

# DECIDER: Leveraging Foundation Model Priors for Improved Model Failure Detection and Explanation

Rakshith Subramanyam<sup>\*1</sup>, Kowshik Thopalli<sup>\*2</sup>, Vivek Narayanaswamy<sup>\*2</sup>, and  
Jayaraman J.Thiagarajan<sup>2</sup>

<sup>1</sup> Axio.ai, Tempe AZ 85281, USA

<sup>2</sup> Lawrence Livermore National Laboratory, Livermore CA 94550, USA  
rakshith.subramanyam@axio.ai  
{kowshik\_thopalli, narayanaswam1}@llnl.gov  
jjthiagarajan@gmail.com

**Abstract.** Reliably detecting when a deployed machine learning model is likely to fail on a given input is crucial for ensuring safe operation. In this work, we propose DECIDER (Debiasing Classifiers to Identify Errors Reliably), a novel approach that leverages priors from large language models (LLMs) and vision-language models (VLMs) to detect failures in image classification models. DECIDER utilizes LLMs to specify task-relevant core attributes and constructs a “debiased” version of the classifier by aligning its visual features to these core attributes using a VLM, and detects potential failure by measuring disagreement between the original and debiased models. In addition to proactively identifying samples on which the model would fail, DECIDER also provides human-interpretable explanations for failure through a novel attribute-ablation strategy. Through extensive experiments across diverse benchmarks spanning subpopulation shifts (spurious correlations, class imbalance) and covariate shifts (synthetic corruptions, domain shifts), DECIDER consistently achieves state-of-the-art failure detection performance, significantly outperforming baselines in terms of the overall Matthews correlation coefficient as well as failure and success recall. Our codes can be accessed at <https://github.com/kowshikthopalli/DECIDER/>

**Keywords:** Failure Detection · Vision-Language Models · Large-language Models

## 1 Introduction

A crucial step in ensuring the safety of deployed models is to proactively identify if a model is likely to fail for a given test input. This enables the implementation

---

\* equal contribution

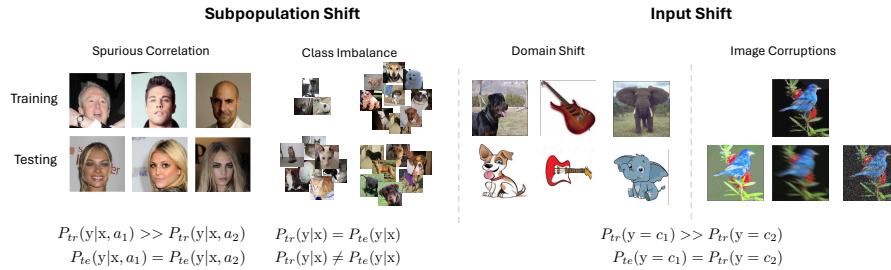
of appropriate correction mechanisms without impacting the model’s operation, or even deferring to human expertise for decision-making. While failures in vision models can be attributed to a variety of factors, the most significant cause is the violation of data distribution assumptions made during training [19], which is the focus of this work. In general, data comprises both task-relevant *core attributes* and irrelevant *nuisance attributes*, and they are never explicitly annotated. Consequently, models can fail to generalize if (i) the training data contains spurious correlations (to nuisance attributes) that do not appear at test time, (ii) class-conditional distribution of nuisance attributes can arbitrarily change between train and test data (e.g., patient race imbalance in clinical datasets), or (iii) novel attributes emerge only at test time (e.g., style changes). Note that, when the class-conditional distributions of core attributes themselves change between train and test data, it leads to the more challenging scenario of *concept shifts*, and is not considered in this work. Nevertheless, detecting failures across all these scenarios is known to be challenging [8, 20, 51], and hence there has been a surge in research interest [6, 10, 15, 17, 21].

We begin by acknowledging that it is not only difficult, but also inefficient, to describe such nuisance attribute discrepancies solely using visual features. In this regard, we explore the utility of large language models (LLMs) and vision-language models (VLMs) in characterizing data attributes through a combination of visual and natural language descriptors. Subsequently, one can leverage these descriptors to design powerful failure detectors that systematically discern gaps in model generalization. Based on this idea, we develop DECIDER (Debiasing Classifiers to Identify Errors Reliably), a new approach for failure detection in vision models. At its core, DECIDER (i) utilizes LLMs (e.g., GPT-3 [1]) to specify task-relevant core attributes, (ii) uses a VLM (e.g., CLIP [37]) to construct a “debiased” version of the task model by aligning its visual features to the core attributes, and (iii) detects failure by measuring disagreement between the original and debiased models for any given test input.

Additionally, DECIDER can be used to provide explanations for failure cases. This is done by employing an attribute-ablation strategy that adjusts the relative importance of core attributes such that the prediction probabilities of the debiased matches the original model. Our extensive empirical evaluation shows that our method achieves state-of-the-art performance in detecting failures across various datasets and test scenarios. In summary, our work provides early evidence for the utility of large-scale foundation models as priors for designing novel safety mechanisms.

## 2 Related Work

**Failure Detection.** Failure detection in classification involves identifying incorrect predictions made by the model [15, 36, 54]. This problem ultimately boils down to identifying an appropriate metric or a *scoring function* that can delineate failed samples from successful ones. Early work involves using simple scores directly derived from the predictions of the model such as Maximum Softmax



**Fig. 1:** A visual illustration of the different failure scenarios we consider. These include scenarios when the model relies on spurious correlations present in the data i.e., when an attribute is spuriously correlated with the label (e.g., color of hair and gender). Another cause of failure is when the training data has class imbalance, leading to poorer generalization on images from the under-sampled class. Lastly, another important cause of failures are when the distribution of the test data is different from the training data. This can range from natural image corruptions to covariate shifts.

