ADD ↑	ADCC ↑	mIoU ↓	BC ↑
CAM	21.2	27.9	67.4	78.8	75.9	0	**12.6**	41.9	49.2	**73.4**	84.4	2	17.4	36.0	**54.8**	**73.6**	**0.1**	**7**
Grad-CAM	21.6	27.5	66.8	77.3	100.0	0	12.7	41.4	48.7	72.9	100.0	0	18.2	35.3	**54.0**	70.6	100.0	1
Grad-CAM++	20.3	28.7	68.9	77.4	100.0	0	13.1	39.6	47.8	72.4	100.0	0	20.1	37.7	52.5	68.4	100.0	0
SG-CAM++	23.7	24.0	64.7	74.2	99.8	0	15.0	35.2	46.2	70.5	99.8	0	31.8	31.5	47.0	70.2	99.7	0
Layer-CAM	20.1	28.7	69.9	77.3	100.0	0	13.1	39.2	48.4	71.4	100.0	0	21.6	37.1	51.8	65.9	100.0	0
FD-CAM	20.5	27.9	**70.9**	78.1	96.7	1	15.8	38.3	49.5	72.5	100.0	0	17.8	38.9	**54.3**	71.9	99.7	2
LIFT-CAM	20.9	25.6	64.5	74.6	83.3	0	12.7	41.1	49.3	72.3	89.8	0	**16.4**	37.8	**54.0**	72.5	**0.1**	**6**
Score-CAM	**16.3**	**33.0**	**73.1**	**80.2**	81.9	**6**	**8.5**	**46.9**	52.6	72.9	80.2	**5**	17.0	**40.3**	48.3	67.7	77.0	1
CAPE (PF)	22.2	26.5	68.7	73.7	**13.4**	3	17.5	**45.2**	**59.7**	69.1	**11.0**	**7**	27.9	35.0	39.9	67.9	4.9	0
CAPE (TS)	27.1	**31.6**	59.1	77.5	**28.5**	3	34.7	34.2	41.3	69.9	**56.8**	2	29.9	27.4	36.3	72.0	**0.8**	1
μ-CAPE (PF)	**15.9**	30.9	69.6	**83.0**	**66.6**	**5**	12.7	43.9	**55.9**	**74.3**	**70.3**	**5**	**16.5**	**43.6**	45.5	**78.2**	**0.6**	**7**
μ-CAPE (TS)	**10.3**	**48.5**	**74.2**	**84.4**	80.9	**12**	**10.7**	**58.3**	**58.7**	**73.5**	89.0	**9**	**14.1**	**48.0**	50.1	**78.4**	5.3	**9**

Table 1. Comparison of CAM interpretation methods using ResNet-50 backbone model. ↓ and ↑ indicate lower or higher is better. "SG-CAM++" denotes Smooth Grad-CAM++. The top-3 scores are marked from darker to lighter green colors.

gions constitute 32.8% of the class prediction of 32.9%. We can analytically say the regions cropped above the threshold maintain 99.7% of the original class confidence.

With CAPE's probabilistic ensemble formulation, we can directly compare the two class maps shown in the Diff(CAPE) column. Furthermore, for the CMML task, predicting CMML from monocyte images is an exploratory and open-ended research question, hence we are particularly interested in understanding where the classifier looks at when the decision is made (*e.g.*, nucleus, cytoplasm, cell exterior region, or their touching boundaries). Each image's attention placement can be significantly different and hard to manually review beyond a few. Using CAPE with the additional help of image segmentation, we can easily compute an empirical summary of the attention placement over all test images such as shown in Table 6 of the supplementary material.

CAPE and μ-CAPE are for different purposes. CAPE is analytical and can explain class discriminative regions which generally show less overlap between the top-2 class explanation maps. In addition, CAPE achieves lower intersection over union (IoU, defined in Sec. 4.3). These characteristics make CAPE useful for understanding the subtle differences between visually similar concepts. μ-CAPE shows significant overlap between the top-2 classes and is more useful when the full class object is needed. An exception is in the CMML example where CAM, Lift-CAM, and μ-CAPE already show the class discriminative characteristic and have 0.0% IoU, meaning that the respective explanation maps do not overlap. This is likely because of the two classes in the CMML problem because the small number of classes trained classifiers are more likely to discard non-discriminative regions [4].

