# Unifying Model Explainability and Robustness via Machine-Checkable Concepts

Vedant Nanda MPI-SWS & University of Maryland vedant@cs.umd.edu

Till Speicher
MPI-SWS
tspeicher@mpi-sws.org

John P. Dickerson University of Maryland john@cs.umd.edu Krishna P. Gummadi MPI-SWS gummadi@mpi-sws.org

#### **Muhammad Bilal Zafar**

Bosch Center for Artificial Intelligence muhammadbilal.zafar@de.bosch.com

## **ABSTRACT**

As deep neural networks (DNNs) get adopted in an ever-increasing number of applications, explainability has emerged as a crucial desideratum for these models. In many real-world tasks, one of the principal reasons for requiring explainability is to in turn assess prediction robustness, where predictions (*i.e.*, class labels) that do not conform to their respective explanations (*e.g.*, presence or absence of a concept in the input) are deemed to be unreliable. However, most, if not all, prior methods for checking explanation-conformity (*e.g.*, LIME, TCAV, saliency maps) require significant manual intervention, which hinders their large-scale deployability. In this paper, we propose a robustness-assessment framework, at the core of which is the idea of using machine-checkable concepts. Our framework defines a large number of concepts that the DNN explanations could be based on and performs the explanation-conformity check at test time to assess prediction robustness. Both steps are executed in an automated manner without requiring any human intervention and are easily scaled to datasets with a very large number of classes. Experiments on real-world datasets and human surveys show that our framework is able to enhance prediction robustness significantly: the predictions marked to be robust by our framework have significantly higher accuracy and are more robust to adversarial perturbations.

## 1 Introduction

Explainability has emerged as an important requirement for deep neural networks (DNNs). Explanations target a number of secondary objectives of model design (in addition to the primary objective of maximizing prediction accuracy), such as informativeness, transferability and audit of ethical values [15, 32, 36]. One of the most important desiderata of explainability is model *robustness*, whereby explanations are used to assess the extent to which some downstream task could rely on the model's predictions. For instance, a prediction classifying an input as a wolf with the explanation that the background contains snow is unlikely to be trusted by the downstream system [40]. A long line of research has focused on rendering DNN predictions explainable with the—often implicit—goal of assessing prediction robustness [3, 11, 28, 29, 30, 31, 33, 40, 41, 43].

However, the scalability of these explanation-based robustness assessment schemes is limited by the need for "humans-in-the-loop". Prediction robustness checks based on explanations operate as following: Given an input, one or more human-interpretable concepts are identified that have a significant impact on the model prediction. Then an *explanation-conformity check* is performed to see whether the concept-prediction relationship matches human-reasoning. In the above example of wolf and snow [40], a human may deem the concept-prediction relationship (snow-wolf) to be unreasonable, and consider the prediction to be non-robust. However, identifying human-interpretable concepts and checking for human-reasoning requires significant human effort by the way of manual annotation of either the inputs (*e.g.*, TCAV [28]), intermediate model components (*e.g.*, LIME [40]) or both (*e.g.*, saliency maps [43]). In practice, human involvement makes many explanation-based robustness assessments unsuitable for large-scale deployment.

**Goals and contributions.** In this paper, our goal is to design a highly scalable robustness assessment framework that *automates the end-to-end process of performing explanation-conformity checks*. At the foundation of our framework are concepts with the following key properties:

- 1. The concepts are *identified automatically* from the training data without any human effort.
- 2. They are *machine-checkable*, *i.e.*, they lend themselves to 'concept–class' style automated explanation-conformity checks without any human involvement.
- 3. They can be *added to off-the-shelf, pretrained DNNs* in a post-hoc manner to assess prediction robustness.

We devise an intuitive procedure for identifying machine-checkable concepts ( *MACCs* ) that satisfy the above key properties. Specifically, our framework automatically defines a large number of MACCs, each corresponding to features shared by some subset of one or more classes (and not shared by other classes) in the training data. At the end of the concept-identification process, each class in the training data has a unique set of corresponding MACCs. Finally, with each prediction of the DNN, our framework performs an automated explanation-conformity check to see if the MACCs corresponding to the predicted class are also detected in the learnt representations of the input (and the MACCs not corresponding to the predicted class are not detected). The predictions passing the explanation-conformity check are deemed robust, even if individual MACCs are hard for humans to recognize.

Experiments and human surveys on real-world image classification datasets show that MACCs help increase the prediction robustness significantly. Specifically, we find that (i) explanation-conformant predictions are not only significantly more accurate, but their corresponding images are also easier for humans to classify confidently than non-conformant predictions, (ii) adversarial attacks against explanation-conformant predictions are significantly harder and in many cases impractical, and (iii) MACCs also provide insights into the potential causes for prediction errors.

# 2 Methodology

In this section, we describe our framework for robust prediction.

Formal problem setup and notation. Let  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$  denote a training dataset of N examples with  $x \in \mathcal{X} = \mathbb{R}^d$  and  $y \in \mathcal{Y} = \{1, 2, \dots, K\}$ . The learning task involves obtaining a mapping  $F_{\text{clf}} : \mathcal{X} \to \mathcal{Y}$ . For a (deep) neural network with L hidden layers, this mapping consists of applying a set of parameterized layers  $f_l(x_l, \theta_l)$ . Here,  $x_l$  and  $\theta_l$  denote, respectively, the input and parameters of the  $l^{th}$  layer. The whole neural network mapping can be expressed as:  $F_{\text{clf}}(x) = f_{\text{clf}}(f_L(f_{L-1}(\dots, f_1(x, \theta_1))))$ , where the output of  $f_{\text{clf}}$ —or the classification layer—is a K-dimensional vector consisting of (potentially un-calibrated) probabilities, generally obtained by applying the softmax function within the layer  $f_{\text{clf}}$ . One then obtains the prediction  $\hat{y} = \operatorname{argmax} F_{\text{clf}}(x)$ . The learning then boils down to minimizing the discrepancy between the predicted and the ground-truth labels. For the sake of computational tractability, this discrepancy is often expressed via the (categorical) cross-entropy loss function, denoted henceforth as  $\mathcal{L}_{\text{clf}}(F_{\text{clf}}(x), y)$ .

## 2.1 Our framework: Robustness via machine-checkable concepts

Our framework, summarized in Figure 1, consists of three main components: Defining machine-checkable concepts (MACCs), leveraging the DNN to detect MACCs, and performing explanation-conformity checks with MACCs to assess prediction robustness. We now describe each of the components individually.

# 2.1.1 Automatically defining MACCs

The first component of our framework automatically defines MACCs that are amenable to explanation-conformity checks without any human intervention. To define MACCs, we leverage the following key insight [25, 36]: one method of composing explanations is to point to presence or absence of *concepts* in the input, where a concept is a feature that is possessed by inputs of a certain set of classes in the dataset, and not possessed by other classes. For instance, in an animal classification task involving zebras, cats and dogs, zebras might have a unique concept *stripes* [28], that is not shared by any other class. Similarly, dogs and cats might share a concept *paws* that is not shared by any other class.

Most prior works detect these concepts by manually annotating (parts of) inputs that contain them (e.g., [28, 40, 41]). Instead of manually annotating the inputs, for every possible subset of one or more classes, we define one MACC that corresponds to the features shared by inputs in that subset. This way of defining MACCs leads to  $M = 2^K - 1$  concepts in a dataset with K classes. For instance, in a datasets with classes cat, dog and zebra, one can define  $2^3 - 1 = 7$  MACCs, as follows { $c_{\text{cat}}$ ,  $c_{\text{dog}}$ ,  $c_{\text{zebra}}$ ,  $c_{\text{cat/dog}}$ ,  $c_{\text{cat/zebra}}$ ,  $c_{\text{cat/dog/zebra}}$ }. Figure 1 shows all overlaps involving two classes. In the figure, the concept  $c_{\text{dog/cat}}$  denotes a property shared by dog and cat, but not by zebras. Similarly,  $c_{\text{dog/zebra}}$  denotes a property possessed by zebras and dogs but not by cats.

