Rethinking Out-of-Distribution Detection

William Yang*, Byron Zhang*, Olga Russakovsky
Princeton University
{williamyang, zishuoz, olgarus}@princeton.edu

Abstract

Out-of-Distribution (OOD) detection is the task of identifying inputs on which a given model should not be trusted to make a prediction. OOD detection is a well-studied problem, commonly formulated for image classifiers as detecting test images from classes that did not appear in the model's training data ("semantic distribution shift"). However, this formulation is limited, as it ignores test images which may come from the familiar classes but may look different from the training examples ("covariate distribution shift"). Incorporating covariate shift into OOD detection poses a challenge because detecting all *such examples may undervalue the model's robustness and ability to generalize beyond its training distribution.*

In this work we propose a new formulation of OOD detection, approaching the problem with respect to the learned model distribution rather than the training data distribution. This formulation seamnlessly incorporates both semantic and covariate shift, and synergizes the perspective of OOD detection with model generalization. Our empirical analysis reveals a number of interesting findings, the most striking being that the simplest OOD detection baseline, Maximum Softmax Probability (MSP), appears to outperform all prior state-of-the-art OOD detection methods under the new formulation. This suggests that the time is ripe to rethink the task of OOD detection, and to adopt the more generalized benchmark which encompasses a broader range of real-world situation.

1. Introduction

Modern computer vision systems have achieved remarkable performance in closed-world classification settings. However, the performance of such systems decreases in real-world settings where images encounter different types of distribution shifts, attributed to camera calibrations, sensory corruptions, introduction of novel object classes, etc. Addressing such distribution shifts plays a crucial role in the safety and practicality of deep learning models in real-world applications such as autonomous vehicles.

(a) In-distribution (b) Covariate Shift (c) Semantic Shift

Figure 1. **Different types of distribution shifts.** Assume a model is trained to predict ImageNet [28] dogs (example on the left). The current out-of-distribution (OOD) detection benchmark specifies that images of cats (example on the right) should be considered OOD, but does not consider the case when encountering images of dogs that do not belong to the distribution of ImageNet dogs (example in the middle). Our benchmark introduce a way of incorporating such cases for evaluation of OOD detectors.

Out-of-distribution (OOD) detection aims to address such distribution shifts by rejecting problematic examples to promote the safety and reliability of deep learning models in real-world applications. Common literature in OOD detection separates distribution shifts into two categories: semantic and covariate [14, 32, 37] (Figure 1). Under semantic shift, images are drawn from classes that do not belong to the in-distribution (ID) dataset (*i.e.* training dataset). An image classifier will make erroneous predictions on images with semantic shift because the concepts of interest are not present. The appropriate action to handle images with semantic shift is thus to reject these images or to flag them for human intervention. Under covariate shift, images are drawn from the classes in the training dataset but exhibit differences in terms of style, contrast, brightness, domain, etc. An image classifier will not necessarily make erroneous predictions because it may have some robustness towards these shifts. Therefore, the appropriate action for handling images with covariate shifts becomes unclear.

Research on detection of covariate shifts remains controversial due to the conflicting objective with OOD generalization [37]. Mainstream research in OOD detection evades the conflict between detection and generalization on covariate shifts by not considering covariate shifts at all [16, 17, 20–22, 30, 35]. Some works attempt to incorporate

covariate shift only from the perspective of the data—by characterizing the all covariate shifted data as only ID [38] or only OOD [14]. However, not every covariate shifted sample leads to an erroneous prediction. As a result, under this characterization, a model can reject too many correctly classified examples, which hurts generalization, or too few incorrectly classified examples, which undermines safety.

We argue that OOD detection should consider distribution shifts with respect to the distribution of the trained classification model instead of the training data, since the notion of safety and generalization is highly dependent on the model behavior. In other words, we reformulate the notion of *in-distribution* as data that the model generalizes to, *i.e*. correctly classifies. The new formulation avoids false rejection of correct predictions, resolving the emerging issues from covariate shifts and unifies task of OOD detection with OOD generalization. Additionally, the task of OOD detection under the new formulation better aligns with the goal of safety, because the optimal decision under safety critical-settings is to abstain from making incorrect predictions.

We use our reformulation to create a general benchmark for OOD detection that is appropriate for any kind of distribution shifts. For covariate shifts, our benchmark better aligns with the ideal behavior by only considering misclassified covariate shifted examples as OOD. For semantic shifts, since a model can never be correct on unknown concepts, our benchmark considers every semantic shifted example as OOD which is consistent with the existing benchmarks. Our benchmark also excludes misclassified examples from the training distribution. If a models fails to learn certain data from the training distribution, these data would be OOD with respect to the fitted model, and hence, should be rejected by OOD detector. We argue that this behavior is more aligned with the goal of building robust computer vision systems in the real world and is akin to detection of outliers in training data.

To empirically motivate our proposed benchmark and better understand current OOD detection algorithms, we performed extensive analysis using covariate shifted dataset ImageNet-C [10], existing semantic shifted dataset OpenImage-O [35], and our newly proposed semantic shifted dataset ImageNet-OOD. We demonstrate several shortcomings of the existing benchmarks and the mitigation of such issues under our proposed benchmark. First, we perform a simple sanity check using randomized models to show that existing formulation breaks the connection between in-distribution and training distribution. Next, we observe that current OOD detection algorithms not only detect covariate shifts but sometimes to a **greater extent** than semantic shifts. Finally, we find that under the new benchmark, the simplest baseline (MSP [11]) outperforms many recent OOD detection algorithms across both semantic shifted and covariate shifted datasets, sometimes to the extent of 5%-7% during evaluation on Area Under Receiver Characteristics Curve (AUROC).

2. Related Work

Datasets for OOD detection. Previous benchmarks in OOD detection distinguishes between "ID" and "OOD" on a dataset level. Common evaluation settings in early benchmarks designate CIFAR-10 [19] as ID and one of CIFAR-100 [19], SVHN [25], MNIST [5], TinyImageNet [28] as OOD. Huang *et al*. [17] pointed out that small datasets do not reflect real-world settings due to low resolution and low number of classes, and suggested to expand OOD detection to larger semantic spaces, designating ImageNet-1K [28] as ID, and one of iNaturalist [13], Texture [4], SUN [36], Places [42] as OOD. Other benchmarks that set ImageNet-1K include ImageNet-O [12] and OpenImage-O [35], which include OOD datasets constructed from subsets of ImageNet-21K and OpenImages [18] through manual verification. Species [9] designates classes from iNaturalist that are disjoint from ImageNet-21K as OOD. Although newer benchmarks come with datasets with higher resolution and more ID classes, we show that some OOD samples in these benchmarks may not truly be OOD due to semantic hierarchies within the label space.

Covariate shift in OOD detection. Covariate shift was first considered in OOD detection by Generalized ODIN [14], where OOD detection performance was evaluated considering all covariate shifted examples as out-of-distribution. Tian *et al*. [32] designed a score function that disentangles detection of semantic vs. covariate shifts. Later, Yang *et al*. [37] pointed out that models should ideally generalize, instead of detect, in the case of covariate shifts, because generalization is the primary goal of machine learning, thus they defined all covariate shifted examples as in-distribution and proposed several benchmarks that include covariate shifted data. Our work share similar motivation as [38], but we do not characterize all covariate shifted examples as in-distribution.