Probability (MSP) [15], predictive entropy [21] and energy [25] to identify failed samples. More recent work focuses on scores that quantify failure by evaluating the local manifold smoothness [33] around a given sample and those that are based on agreement of a sample between different components of an ensemble [18, 45]. However, such metrics can become unreliable to characterize failure as the model used to derive them can be potentially mis-calibrated and unreliable [11, 29]. Failure detection has also been studied under the lens of generalization gap estimation [10, 32] where the goal is to predict the accuracy of the model on an unlabeled target distribution using distributional metrics derived from a number of calibration datasets.

**Failure Detection with Vision Language Foundation Models.** Visual-Language Models (VLMs) [24, 38] are pre-trained on a large-corpora of image-text captions using a self-supervised objective. VLMs facilitate flexible adaptation to downstream tasks through zero-shot transfer or fine-tuning, demonstrating enhanced performance in zero-shot classification and OOD detection [5, 9, 28, 30, 47, 49, 50]. Recently, VLMs have been used as a lens to understand the failure modes and weaknesses of any pre-trained model. For instance, the authors of [17] fit a post-hoc failure detector on the latent spaces of the VLM to estimate whether a sample has been correctly identified or not by the pre-trained classifier. The detector is then used to identify the directions of classifier failure modes. However, this approach requires a carefully tailored calibration set to fit the detector which is often unavailable in practice. On the other hand, the authors of [4] demonstrate that the latent space agreement between the pre-trained model and the VLM is a potential indicator for failure. In contrast, our paper aims to perform failure detection by first designing an improved classifier leveraging the VLM latent space and assessing the agreement between the classifier and its enhanced version while providing explanations for failure.

### 3 Background

**Preliminaries.** Let  $\mathcal{F}$  denote a multi-class classifier with parameters  $\theta$ , trained on a dataset  $\mathcal{D} = (x_i, y_i)_{i=1}^M$  comprising  $M$  samples. Here,  $x_i \in \mathcal{X}$ , is a 3 channel, input RGB image, and  $y_i \in \mathcal{Y}$  is the corresponding label, where  $\mathcal{Y}$  is the set of class labels i.e.,  $\mathcal{Y} = \{1, 2, \dots, C\}$ . Here,  $C$  denotes the total number of distinct classes. The classifier  $\mathcal{F}$  operates on the input to produce the logits  $\mathcal{F}(x)$  corresponding to every class which is followed by a `softmax` operation to estimate output probabilities  $p(y = c|x)$  where  $c$  corresponds to the class index.

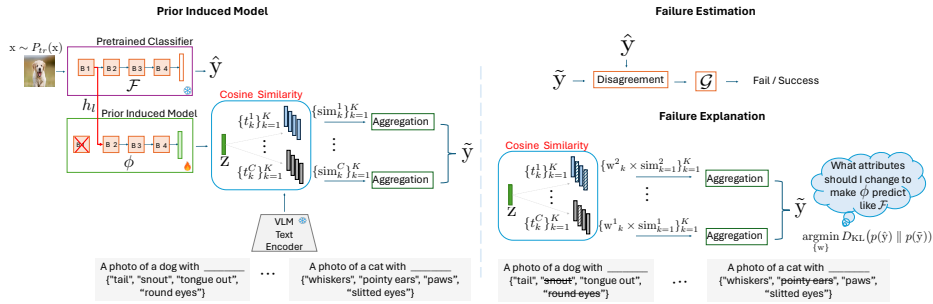
In this paper, we consider the problem of failure detection in classification models, where the source of failure arises due to the following scenarios (Fig. 1) - (i) Input level shifts where the training and test images share identical conditional output distributions i.e.,  $P_{tr}(y|x) = P_{te}(y|x)$  but different input marginals  $P_{tr}(x) \neq P_{te}(x)$ . Here, the test data can correspond to domain variations or image corruptions. (ii) Sub-population shifts (a) Spurious correlation where the labels are non-causally associated [51] with certain input characteristics or attributes in the training data over others leading to learning non-generalizable decision rules. For instance, let  $a_1$  and  $a_2$  correspond to two attributes of an image  $x$  and the training distribution is such that  $P_{tr}(y|x, a_1) \gg P_{tr}(y|x, a_2)$ . This model is susceptible to spurious correlations between the inputs and the targets and can fail during test time when  $P_{te}(y|x, a_1) = P_{te}(y|x, a_2)$ , (b) Class imbalance where the number of examples in a given class can be significantly greater than those present in another i.e.,  $P_{tr}(y = c_1) \gg P_{tr}(y = c_2)$ . This does not allow the classifier to optimally capture the image statistics and semantics of class  $c_2$  leading to sub-optimal generalization performance.

**Failure Detector Design.** Failure detection is a binary classification problem of identifying whether an input sample has been correctly predicted or not by the model. We define our failure detector  $\mathcal{G}$  as follows,

$$\mathcal{G}(x; \theta, \tau) = \begin{cases} \text{failure,} & \text{if } s(x; \theta) < \tau, \\ \text{success,} & \text{if } s(x; \theta) \geq \tau. \end{cases} \quad (1)$$

Here,  $s(\cdot)$  is a scoring function derived from the classifier  $\mathcal{F}$  that assigns higher values for correctly identified samples and vice-versa and  $\tau$  is the user-controlled threshold for detection. Following standard practice from the generalization gap literature [7, 45], we identify  $\tau$  such that  $\sum_i \mathbb{I}(s(x_i; \theta) \geq \tau)$  approximates the true accuracy of the held-out validation dataset.

**Contrastive Language-Image Pre-training (CLIP).** CLIP [37] is a vision-language model trained on large corpus of image-text pairs with self-supervised learning. It aligns images with natural language descriptions in a shared embedding space, enabling zero-shot learning and fine-tuning for downstream tasks such as image captioning [43] and visual question answering [12, 41, 42, 52]. CLIP employs image ( $I(\cdot)$ ) and text ( $T(\cdot)$ ) encoders to generate embeddings ( $z_I$  and  $z_T$ ). For zero-shot inference, it computes the cosine similarity (`cos sim`) between image and text embeddings. This similarity yields class-specific logit scores for



**Fig. 2: DECIDER for failure detection.** (Left) DECIDER trains a Prior Induced Model (PIM)  $\phi$ , identical to the architecture of the pre-trained classifier  $\mathcal{F}$ , utilizing priors from a VLM model. (Top Right) The disagreement between the predictions of  $\phi$  and  $\mathcal{F}$  serves as an indicator for failure detection. (Bottom Right) By adjusting attribute level weights, DECIDER offers explanatory insights into failures.

zero-shot classification, where the prediction probability  $p(y|x)$  is calculated using softmax.