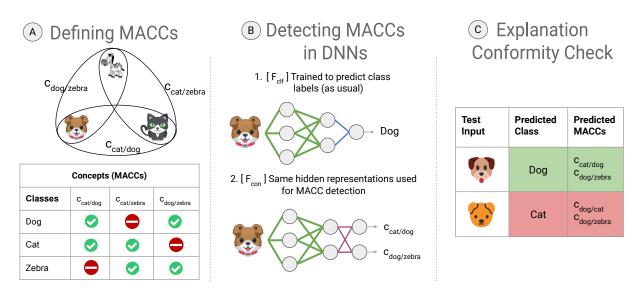


Figure 1: [System overview] We propose the use of (a) Machine-checkable concepts (MACCs), that are defined as concepts shared between inputs of one or more classes (Section 2.1.1)—the figure only shows the MACCs shared between two classes. (b) Automatically detecting MACCs involves adding an additional classification layer (to any hidden layer) of an existing DNN (Section 2.1.2). (c) At test time, we perform the explanation-conformity check to ensure that the MACCs corresponding to the predicted class are also detected in the image (Section 2.1.3). The predictions not passing the explanation-conformity check are deemed non-robust.

#### 2.1.2 Detecting MACCs in DNNs

Given a DNN  $F_{\rm clf}$  as in the formal setup, trained to predict the class labels, we express the MACC detector  $F_{\rm con}$  as:  $F_{\rm con}(\boldsymbol{x}) = f_{\rm con}(f_L(f_{L-1}(\dots,f_1(\boldsymbol{x},\boldsymbol{\theta}_1))))^{-1}$  where the output of  $f_{\rm con}$  is an M-dimensional vector consisting of (potentially un-calibrated) probabilities,  $p(c_i=1|\boldsymbol{x})$ . Since  $F_{\rm con}$  attempts a multilabel classification task, we obtain the probabilities using the sigmoid function  $\sigma(z) = (1+e^{-z})^{-1}$ . Finally, one obtains a predicted MACC vector  $\hat{\boldsymbol{c}} = [\hat{c}_1,\dots,\hat{c}_M]$  with  $\hat{c}_i = 1$  indicating the predicted presence/absence of each MACC in the input. Here,  $F_{\rm con}(\boldsymbol{x})_i > 0.5$ , else  $\hat{c}_i = 0$ . Learning  $F_{\rm con}$  can be done by optimizing the sum of M individual binary cross-entropy loss functions, with one loss function for each MACC. We refer to this sum of loss functions as  $\mathcal{L}_{\rm con}$ .

Our framework allows for the flexibility to be trained in two different ways: (1) **Post-hoc training:** Taking a pretrained DNN  $F_{\text{clf}}$  as described in the formal setup, and training the MACC detection layer,  $f_{\text{con}}$  by attaching it to one of the hidden layers of  $F_{\text{clf}}$ . With this method, the pre-learnt representations of  $F_{\text{clf}}$  are used and *only* the parameters of  $f_{\text{con}}$  are learnt. (2) **Joint training:** Training all the parameters of the network from scratch, that is, training the hidden layers  $f_i$ ,  $\forall i \in \{1 \dots L\}$ , the class label layer  $f_{\text{clf}}$ , and the MACC layer  $f_{\text{con}}$  by minimizing the joint loss  $\lambda \mathcal{L}_{\text{clf}} + (1 - \lambda)\mathcal{L}_{\text{con}}$ . Here the parameter  $\lambda$  trades-off the accuracy between the class labels prediction accuracy and the MACC detection accuracy, and can be determined via cross-validation. Finally, a combination of these two techniques (*e.g.*, selectively training only some hidden layers) can also be used.

#### 2.1.3 Explanation-conformity checks with MACCs

The final component of our framework constitutes of performing an explanation-conformity check with MACCs to assess prediction robustness. Our intuition is that predictions passing the check would be more robust. Our explanation-conformity check proceeds as follows: Given an input instance  $\boldsymbol{x}$ , let  $\hat{y} = F_{\text{clf}}(\boldsymbol{x})$  be the class prediction and  $\hat{c} = F_{\text{con}}(\boldsymbol{x})$  be the MACC prediction. Then, the explanation-conformity check probes if the MACCs corresponding to the predicted class are also detected (and the MACCs not related to the predicted class are not detected). The prediction is deemed robust if,  $\frac{\sum_i \mathbb{I}[\hat{c}_i = c_i^{\hat{y}}]}{M} \ge t_{\text{con}}$ , for some  $t_{\text{con}} \in [0,1]$ . A higher value of  $t_{\text{con}}$  means that fewer predictions would pass the explanation-conformity check, however, the degree of robustness for these predictions is expected to be higher (see Section 3 for details).

<sup>&</sup>lt;sup>1</sup>Note that  $f_{con}$  can be attached to any intermediate layer between  $f_1$  to  $f_L$ .

#### 2.2 Discussion: Salient properties of MACCs

MACCs and Human-interpretability. Most of the existing works on concept-centered explanation-conformity checks [28, 40] use human supervision to annotate images as containing a certain concept. Such concepts often correspond to features that are (i) shared by certain classes and not shared by other classes in the data, and, (ii) can be easily recognized and named by humans (e.g., stripes on zebras, paws on cats and dogs). While MACCs are not explicitly human recognizable, and hence do not satisfy criterion (ii), their definition procedure (Section 2.1.1) ensures that they do indeed satisfy criterion (i). In this sense, MACCs subsume the concepts defined in prior work on concept-centered explainability. However, our framework trades-off human recognizability of MACCs to enable end-to-end automation of robustness assessments from MACC definition  $\rightarrow$  detection  $\rightarrow$  explanation-conformity checks.

**Pruning MACCs.** It is quite possible that in a K-class classification task, some classes may not share any meaningful features, and their corresponding MACCs, may not correspond to any useful concepts. For instance, the class cat may not share any similarities with class kite, and hence, the corresponding MACC might be meaningless. We expect these MACCs to have low detection accuracy. During the training procedure (Section 2.1.2), such MACCs can be dropped. Moreover, since the possible space of MACCs is very large (for a dataset of K=100 classes, there are a total of  $2^{100}-1\approx 10^{30}$ ) possible MACCs, one could use a random subset of MACCs, or only consider MACCs that represent properties shared by exactly two, or exactly three classes (e.g., in Figure 1). Finally, MACCs that uniquely correspond to a class may be redundant in conformity checks and can be safely pruned.

#### 3 Evaluation of the robustness framework

In this section, we conduct experiments and human surveys on real-world datasets to evaluate the effectiveness of our MACC framework. Specifically, we ask whether the predictions passing the MACC explanation-conformity achieve better robustness.

**Evaluation metrics.** Inspired by usage of explanation-conformity checks in practice [6, 15, 44], we use the following evaluation metrics to quantify prediction robustness: (i) *Error Estimability*, *i.e.*, accuracy on explanation-conformant predictions, (ii) *Error Vulnerability*, *i.e.*, resistance to adversarial attacks on explanation-conformant predictions, and, (iii), *Error Explainability*, *i.e.*, ability to map errors to potential issues in the input.

**Setup.** We conduct experiments on CIFAR-10, CIFAR-100 and Fashion MNIST datasets. We define MACCs such that each class in CIFAR-10 and Fashion MNIST data is accompanied by 9 MACCs whereas in CIFAR-100 data, this number is 99.

We use simple deep CNN architectures, that have publicly available implementations, and provide comparable performance to state-of-the-art. Additional details on data preprocessing, MACC definition, picking  $t_{\rm con}$ , and training architectures can be found in Appendix A.

Training the models to maximize the classification accuracy leads to a test set accuracy of 88.8%, 92.49% and 59.41% on CIFAR-10, Fashion MNIST and CIFAR-100 datasets, respectively. We refer to this model as the *vanilla* model. For the training of  $F_{con}$ , we consider the post-hoc training alternative considered in Section 2. The joint training alternative leads to similar statistics. For the detailed analysis, we focus on the performance of post-hoc training and leave detailed comparison between different training schemes for a future study. For *performance comparison*, we use the probability calibration method of Guo et al. [24] (see Section 3.4).

We now present the performance of MACCs in improving prediction robustness.

#### 3.1 Do MACCs provide reliable Error Estimability?

We propose and test two hypotheses related to reliable error estimability: (i) predictions that pass the MACC explanation-conformity check are more likely to be accurate, and, (ii) predictions that are not explanation-conformant might consist of inputs with high aleatoric uncertainty [13] and might be more difficult for even humans to classify.