OOD Detection Methods. OOD detection methods can be considered as scoring functions that assign higher scores to ID examples and lower scores to OOD examples. Logit-based methods (Maximum Softmax Probabiliy (MSP) [11], Maximum Logits [9], and Energy [22]) derive scoring functions from classification logits with the intuition that OOD examples tend to have lower activations across all classes. Feature-based methods operate on the penultimate layer of the model by comparing to the features from the training data using distance metrics such as the Mahalanobis distance [20], k-th nearest neighbor distance [31], and distance to a fitted principal space [35]. Gradient-based methods incorporate information in the gradient space to generate more gap between ID and OOD data, either implicitly

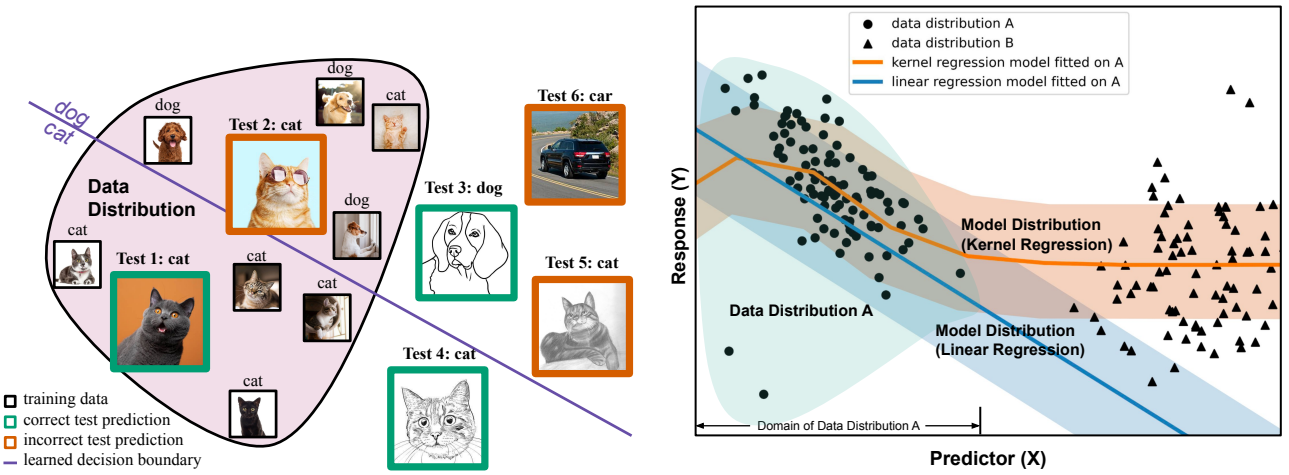

Figure 2. **Failure cases of data distribution shift.** *Left.* Consider a binary linear classifier that learned to distinguish dogs and cats. The training data distribution is highlighted in pink, and test examples that come from the *same* distribution may be classified correctly (test 1) or incorrectly (test 2). All other test examples (test 3-6) do *not* come from the same distribution and thus would be considered OOD under the data distribution shift formulation. However, since the model is able to generalize to test examples 3 and 4, our model-centric OOD formulation would consider them ID. *Right.* Now consider a more complex scenario where given features x, a regression model is tasked with predicting the real-valued output y. The kernel regression model successfully generalizes outside its training distribution (data distribution A) to data distribution B, which as we argue should thus be considered ID with respect to that model. This, however, is not the case for linear regression, so distribution B would remain OOD. Our work grounds the OOD definition in the model distribution, making it more practically relevant and also (as we argue in the experiments) more empirically informative.

through adversarial perturbation [21], or explicitly through backpropogation of the gradient norm [16]. Previous works have also explored modification of the training process or model architectures, through decomposing the linear prediction head to incorporate prior knowledge [14] and using group labels to simplify decision boundary between ID vs. OOD examples [17]. We perform evaluation on our new benchmark using representative logit-based and feature-based detectors, which require minimal computational cost [37].

3. General Benchmark for OOD Detection

OOD detection attempts to prevent failures in computer vision systems by sensing when distribution shifts occur. However, current OOD detection benchmarks only consider semantic shifts. Introduction of covariate shifts proves to be problematic due to the conflicting nature between the task of OOD detection and OOD generalization. We propose a new notion of distribution shift with respect to the model distribution instead of the data distribution. With new formulation, we build a general benchmark that unifies OOD generalization and OOD detection for a more realistic and comprehensive analysis of OOD detection algorithms.

3.1. Preliminaries

For the task of image classification , a dataset D_{in} sampled from distribution $P_{in}(x, y)$ is used to train some classifier C. In real-world deployments, distribution shift occurs when classifier C receives data from distribution $P_{out}(x, y)$ where $P_{in}(x, y) \neq P_{out}(x, y)$ [23]. An out-of-distribution detector is a scoring function s that maps an image x to a real number $\mathbb{R}$ such that for some threshold τ, we arrive at a detection rule f

$$f(x) = \begin{cases} \text{in-distribution} & \text{if } s(x) \geq \tau \\ \text{out-of-distribution} & \text{if } s(x) < \tau \end{cases} \quad (1)$$

3.2. Data Distribution Shift

Definition. Data distribution shifts fall into two categories: covariate shift and semantic shift. Covariate shift occurs when the marginal distribution with respect to the image differs: $P_{in}(x) \neq P_{out}(x)$ [37]. Semantic shift occurs when the semantic labels in joint distribution $P_{in}(x, y)$ and $P_{out}(x, y)$ are disjoint [14]. Rigorously, given sets of semantic labels Y_{in} from the training distribution and semantic labels Y_{out} from a semantic shifted distribution, the two sets have the following property: $Y_{in} \cap Y_{out} = \emptyset$.

To evaluate OOD detection under data distribution shift, images from the ID test dataset is considered to be drawn from P_{in} and images from a different dataset is considered to be drawn from P_{out}. The notion of data distribution shift is acceptable when only semantic shift is present. However, with the introduction of covariate shift, detecting data distribution shift becomes ill-motivated.

Problems with the Formulation. OOD detection in the presence of covariate shift indicates a clear conflict with model generalization under the formulation of data distribution shift. It is unreasonable to detect and reject every image from a covariate shifted distribution as OOD. With modern deep neural networks, the trained model is expected to generalize to unseen distributions. The field of model robustness and out-of-distribution generalization specifically tackles this problem and aims to improve model performance under the presence of covariate shifts.

3.3. Model Distribution Shift

Introduction of covariate shift requires rethinking the notion of in-distribution vs. out-of-distribution. We propose a new notion of distribution shift with respect to the fitted model distribution instead of the data distribution. Formulating distribution shift with respect to the model synergizes the task of generalization with detection.

Definition. The goal of the image classification task is to use data pairs (x, y) sampled from some distribution P_{in} to build a classifier $P_\theta(y|x)$ with the objective of matching $P_\theta(y|x) = P_{in}(y|x)$. Model distribution shift only occurs when the training objective is violated. For example, the classifier may encounter images from other data distributions P_{out} where $P_{out}(x) \neq P_{in}(x)$ but $P_{out}(y|x) = P_\theta(y|x)$. Such images are considered OOD under the data distribution shift paradigm but not under the model distribution shift paradigm (Figure 2). Concretely, we define the concept of in-distribution with respect to the model as images that the model generalizes to. In other words, an image is in-distribution if the model correctly predicts its class label. As a result, goal of OOD detection under the formulation of model distribution shift becomes the detection of incorrect predictions, which resolves issues introduced by covariate shift. Additionally, the goal of model robustness and OOD generalization can now be viewed as the reduction of model distribution shift which synergizes with the task of OOD detection.

Connection with Previous Works. Our formulation of model distribution shift is similar Nalisnick *et al*. [24] but with respect to a discriminator model $P_\theta(y|x)$ instead of a generative model $P_\theta(x)$. Their definition leverages the idea of the typical set of the model distribution, which has the problem of being ambiguous in the context of OOD detection [40]. Our definition is not based on typical sets, and therefore, does not suffer the same issue. Additionally, their work presents the notion of model distribution to build a novel OOD detector for data distribution shift instead of resolving the issues present in the current formulation with data distribution detection.

Connection with Detecting Failures. OOD detection under our proposed formulation of model distribution shift is similar to existing works on the task of failure detection in computer vision systems [41]. Unlike previous works however, our work expand the idea to different types of data distribution shifts (*i.e*. covariate and semantic), which is more reflective of real-world classification settings.

Connection with Outliers Detection. A datapoint is considered an outlier if it has vastly different characteristics and behavior compared to other datapoints [34]. The characteristics of images can be challenging to describe because images exist in very high dimensional spaces. Instead, an image classifier can be considered as a composition of two functions $f(g(x))$ where f is a logistic regression classifier and g is a feature extractor. The notion of outlier can be defined as datapoints with different characteristics in the feature space of $g(x)$ and behavior under the classifier f. Therefore, OOD detection under the introduced notion of model distribution shift encompasses outlier detection.

Connection with Data Distribution Shift. Our formalization of model distribution shift has the same definition as data distribution shift in the context of semantic shifts. Under semantic shift, the semantic label of the image is disjoint from the set of labels that the classifier is trained on. As a result, given a semantic shifted distribution P_{out}, $P_\theta(y|x) \neq P_{out}(y|x)$ by definition, indicating that model distribution shift is equivalent to data distribution shift under semantic shift.

3.4. Evaluation

With the introduction of model distribution shift, we propose a general benchmark with metrics that considers two facets of the OOD detection problem: detection performance under model distribution shift and generalization ability in the presence of the detection mechanism.