## 4 Proposed Approach

### 4.1 Motivation

Typically, a classifier  $\mathcal{F}$  is trained on a dataset  $\mathcal{D}$  to learn the mapping between inputs and target labels. The datasets contain both task-relevant *core attributes* and irrelevant *nuisance attributes*, which are not explicitly annotated. Consequently, the decision rules of the classifier could rely on nuisance attributes leading to poor generalization. For e.g., the model can fail to generalize if the training data contains spurious correlations with nuisance attributes that do not appear during testing. We underscore that this problem of reliance on nuisance attributes arises due to the difficulty in describing them solely using visual features.

To address this, we go beyond using only visual features and propose to leverage a combination of vision and language descriptors through the use of LLMs and VLMs and design failure detectors that discern the gap in model generalization. In this section we describe our novel strategy for failure detection which involves training a classifier referred to as the Prior Induced Model (PIM)  $\phi$  with the aid of LLMs and VLMs. We believe that the prior knowledge induced by VLMs will help PIM associate task-relevant core attributes. We first describe our paradigm that incorporates foundation models in classifier training. We then develop a prediction disagreement based strategy between PIM and the original classifier to conduct failure detection. Finally, we elucidate the capability of our approach in extracting failure explanations in order to support interpretability.

## 4.2 Incorporating Foundation Model Priors

A key challenge in traditional classification models is the direct mapping of images to coarse labels which encapsulate several attributes. For instance, in distinguishing between a dog and a cat, the label “dog” encompasses attributes like “wagging tail” and “snout”, while “cat” includes “whiskers” and “pointy ears”. Without explicit access to such detailed attribute information and due to potential biases in the training data, models are susceptible to rely on overly simplistic decision rules. In contrast, VLMs such as CLIP offer capabilities to encode both image and textual attribute descriptions into a unified latent space that is enriched to support meaningful image-text attribute associations.

To improve the effectiveness of classification model training, we hypothesize that aligning the model’s visual features with the textual descriptions of core attributes related to the class of interest in the VLM latent space can enhance training. This alignment is expected to equip the classifier with the ability to develop decision-making rules that are both more reliable and generalizable, while also reducing the influence of existing biases.

To achieve this, we introduce the PIM model  $\phi$ , which is guided by the LLM and VLM based priors (see Fig. 2 left). The architecture of PIM closely resembles that of its counterpart  $\mathcal{F}$ , with the notable distinction being that its final layer projects onto the VLM latent space. This projection supports the alignment with the textual descriptions of class-level attributes, thereby harnessing the linguistic capabilities of foundational models. PIM is specifically engineered to accept early-stage features from  $\mathcal{F}$ , denoted as  $h_l$ , which are then processed through PIM’s analogous layers to produce the image encoding  $z$  within the VLM latent space. For instance, when both  $\mathcal{F}$  and  $\phi$  are based on the ResNet architecture [13], the output from block 1 of  $\mathcal{F}$  serves as the input for block 2 in  $\phi$ .

It must be noted that the success of our approach relies upon the quality of the fine-grained text attributes extracted for every class. While there exists strategies [27] that are capable of extracting image-level textual descriptions, they usually involve the text decoders in the loop which can be computationally expensive. Therefore, we resort to using Large Language Models (LLMs) to compute task-specific attribute descriptions offline.

## 4.3 Generating Task-specific Core-attribute Descriptions

LLMs [1, 44] have demonstrated their utility across a range of language tasks [31, 35, 39, 48] and are particularly adept at contextual understanding, and generating coherent text even with descriptive prompting. To extract the class-specific attribute descriptions, we query GPT-3 [1] with the prompts “List visually descriptive attributes of <CLASS>.” This allows us to gather a set of  $K$  attributes  $\mathcal{A}^c = \{a_k^c\}_{k=1}^K$  for every class  $c$ .

#### 4.4 Training PIM

**(i) Computing Cosine Similarities.** We first compute the cosine similarity scores between the image embedding  $z$  produced by PIM for a given image and the text embeddings associated with attribute  $k$  from each class  $c$ . It is given by,

$$\Omega_{\mathcal{A}^c} = \{\omega_k^c\}_{k=1}^K \text{ where } \omega_k^c = \cos \text{sim}(z, e_k^c) \quad (2)$$

Here, the text embeddings  $E_{\mathcal{A}^c} = \{e_k^c\}_{k=1}^K$  for each attribute of every class are obtained using the CLIP text encoder.

**(ii) Attribute Similarity Aggregation.** Subsequently, we aggregate these attribute similarity scores,  $\Omega_{\mathcal{A}^c}$ , for each class  $c$  to obtain coarse prediction logits corresponding to the class label  $y \in \mathcal{Y}$ . We investigate two aggregation strategies namely - (i) Class-level mean and (ii) Class-level max to consolidate these scores into final class predictions which are eventually normalized using `softmax`. These strategies enable a more refined and attribute-aware determination of classification outcomes.

**(iii) Optimization Objective.** The optimization is primarily guided by the cross-entropy loss which evaluates the discrepancy between the predicted probabilities from PIM and the ground truth label. In addition, we include consistency driven augmentations namely CutMix [53] and AugMix [16] to improve its robustness. Additionally, we upweight the losses corresponding to the instances where (i) the biased classifier  $\mathcal{F}$  predicts accurately, but  $\phi$  does not and (ii) the biased classifier  $\mathcal{F}$  does not predict accurately, as well as  $\phi$  does not, within a training batch.