Table 1 shows that on all three datasets, the prediction accuracy on explanation-conformant predictions is significantly higher than non-conformant predictions, validating our hypothesis (i). To confirm our hypothesis (ii), we show images from CIFAR-10 data to human annotators at Amazon Mechanical Turk (AMT). The AMT annotators are shown an

<sup>&</sup>lt;sup>2</sup>Instead, MACCs may represent complex *polymorphic and composite* features in practice, i.e., the MACC corresponding to 'features shared by cats and dogs but not zebras' could correspond to a paw or the non-existence of stripes, or any combination of such distinguishing features.

Table 1: [Error Estimability] Accuracy of the vanilla DNN with no explanation-conformity check (Vanilla), accuracy on samples passing the explanation-conformity check (explanation-conf.) and on samples not passing the check (non explanation-conf.). Numbers in parentheses show the fraction of samples in each category. Accuracy on explanation-conformant predictions is significantly higher.

	Vanilla	Explanation-conf.	Non explanation-conf.
CIFAR-10	0.89 (1.00)	0.93 (0.91)	0.48 (0.09)
Fashion-MNIST	0.92 (1.00)	0.99 (0.70)	0.77 (0.30)
CIFAR-100	0.59 (1.00)	0.65 (0.84)	0.30 (0.16)

Table 2: [Error Vulnerability] Attacking class labels. Fraction of adversarially perturbed inputs that fail the explanation-conformity check (meaning the adversarial attack is detected.) On CIFAR-10 and Fashion-MNIST data, > 98% of the attacks are detected, except for DeepFool on CIFAR-10 where around 40% are detected. On CIFAR-100 data, around half of the adversarial attacks are detected.

	FGSM [21]	DeepFool [37]	C&W (L2) [9]	PGD [35]
CIFAR-10	0.98	0.41	1.00	0.99
Fashion-MNIST	1.00	0.99	1.00	1.00
CIFAR-100	0.50	0.45	0.49	0.50

image and asked to choose the class that the image belongs to from the list of 10 classes. Each image is annotated by 30 users. Further details on the experiment can be found in Appendix C.

The results show that for explanation-conformant images, humans are able to detect the correct class 91.25% of the time, whereas accuracy for non-conformant images is 83.19%. Moreover, the worker disagreement—as measured via average Shannon Entropy—is 0.22 and 0.39 for explanation-conformant, and non-conformant images. The difference in accuracy and worker agreement shows that the non explanation-conformant images are harder not only for the DNN, but also human annotators to classify. We expand on the difficulty of human annotators in Section 3.3.

#### 3.2 Do MACCs defend against Error Vulnerability?

We now ask if an explanation-conformity check can help defend against adversarial perturbations. Specifically, we start off with a 50% random subset of test images that were correctly classified by the vanilla DNN and adversarially perturb them w.r.t.  $F_{\rm clf}$  so that they are now incorrectly classified. We use a number of popular adversarial attacks (see Table 2). Next, we check if these adversarial perturbation designed to change the class labels also resulted in a corresponding change in the detected MACCs. If that is not the case, then MACC explanation-conformity check could be used as a method to detect adversarial perturbations.

Table 2 shows the fraction of adversarially attacked inputs that fails the MACC explanation-conformity check, revealing that the check is able to detect a vast fraction of adversarial attacks.

While MACCs are able to defend against a significant proportion of attacks on class labels, a determined adversary could *additionally* attack the MACC detection component ( $F_{con}$  in Section 2) such that not only does the class label get switched, the MACC prediction is also changed such that the explanation-conformity check is passed. We now study the nature of such adversarial perturbations. To perform this attack, we modify the PGD attack (details in Appendix D.1) such that the class labels and MACCs are changed in a consistent manner to pass the explanation-conformity check.

We note that the perturbation required to perform an explanation-conformant attack is significantly higher than the one required for an attack that aims to change the class label only. Specifically, while the class-only attacks in Table 2) require a perturbation (based on L2 distance from the original image) of  $0.31 \pm 0.20$  and  $0.26 \pm 0.14$  on CIFAR-10 and Fashion-MNIST datasets respectively, the explanation-conformant perturbations have a magnitude of  $5.31 \pm 5.62$  and  $3.16 \pm 2.85$ . In other words, explanation-conformant attacks require perturbations that are more than an order of magnitude larger.

Are the perturbations still imperceptible to humans? We suspect that the magnitude of the explanation-conformant perturbations is so large that they might not be imperceptible to humans anymore. Perturbations being imperceptible to humans is often considered as a major property adversarial perturbations [10, 38]. To test this hypothesis, we set up a human survey on AMT where the humans are shown three kinds of images: (i) the *original*, unperturbed image, (ii) image with *class-only* perturbation that aims to change the predicted class label, and, (iii) the image with *explanation-conformant* perturbation that aims to change the predicted class label as well as predicted MACCs such



Figure 2: [Error Vulnerability] Attacking both class labels and MACCs. The figure shows some randomly selected Original Images from CIFAR-10 that were correctly classified by the vanilla DNN. Also shown are the perturbed images obtained by performing a conventional adversarial attack, using the PGD method [35], aimed at switching the predicted class label on the input (Class-only). As is expected of such attacks [10, 38], the perturbations are imperceptible to the human eye. Finally, we show the perturbed images where the adversarial attack not only changes the predicted class labels, but also the MACCs such that the predictions are explanation-conformant (Explanation-conf.). Explanation-conformant perturbations are so large that they are clearly perceptible to the human eye.

Human agreement	< 6 MACCs detected	≥ 6 MACCs detected
$\leq 50\%$	75.4%	24.6%
> 50%	47.7%	52.3%





(a) Human agreement & num. detected MACCs

(b) Images with lowest agreement

Figure 3: [Insights into causes of errors] The table (3a) shows that images with less human agreement also tend to have few detected MACCs. The figure (3b) shows the images with lowest human agreement. For more examples and details, see Appendix C.

that the prediction passes the explanation-conformity check. AMT workers were then asked to label if the image contained an adversarial perturbation or not. Details of the survey can be found in Appendix D.2.

The results show that for class-only category, humans are able to detect the images with adversarial perturbations around 49.8% of the time, *i.e.*, the human accuracy is as good as a random guess. On the other hand, for images in the explanation-conformant category, the humans are able to detect the adversarially perturbed images 85% of the time. This vast difference in human detection accuracy shows that explanation-conformant perturbations are much more noticeable to the human eyes that class-only perturbations. Figure 2 also shows some examples of the explanation-conformant perturbations (more examples in Appendix D). In summary, the survey shows that *it is difficult to attack the MACC explanation-conformity check in a manner that is undetectable by humans*.

#### 3.3 Do MACCs provide insights into the causes of errors?

Inspired by the insight in Section 3.1 that even humans tend to make more errors on non explanation-conformant inputs, we now further explore these cases.

Specifically, we note that some non explanation-conformant inputs consists of cases where  $F_{con}$  is able to detect very few MACCs (see Appendix C.3 for a full distribution).<sup>3</sup> This means that the DNN is struggling to identify concepts related to *any* class in the input. We hypothesize that low concept detection rate might mean that these inputs might consist of cases where even humans might find it hard to identify the class of the image.

To test this hypothesis, we divide the non explanation-conformant images from the annotation task described in Section 3.1 into different categories based on the (dis)agreement between human annotators. The agreement here is

<sup>&</sup>lt;sup>3</sup>An explanation-conformant prediction, with  $t_{\rm con}=1$  in CIFAR-10 data would mean that  $F_{\rm con}$  detects 9 exactly MACCs in the input. See Appendix A for details on MACCs for each class.

measured as the fraction of the votes obtained by the class with most votes. Hence, an agreement value of 1.0 means that all humans annotated the image with the same class, whereas a value of 0.1 means that the most-voted-for class received votes that are no better than a random assignment (as the CIFAR-10 dataset consists of 10 classes).

Next, we divide the images into two categories: images where < 6 MACCs were detected and where  $\ge 6$  MACCs were detected. Figure 3a shows the relative fraction of these two categories against the human agreement. The figure shows that the images with small degree of agreement tend to mostly consist of cases where very few (< 6 MACCs) are detected. Specifically, out of the images with less or equal to 50% agreement, 75.4% of them have 5 or less MACCs detected. Figure 3b shows the images with lowest human agreement.