Detection Metric. The existing OOD detection evaluation metric can be used for model distribution shift detection. Area under the Receiver Characteristic Curve (AUROC) is a threshold free way of estimating detection performance by measuring $P(s(x_{in}) > s(x_{out}))$ for some OOD detection scoring function s, in-distribution image x_{in} and out-of-distribution image x_{out}.

Robustness Metric. We include a robustness metric that is formulation-independent and provide a better connection between the OOD detector's performance and the model's task accuracy. Accuracy vs. Declaration Rate (ADR) is an existing threshold free metric that quantifies the trade-off between model generalization and refusal of predicting uncertain images [41]. We extend this metric to the context of out-of-distribution detection by analyzing the trade-off between the strength of detection vs. model generalization on the remaining images. Concretely, ADR calculates the area under the curve between proportion of images an OOD

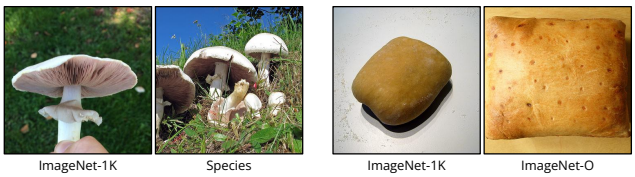

Figure 3. **OOD class as hyponym of ID class in previous datasets.** *Left*: "Agaricus" from ImageNet-1K vs. "Agaricus Xanthodermus" from Species Dataset [9]. *Right*: "Dough" from ImageNet-1K and "Pastry Dough" from ImageNet-O [12].

detector declare as OOD vs. the classification accuracy of the model on the remaining undetected images. Overall, this metric accounts for model generalization where higher value indicate better classification performance.

Sample Reweighting. The issue of data imbalance emerges under the model distribution shift formulation because positive and negative instances are retrieved from multiple datasets. Data imbalance was not an issue in the formulation of data distribution shift formulation because each dataset is either entirely filled with positive instances or negative instances. Therefore, false positive rate (FPR) and true positive rate (TPR) are unaffected by the number of images in each of the datasets. With the proposed benchmark, positive and negative examples can come from different data sources. As a result, the proposed metrics will favor larger datasets. To counteract this undesired property, each datapoint from some dataset D is weighted by $\frac{1}{|D|}$ during evaluation. This will provide balance the effect each dataset contributes to the performance on the metrics.

4. ImageNet-OOD

Under both data and model distribution shift formulations, a semantic shifted OOD example cannot contain a class that belongs to the training dataset. With data distribution shift, the previous statement follows by definition. With model distribution shift, the assumption that all OOD examples result in incorrect predictions may become invalid if an OOD dataset includes images from a training class. As OOD detection steers toward larger, fine-grained semantic spaces [9, 12, 17, 35], evaluation datasets may mislabel examples from training classes as OOD by overlooking the hierarchical structure of semantic labels. Figure 3 shows problematic examples in current datasets

To facilitate with evaluation under both data and model distribution shifts, we introduce ImageNet-OOD, a diverse OOD detection dataset leveraging properties of the WordNet Semantic Tree to account for hierarchical relations of semantic labels, encompassing images from 1000 classes. Setting ImageNet-1K synsets as ID classes, and we construct ImageNet-OOD to be semantic shifted from ImageNet-1K by carefully selecting a subset of 1000

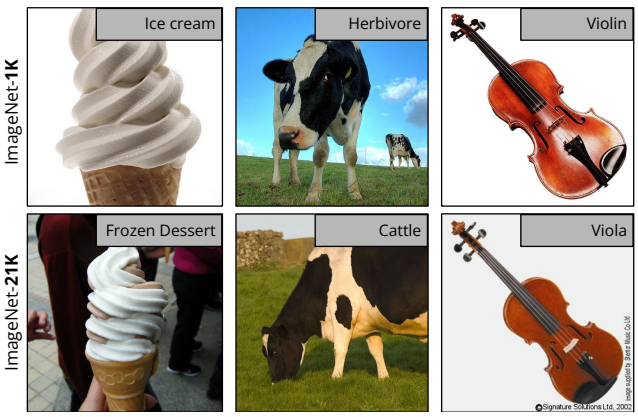

Figure 4. **Considerations when designing ImageNet-OOD.** Assume ImageNet-1K classes as the training distribution, we attempt to use ImageNet-21K classes to construct ImageNet-OOD. The following pairs of classes can lead to images that do not actually consider semantic shift: *Left*. An image of "frozen desert" from [12] can come from the ID class "ice cream", so it is not truly OOD. *Middle*. "Herbivore" should not be considered OOD, even though it is not labeled as a semantic parent of "cattle" in WordNet. *Right*. "Viola" and "violin" are visually indistinguishable if not present in the same image; "viola" cannot be considered OOD because images of viola may already exist in "violin" due to labeling errors.

synsets from ImageNet-21K with three considerations.

First, we filter out ImageNet-1K classes, their hypernyms (semantic ancestors) and hyponyms (semantic descendants). Classes such as "Pastry Dough" should not be considered as OOD when images of "Dough" belongs to the training distribution (Figure 4 *Left*), as predicting an image of "Pastry Dough" as "Dough" is not incorrect.

Second, we avoid classes of natural beings because the construction of WordNet hierarchy within the subtree of "organism" is unreliable. In particular, natural beings are separated by both technical biological levels (family, genus, species, etc.), as well as non-technical categories (by colors, patterns, etc.), resulting in an inconsistency in hierarchical relations. For example, the class "herbivore" is still considered as OOD because the ID class "cattle" is not specified as a semantic descendant of herbivore, even though it should be (Figure 4 *Middle*).

Third, from the remaining synsets, we manually select 1000 classes that are visually unambiguous to humans. The ImageNet authors reported errors persistent in ambigous classes that human annotators could not accurately differentiate, such as violin and viola (Figure 4 *Right*) [28]. As a result, certain ambiguous classes with OOD labels may contain images leaked from ID classes, and manual verification is required to circumvent this issue.

To standardize the evaluation settings of ImageNet-OOD for future work, we select 50,000 images by randomly sam-

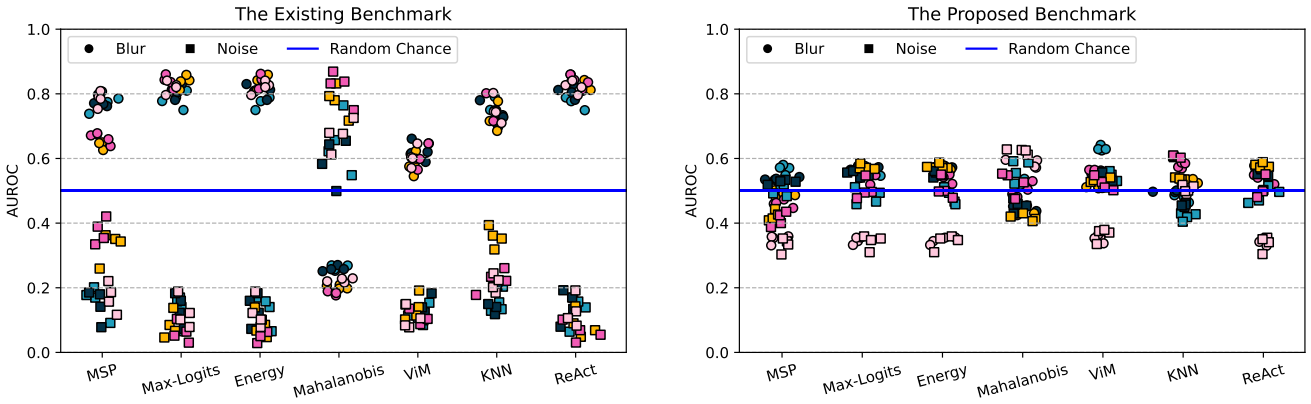

Figure 5. **Performance of OOD detection under random models.** AUROC performance of 5 ResNet-50 [8] models with **random**, untrained parameters on subsets of ImageNet-C [10] under the existing benchmark vs our proposed benchmark. Colors indicate the specific random model and the markers indicate the corruption type. Results reveal that OOD detection methods confidently identify corruptions as being "out-of-distribution" for these models – despite the models not being trained on *any* data! Thus, the existing OOD benchmark appears to be ill-defined. In contrast, under our model-centric benchmark all OOD detectors perform near random chance for these untrained models, as expected.

pling 50 images from each of the the 1000 manually selected classes. This also maintains the same number of images as the ID dataset (ImageNet-1K validation). Note that large-scale evaluation can be performed using any images from the 1000 ImageNet-OOD classes and their hypernyms. For transparency and reproducibility, we used the processed version of ImageNet-21K [27] and its associated semantic tree to construct ImageNet-OOD.