#### 4.5 DECIDER: Failure Estimation Using PIM

To assess the failure of the biased classifier  $\mathcal{F}$ , we compute the disagreement between PIM and  $\mathcal{F}$  based on the discrepancy between their predictions. This disagreement score is calculated as the cross-entropy between the sample-level probability distributions between the two models with PIM being the reference distribution given by  $s(x) = -\sum_{c=1}^C p(y=c|x) \cdot \log(q(y=c|x))$  where  $p(\cdot)$  and  $q(\cdot)$  represent the predicted probabilities of  $\mathcal{F}$  and PIM, respectively.

#### 4.6 Extracting Explanations for Failure

Our failure explanation protocol is designed to elucidate the underlying reasons behind the discrepancies between predictions of  $\mathcal{F}$  and  $\phi$ . The primary objective is to identify the optimal subset of attributes necessary for aligning the PIM’s prediction probabilities with those of the task model. To achieve this, we implement an attribute ablation strategy where we iteratively adjust a group of weights corresponding to each attribute across all classes. Our iterative process begins by initially assigning uniform weights to every attribute for each class within a batch. These weights are then optimized by minimizing the Kullback-Leibler (KL) divergence between the probability distributions predicted by  $\mathcal{F}$

and those adjusted by PIM, accounting for the influence of the weighted attributes. As the algorithm converges, the weights will highlight those attributes that have significant impact on the predictions of  $\mathcal{F}$ , providing insights into the features considered by  $\mathcal{F}$  when making decisions. Fig. 2 right illustrates our failure explanation mechanism.

## 5 Empirical Evaluation

We conduct comprehensive evaluations of DECIDER using various classification benchmarks and assess performance under various failure scenarios with different architectures. We employ OpenAI’s CLIP ViT-B-32 model in all experiments [38].

### 5.1 Experimental Setup

**Datasets.** Our experiments are centered around datasets reflecting four common sources of model failure:

- **Input-Level Shifts:** CIFAR100-C [14], comprising 19 types of corruptions at five severity levels over the CIFAR100 test images across 100 categories.
- **Spurious Correlations:** (1) Waterbirds [51] involves classifying images as ‘water bird’ or ‘land bird’. The training data offers biases tied to the background (water/land). (2) CelebA [26, 51] involves classifying if individuals have blond hair or not, with labels spuriously correlated with gender.
- **Class Imbalance:** We modify the Kaggle Cats vs Dogs dataset [3], adjusting the distribution to create a training imbalance with 5,989 cat and 19,966 dog images for training, while maintaining balanced test data.
- **Distribution Shifts:** (1) PACS [23] includes images from four domains (Photo, Art-painting, Cartoon, Sketch), to be classified into seven categories. As two large-scale benchmarks, we consider (2) DomainNet [34] which contains images from 345 categories from 6 domains (Real, Painting, Infograph, Quickdraw, Cartoon and Sketch) and (3) ImageNet-Sketch [46] benchmark which contains sketch images from 1000 ImageNet [40] classes.

**Model Architectures.** We consider the ResNet-50 architecture for CelebA dataset and for all other datasets, we employ ResNet-18 trained on their respective datasets as the original classifier  $\mathcal{F}$ . In the supplementary, we study the performance of DECIDER on more architectures and we provide additional training details.

### 5.2 Baselines

We consider different baselines that use sample-level scores  $s$  for failure estimation:-

- (i) Maximum Softmax Probability (MSP) [15] which is given by  $s(x) = \max_j p(y = j|x)$ , (ii) Predictive Entropy (Ent) is essentially the entropy among the predictions of a sample and is given by  $s(x) = -\sum_{j=1}^K p(y = j|x) \cdot \log(p(y = j|x))$ ,



Dataset	Method	FR	SR	MCC	
CIFAR100	MSP	0.6835	0.809	0.4943	
	Energy	0.6776	0.7965	0.4747	
	Ent	0.6894	0.8105	0.514	
	DECIDER				
	+ mean	0.7949	0.7436	0.5267	
	+ max	0.7933	0.7474	<b>0.5292</b>	
	CIFAR100-C	MSP	0.7448	0.6345	0.3593
		Energy	0.8145	0.5442	0.3577
		Ent	0.7761	0.616	0.3766
		DECIDER			
+ mean		0.8507	0.5393	0.4007	
+ max		0.8448	0.5506	<b>0.4015</b>	
Waterbirds		MSP	0.3166	0.8891	0.2419
		Energy	0.4803	0.8047	0.2814
		Ent	0.4878	0.8022	0.2827
		DECIDER			
	+ mean	0.5303	0.8310	0.3580	
	+ max	0.6063	0.8580	<b>0.4598</b>	
	CelebA	MSP	0.4058	0.9653	0.3634
		Energy	0.4292	0.9616	0.3677
		Ent	0.4214	0.9631	0.3675
		DECIDER			
+ mean		0.5443	0.9701	<b>0.4928</b>	
+ max		0.4390	0.9621	0.3738	
Cats and Dogs		MSP	0.4076	0.9235	0.3316
		Energy	0.4303	0.9196	0.3428
		Ent	0.4233	0.9212	0.3402
		DECIDER			
	+ mean	0.5993	0.9468	0.544	
	+ max	0.5783	0.9554	<b>0.5532</b>	

(a)

(b)

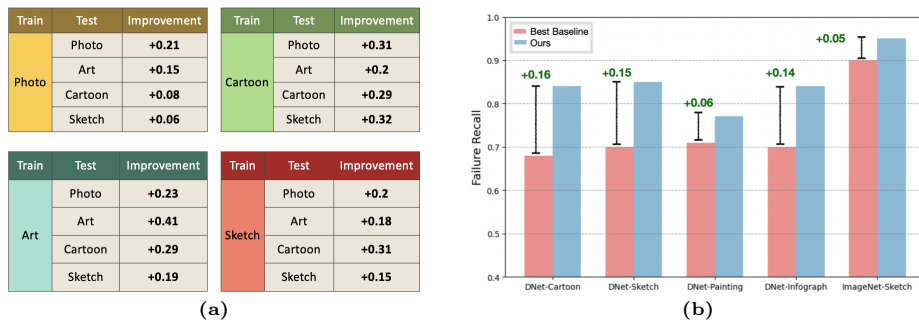
**Fig. 3:** Results on failure detection across different benchmarks - (a) CIFAR100, and image corruptions on CIFAR-100-C, and (b) subpopulation shifts from spurious correlations on Waterbirds, CelebA datasets, and class imbalance on Cats vs Dogs. DECIDER consistently outperforms baselines in terms of the overall Matthew’s Correlation Coefficient (MCC) as well as achieving higher failure and success recalls.