These results show that detection of very few MACCs in an image correlates with the fact that even human judges (who are often the source of ground truth in image classification tasks) would find it difficult to classify these images. Hence, *MACCs can serve as a useful tool to pinpoint problematic inputs in the data*. However, we do note that MACCs are not able to explain causes of errors for *all* the misclassified inputs, rather they only explain errors for a certain category of the data (with very few concepts detected).

#### 3.4 Discussion

The results show that MACCs can be used to perform explainability checks that significantly enhance predictions' robustness along a wide range of measures. In this section, we discuss some more pertinent points related to the implementation of MACCs.

Effect of varying  $t_{\rm con}$ . As described in Section 2, varying  $t_{\rm con}$  can be thought of as a flexible parameter to fine-tune prediction robustness. We further investigate the effect of  $t_{\rm con}$  on the fraction of samples deemed explanation-conformant and the prediction accuracy on these samples. Results in Appendix B shows that increasing  $t_{\rm con}$  leads to more samples being marked as explanation-conformant, however, the classification accuracy on explanation-conformant samples decreases.

Other methods for assessing prediction robustness. We also compare the robustness estimates obtained using the MACC explanation-conformity check with the more traditional method of probability calibration. Specifically, we use the temperature scaling method of Guo et al. [24] to calibrate the softmax probabilities.<sup>4</sup> We then predictions to be robust if the (calibrated) predicted class probability is above X, where X is chosen such that the same fraction of predictions are marked robust as by our method in Table 1.<sup>5</sup>

The comparison reveals that (i) both the robustness check based on calibrated probabilities and the MACC explanation-conformity check achieve comparable performance in terms of the tradeoff between predictions marked robust and classification accuracy on these predictions, however, (ii) the calibration method leads to much lower performance in terms of Error Vulnerability, *i.e.*, the amount of perturbation required to pass the calibration robustness check is almost an order of magnitude smaller. More details on the comparison can be found in Appendix D.

## 4 Related work

Most prior approaches to DNN explainability and robustness operate by identifying important features, concepts, or training data instances [7, 11, 14, 15, 23, 26, 27, 31, 40, 43]. The main differences between these studies and our approach is that we target a specific application of concept explainability, *i.e.*, the explanation-conformity check, and automate the robustness assessment procedure.

A line of work closely related to ours is that of concept-based explanations. Kim et al. [28] propose a method to evaluate how important a user-defined concept is in predicting a specific class. Yeh et al. [45] propose ways to find concepts that are enough to explain a given prediction. Ghorbani et al. [20] proposed ways to automatically extract concepts from visual data while Bouchacourt and Denoyer [8] proposed a similar approach for textual data. Goyal et al. [22], Shi et al. [42] focus on identifying human-interpretable concepts that have causal relationships with model's predictions. However, none of these methods proposes automation of explanation-conformity checks.

Some recent studies [19, 44] have focused on linking explainability and adversarial robustness. Ghorbani et al. [19] show that saliency map based explanations are easy to fool via adversarial attacks. On the other hand, MACCs are quite resistant to adversarial perturbations (Section 3.2). Tao et al. [44] propose an explanation-based check to detect adversarial perturbations. However their approach is limited to hand-crafted features and is specialized for facial

<sup>&</sup>lt;sup>4</sup>We use the implementation provided by the authors: github.com/gpleiss/temperature\_scaling

<sup>&</sup>lt;sup>5</sup>Comparison with more thresholds reveals similar insights. Details in Appendix B.

recognition, whereas our approach can be extended to more general image recognition tasks and also other classification tasks.

Prediction robustness has also been studied in the context of calibration and prediction uncertainty [12, 16, 17, 24, 34]. Empirical comparison with a recent calibration technique [24] shows that while the robustness check based on this technique provides comparable accuracy, MACCs are far more robust to adversarial perturbations (Section 3.4), and additionally help provide insights into the causes of errors (Section 3.3). Moreover, unlike many prior works in this line of research, *e.g.*, [17, 34], our proposed framework can be easily plugged into an existing trained model in a post-hoc manner.

Finally, MACCs also share some similarities with redundant output encoding and error correcting output codes (ECOC) [5, 18]. However, unlike MACCs, ECOCs do not provide an explanation-conformity check with a built-in reject option.

# 5 Conclusion, limitations & future work

In this work, we proposed a robustness assessment framework that uses Machine-checkable Concepts, or MACCs, to automate the end-to-end process of performing explanation-conformity checks. The automation means that our framework can be scaled to a large number of classes. MACCs partly achieve this scalability by focusing on a specific explainability desideratum—*i.e.*, assessment of prediction robustness—and potentially sacrificing some other desiderata (details in Section 1). Experiments and human-surveys on several real-world datasets show that the MACC explanation-conformity check facilitates higher prediction accuracy (on predictions passing the explanation-conformity check), adds resistance to adversarial perturbations, and can also help provide insights into the source of errors.

Our work opens several avenues for future work: For now, MACCs are defined such that they are shared between all images of the same class. A useful follow-up would be to consider *multiple sets* of MACCs per class to account for intra-class variability. Moreover exploring the MACC pruning strategies, analyzing the effect of the number of MACCs on the robustness, and a deeper exploration of the tradeoffs provided by different training methodologies mentioned in Section 2 (post-hoc, joint, or a combination) are also promising future directions.

We believe that our work has potential to provide significant positive impact for the society. As machine learning models are deployed in a wide array of real-world domains, the issue of prediction robustness has become increasingly relevant. The ability of our methods to provide improved uncertainty estimates, offer resistance to adversarial perturbations, and the capability to potentially debug the model errors is a useful tool for many societal applications. Examples of these applications include image search in online databases and driver-assistance systems in the automotive domains.

On the flip side, our methods are evaluated empirically and do not come with theoretical performance guarantees. As a result, appropriate care should be applied before using them in critical life-affecting domains. An analysis exploring the performance guarantees remains an important future research direction.

Most of the prior work on concept-based explanations restricts itself to concepts that can be explicitly named by humans (see Section 2.2 for a discussion). Our framework represents a departure from this restriction, and places more emphasis on machine-checkability (much like the line of work on machine-checkable theorem proving [47]). As a result, while our machine-checkable concepts (MACCs) are able to meet the goal that they were designed for, it should be noted that they may not fulfil some other explainability criteria [6, 15, 32]. Combining machine-checkability with human-interpretability would be a worthwhile future research direction.

## 6 Acknowledgements

Dickerson and Nanda were supported in part by NSF CAREER Award IIS-1846237, DARPA GARD #HR00112020007, DARPA SI3-CMD #S4761, DoD WHS Award #HQ003420F0035, and a Google Faculty Research Award. This work was supported in part by an ERC Advanced Grant "Foundations for Fair Social Computing" (no. 789373).

#### References

- [1] https://gist.github.com/Noumanmufc1/60f00e434f0ce42b6f4826029737490a,. Accessed: 2020-01.
- [2] https://github.com/aaron-xichen/pytorch-playground/blob/master/cifar/model.py, . Accessed: 2020-05.
- [3] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. How to explain individual classification decisions. *JMLR*, page 1803–1831, 2010.