5. Experiments

To empirically motivate the proposed benchmark under the formulation of model distribution shift, we utilize the covariate shifted dataset ImageNet-C [10] for evaluation along with the semantic shifted datasets OpenImage-O [35] and the newly proposed ImageNet-OOD. Additionally, we evaluate OOD detectors using random models to question the connection between the concept of in-distribution with the training distribution. Finally, we analyze the differences in outcomes between the existing benchmark vs. the proposed benchmark.

5.1. Old Benchmark Fails Sanity Check

The concept of in-distribution is only defined with regard to the data used to train the model. Therefore, for a randomly initialized model, the concept of in-distribution does not exist. Given that every data should be considered OOD for a random model, a well-behaved OOD detection algorithm should perform around random chance [11]. We design a simple sanity check around this idea and found that OOD detection under the current formulation of data distribution shift fails this sanity check and is highly biased toward certain corruptions in ImageNet-C.

Experiment Setup. Five random models are used for evaluating out-of-distribution detection on blurring and noise corruptions with the ImageNet-C dataset. Performance of seven commonly used OOD detection methods is evaluated on both the existing benchmark (under data distribution shift) and our proposed benchmark (under model distribution shift). This experiment is performed on Resnet-50 [8] initialized with the Pytorch's default Kaiming normal initialization [7]. We use only the most severe corruptions in ImageNet-C.

Bias under the Old Formulation. Figure 5 reveals that existing OOD formulation and evaluation exhibits strong bias in most of the OOD detectors as they detect blurry images as out-of-distribution and noisy images as in-distribution. This effect is most evident in logits-based out-of-distribution detectors: Maximum Softmax Probability (MSP) [11], Maximum Logits [9], and Energy [22]. Most interestingly, although Energy and Maximum Logits are well established out-of-distribution detectors that outperform the MSP baseline, they are more biased toward detecting certain corruptions. Methods that condition on the training data do not remedy this issue. ViM [35] and KNN [31], albeit better than Energy and Maximum Logits, is still significantly biased. Mahalanobis distance [20] seems to only reverse the direction of the bias, not the magnitude of the bias. Limiting over-activation due to noisy images through rectifying the activations (ReAct) [30] before applying Energy has no effect on this phenomenon. The bias toward detecting certain corrupted images illustrates that detecting data distribution shift does not properly align the idea of in-distribution with the training distribution.

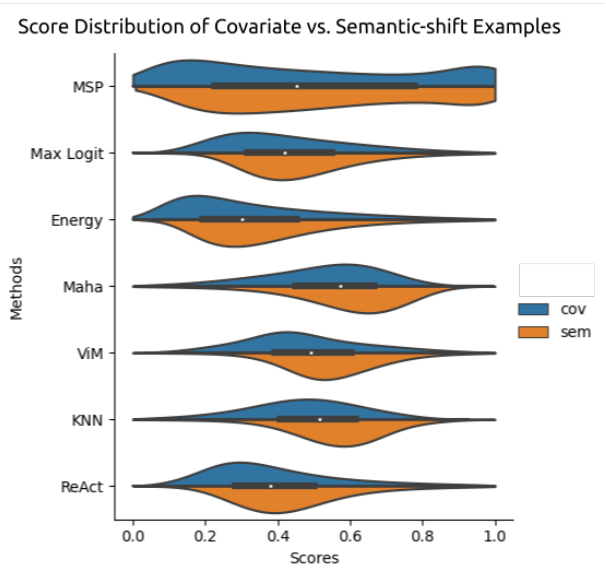

Figure 6. **Distribution of Semantic vs. Covariate OOD detection scores.** Density of OOD detection scores from semantic shifted dataset ImageNet-OOD, OpenImage-O, Places, iNaturalist, and SUN and covariate shifted data ImageNet-C. Result reveals that covariate-shift scores tend to fall lower than semantic-shift scores for most OOD detectors.

Robustness under the New Benchmark. Figure 5 reveals that under proposed benchmark, the sanity check passes. In particular, the expected OOD detection performance is much closer to random chance. Additionally, the detection performance is grouped by the specific model rather than the corruption type. Therefore, the deviation from random chance is more likely attributed to noise in the random initialization than the OOD detection method itself. Overall, the sanity check suggests that our proposed benchmark is more reflective of the model behavior and properly aligns with the notion of in- and out-of-distribution.

Implications. Previous works demonstrated the power of random models as a feature extractor for tasks such as classification, in-painting, denoising, super resolution, or interpretability. [1, 2, 29, 33]. While having this power is acceptable for other tasks, it is very problematic in the context of out-of-distribution detection as it challenges the fundamental concept of in- and out-of-distribution. The idea of in-distribution becomes ill-defined on random models as the model has not encountered or learned from data from any distributions.

5.2. OOD Detectors Detect Covariate Shift

We observe that OOD detection scores are heavily influenced by covariate shifts, which illustrates that current OOD detectors do not ignore covariate shifts, and it is important to consider covariate shifted datasets for evaluation.

To conduct the experiment, we first obtain detection scores of covariate and semantic shifted examples on a pretrained ResNet-50 model for seven OOD detection algorithms. Examples with semantic shifts come from ImageNet-OOD (see Section 4), OpenImage-O [18], Places [42], iNaturalist [13] and SUN [36]. Examples with covariate shifts come from ImageNet-C [10]. We perform Kernel Density Estimation (KDE) for detection scores of semantic shifted examples and covariate shifted examples separately. The scores are normalized for each detector to the range [0, 1] for better visualization.

It is evident from the distribution of detection scores in Figure 6 that OOD detection methods output similar, and sometimes, even lower scores for covariate shifts than semantic shifts. In particular, we see this effect more pronounced in recently proposed methods. For example, the density of ViM [35] when compared to MSP tend to be more shifted towards zero for covariate shifted examples than semantic shifted examples. The analysis reveals that it is impossible for current OOD detectors to detect only semantic shift. In consequence, the generalization and detection trade-off is always present. Therefore, it is important to incorporate covariate shifts in evaluation of OOD detectors, motivating the needed for our proposed general benchmark.

5.3. New Benchmark Reaches New Conclusions

To analyze the difference of OOD detection under the old benchmark detecting data distribution shift with the newly proposed benchmark detecting model distribution shift, we evaluated a pretrained ResNet-50 using seven popular OOD detection methods. Results shown in table 6 reveals that the new benchmark arrives as a different conclusion for the best OOD detector. In particular, we see that across all three dataset scenarios, the baseline MSP outperforms every other method on AUROC_M for model distribution shift detection and ADR for calibrating model robustness. This result reorders the state-of-the-art under dataset distribution shift detection where ViM, Maximum Logits, or ReAct outperformed other methods under AUROC_D. This revelation challenges many established intuitions in the OOD detection community such as superior performance of Energy and Maximum Logits with respect to MSP.

5.4. Analysis Between the Benchmarks

To analyze the factors that led to differences between the old benchmark and our proposed benchmark, we compare the ranking of OOD detection scores on OOD examples with correctly predicted (Correct ID) and incorrectly predicted (Incorrect ID) examples from the training distribution. Our experiment reveals that recent OOD detection methods (1) mainly improve performance under the old benchmark through better separation between incorrect ID examples and OOD examples, and (2) show worse separa-

Dataset	ImageNet-C [10]			ImageNet-OOD			OpenImage-O [35]		
Metric	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR
MSP [11]	77.3	**88.6**	**78.2**	79.2	**86.8**	**65.9**	84.0	**89.9**	**74.1**
Max-logit [9]	81.3	84.7	75.3	**80.4**	84.6	63.8	87.4	89.4	73.4
Energy [22]	81.4	83.6	74.7	79.9	83.6	63.3	87.1	88.6	72.8
Mahalanobis [20]	68.7	64.0	66.8	57.8	59.1	48.2	71.6	69.8	58.1
ViM [35]	**84.0**	82.3	73.7	78.6	82.3	63.3	88.8	89.1	73.8
KNN [31]	77.3	67.0	66.7	61.6	58.1	44.1	78.5	71.8	56.2
ReAct [30]	80.1	81.4	74.0	75.6	79.4	61.2	**89.2**	88.6	73.3

Table 1. **Performance differences under different metrics.** OOD detection performance under the existing benchmark with the data distribution shift formulation is labeled as AUROC$_D$, OOD detection performance under our proposed benchmark with the model distribution shift formulation is labeled as AUROC$_M$, and robustness under OOD detection using Accuracy vs. Declaration Rate labeled as ADR. The data indicates MSP outperforms every other method under our proposed benchmark.