(iii) **Energy** [25] score is defined by  $s(x) = -T \cdot \log \sum_{j=1}^K \exp^{\mathcal{F}_\theta(x_j)}$ . Following standard practice, we consider  $T = 1$  in all our experiments. (iv) Generalized Model Disagreement (GDE) [2, 18] - Let  $\mathcal{F}_{\theta_1}, \mathcal{F}_{\theta_2} \dots \mathcal{F}_{\theta_r}$  denote  $r$  models trained with different random seeds. Let  $\mathcal{F}_{\theta_1}$  denote the base classifier. Then the score is computed as  $s(x) = \frac{1}{r} \sum_{i=1}^r \frac{1}{r-1} \sum_{j \neq i}^r \mathbb{I}(\mathcal{F}_{\theta_i} \neq \mathcal{F}_{\theta_j})$ . We set  $r$  to 5.

It must be noted that we utilize negative versions of entropy and energy to reflect the fact the samples that are correctly predicted are associated with higher scores.

### 5.3 Metrics

We consider the following metrics to evaluate failure detection performance: (i) **Failure Recall (FR)** which corresponds to the fraction of samples that have been correctly identified as failure, (ii) **Success Recall (SR)** corresponds to the fraction of samples that have been correctly predicted as successful. The trade-off between the two metrics is indicative of how aggressive or conservative the failure detector is. (iii) **Matthew’s Correlation Coefficient (MCC)** holistically assesses the quality of the binary classification task of failure detection and provides a balanced measure when the class sizes are different. It takes into account both true and false positives and negatives respectively while assessing performance.



**Fig. 4: DECIDER produces the best performance on covariate shifts..** (left) Comparison of DECIDER against the best baseline in terms of the difference in MCC on the PACS dataset involving covariate shifts across 4 different visual domains. (Right) Improvement in failure recall performance of the best performing baseline and DECIDER on large-scale covariate shift benchmarks- DomainNet (DNet) and ImageNet-Sketch. The classifiers and PIMs are trained on DomainNet Real and Imagenet train sets respectively and evaluated on the different distribution shift datasets.

## 5.4 Findings

**Input Shifts.** Fig. 3(a) showcases the results on the CIFAR100 and CIFAR100-C datasets. On the clean CIFAR100, DECIDER outperforms the baselines with a superior MCC of 0.5292 for the max variant (versus 0.514 for the best baseline), attributed to higher failure recall (0.7933) and success recall (0.7474). On the more challenging CIFAR100-C (severity level 4), DECIDER further highlights its efficacy by achieving an MCC of 0.4015 with max aggregation, exceeding the top baseline (entropy) which has an MCC of 0.3766. This is due to a balanced trade-off between failure recall (0.8448) and success recall (0.5506), distinguishing DECIDER from other baselines that fail to maintain such balance. These findings clearly demonstrate DECIDER as robust in detecting classifier failures amid input-level shifts, surpassing other baselines in performance metrics.

**Subpopulation Shifts.** Our comprehensive evaluation addresses datasets affected by various subpopulation shifts. The summarized results in Fig. 3(b) underline the effectiveness of DECIDER in navigating these challenges:

**Waterbirds:** DECIDER achieves a high failure recall of 0.6063, outperforming the best baseline (entropy) which has a recall of 0.4878. Importantly, DECIDER maintains a high success recall (0.858) with minimal compromise compared to MSP (0.8891). The outcome is a leading MCC of 0.4598, attesting to DECIDER’s balanced detection ability in environments with misleading background cues.

**CelebA:** With mean aggregation, DECIDER delivers the highest MCC of 0.4928, combining a failure recall of 0.5443 with a success recall of 0.9701, showcasing its strength in addressing gender and hair color spurious correlations.

**Cats vs Dogs:** Exhibiting strong performance in class imbalance, DECIDER (max aggregation) achieves an MCC of 0.5532, significantly surpassing the top baseline









(energy) with an MCC of 0.3428, underlining its efficacy in balanced success and failure recall. DECIDER not only demonstrates high failure detection capability but also ensures high success recall rates above 0.94, highlighting its proficiency in class-imbalanced settings.

**Covariate Shifts.** In this section, we evaluate the performance of DECIDER in the challenging setting of identifying failure due to covariate shifts. We first consider the PACS dataset which contains 4 different domains. We train PIM and derive individual thresholds for each of the four domains and evaluate its performance across all domains. While we present detailed results for baselines and metrics in the supplementary, in Fig. 4(a), we report the gain in MCC scores between the best performing baseline and DECIDER. It can be seen that DECIDER outperforms the baselines by a large margin across all the domains. To further validate the effectiveness of DECIDER, we conducted experiments on large-scale covariate shift benchmarks, including DomainNet and ImageNet. In the DomainNet case, we trained the classifier and PIM on images from the real domain and evaluated their performance on four different target domains: Cartoon, Sketch, Painting, and Infograph. For ImageNet, we trained on the ImageNet training dataset and assessed the performance on the challenging ImageNet-Sketch benchmark. Fig. 4(b) presents the failure recall performance of the best-performing baseline and DECIDER, clearly demonstrating the superiority of our approach even when applied to large-scale datasets.