- [4] Rodrigo Benenson. Are we there yet? https://rodrigob.github.io/are\_we\_there\_yet/build/classification\_datasets\_results.html#43494641522d3130, 2015. Accessed: 2019-09.
- [5] Adam Berger. Error-correcting output coding for text classification. Citeseer.
- [6] Umang Bhatt, Alice Xiang, Shubham Sharma, Adrian Weller, Ankur Taly, Yunhan Jia, Joydeep Ghosh, Ruchir Puri, José MF Moura, and Peter Eckersley. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657, 2020.
- [7] Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, 5(4):2403–2424, 2011.
- [8] Diane Bouchacourt and Ludovic Denoyer. Educe: Explaining model decisions through unsupervised concepts extraction, 2019.
- [9] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In 2017 IEEE Symposium on Security and Privacy (SP), pages 39–57, 2017.
- [10] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In 2017 ieee symposium on security and privacy (sp), pages 39–57. IEEE, 2017.
- [11] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnett, Jonathan Su, and Cynthia Rudin. This looks like that: deep learning for interpretable image recognition. In *NeurIPS*, 2019.
- [12] Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In ALT, 2016.
- [13] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2): 105–112, 2009.
- [14] Amit Dhurandhar, Karthikeyan Shanmugam, Ronny Luss, and Peder A Olsen. Improving simple models with confidence profiles. In *NeurIPS*. 2018.
- [15] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. In *eprint* arXiv:1702.08608, 2017.
- [16] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *ICML*, 2016.
- [17] Yonatan Geifman and Ran El-Yaniv. SelectiveNet: A deep neural network with an integrated reject option. In *ICML*.
- [18] Rayid Ghani. Using error-correcting for efficient text classification. In ICML, 2000.
- [19] Amirata Ghorbani, Abubakar Abid, and James Zou. Interpretation of neural networks is fragile, 2017. URL <a href="http://arxiv.org/abs/1710.10547">http://arxiv.org/abs/1710.10547</a>. cite arxiv:1710.10547Comment: Published as a conference paper at AAAI 2019.
- [20] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, pages 9273–9282, 2019.
- [21] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.
- [22] Yash Goyal, Amir Feder, Uri Shalit, and Been Kim. Explaining classifiers with causal concept effect (cace), 2019.
- [23] Sadaf Gulshad, Jan Hendrik Metzen, Arnold Smeulders, and Zeynep Akata. Interpreting adversarial examples with attributes. *arXiv* preprint arXiv:1904.08279, 2019.
- [24] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *ICML*, 2017.
- [25] Germund Hesslow. The problem of causal selection. *Contemporary science and natural explanation: Commonsense conceptions of causality*, pages 11–32, 1988.
- [26] Rajiv Khanna, Been Kim, Joydeep Ghosh, and Oluwasanmi Koyejo. Interpreting black box predictions using fisher kernels. In *AISTATS*, 2019.
- [27] Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *NeurIPS*, 2016.
- [28] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *ICML*, 2018.
- [29] Jinkyu Kim, Anna Rohrbach, Trevor Darrell, John Canny, and Zeynep Akata. Textual explanations for self-driving vehicles. In *ECCV*, pages 563–578, 2018.

- [30] Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: Patternnet and patternattribution. In *ICLR*, 2017.
- [31] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In ICML, 2017.
- [32] Zachary C. Lipton. The mythos of model interpretability. *Commun. ACM*, 61(10):36–43, September 2018. ISSN 0001-0782. doi: 10.1145/3233231. URL https://doi.org/10.1145/3233231.
- [33] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 4768–4777, 2017.
- [34] David Madras, Toni Pitassi, and Richard Zemel. Predict responsibly: improving fairness and accuracy by learning to defer. In NeurIPS, 2018.
- [35] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2017.
- [36] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267: 1–38, 2019.
- [37] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *CVPR*, 2016.
- [38] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519, 2017.
- [39] Jonas Rauber, Wieland Brendel, and Matthias Bethge. Foolbox: A python toolbox to benchmark the robustness of machine learning models, 2017.
- [40] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *KDD*, pages 1135–1144. ACM, 2016.
- [41] Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. *arXiv preprint arXiv:1703.03717*, 2017.
- [42] Tian Shi, Xuchao Zhang, Ping Wang, and Chandan K. Reddy. A concept-based abstraction-aggregation deep neural network for interpretable document classification, 2020.
- [43] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.
- [44] Guanhong Tao, Shiqing Ma, Yingqi Liu, and Xiangyu Zhang. Attacks meet interpretability: Attribute-steered detection of adversarial samples. In *NeurIPS*, 2018.
- [45] Chih-Kuan Yeh, Been Kim, Sercan O. Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On completeness-aware concept-based explanations in deep neural networks, 2019.
- [46] Sergey Zagoruyko. 92.45% on CIFAR-10 in torch. http://torch.ch/blog/2015/07/30/cifar.html. Accessed: 2019-09-17.
- [47] Vincent Zammit. On the readability of machine checkable formal proofs. PhD thesis, 1999.

# A Implementation details and reproducibility

In this section, we report relevant implementation details from Section 3.

## A.1 Extracting MACCs

As discussed in section 2.1.1, one could define MACCs as concepts shared between one or more classes. Since this quantity scales exponentially with the number of classes, in the paper, we restrict ourselves to MACCs that are shared by every pair of classes. Table 3 shows some example of MACCs for each of the datasets used in Section 3 (*i.e.*, CIFAR-10, CIFAR-100 and Fashion-MNIST). By restricting ourselves to MACCs shared by each pair of classes, we get  ${}^{N}C_{2}$  unique MACCs, and each class has N-1 unique MACCs. Appendix B provides some initial analysis on MACCs defined as shared concepts between each triplet of classes, thus leading to  ${}^{N}C_{3}$  concepts. However, we leave a detailed analysis of different combinations of concepts to future work.

Table 3: [MACCs details] Details and examples of MACCs extracted for each dataset. MACCs here correspond to concepts shared between each pair of classes.

	Total # of unique MACCs	# MACCs per class	Examples of MACCs
CIFAR-10	45	9	$c_{dog/cat}, c_{dog/airplane}, \ c_{dog/truck}, c_{dog/automobile}$
CIFAR-100	4095	99	$c_{baby/boy}, c_{baby/girl}, \ c_{baby/man}, c_{baby/woman}$
Fashion-MNIST	45	9	$c_{trouser/pullover}, c_{trouser/dress}, \\ c_{trouser/coat}, c_{trouser/sandal} \dots$

## A.2 Model Architectures

Our goal is to use architectures that are: (i) easily implementable and are widely used in the community, and, (ii) are able to achieve close to state-of-the-art accuracy on the respective datasets. To this end, we use the following architectures.

- CIFAR-10: The architecture by Zagoruyko [46] that achieves competitive accuracy [4].
- Fashion-MNIST: The architecture as described in [1], taken from the official Github repository of the dataset.<sup>6</sup>
- **CIFAR-100:** The architecture as described in [2].

Furthermore, for the sake of simplicity, we chose to train these models from scratch, and did not rely on transfer learning or pretrained feature extractors. We leave the detailed analysis of MACCs under these training paradigms to a separate future study.

#### A.3 Data Preprocessing

For both CIFAR10 and Fashion-MNIST we do mean-std. normalization. For CIFAR-10 and CIFAR-100 we use a mean and std of (0.5,0.5,0.5) while for Fashion-MNIST we use a mean and std of 0.1307 and 0.3081 respectively. These are commonly used values for the respective datasets. While all 3 datasets come with a pre-defined train-test split, we further split the train set into train and validation set with 10k samples in the validation set for each of the datasets. Table 4 shows the number of samples in train, val and test sets for each of the datasets.

#### A.4 Hyperparameters

The optimizer used throughout the experiments is SGD with a momentum of 0.9 and a learning rate of 0.01.

We select  $t_{\rm con}$  to be 1.0 for CIFAR-10 and Fashion-MNIST datasets, and 0.98 for CIFAR-100 dataset. This value is chosen to ensure that a reasonably high number of predictions are deemed explanation-conformant. A detailed trade-off between  $t_{\rm con}$  and classification accuracy is shown in Figure 4.

<sup>&</sup>lt;sup>6</sup>https://github.com/zalandoresearch/fashion-mnist

 Train
 Validation
 Test

 CIFAR-10
 40,000
 10,000
 10,000

 CIFAR-100
 40,000
 10,000
 10,000

 Fashion-MNIST
 50,000
 10,000
 10,000

Table 4: [Dataset splits] Training, validation and test set splits for each dataset.

#### A.5 Hardware

We use a machine with a NVIDIA Tesla p100-sxm2 16GB GPU, with 80 CPU cores and 512GB of RAM. However, all of our models and data are fairly common and can run on any standard machine without compromising the training time significantly. Depending on batch size, one can run our code in as little as 500MB of GPU space and by loading each batch from disk, RAM usage can be reduced to as little as 16GB.