4.3. Quantitative Analysis

We use four common CAM interpretability evaluation metrics with an additional metric in our quantitative analysis. Let $\mathbf{E}_c = \Phi(\mathbf{x}, c)$ denote the overall process that generates an explanation map $\mathbf{E}_c$ from an image given class c, and $\mathbf{p}_c = \Psi(\mathbf{x}, c)$ denotes the model prediction generation process. The measurements are defined below.

Average Drop in Confidence (AD) [3]. For a single image with target class c, $\text{AD}(\mathbf{x}) = \frac{\max(y_c - o_c, 0)}{y_c}$; where $y_c = \Psi(\mathbf{x}, c)$ and $o_c = \Psi(\mathbf{E}_c \odot \mathbf{x}, c)$, $\odot$ defines the element-wise production, and $c = \text{argmax}_{c' \in |\mathcal{C}|}(\mathbf{p}_{c'})$.

Average Increase in Confidence (IC) [3] measures the confidence gain when the explanation map is applied: $\text{IC}(\mathbf{x}) = \mathbb{1}(y_c < o_c)$, where $\mathbb{1}$ is an indicator function.

AD in Deletion (ADD) [12] overcomes the drawbacks that IC and AD give good scores when an interpretation method always gives an over-confident explanation. $\text{ADD}(\mathbf{x}) = \frac{\max(y_c - d_c, 0)}{y_c}$, where $d_c = \Psi\big((1 - \mathbf{E}_c) \odot \mathbf{x}, c\big)$.

AD, Coherency, and Complexity (ADCC) [20] was introduced as a robust measurement in comparison to AD and IC. ADCC represents the harmonic mean of different metrics. $\text{ADCC}(\mathbf{x}) = \frac{3}{\text{coh}(\mathbf{E}_c, \mathbf{E}'_c)^{-1} + (1 - \text{com}(\mathbf{E}_c))^{-1} + (1 - \text{AD}(\mathbf{x}))^{-1}}$. $\text{coh}(\mathbf{E}_c, \mathbf{E}'_c) = 2 \cdot \text{corr}(\mathbf{E}_c, \mathbf{E}'_c) + 1$ measures the min-max normalized Pearson Correlation Coefficient (corr) between $\mathbf{E}_c$ and $\mathbf{E}'_c = \Phi(E_c \odot \mathbf{x}, \mathbf{c})$. $\text{com}(\mathbf{E}_c) = |\mathbf{E}_c|$ measures the complexity of an explanation map by its $\mathcal{L}_1$-norm.

Intersection over Union (IoU) measures the overlap between the explanation maps of top-2 predicted classes. We first create a mask $\mathbf{S}_c = \mathbf{E}_c > 0.2 \cdot \max(\mathbf{E}_c)$, then $\text{IoU}(\mathbf{x}) = \frac{|\mathbf{S}_{c_1} \cap \mathbf{S}_{c_2}|}{|\mathbf{S}_{c_1} \cup \mathbf{S}_{c_2}|}$, for the top-2 classes c_1 and c_2. We report the mean IoU (mIoU) in Table 1.

Borda Count (BC) is a voting method to give a score based on multiple rankings. We assign a 1st ranking a score of 3, a 2nd ranking a score of 2, and a 3rd ranking a score of 1. The rest ranks are scored 0. Our BC ranking sums over the scores from the above measurements.

The quantitative analysis is shown in Table 1. We show the following observations.

1. μ-CAPE explanations hold the top BC rankings across all datasets because of their AD, IC, ADD, and ADCC scores. This illustrates the advantage of the μ-CAPE explanation in terms of the capability to include both class discriminative and class mutual regions. In contrast, the CAPE explanation highlights class discrimina-

tive regions hence leading the mIoU measurement.