Method	Correct ID	Incorrect ID	OOD
MSP [11]	33.5	16.4	12.5
Max-logit [9]	32.9	20.1	11.2
Energy [22]	32.7	20.6	11.3
ViM [35]	32.7	21.6	10.7
ReAct [30]	32.6	22.2	10.5

Table 2. **Average percentile of OOD detection scores across semantic shift, incorrect data-ID, and correct data-ID.** The average percentile of OOD scores across recently proposed OOD detectors reveals that detectors mostly improve relative to MSP in the existing benchmark by pushing the percentile of incorrect ID examples up. The finding reveal the cause for the discrepancy between existing benchmark and the proposed benchmark.

bility between correct ID and incorrect ID.

For the experiment, we compare percentile of the detection scores of four OOD detectors (those that perform better than MSP on AUROC$_D$) on ImageNet-1K vs. OpenImage-O using the same ResNet-50 model from Table 2. The three groups are described as follows: ImageNet-1K images that the model correctly classifies, ImageNet-1K images that the model incorrectly classifies, and OpenImage-O (OOD). Table 2 shows a common trend across all four detectors: incorrect ID examples showed an increase in average percentile, while correct ID examples showed a decrease in average percentile. The trend has two implications.

First, the improvements on AUROC$_D$ from these four detectors are mainly attributed to the better separation between the Incorrect ID and OOD examples. Under the existing benchmark, all ID examples, including incorrect ones, are considered as positive examples when computing AUROC$_D$. Since AUROC$_D$ can be interpreted as $P(s(x_{in}) > s(x_{out}))$, ranking any ID example higher than OOD examples induces a positive change in AUROC$_D$, and lower will penalize the performance. Although the four methods rank correct examples lower than MSP on average, the higher ranking of incorrect examples is significant enough to improve the detection performance on AUROC$_D$. Under our new benchmark, ranking incorrect ID examples higher than OOD examples is not penalized.

Second, as a side effect of increasing the ranking of incorrect ID examples, all four methods showed lower separability between incorrect vs. correct ID examples. Concretely, the gap in percentile of correct ID and incorrect ID scores is smaller in all four methods than MSP. Therefore, it is more likely for these methods to rank an incorrect example higher than a correct example. Under the existing benchmark, ranking an incorrect example higher than a correct example is not penalized. However, under our new benchmark, this behavior is penalized, and therefore, the four methods other than MSP in Table 2 to lower performance in AUROC$_M$.

6. Conclusion and Discussion

We provide analysis of OOD detection algorithms across different distribution shifts and identified the key limitations in the existing benchmark for OOD detection with the introduction of covariate shift. We introduce a general benchmark that reformulates OOD detection with regard to model distribution to overcome limitations of the existing formulation and provide a more comprehensive evaluation of existing OOD detection algorithms. We also designed a new diverse OOD detection dataset *ImageNet-OOD* that accounts for the hierarchical nature of semantic concepts and potential annotation errors from human labelers.

Broader Impact. In safety critical applications, OOD detection becomes an important task to prevent catastrophic failures in computer vision systems. We seek to better understand if the current OOD detection algorithms accomplish the goal of failure prevention. We believe this will expand the capabilities of visual recognition systems and enable their deployments in high stake scenarios where catastrophic failures are not an option.

7. Acknowledgements

This material is based upon work supported by the National Science Foundation under Grants 2112562 and 2145198. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors' and do not necessarily reflect the views of the National Science Foundation. We would also like to thank Ye Zhu, Kaiqu Liang, Sunnie Kim, Xindi Wu, Zeyu Wang, Tinghao Xie, Tong Wu, John Yang, and Christiane Fellbaum for helpful feedback.

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *Advances in neural information processing systems*, 31, 2018. 7

[2] Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016. 7

[3] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1563–1572, 2016. 14

[4] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[5] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012. 2

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 13

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 6

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 6

[9] Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joe Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. *ICML*, 2022. 2, 5, 6, 8, 13, 14

[10] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 2, 6, 7, 8, 13, 14, 15

[11] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of International Conference on Learning Representations*, 2017. 2, 6, 8, 13, 14

[12] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021. 2, 5

[13] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alexander Shepard, Hartwig Adam, Pietro Perona, and Serge J. Belongie. The inaturalist species classification and detection dataset. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8769–8778, 2018. 2, 7

[14] Y. C. Hsu, Y. Shen, H. Jin, and Z. Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10948–10957, 2020. 1, 2, 3, 11

[15] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 13

[16] Rui Huang, Andrew Geng, and Yixuan Li. On the importance of gradients for detecting distributional shifts in the wild. In *Advances in Neural Information Processing Systems*, 2021. 1, 3, 11, 14

[17] Rui Huang and Yixuan Li. Mos: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 1, 2, 3, 5

[18] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2017. 2, 7

[19] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 2

[20] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 1, 2, 6, 8, 13

[21] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018. 1, 3, 11, 14

[22] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. Energy-based out-of-distribution detection. *Advances in Neural Information Processing Systems*, 2020. 1, 2, 6, 8, 13, 14

[23] Jose G Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. *Pattern recognition*, 45(1):521–530, 2012. 3

[24] Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, and Balaji Lakshminarayanan. Detecting out-of-distribution inputs to deep generative models using typicality. *In NeurIPS Workshop on Bayesian Deep Learning*, 2019. 4

[25] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. 2

[26] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017. 11

[27] Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 6

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 1, 2, 5, 12

[29] Andrew M Saxe, Pang Wei Koh, Zhenghao Chen, Maneesh Bhand, Bipin Suresh, and Andrew Y Ng. On random weights and unsupervised feature learning. In *Icml*, 2011. 7

[30] Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. 1, 6, 8, 13, 14

[31] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. *arXiv preprint arXiv:2204.06507*, 2022. 2, 6, 8, 13

[32] Junjiao Tian, Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Exploring covariate and concept shift for out-of-distribution detection. In *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021. 1, 2

[33] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454, 2018. 7

[34] Hongzhi Wang, Mohamed Jaward Bah, and Mohamed Hammad. Progress in outlier detection techniques: A survey. *Ieee Access*, 7:107964–108000, 2019. 4

[35] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 5, 6, 7, 8, 13, 14

[36] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, 2010. 2, 7

[37] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *arXiv preprint arXiv:2110.11334*, 2021. 1, 2, 3

[38] Jingkang Yang, Kaiyang Zhou, and Ziwei Liu. Full-spectrum out-of-distribution detection, 2022. 2

[39] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In Edwin R. Hancock Richard C. Wilson and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12. BMVA Press, September 2016. 13

[40] Lily Zhang, Mark Goldstein, and Rajesh Ranganath. Understanding failures in out-of-distribution detection with deep generative models. In *International Conference on Machine Learning*, pages 12427–12436. PMLR, 2021. 4

[41] Peng Zhang, Jiuling Wang, Ali Farhadi, Martial Hebert, and Devi Parikh. Predicting failures of vision systems. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3566–3573, 2014. 4

[42] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 2, 7

In this appendix, we will provide more detailed analysis on claims made in the main paper.

- **Section A**: We extend Section 5.3 of the main paper and examine performance of OOD detectors under our benchmark using different model architectures.
- **Section B**: We supplement Section 5.3 of the main paper and evaluate our benchmark using gradient-based OOD detection methods.
- **Section C**: We analyze the behavior of methods developed for Open-Set Recognition, a closely related task to OOD detection, under our new benchmark.
- **Section D**: We supplement Section 5.4 of the main paper and provide more analysis on the separability between correct and incorrect in-distribution examples.
- **Section E**: We supplement Section 5.1 of the main paper and perform the sanity check to examine covariate shift bias from different model architectures.
- **Section F**: We supplement Section 4 of the main paper and elaborate on the process of manually selecting classes to construct ImageNet-OOD.
- **Section G**: We supplement Section 4 of the main paper and provide class names and synset IDs for the 1000 ImageNet-OOD classes.