# **B** MACC fine-tuning & comparison to calibration methods

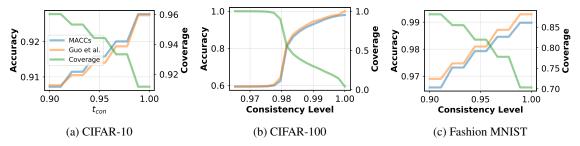


Figure 4: [Error Estimability] A higher value of  $t_{con}$  results in fewer samples being marked explanation-conformant, but the accuracy on these samples increases. Also shown is the accuracy of the probability calibration method of Guo et al. [24]. For the same level of coverge, both methods achieve very similar accuracy. For MACCs, coverage is the fraction of samples marked explanation-conformant. For Guo et al. [24], coverage is the fraction of samples with the calibrated prediction probability higher than the corresponding threshold (Section 3.4).

**Effect of**  $t_{\text{con}}$ . Figure 4 shows the trade-off between the accuracy on explanation-conformant samples and the amount of samples which are marked explanation-conformant (coverage). As described in Section 2.1.3, and (Section 3.4), this trade-off is achieved by varying  $t_{con}$ . The figure shows that as expected, increasing  $t_{con}$  results in fewer samples being marked explanation-conformant, however, the classification accuracy on explanation-conformant sections is higher.

Comparison with calibration method (Guo et al. [24]) Figure 4 also shows the trade-off between fraction of samples marked robust and accuracy on these samples achieved by using the probability calibration method of Guo et al. [24] (details in Section 3.4). The figure shows that MACCs are able to achieve a trade-off very similar to that of Guo et al. [24], highlighting the competitive calibration capability of MACC explanation-conformity check.

**Effect of number of concepts** Table 5 shows the comparison on CIFAR-10 between different sets of MACCs. With a higher number of MACCs, we see that a slightly lower fraction of samples are marked as explanation-conformant, however, the accuracy on this set is slightly higher. We leave a more in-depth analysis of different sets of MACCs to future work.

# C MACCs and error interpretability

We describe the human surveys conducted in Sections 3.1 and 3.3.

In the following, we describe the dataset used in the survey, the survey setup and metrics used and finally show the obtained results.

## C.1 Details of Human Experiments

**Dataset.** We are interested in examining links between MACC explanation conformity checks and difficulty that humans (who are often sources of ground truth for such tasks) would face in classifying the image. We therefore use the

Table 5: [Error Estimability on different sets of MACCs ] Performance of CIFAR-10 with different sets of MACCs shows a slight trend of fewer samples being marked explanation-conformant with a higher number of MACCs. The first number is accuracy and the second number in parentheses is coverage.

	Vanilla	Explanation-conf.	Non explanation-conf.
CIFAR-10 with			
$^{N}C_{2}=45~\mathrm{MACCs}$	0.89 (1.00)	0.93 (0.91)	0.48 (0.09)
$t_{con} = 1.0$			
CIFAR-10 with			
$^{N}C_{3}=120\mathrm{MACCs}$	0.89 (1.00)	0.94 (0.89)	0.51 (0.11)
$t_{con} = 1.0$		· · ·	

Table 6: [Survey dataset statistics] Number of images from each partition used in the survey and share of the partitions in the original CIFAR-10 test set.

Image Group	Number of images used in the survey	Fraction in the dataset
Correct and Explanation Conformant (C+)	50	84.6%
Incorrect and Explanation Conformant (C-)	50	6.6%
Less Than 6 MACCs (NC<6)	72	0.7%
Between 6 and 12 MACCs (NC-6-12)	50	6.9%
Greater Than 12 MACCs (NC>12)	122	1.2%

setup described in Appendix A to obtain MACC- and class-predictions for the images in the CIFAR-10 test set. Next, we divide the dataset according to explanation-conformity into five types of images:

- 1. C+: correctly predicted by DNN and explanation-conformant
- 2. C-: incorrect and explanation conformant
- 3. NC<6: non- explanation-conformant with fewer than 6 predicted MACCs
- 4. NC-6-12: non- explanation-conformant with between 6 and 12 predicted MACCs
- 5. NC>12: non-conformant with more than 12 predicted MACCs

From each partition we randomly sample 50 images, except for NC<6 and NC>12 which consist of very few images (72 and 122 images, respectively), for which we therefore include all images. This gives us a total of 344 images used in the survey. The composition of the set of images used in the survey as well as the fraction of images in the original CIFAR-10 test set that belong to each of the groups is shown in Table 6.

Table 7: [Human accuracy for different viewing duration timeout values] As the viewing time is reduced from 5 to 1 seconds in our prior studies we see a decrease in human classification accuracy. This reduction, however, is small at only about 1.6% from 5 to 1 seconds, so we choose an image viewing timeout of 1 second in our main survey.

Viewing duration timeout	Human Accuracy
5s	93.4%
3s	92.2%
2s	91.8%
<b>1</b> s	91.8%

**Survey setup.** We run a survey where the task is to classify the given image into one of the 10 classes in the data. We recruit 150 workers from Amazon Mechanical Turk (AMT) for the survey. To keep the workload for each worker

<sup>&</sup>lt;sup>7</sup>A complete random subsampling without regard to the categories would result in near-zero images from sparsely populated categories.

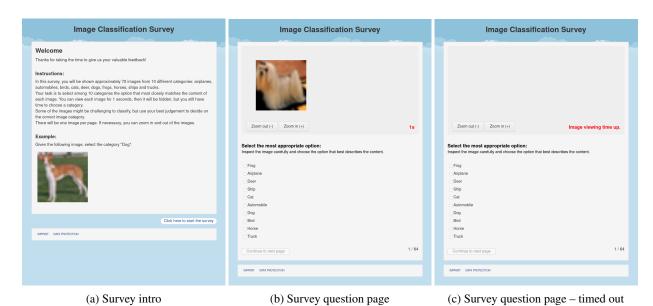


Figure 5: [Error Explainability Survey, Screenshots] Figure 5a shows the survey's intro page containing instructions for the participants. Figure 5b shows the web app interface used by participants to view and classify images and Figure 5c shows how the interface changes after the 1s viewing time has expired.

manageable, we create multiple random partitions of the set of 344 images into five subsets, four with size 70 and one with size 64, such that each subset contains 10 randomly chosen images from each of the C+, C- and NC-6-12 categories, between 11 and 15 images from NC<6 and between 22 and 25 images from NC>12 (the goal was to show each category to a similar number of workers, and also to ensure that each worker sees a proportional fraction of images from each category).

After an introduction page containing instructions for the image classification task, we show one of the randomly chosen subsets of images to each worker in a web app which we created for the study.

By default, we enlarge each image from its original 32x32 pixel size to 256x256 pixels, however, the workers could zoom out the image to the original size or further zoom into the image. Screenshots of the web app interface seen by the AMT workers are shown in Figure 5. As a result of assigning workers to randomly chosen subsets of images, we obtain a varying number of responses per image, ranging from 20 to 40, with an average of 30 responses per image.

To understand which images are *easy* for humans to classify, we limit the time that each worker can view each of the images.<sup>8</sup> However, after an image is hidden, there is unlimited time to choose among the 10 categories.

We assume that correct and explanation-conformant images which make up 84.6% of the dataset are most representative of easy to classify images and therefore use them to calibrate the appropriate amount of viewing time necessary for correctly classifying an image. To validate the viewing duration value, we conducted a four-part prior study on a randomly chosen set of 50 correct and explanation-conformant images not used in the main study, which we showed for 5, 3, 2 and 1 seconds to 25 AMT workers each and asked them to choose the correct class, while keeping all other parameters the same as in the main study. We find only small differences in performance for the four viewing durations, as shown in Table 7. Consequently, use a viewing duration value of 1 second in our main study.

**Workers and compensation.** For each survey, we recruit 25 workers from AMT. We only admit workers (i) from the US, who (ii) have the master qualification, (iii) have at least 95% previous HIT approval rate, and (iv) at least 100 approved assignments on AMT. The compensation was set to 8 USD per participant. The average completion time of the survey was less than 25 minutes.

<sup>&</sup>lt;sup>8</sup>This choice is based on the well-known phenomenon in psychology and neuroscience called the speed-accuracy tradeoff. We hypothesize that images which are easy to classify for humans do not require a lot of time to make a decision and therefore setting a limit on viewing time enables us to distinguish between images that are easy and hard to classify for humans. See: Heitz RP. The speed-accuracy tradeoff: history, physiology, methodology, and behavior. Front Neurosci. 2014;8:150. Published 2014 Jun 11. doi:10.3389/fnins.2014.00150

Table 8: [Human performance on the different groups of images used in the survey] Humans perform better – *i.e.* have higher accuracy and lower confusion – on images that are correct and explanation conformant, which are also the images that are easy for the machine to classify. Additionally, as the number of detected MACCs increases, humans are less confused, even though human accuracy is similar. This correlates with an increase in machine accuracy.