2. All μ-CAPE (TS) measurements are generally better than the (PF) model measurements but the PF models are much cheaper to run, especially on large datasets. Comparing CAPE (TS) and (PF), the (PF) version leads to better mIoU on the CUB and ImageNet datasets, but the opposite is observed on the CMML dataset.
3. Score-CAM has a good BC ranking based on high AD, IC, ADD, and ADCC rankings on CUB and ImageNet. Notably, it has a significantly lower AD score on ImageNet. Score-CAM explanation map for an image and a target class pair requires the computation of explanation maps for all classes, which is computationally intensive. In contrast, μ-CAPE and CAPE only need a simple feed-forward inference that incurs trivial computation overhead compared to the original CAM. For instance, CAPE and CAM take around 150 milliseconds to compute for one CUB image on our hardware, and Score-CAM takes 15 seconds.

4.4. Ablation Study on Classification Performance

In Table 2, we show the classification performance using the vanilla classification layer and the CAPE classification layer on the same ResNet-50 model with different settings. The Naive AVG and Off-the-shelf CAPEs reuse the vanilla classification layer's parameters where their difference is that Naive AVG CAPE aggregates all pixel probability distributions by averaging (Eq. (4)) while Off-the-shelf CAPE employs the image region importance (Eq. (6)). Both models can be used as post hoc *visual* interpretation methods like CAMs, but they have classification performance gaps toward the vanilla classifier. This leads to our proposal of training the CAPE model to mitigate the gap. The Direct CE CAPE (TS) employs full course training using the cross-entropy loss $\mathcal{H}(\hat{\mathbf{p}}, \mathbf{q})$ but does not show a significant improvement from Off-the-shelf CAPE. Both Bootstrap-trained (TS and PF) models get closer performance to the vanilla classification model but arguably there is a marginal performance gap. Finally, we stress that our μ-CAPE and CAPE explanations share the same model and only differ in their explanation map formation (*i.e.*, $\mathbf{E}_c^{\mu\text{-CAPE}}$ *vs.* $\mathbf{E}_c^{\text{CAPE}}$).

Model		CMML	CUB	ImageNet
# Classes ($\lvert\mathcal{C}\rvert$)		2	200	1,000
$H \times W$		11×11	14×14	7×7
$H \cdot W \cdot \lvert\mathcal{C}\rvert$		242	39,200	49,000
CAPE	Naive AVG	89.5	79.01	74.01
	Off-the-shelf	87.4	80.62	74.01
	Direct CE (TS)	88.8	80.51	72.95
	Bootstrap (PF)	90.3	82.12	74.42
	Bootstrap (TS)	89.8	82.19	74.64
Vanilla classification		90.5	83.34	76.13

Table 2. Classification accuracy evaluated on ResNet50 model for different CAPE configurations and vanilla classification layer.

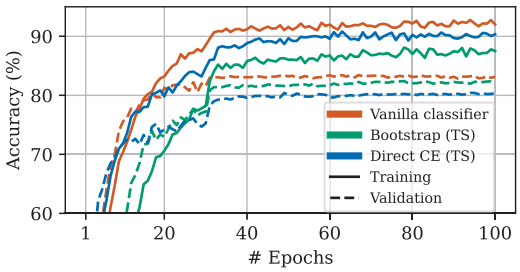

Figure 4. The ResNet-50 training and validation classification accuracy recorded during the training course for the CUB dataset.

5. Discussion and Conclusion

We proposed CAPE, a novel DNN interpretation method that is powerful in visualizing and analyzing DNN model attention. It enables us to probabilistically understand how the model predicts, and provides novel insights into meaningful and analytical interpretations. CAPE is a simple reformulation of the softmax classification layer that adds a trivial cost to classification inference and visual explanation compared to the vanilla classifier and CAM explanation. We conclude with CAPE's characteristics and limitations to motivate future work.