A. The New Benchmark is Less Sensitive to Model Choice

We expand the evaluation of seven out-of-distribution (OOD) detection methods to three additional model architectures: DenseNet-121, Wide ResNet-50, and ViT-L-32. All of the models trained on ImageNet are provided by the Torchvision library [26]. Results for the three models are shown in Table 3, Table 4, and Table 5. When analyzing the performance using AUROC$_D$, the conclusion on the best-performing OOD detector is not clear. Results from Table 3 show that Energy is the best detector under ImageNet-C with DenseNet, but results from Table 4 indicate that ViM is the best detector under ImageNet-C with Wide Resnet-50. Such a pattern indicates that under the old benchmark, not only is the detection performance dependent on the dataset, but it is also dependent on the model architecture. In contrast, under the new benchmark with model distribution shift, MSP reveals to be the best detector across all the models except for ViT on ImageNet-OOD and OpenImage-O. Therefore, the results suggest that the new benchmark under model distribution shift provides a measurement of the performance of OOD detection algorithms that is less sensitive to model architecture choice.

B. Gradient-Based OOD Detection Algorithms

We expand our analysis on ResNet-50 to gradient-based OOD detection algorithms, which are more computationally expensive than feature- or logits-based OOD detection algorithms. We tested two popular gradient-based methods: ODIN [21] and GradNorm [16]. Our findings in Table 6 reveal that the gradient-based methods follow similar conclusions to feature and logits based methods.

C. Connection with Open-Set Recognition

Open-Set Recognition (OSR) is a similar task to semantic OOD detection, where the goal is to identify unseen classes. However, unlike OOD detection, OSR algorithms are evaluated by class splits from a single dataset, which minimizes spurious covariate shifts during evaluation. We perform the same analysis on a popular OSR algorithm that does not require any retraining: OpenMax [14]. Results from Table 6 uncovers that despite the differences in evaluation, the same concern still exists. OpenMax does detect covariate shift under ImageNet-C, given that it achieves 81.4% AUROC$_D$ on ImageNet-C, which is a misalignment with the OSR task. Additionally, the performance gains when compared to MSP disappears and reverse when evaluated under the new benchmark. These findings suggest that OSR algorithms follow similar behaviors to OOD detection algorithms.

D. Additional Experiments on Correct and Incorrect Examples

Besides the analysis on ranking, we also use AUROC on OpenImage-O to measure the separability between three groups of examples: correctly classified in-distribution (ID) examples, incorrectly classified ID examples, and semantic shifted OOD examples.

Specifically, since AUROC is a measure of separability (*i.e.* the probability of scoring a positive example higher than a negative example), we examine the AUROC among these three groups of examples. Using the probabilistic interpretation of AUROC, we can decompose AUROC between ID vs. OOD by correct ID and incorrect ID using the law of total probability:

$$\begin{aligned}&(f(x_{in}) > f(x_{out})) = \\ &(f(x_{in}) > f(x_{out}) | C(x_{in}) = 1)(C(x_{in}) = 1) + \\ &(f(x_{in}) > f(x_{out}) | C(x_{in}) = 0)(C(x_{in}) = 0)\end{aligned} \quad (2)$$

where x_{in}, x_{out} refer to semantic ID and OOD examples, respectively, f is the OOD scoring function, and C is an indicator function with $C(x) = 1$ if x is predicted correctly, and 0 otherwise.

Table 7 reports $(f(x_{in}) > f(x_{out}) | C(x_{in}) = 1)$ (*i.e.* column Correct ID (+) vs. OOD (-)) and $(f(x_{in}) >$

$f(x_{out})|C(x_{in}) = 0)$ (*i.e.* column Incorrect ID (+) vs. OOD (-)). The results reveal increased separability between incorrect ID vs. OOD from MSP to ViM and ReAct, decreased separability between correctly classified ID vs. incorrectly classified ID, and roughly the same separability between correctly classified vs. OOD. Since $(C(x_{in}) = 1)$ in Equation 2 is the classification accuracy on the ID dataset, we see that for 71.6% of increase in AUROC_D from ViM can be attributed to an increase in performance of detecting Incorrect ID vs. OOD predictions. This pattern supports the claim that many advanced OOD detection methods improve under the old benchmark by detecting more incorrectly classified examples as ID rather than a balanced improvement across the ID set.

Additionally, we also use AUROC to approximate the probability of scoring correct ID higher than incorrect ID examples, reported in the Correct ID (+) vs. Incorrect ID (-) column in Table 7. We found that advanced OOD detection methods have lower separability between correctly classified ID vs. incorrectly classified ID. Having lower separability between correct vs. incorrect hurts the performance of the model from a safety perspective, as more problematic examples can pass through the OOD detection filter.

Furthermore, we repeated the analysis with ImageNet-OOD and found similar conclusions as shown in Table 8.

E. Sanity Check Fails on Other Architectures

We expand our analysis using the sanity check to random DenseNet-121, Wide ResNet-50, and ViT-L-30 architectures. Results from Figure 7 and Figure 8 reveal that the same issue with sanity check occurs on DenseNet, except ViM, and Wide Resnet-50, suggesting that this issue applies to convolution based architecture. Interestingly, Figure 9 reveals that ViT, a transformer-based architecture, does not suffer such issues with logits-based OOD methods. Still, ViM, KNN, and Mahalanobis still suffer the same issue under ViT. Overall, sanity checks on DenseNet, Wide ResNet, and ViT pass on our new benchmark, suggesting that our benchmark is reflective of the model's behavior even on different architectures.

F. Manual Selection of ImageNet-OOD Classes

In this section, we provide details on the manual selection process of ImageNet-OOD. Because images from ID classes may leak into OOD classes if human labelers are unable to disambiguate two classes, such as "violin" and "viola" [28], we manually selected 1000 classes from ImageNet-21K to construct ImageNet-OOD. However, even after excluding hypernyms, hyponyms, and the "organism" subtree, there are still 5074 remaining candidate ImageNet-21K classes. It is simply infeasible to check all 5074 classes against all 1000 ImageNet-1K classes, which would require $5074 \times 1000 = 5,074,000$ manual comparisons. Therefore, we need to employ another mechanism that can pass through the 5074 classes in linear time.

To pick out the 1000 classes, We first gathered the sister classes for each ImageNet-1K class. A sister class c_s^i is defined as a class that shares a direct parent with an ImageNet-1K class c^i. For example, the sister classes for the ImageNet-1K class "microwave" has sister classes "food_processor", 'ice_maker", "hot_plate", "coffee_maker", and "oven", because these classes all have the same direct parent "kitchen_appliance." Considering only sister classes allowed us to further reduce the search space down to 2874 candidate classes.

Once we had obtained the sister classes, we examined the visual and semantic ambiguity between each sister class and its corresponding ImageNet-1K class through example images. Unambiguous classes are added to the final list of ImageNet-OOD classes. Since the ambiguity between classes was considered during the curation of ImageNet-1K classes, we assume that there exists minimal ambiguity between classes under different subtrees that contains an ImageNet-1K class. This assumption allowed us to only examine the relationship of sister classes with their corresponding ImageNet-1K class instead of with all 1000 ImageNet-1K classes. In the end, we only needed to compare 13,831 pairs of classes.

Dataset	ImageNet-C [10]			ImageNet-OOD			OpenImage-O [35]		
Metric	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR
MSP [11]	74.9	**87.9**	**78.4**	78.1	**86.3**	**64.7**	83.1	**89.4**	**72.8**
Max-logit [9]	78.1	83.9	75.5	**79.1**	84.0	62.7	**87.2**	89.3	72.5
Energy [22]	**78.2**	82.5	74.8	78.3	82.7	62.0	87.1	88.4	71.8
Mahalanobis [20]	70.0	64.8	67.6	63.3	64.0	50.1	77.6	75.0	60.6
ViM [35]	75.2	73.7	71.1	69.1	70.7	55.4	78.8	78.0	65.4
KNN [31]	67.7	61.3	64.7	59.9	56.9	43.4	78.1	71.2	55.6
ReAct [30]	74.9	70.2	69.1	58.9	61.0	47.7	81.5	78.1	61.1

Table 3. **OOD detection performance under different metrics for DenseNet-121 [15].** OOD detection performance under the existing benchmark with the data distribution shift formulation is labeled as AUROC$_D$, OOD detection performance under our proposed benchmark with the model distribution shift formulation is labeled as AUROC$_M$, and robustness under OOD detection using Accuracy vs. Declaration Rate labeled as ADR.