In summary, these results highlight the importance of leveraging language priors together with priors from the VLM to construct debiased models that reliably help detect failures across different scenarios.

## 6 Failure Explanation

Having empirically demonstrated the superior failure detection capabilities of DECIDER, we now turn our attention to the task of explaining the reasons behind failures. To that end, we consider the max variant of DECIDER and adjust the influence of individual attributes to ensure that the prediction probabilities generated by DECIDER closely mirror those of the original model as explained in Section 4. This manipulation offers evidence of what attributes the task model uses. For e.g., on the top left of Fig. 5, the task is to correctly identify the hair color. Here, the classifier  $\mathcal{F}$  incorrectly classifies the image, while PIM accurately identifies the same. We observe that our optimization process reduces the influence of core attributes such as "Browning Tresses" and "Red Highlights" on PIM's predictions. This manipulation serves as evidence that the biased classifier  $\mathcal{F}$  may not have considered these crucial attributes in its decision-making process. Similarly, in the example shown in Fig. 5,  $\mathcal{F}$  misclassifies a Cat as a Dog (top right) and the proposed optimization shows that the classifier has not focused enough on the important core attributes such as "Thin Whiskers" thus making the erroneous classification.

TM: Task Model		PIM: Prior Induced Model	
	<b>TM: Landbird</b> <b>PIM: Waterbird</b>		<b>TM: Dog</b> <b>PIM: Cat</b>
	<b>TM: Waterbird</b> <b>PIM: Landbird</b>		<b>TM: Cat</b> <b>PIM: Dog</b>
	<b>TM: Blond</b> <b>PIM: Not Blond</b>		<b>TM: Blond</b> <b>PIM: Not Blond</b>
	<b>TM: Not Blond</b> <b>PIM: Blond</b>		<b>TM: Not Blond</b> <b>PIM: Blond</b>

**Fig. 5: Failure Explanations.** We explain the failures of the biased classifier  $\mathcal{F}$ , by manipulating the influence of individual attributes in PIM, such that the prediction probabilities of PIM match that of  $\mathcal{F}$ . The knowledge of the attributes whose influence was needed to be reduced provides an indication that  $\mathcal{F}$  has not focused on those attributes to make its decisions. We show qualitative examples on Water birds in top left, Cats vs dogs in top right and from CelebA dataset in bottom.

## 7 Analyses

**Biases or insufficiency of GPT-3 attributes.** The success of DECIDER relies on the quality of the attributes generated by the LLM. To study the impact on failure detection on the quality of text attributes, we consider two practical scenarios: (i) GPT-3 generates irrelevant attributes: In this case, the PIM model has the risk of learning noisy decision rules that the even the classifier might not have; (ii) GPT provides insufficient attributes: With only partial attributes, PIM’s predictive performance can be limited. To comprehensively evaluate the impact of both scenarios, we employ the following protocol on the Waterbirds dataset. For scenario (i), we add 5 randomly sampled core attributes from the other class to the attribute set of each class. For case (ii), we remove 5 randomly selected attributes from the attribute set of each class. We train PIM under both these scenarios. As the results in Table 1 show, although there is a noticeable drop in performance due to the severe attribute corruptions, DECIDER still outperforms the best baseline (Ent) method. This demonstrates the robustness of DECIDER to imperfect attribute sets.

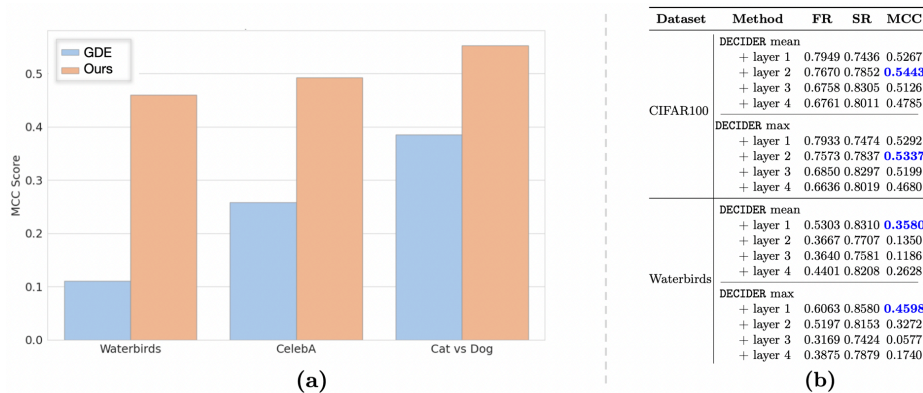
**Impact of Layer Selection of  $\mathcal{F}$  on  $\phi$ .** In this study, we explore how the performance of the PIM model  $\phi$  is influenced by the specific layer in  $\mathcal{F}$  from which we extract features. This experiment uses the ResNet-18 architecture,

**Table 1: Impact of attribute quality** – (i) *irrelevant*: add 5 nuisance attributes; (ii) *insufficient*: remove 5 core attributes. Although there is a drop in performance under attribute corruptions, DECIDER still outperforms existing baselines.

Metric	Baseline (Ent)	DECIDER	DECIDER (irrelevant)	DECIDER (insufficient)
Failure Recall	0.48	<b>0.60</b>	0.54	0.49
Success Recall	0.80	<b>0.85</b>	0.81	0.83
MCC	0.28	<b>0.45</b>	0.34	0.33

with models trained on the CIFAR100 and Waterbirds datasets. From the results presented in the table in Fig. 6, using features from the early layers (layer 1 and layer 2) of ResNet-18 yields the highest MCC (Matthews Correlation Coefficient) scores. In contrast, leveraging features from the later layers leads to a noticeable decline in performance. This observation suggests that the initial layers of the network are less prone to carrying biases than the later ones, supporting the findings from previous research [22].