	<b>Human Accuracy</b>	Human Confusion (Shannon Entropy)	Machine Accuracy
Correct and Explanation Conformant (C+)	91.8%	0.212	100.0%
Incorrect and Explanation Conformant (C-)	83.7%	0.407	0.0%
Less Than 6 MACCs (NC<6)	83.1%	0.418	31.9%
Between 6 and 12 MACCs (NC-6-12)	82.9%	0.4	42.0%
Greater Than 12 MACCs (NC>12)	84.7%	0.325	47.5%

#### C.2 Measures

**Human Accuracy.** We measure the average human accuracy, *i.e.* the fraction of workers who choose the correct category for each image, averaged over all images.

**Human confusion via Shannon Entropy.** We use Shannon Entropy as a measure of confusion among the humans in predicting the class of the image.

We randomly subsample the votes for each image to the minimum number of responses of 20 for any image to make the entropy computationally comparable. We repeat this process 10 times with different random seeds to obtain robust results. If our entropy-based confusion measure is high for an image it means, that votes for the different classes are distributed relatively uniformly and therefore there is high confusion among humans.

**Human Agreement.** To measure how much humans agree on predicting the class of a given image, we compute what share of all votes is allocated to the class that receives the majority of all votes. In the CIFAR-10 data with 10 classes, completely random votes would results in a human agreement of 10%, whereas all votes for the same class would results in a human agreement of 10%.

**Reweighting** The set of images used in the survey is made up of five types of images, C+, C-, NC<6, NC-6-12 and NC>12, which are sampled out of proportion from the original dataset (in order to ensure ample participation from each category). Therefore, when we report results for images from multiple of these groups such as in Table 9, we compute the metrics for each set of images separately and then perform a weighted aggregation according to how prevalent each of the groups is in the original CIFAR-10 test set is. The prevalence of each group of images can be seen in Table 6.

#### **C.3** Human Experiment Results

First, we measure human accuracy and confusion on each of the five subsets of images used in the survey and report the results in Table 8. For comparison we report machine accuracy as well. We observe that humans are considerably more accurate and show less confusion on correct and explanation conformant images, indicating that there seems to be an overlap between the images deemed easy by humans and on which the machine performs well. Another finding is that for non-conformant images, although exhibiting similar error rates, humans seems to be less confused as the number of detected MACCs increases. This correlates with increased machine accuracy as well.

In Table 9 we further analyze human classification performance by grouping images based on explanation conformity and also on whether they are correctly classified by the DNN. Additionally we show performance on the entire set of 344 images. We observe that humans make more accurate classifications and show less confusion on explanation conformant and correctly classified images, compared to non-conformant and incorrectly classified images respectively. For explanation conformity, however, the accuracy and confusion differences are larger then for correctness, meaning

Table 9: [Human performance on different divisions of the dataset] Humans exhibit higher classification accuracy and lower confusion on explanation conformant and on images correctly classified by the machine, compared to explanation non-conformant and incorrectly classified images, respectively. However, for explanation conformance this gap is larger for both metrics than for machine correctness. The human and machine performance numbers are extrapolated to the entire CIFAR-10 test set by reweighting the performance achieved on each of the image partitions according to their prevalence in the test set (see Table 6).

	Reweighed Human Accuracy	Reweighed Human Confusion (Shannon Entropy)	Reweighed Machine Accuracy
All Images	90.5%	0.24	88.3%
Explanation Conformant	91.3%	0.215	92.8%
Explanation Non-Conformant	83.2%	0.387	42.0%
<b>Machine Correct</b>	90.7%	0.233	100.0%
<b>Machine Incorrect</b>	85.0%	0.388	0.0%

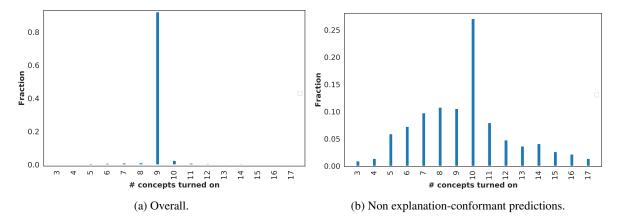


Figure 6: [# MACCs turned on, CIFAR-10] Figure 6a shows the number of MACCs detected for the entire dataset. The peak at 9 MACCs aligns with our observation that most of the predictions (about 91%, Table 1) are explanation-conformant and hence have exactly 9 MACCs detected. Figure 6b shows the distribution for just non explanation-conformant predictions. Notice that there are a few samples where the model detected very few (<6) and very large (>12) number of MACCs.

that explanation conformity seems to be a better indicator of difficulty in image classification for humans compared to correctness of machine predictions.

Finally, to further examine the link between the number of detected concepts and human image classification performance, we test whether as the number of detected MACCs increases, human confusion in classification reduces. Figure 6 shows that for some images the machine classifier detects very few concepts. We divide the set of explanation non-conformant images into ones with less than 6 and 6 or more detected MACCs. For both sets we measure human agreement as described in section C.2. Figure 7 shows that images with low human agreement primarily have less than 6 MACCs detected and images with higher agreement mostly have 6 or more MACCs detected. This indicates that human agreement and MACC detection are indeed connected. Figure 8 shows the images with the lowest agreement among humans.

# D Additional experiments on Error Vulnerability

We use the implementation of FGSM, DeepFool, CarliniWagner and PGD provided by the widely used (and publicly available) repository Foolbox (v1.8.0) [39].

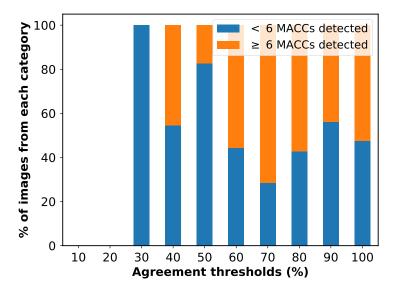


Figure 7: [Human agreement in image classification for explanation non-conformant images] Each agreement threshold value corresponds to a bin containing explanation non-conformant images where human agreement is in the range of (10%, 20%], (20%, 30%], etc. where the values on the x-axis mark the upper boundary of the bins. For each bin we show its composition in terms of images with less than 6 and 6 or more detected MACCs by the machine. Since there are 172 images with 6 or more MACCs but only 72 images with less than 6 MACCs detected, we normalize the contributions of each of the groups such that if a bin would contain less than 6 and 6 or more detected MACC images in a ratio of 72:172, we would show a 50% share each.

The figure shows, that images with low human agreement mostly are ones with less than 6 detected MACCs. This is an indication, that human agreement and perception and MACC detection are correlated.



Figure 8: [Images with least human agreement] Explanation non-conformant images that exhibit the least agreement among humans. The images are ordered with agreement increasing from left to right, top to bottom.

# D.1 Performing an explanation-conformant adversarial perturbation and comparison with Guo et al. [24]

To perform an explanation-conformant adversarial attack, we build on PGD by Madry et al. [35]. Throughout the description of the algorithm, we use the following notation:  $\hat{y}$  is the predicted class label, y is the ground truth, r is the vector of ground truth MACCs,  $\hat{r}$  is the vector of predicted MACCs, and  $\hat{r}_{const}$  are the MACCs which pass the explanation-conformity check (corresponding to the predicted class) i.e.,  $(\hat{y}, \hat{r}_{const})$  is an explanation-conformant prediction.  $\rho$  is the budget given to the attacker i.e., if the perturbed image is x' and the original image is x then  $||x-x'||_2 \le \rho$ .