Dataset	ImageNet-C [10]			ImageNet-OOD			OpenImage-O [35]		
Metric	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR
MSP [11]	76.0	**88.6**	**81.0**	79.9	**86.8**	**66.7**	86.2	**91.0**	**76.2**
Max-logit [9]	77.5	83.7	78.1	**80.0**	83.4	63.6	88.6	89.6	74.5
Energy [22]	77.4	82.5	77.6	79.4	82.4	63.1	88.2	88.8	73.9
Mahalanobis [20]	71.1	69.7	72.6	63.1	65.2	54.6	75.6	74.8	63.8
ViM [35]	**81.5**	82.2	76.9	79.5	82.6	64.6	**90.3**	89.9	75.6
KNN [31]	64.0	53.3	62.9	49.1	43.0	35.8	69.3	59.9	48.4
ReAct [30]	67.6	61.6	68.5	47.6	48.4	40.4	71.3	67.6	52.3

Table 4. **OOD detection performance under different metrics for Wide ResNet-50 [39].** OOD detection performance under the existing benchmark with the data distribution shift formulation is labeled as AUROC$_D$, OOD detection performance under our proposed benchmark with the model distribution shift formulation is labeled as AUROC$_M$, and robustness under OOD detection using Accuracy vs. Declaration Rate labeled as ADR.

Dataset	ImageNet-C [10]			ImageNet-OOD			OpenImage-O [35]		
Metric	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR
MSP [11]	70.5	**85.3**	**84.9**	73.6	82.1	59.5	81.0	87.4	68.7
Max-logit [9]	71.7	79.8	81.5	68.1	74.8	51.9	77.2	81.9	60.3
Energy [22]	71.8	72.4	78.0	62.4	67.0	46.7	71.7	74.5	53.9
Mahalanobis [20]	70.5	82.8	83.9	**79.3**	**85.0**	**65.3**	**89.6**	**91.4**	**75.8**
ViM [35]	68.2	72.5	79.1	69.1	72.2	54.9	81.7	81.8	65.3
KNN [31]	59.5	49.3	67.0	49.7	45.2	38.0	60.1	53.5	45.6
ReAct [30]	**73.6**	80.4	82.4	74.7	81.0	60.2	81.2	85.9	69.3

Table 5. **Performance differences under different metrics for ViT [6].** OOD detection performance under the existing benchmark with the data distribution shift formulation is labeled as AUROC$_D$, OOD detection performance under our proposed benchmark with the model distribution shift formulation is labeled as AUROC$_M$, and robustness under OOD detection using Accuracy vs. Declaration Rate labeled as ADR.

Dataset	ImageNet-C [10]			ImageNet-OOD			OpenImage-O [35]		
Metric	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR	AUROC$_D$	AUROC$_M$	ADR
MSP [11]	77.3	**88.6**	**78.2**	79.2	**86.8**	**65.9**	84.0	**89.9**	**74.1**
ODIN$_\epsilon$ [21]	76.3	83.1	75.4	80.1	83.5	63.0	86.5	87.9	71.5
ODIN [21]	81.3	84.7	75.2	**80.4**	84.6	63.8	87.4	89.3	73.4
GradNorm [16]	78.8	73.3	69.8	73.8	72.2	54.3	80.4	77.2	59.1
OpenMax [3]	**81.4**	83.3	74.2	80.2	83.4	62.3	**87.4**	88.5	71.8

Table 6. **OOD detection performance under different metrics for ResNet-50 under Gradient-based OOD and OpenMax.** OOD detection performance under the existing benchmark with the data distribution shift formulation is labeled as AUROC$_D$, OOD detection performance under our proposed benchmark with the model distribution shift formulation is labeled as AUROC$_M$, and robustness under OOD detection using Accuracy vs. Declaration Rate labeled as ADR. ODIN$_\epsilon$ uses $T = 1000$ and $\epsilon = 0.0014$. ODIN uses $T = 1000$ with no adversarial perturbations. GradNorm uses $T = 1$.

	Correct ID (+) vs. OOD (-)	Incorrect ID (+) vs. OOD (-)	Correct ID (+) vs. Incorrect ID (-)
MSP [11]	90.7	62.4	86.4
Max-Logit [9]	92.2	72.0	77.7
Energy [22]	91.5	72.8	76.2
ViM [35]	92.5	76.9	74.6
ReAct [30]	92.1	79.9	73.8

Table 7. **Detailed breakdown of AUROC$_D$ performance with respect to OpenImageO.** Breakdown reveals that the improvement observed in ReAct is due to better separation between incorrect ID predictions and OOD predictions, shown by the large margin of increase in AUROC between Incorrect ID (+) vs. OOD (-). Performance between correct in-distribution predictions and out-of-distribution predictions are similar. Additionally, when evaluating incorrect ID predictions vs. correct ID predictions, separability decreases for other algorithms compared to MSP.

	Correct ID (+) vs. OOD (-)	Incorrect ID (+) vs. OOD (-)	Correct ID (+) vs. Incorrect ID (-)
MSP [11]	86.9	54.7	86.4
Max-Logit [9]	86.3	61.8	77.7
Energy [22]	85.4	62.4	76.2

Table 8. **Detailed breakdown of AUROC$_D$ performance with respect to ImageNet-OOD.** Breakdown reveals that the improvement observed in both Max-Logit and Energy is attributed to better separation between incorrect ID predictions and OOD predictions, shown by the large margin of increase in AUROC between Incorrect ID (+) vs. OOD (-). Performance between correct in-distribution predictions and out-of-distribution predictions are similar. Additionally, when evaluating incorrect ID predictions vs. correct ID predictions, separability decreases for MaxLogit and Energy compared to MSP.

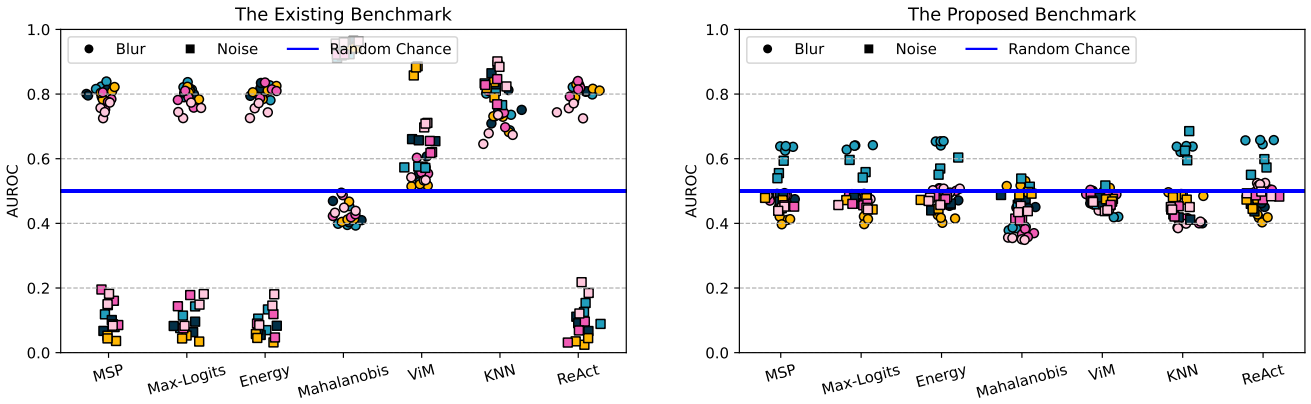

Figure 7. **Performance of OOD detection under random models.** AUROC performance of 5 DenseNet-121 models with **random**, untrained parameters on subsets of ImageNet-C [10] under the existing benchmark vs our proposed benchmark. Colors indicate the specific random model and the markers indicate the corruption type. Results reveals mostly the same conclusion as the ResNet-50 sanity check.