**Model Ensembles for Disagreement Analysis.** It has been shown that the prediction disagreement between different constituent members of a model ensemble can serve as an indicator of failure [18,45]. In this experiment, we compare the failure estimation performance obtained through the disagreement between PIM and  $\mathcal{F}$  to the performance obtained by the disagreement between an ensemble (GDE). To that end, we trained five different classifiers with different initial seeds on three different datasets: Waterbirds, CelebA, and Cat vs Dogs. Figure 6, evidences the superiority of the proposed approaches compared to GDE.



**Fig. 6:** (a) Comparison of DECIDER against the failure detection performance obtained through disagreement between predictions from an ensemble of multiple instances of  $\mathcal{F}$  on Waterbirds, CelebA and Cats vs Dogs datasets respectively. (b) Ablation study analyzing the impact of using features from different layers of the base model  $\mathcal{F}$  as input to the Prior Induced Model (PIM)  $\phi$  on CIFAR-100 and Waterbirds datasets.

**Impact of PIM accuracy on failure detection.** Since we attempt to train a debiased classifier, in this section, we study the impact of its accuracy on failure detection. Table 1 in the appendix reveals that, despite the occasional slight decrease in the predictive performance of the debiased model PIM, the core-nuisance attribute disambiguation, which is crucial for failure detection, is not compromised. Consequently, DECIDER consistently achieves superior failure recall compared to the baselines.

**Replacing PIM with CLIP classifiers.** Given that we propose leveraging the priors from CLIP to obtain a debiased version of the classifier, it is natural to consider utilizing CLIP’s zero-shot classifier directly as PIM. Table 2 in appendix demonstrates that such an approach yields poor failure detection performance when CLIP’s zero-shot classifier is employed as PIM. This is because the visual features and their correlations to the core attributes of CLIP can differ significantly from the task model, thus rendering the model disagreement based failure detection highly ineffective.

## 8 Conclusion

In this work, we introduced DECIDER, a novel approach that leverages LLMs and vision-language foundation models to detect failures in pre-trained image classification models. Our key insight was to train an improved version of the pre-trained classifier, PIM, that learns robust associations between visual features and class-level attributes by projecting into the shared embedding space of a VLMs such as CLIP. By analyzing the disagreement between PIM’s predictions and the original biased model, DECIDER can reliably identify potential failures while offering human-interpretable explanations. Extensive experiments across multiple benchmarks evidences the consistent superiority of DECIDER over baselines, achieving substantially higher overall scores and better trade-offs between failure and success recalls. Our work highlights the promise of integrating vision-language priors into model failure analysis pipelines to facilitate more reliable and trustworthy deployment of vision models in safety-critical applications. Extending DECIDER to other vision-language models and exploring its application to other failure modes such as adversarial attacks constitute our future work.

## Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344, Lawrence Livermore National Security, LLC. Supported by LDRD project 24-FS-002. LLNL-CONF-862086

## References

1. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. *Advances in neural information processing systems* **33**, 1877–1901 (2020)
2. Chen, J., Liu, F., Avci, B., Wu, X., Liang, Y., Jha, S.: Detecting errors and estimating accuracy on unlabeled data with self-training ensembles. *Advances in Neural Information Processing Systems* **34**, 14980–14992 (2021)
3. Cukierski, W.: Dogs vs. cats (2013), <https://kaggle.com/competitions/dogs-vs-cats>
4. Deng, A., Xiong, M., Hooi, B.: Great models think alike: Improving model reliability via inter-model latent agreement. In: *Proceedings of the 40th International Conference on Machine Learning. Proceedings of Machine Learning Research*, vol. 202, pp. 7675–7693. PMLR (23–29 Jul 2023)
5. Esmaeilpour, S., Liu, B., Robertson, E., Shu, L.: Zero-shot out-of-distribution detection based on the pre-trained model clip. In: *Proceedings of the AAAI conference on artificial intelligence*. vol. 36, pp. 6568–6576 (2022)
6. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: *international conference on machine learning*. pp. 1050–1059. PMLR (2016)
7. Garg, S., Balakrishnan, S., Lipton, Z.C., Neyshabur, B., Sedghi, H.: Leveraging unlabeled data to predict out-of-distribution performance. In: *International Conference on Learning Representations* (2022), [https://openreview.net/forum?id=o\\_HsiMPYh\\_x](https://openreview.net/forum?id=o_HsiMPYh_x)
8. Geirhos, R., Jacobsen, J.H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., Wichmann, F.A.: Shortcut learning in deep neural networks. *Nature Machine Intelligence* **2**(11), 665–673 (2020)
9. Goyal, S., Kumar, A., Garg, S., Kolter, Z., Raghunathan, A.: Finetune like you pretrain: Improved finetuning of zero-shot vision models. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 19338–19347 (2023)
10. Guillory, D., Shankar, V., Ebrahimi, S., Darrell, T., Schmidt, L.: Predicting with confidence on unseen distributions. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 1134–1144 (2021)
11. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: *International conference on machine learning*. pp. 1321–1330. PMLR (2017)
12. Guo, J., Li, J., Li, D., Tiong, A.M.H., Li, B., Tao, D., Hoi, S.: From images to textual prompts: Zero-shot visual question answering with frozen large language models. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 10867–10877 (2023)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 770–778 (2016)
14. Hendrycks, D., Dietterich, T.G.: Benchmarking neural network robustness to common corruptions and perturbations. In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net* (2019), <https://openreview.net/forum?id=HJz6tiCqYm>
15. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations* (2017)