Training convergence and soft prediction confidence. Fig. 4 illustrates that the training convergence issue affects the CAPE model's accuracy. We believe the convergence issue is caused by the soft prediction confidence characteristic of CAPE (see Table 1 of the supplementary material). We suspect that the softened predictions in the CAPE formulation are a result of the large number ($H \times W \times |\mathcal{C}|$) of voxels accumulated in the denominator of the softmax function (see Eq. (9)). It is commonly known that interpretable models often have to trade accuracy for improved explainability [18, 30]. We believe for CAPE, the trade-off is between the probability computation capacity (leading to improved explainability and analytical ability) and the soft prediction confidence (causing training convergence issues), both resulting from the usage of softmax normalization. Bootstrap training was introduced to soften the classification confidence scores of the vanilla classifier and therefore mitigate the optimization difficulty of CAPE training.

CAPE explains itself. Even though the CAPE module's training was bootstrapped from the vanilla classifier and the CAPE models' classification performance approaches to the vanilla classifier's performance, Table 2 in the supplementary material shows an in-negligible prediction disagreement between the two classification layers. Hence, CAPE's probabilistic explanation should not be used to explain the decision process of the vanilla classification classifier.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014. 3

[2] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1721–1730, 2015. 1

[3] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 839–847. IEEE, 2018. 2, 6, 7

[4] Zhaozheng Chen and Qianru Sun. Extracting class activation maps from non-discriminative features as well. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3135–3144, 2023. 7

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[6] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, pages 3429–3437, 2017. 1, 2

[7] Ziteng Gao, Limin Wang, and Gangshan Wu. Lip: Local importance-based pooling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3355–3364, 2019. 3

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[9] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *NIPS 2014 Deep Learning Workshop*, 2015. 5

[10] Peng-Tao Jiang, Chang-Bin Zhang, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. Layercam: Exploring hierarchical class activation maps for localization. *IEEE Transactions on Image Processing*, 30:5875–5888, 2021. 2, 6

[11] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *IEEE Intl. Conf. on Computer Vision*, pages 2106–2113. IEEE, 2009. 1, 2

[12] Hyungsik Jung and Youngrock Oh. Towards better explanations of class activation mapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1336–1344, 2021. 3, 5, 6, 7

[13] Alexander Kolesnikov and Christoph H Lampert. Seed, expand and constrain: Three principles for weakly-supervised image segmentation. In *European Conf. on Computer Vision*, pages 695–711. Springer, 2016. 1

[14] Hui Li, Zihao Li, Rui Ma, and Tieru Wu. Fd-cam: Improving faithfulness and discriminability of visual explanation for cnns. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 1300–1306. IEEE, 2022. 2, 5, 6

[15] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022. 5

[16] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017. 3

[17] Daniel Omeiza, Skyler Speakman, Celia Cintas, and Komminist Weldermariam. Smooth grad-cam++: An enhanced inference level visualization technique for deep convolutional neural network models. In *Intelligent Systems Conference*, 2019. 2, 6

[18] Konstantinos Panagiotis Panousis, Dino Ienco, and Diego Marcos. Sparse linear concept discovery models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2767–2771, 2023. 8

[19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 5

[20] Samuele Poppi, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Revisiting the evaluation of class activation mapping for explainability: A novel metric and experimental analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2299–2304, 2021. 7

[21] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016. 1, 2, 3

[22] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules. *Advances in neural information processing systems*, 30, 2017. 3

[23] Neil Savage. Breaking into the black box of artificial intelligence. *Nature*, 2022. 1

[24] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *IEEE Intl. Conf. on Computer Vision*, pages 618–626, 2017. 1, 2, 6

[25] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Intl. Conf. on Learning Representations Workshop*, 2014. 1, 2

[26] Bas HM Van der Velden, Hugo J Kuijf, Kenneth GA Gilhuijs, and Max A Viergever. Explainable artificial intelligence (xai) in deep learning-based medical image analysis. *Medical Image Analysis*, page 102470, 2022. 1

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia

Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. 3

[28] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. *Technical Report CNS-TR-2011-001*, 2011. 5

[29] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-cam: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 24–25, 2020. 2, 6

[30] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023. 8

[31] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2921–2929, 2016. 6