In the paper we use "post-hoc" training as described in section 2.1.2. This is a 2 step process, where we first train the model to predict class labels, and in the second step we add a head to predict MACCs, keeping all the previous layers fixed from the first training step. We denote the neural network until the penultimate layer by  $f_{rep}$ , the final layer for predicting class labels by  $f_{class}$  and the final layer for predicting MACCs by  $f_{concept}$ . Under our framework, for a given input x, class label prediction is given by  $f_{class}(f_{rep}(x))$  and MACC prediction is given by  $f_{concept}(f_{rep}(x))$  (notice that class and MACCs predictions both share common hidden layer representations via  $f_{rep}$ ). The losses for class and MACCs are given by  $f_{class}(f_{class}(f_{rep}(x)))$  (we use cross-entropy loss in our implementation) and  $f_{concept}(f_{concept}(f_{rep}(x)))$  (we use binary cross entropy loss in our implementation). Given these notations, Algorithm

1 proposes a method to generate explanation-conformant adversarial samples. Note that the algorithm is same as that of Madry et al. [35], however the only change required to generate explanation-conformant adversarial samples is the change in the loss function.

## **Algorithm 1:** Algorithm for generating consistent adversarial samples

```
Data: Ground-truth class label y, predicted class label \hat{y}, ground truth MACCs r, predicted MACCs \hat{r}, input images x,
   f_{class}, f_{concept}, f_{rep}, \rho Result: Perturbed image x'
 x^0 = x, k = 0;
 2 while y == \hat{y} || not explanation-conformant(\hat{r}, y) do
         L := L_{class}(f_{class}(f_{rep}(x)), y) - L_{concept}(f_{class}(f_{rep}(x)), \hat{r}) \triangleright \hat{r} represents the consistent concept
            labels and we want to minimize this loss, hence the negative sign
         x^{k+1} := x^k + \eta * \nabla_x L;
        \begin{array}{l} \text{if } ||x^{k+1}=x^0||_2>\rho \text{ then} \\ ||x^{k+1}:=\text{projection of } x^{k+1} \text{ onto the 2-norm ball represented by } ||x^{k+1}-x^0||_2=\rho; \end{array}
 5
 6
 7
         \hat{y} := f_{class}(f_{rep}(x^{k+1});
 8
         \hat{r} := f_{concept}(f_{rep}(x));
         k := k + 1;
10
11 end
```

Table 10: [Error Vulnerability] Mean L2 distances for attacks on class labels only (*class-only*), on MACC conformity check in addition to the class label (*explanation-conf*.) and on the method of Guo et al. [24] (Guo et al.—see Section 3.4). Explanation-conformant attacks require much larger perturbation than the class-only attack and the attack on Guo et al. [24].

	Class-only	Explanation-conf.	Guo et al.
CIFAR-10	$0.31 \pm 0.20$	$\textbf{5.31} \pm \textbf{5.62}$	$0.35 \pm 0.20$
Fashion-MNIST	$0.26 \pm 0.14$	$\textbf{3.16} \pm \textbf{2.85}$	$0.49 \pm 0.22$

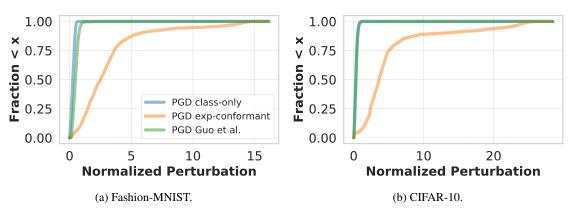
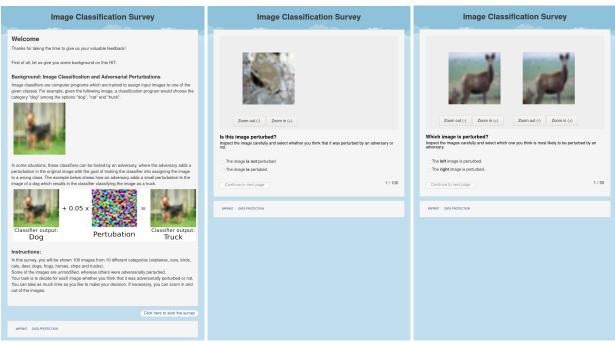


Figure 9: [Error Vulnerability] Shows the entire distribution of (L2) perturbations in Table 10. In order to attack the explanation-conformity check, the required perturbations are much larger than that of class only and Guo et al..

**Results.** As discussed in Section 3.2, Figure 9 and Table 10 show that the amount of perturbation required for explanation conformant attacks is much higher than the traditional attack on just class labels. Moreover, attacking the calibrated probability-based method of Guo et al. [24] also requires order of magnitude lesser perturbation (than MACC explanation-conformant attack), thus showing that adding MACCs lead to higher robustness against adversarial attacks. Figures 11 & 12 show some explanation-conformant adversarial samples generated using Algorithm 1 as compared to the traditional class-only PGD attack. The explanation-conformant adversarial samples seem to show a much more visible perturbation. We explore the human perceptability of these perturbations in Appendix D.2.



(a) Survey intro

(b) Survey question page: Individual image (c) Survey question page: Image-pair sursurvey vey

Figure 10: [Adversarial Survey, Screenshots] Figure 10a shows the survey's landing page that introduces the workers to the task. Figures 10b and 10c show the question pages for the *individual image survey* and *image-pair survey*, respectively.

#### D.2 Human perceptibility of explanation-conformant adversarial perturbations

Here we describe the human survey conducted in Section 3.2. The goal of the survey was to test whether humans find any difference between the *class-only* and *explanation-conformant* perturbations conducted in Section 3.2. We use Amazon Mechanical Turk (AMT) to recruit participants for the surveys.

**Survey data.** We created two groups of perturbed images (i) class-only and (ii) explanation-conformant. Each group consists of 50 randomly selected images.

**Survey Setup.** For each of the two sets of attacked images we run two surveys.

- 1. In first *individual image survey*, we show one image at a time, that is either adversarially perturbed or an unmodified original. The survey consists of 100 images in total: 50 perturbed images and the other 50 being their original counterpart.
- 2. In the second survey, we show *image-pair survey*. One image is unmodified and the other one is its adversarially attacked counterpart. We then ask each participant for each of the 50 image pairs which of the two images they think was perturbed.

For both surveys, workers had unlimited time to make a decision. By default, we enlarge each image from its original 32x32 pixel size to 256x256 pixels, however, the workers could zoom out the image to the original size or further zoom into the image. Screenshots of the web app interfaced shown to participants are shown in Figure 10.

**Workers and compensation.** For each survey, we recruit 25 workers from AMT. We only admit workers (i) from the US, who (ii) have the master qualification, (iii) have at least 95% previous HIT approval rate, and (iv) at least 100 approved assignments on AMT. The compensation was set to 8 USD per participant.

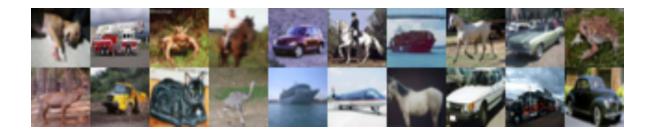
**Results.** The average completion time of the surveys was around 32 minutes.

Table 11 shows the human accuracy for both survey types and image types. The results show that while the human detection accuracy is close to random for class-only perturbations, it is much higher for explanation-conformant images. This result shows that the explanation-conformant perturbations are indeed so large that humans can detect them in

Table 11: [Human performance in adversarial perturbation detection surveys.] The table shows human accuracy in detecting whether an image has been adversarially perturbed or not. We report results for two different perturbation types (class-only and explanation-conformant) and two survey types (individual image survey and image pair survey) as defined in Appendix D.2. For both survey types, when annotating class-only perturbations, the human detection accuracy is close to random. On the other hand, for explanation-conformant perturbations, the human accuracy is much higher. Furthermore, humans perform even better on image-pair survey (for both image types).

Image / Survey Type	Human Accuracy
Class-only / Individual image survey	49.8%
Class-only / Image-pair survey	58.8%
Explanation-conf. / Individual image survey	85.0%
Explanation-conf. / Image-pair survey	95.7%

most cases. Second, we see that images that were attacked in a consistent manner are significantly easier for humans to detect compared to traditionally attacked ones where detection rates are close to random.



(a) Original Images



(b) PGD class-only



(c) PGD explanation-conformant

Figure 11: [CIFAR-10] Visual quality of samples generated by explanation-conformant PGD attack compared to the traditional PGD attack (changes class label only).



(a) Original Images



(b) PGD class-only



(c) PGD explanation-conformant

Figure 12: [Fashion-MNIST] Visual quality of samples generated by explanation-conformant PGD attack compared to the traditional PGD attack (changes class label only).