16. Hendrycks, D., Mu, N., Cubuk, E.D., Zoph, B., Gilmer, J., Lakshminarayanan, B.: AugMix: A simple data processing method to improve robustness and uncertainty. *Proceedings of the International Conference on Learning Representations (ICLR)* (2020)
17. Jain, S., Lawrence, H., Moitra, A., Madry, A.: Distilling model failures as directions in latent space. In: *The Eleventh International Conference on Learning Representations (2023)*, <https://openreview.net/forum?id=99RpBVpLiX>
18. Jiang, Y., Nagarajan, V., Baek, C., Kolter, J.Z.: Assessing generalization of SGD via disagreement. In: *International Conference on Learning Representations (2022)*, <https://openreview.net/forum?id=Wv0GCEAQhx1>
19. Jiang, Y., Neyshabur, B., Mobahi, H., Krishnan, D., Bengio, S.: Fantastic generalization measures and where to find them. *arXiv preprint arXiv:1912.02178* (2019)
20. Joshi, N., Pan, X., He, H.: Are all spurious features in natural language alike? an analysis through a causal lens. *arXiv preprint arXiv:2210.14011* (2022)
21. Kirsch, A., Mukhoti, J., van Amersfoort, J., Torr, P.H.S., Gal, Y.: On pitfalls in ood detection: Entropy considered harmful (2021), *uncertainty & Robustness in Deep Learning Workshop, ICML*
22. Lee, Y., Chen, A.S., Tajwar, F., Kumar, A., Yao, H., Liang, P., Finn, C.: Surgical fine-tuning improves adaptation to distribution shifts. *arXiv preprint arXiv:2210.11466* (2022)
23. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.M.: Deeper, broader and artier domain generalization. In: *Proceedings of the IEEE international conference on computer vision*. pp. 5542–5550 (2017)
24. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: *International Conference on Machine Learning*. pp. 12888–12900. PMLR (2022)
25. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. *Advances in neural information processing systems* **33**, 21464–21475 (2020)
26. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: *Proceedings of International Conference on Computer Vision (ICCV)* (December 2015)
27. Merullo, J., Castriato, L., Eickhoff, C., Pavlick, E.: Linearly mapping from image to text space. *arXiv preprint arXiv:2209.15162* (2022)
28. Michels, F., Adaloglou, N., Kaiser, T., Kollmann, M.: Contrastive language-image pretrained (clip) models are powerful out-of-distribution detectors. *arXiv preprint arXiv:2303.05828* (2023)
29. Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., Lucic, M.: Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems* **34**, 15682–15694 (2021)
30. Ming, Y., Cai, Z., Gu, J., Sun, Y., Li, W., Li, Y.: Delving into out-of-distribution detection with vision-language representations. In: Oh, A.H., Agarwal, A., Belgrave, D., Cho, K. (eds.) *Advances in Neural Information Processing Systems* (2022), <https://openreview.net/forum?id=KnCS9390Va>
31. Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., et al.: Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332* (2021)
32. Narayanaswamy, V., Anirudh, R., Kim, I., Mubarka, Y., Spanias, A., Thiagarajan, J.J.: Predicting the generalization gap in deep models using anchoring. In: *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. pp. 4393–4397 (2022)



33. Ng, N., Cho, K., Hulkund, N., Ghassemi, M.: Predicting out-of-domain generalization with local manifold smoothness. arXiv preprint arXiv:2207.02093 (2022)
34. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 1406–1415 (2019)
35. Pratt, S., Covert, I., Liu, R., Farhadi, A.: What does a platypus look like? generating customized prompts for zero-shot image classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15691–15701 (2023)
36. Qu, H., Li, Y., Foo, L.G., Kuen, J., Gu, J., Liu, J.: Improving the reliability for confidence estimation. In: European Conference on Computer Vision. pp. 391–408. Springer (2022)
37. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
38. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
39. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog **1**(8), 9 (2019)
40. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* **115**(3), 211–252 (2015). <https://doi.org/10.1007/s11263-015-0816-y>
41. Schwenk, D., Khandelwal, A., Clark, C., Marino, K., Mottaghi, R.: A-okvqa: A benchmark for visual question answering using world knowledge. In: European Conference on Computer Vision. pp. 146–162. Springer (2022)
42. Song, H., Dong, L., Zhang, W.N., Liu, T., Wei, F.: Clip models are few-shot learners: Empirical studies on vqa and visual entailment. arXiv preprint arXiv:2203.07190 (2022)
43. Subramanyam, R., Jayram, T., Anirudh, R., Thiagarajan, J.J.: Crepe: Learnable prompting with clip improves visual relationship prediction. arXiv preprint arXiv:2307.04838 (2023)
44. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Baidykov, N., Batra, S., Bhargava, P., Bhosale, S., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
45. Trivedi, P., Koutra, D., Thiagarajan, J.J.: A closer look at scoring functions and generalization prediction. In: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1–5. IEEE (2023)
46. Wang, H., Ge, S., Lipton, Z., Xing, E.P.: Learning robust global representations by penalizing local predictive power. In: Advances in Neural Information Processing Systems. pp. 10506–10518 (2019)
47. Wang, H., Li, Y., Yao, H., Li, X.: Clipn for zero-shot ood detection: Teaching clip to say no. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1802–1812 (2023)
48. Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A.W., Lester, B., Du, N., Dai, A.M., Le, Q.V.: Finetuned language models are zero-shot learners. In: International Conference on Learning Representations (2022), <https://openreview.net/forum?id=gEZrGCozdqR>

49. Wei, Y., Hu, H., Xie, Z., Liu, Z., Zhang, Z., Cao, Y., Bao, J., Chen, D., Guo, B.: Improving clip fine-tuning performance. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5439–5449 (2023)
50. Wortsman, M., Ilharco, G., Kim, J.W., Li, M., Kornblith, S., Roelofs, R., Lopes, R.G., Hajishirzi, H., Farhadi, A., Namkoong, H., et al.: Robust fine-tuning of zero-shot models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7959–7971 (2022)
51. Yang, Y., Zhang, H., Katabi, D., Ghassemi, M.: Change is hard: A closer look at subpopulation shift. In: International Conference on Machine Learning (2023)
52. Yu, S., Cho, J., Yadav, P., Bansal, M.: Self-chained image-language model for video localization and question answering. *Advances in Neural Information Processing Systems* **36** (2024)
53. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: International Conference on Computer Vision (ICCV) (2019)
54. Zhu, F., Cheng, Z., Zhang, X.Y., Liu, C.L.: Rethinking confidence calibration for failure prediction. In: European Conference on Computer Vision. pp. 518–536. Springer (2022)