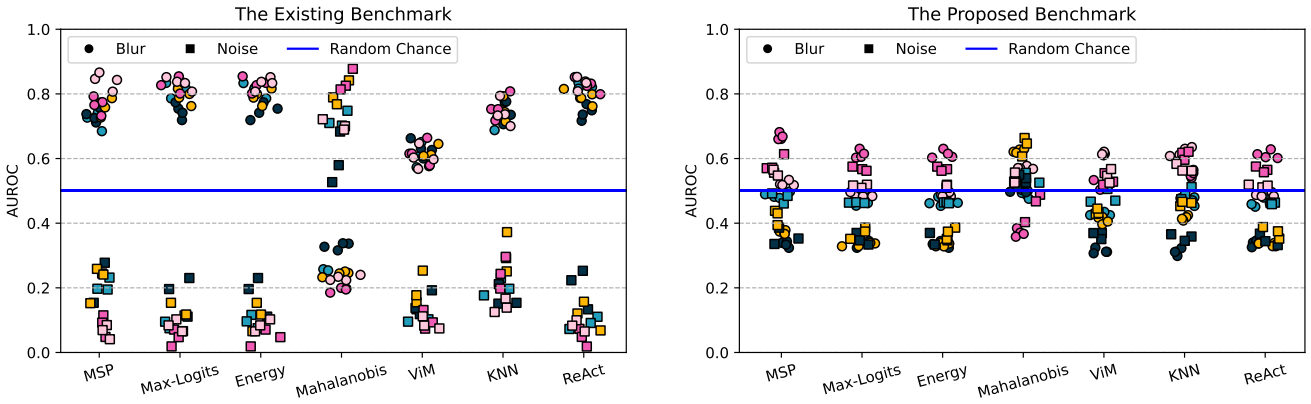

Figure 8. **OOD detection performance under random models.** AUROC performance of 5 Wide Resnet-50 models with **random**, untrained parameters on subsets of ImageNet-C [10] under the existing benchmark vs our proposed benchmark. Colors indicate the specific random model and the markers indicate the corruption type. Results reveals the same conclusion as the ResNet-50 sanity check.

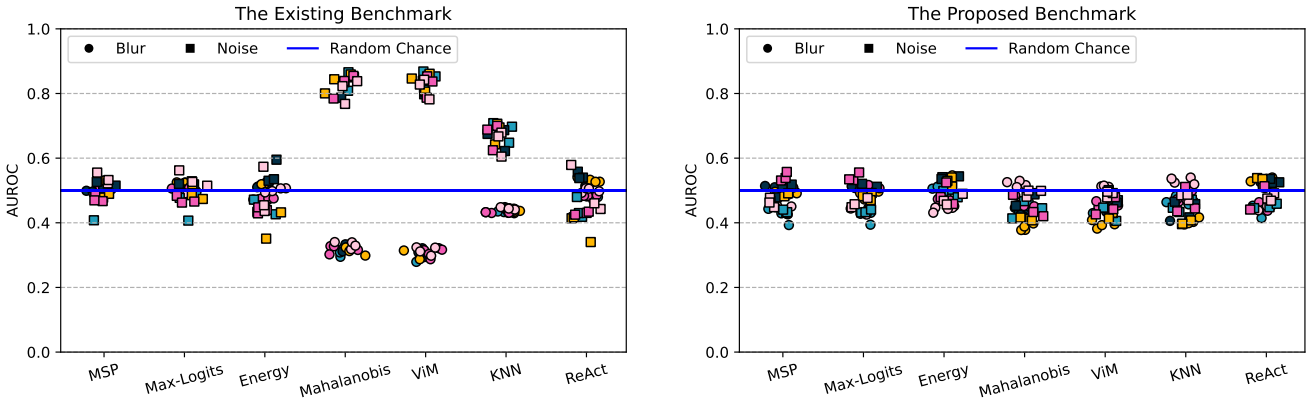

Figure 9. **Performance of OOD detection under random models.** AUROC performance of 5 ViT-L-32 models with **random**, untrained parameters on subsets of ImageNet-C [10] under the existing benchmark vs our proposed benchmark. Colors indicate the specific random model and the markers indicate the corruption type. Results from transformers slightly differs from convolution-based methods. Logits-based OOD methods performs around random chance under the old benchmark with data distribution shift.

G. ImageNet-OOD Classes

pincer, tape_deck, convenience_store, portmanteau, batting_glove, monstrance, linseed, recycling_bin, baked_potato, drawknife, pocket_watch, leek, horn, chicken_Kiev, rest_house, pantheon, remote_terminal, buzzer, anvil, hatpin, net, medical_building, trigger, office_building, bone-ash_cup, mattress_cover, cookie_jar, guava, pinball_machine, fish_cake, hotel-casino, map, sumo_ring, ghetto_blaster, persimmon, binnacle, pinata, gas_heater, brown_bread, yogurt, sukiyaki, magazine_rack, tender, bushel_basket, beach, battery, rasp, scrub_brush, windshield_wiper, censer, alpenstock, kohlrabi, relay, window_box, omelet, spoiler, pomelo, protractor, kaiser_roll, riot_gun, field_lens, french_fries, streetlight, mantilla, piston, glockenspiel, falafel, fricassee, monocle, hairdressing, motor, gearing, skyscraper, apartment_building, pylon, grinder, hubcap, triangle, ham_sandwich, free_house, sushi, checker, taco, dressing_table, tape, pizzeria, pile_driver, steak_au_poivre, tweed, bagpipe, deviled_egg, cashmere, sallet, bedspread, coffee_bean, matzo, urn, harmonium, machete, windshield, submersible, pasta, catcher's_mask, bean, winch, phial, coonskin_cap, tongs, pottage, chancellery, pantyhose, beanbag, kabob, vise, flap, television_equipment, pillory, butter_dish, headrest, C-clamp, toaster_oven, flannel, tamarind, gorget, wrench, gravy_boat, energizer, beekeeper, covered_bridge, hydrofoil, tumbler, bumper, risotto, breadstick, inhaler, stapler, paper_feed, densitometer, bacon_and_eggs, searchlight, avocado, clothes_dryer, coffee_filter, passion_fruit, air_hammer, bilge_pump, casket, ramp, semiconductor_device, potherb, beret, broiler, telephone_pole, referee, lentil, lute, ministry, spark_plug, contact, won_ton, andiron, duffel, ballet_dancer, snowboard, chicken_and_rice, lace, soldering_iron, goal, wire, corbel, denture, conference_center, eyebrow_pencil, patch, caftan, shackle, cantilever_bridge, peach, water_cooler, poached_egg, jello, flywheel, pavior, cockpit, terrine, weathervane, fuse, drill_press, fondue, chicken_soup, canal_boat, skewer, cymbal, aqualung, hovel, cranberry, pumpkin_seed, rubber_band, circular_saw, seafood_Newburg, paintball_gun, mustard, garrison_cap, bowler_hat, dolmas, tomatillo, fish_and_chips, barbecued_wing, rink, white, keyhole, analyzer, whiskey_bottle, mixer, fireman, junk, snifter, couscous, yam, egg_roll, psaltery, curry, chessman, bat, layette, schnitzel, short_pants, hanger, chocolate_kiss, ink_bottle, erecting_prism, peeler, bristle_brush, flask, blindfold, tugboat, fried_rice, hip_boot, supernova, garnish, ratatouille, balalaika, corn_pudding, hoist, console_table, tunic, Mason_jar, cafe_au_lait, hurricane_deck, pestle, grape, solar_array, shadow_box, muffler, alehouse, lime, beef_Bourguignonne, semi-detached_house, latch, condensation_pump, pier_glass, ticking, chandelier, tomato, tepee, dulcimer, laser, bacon-lettuce-tomato_sandwich, carabiner, pillow_block, ice_ax, microtome, car_seat, olive, mango, battle_cruiser, ejection_seat, sandbag, gaming_table, fire_extinguisher, dumpcart, stanchion, chili, chlamys, ordinary, tailstock, boysenberry, rye_bread, hand_truck, felt, headscarf, circuitry, wand, battle-ax, ball, chanter, simulator, barbecued_spareribs, step, kurta, guided_missile_cruiser, float, pilaf, nuclear_reactor, mandolin, graduated_cylinder, rouge, lentil_soup, earphone, table-tennis_table, rumble_seat, chuck, tailpipe, bandsaw, tostada, horizontal_stabilizer, crusher, escapement, phonograph_needle, vichyssoise, slide_fastener, amplifier, steeple, man-of-war, clothespin, bath_oil, balaclava, jamb, fipple_flute, afterburner, Chinese_cabbage, washboard, sconce, ventilator, trestle_bridge, kettle, gumbo, Scotch_egg, copyholder, medlar, baseball_cap, parimutuel_machine, cruet, millstone, chin_rest, litchi, punnet, back_brace, stile, treadmill, stool, dark_bread, tie_rack, roulette_wheel, papaw, drugstore, crater, buffer, brocade, bones, resistor, lever, casserole, camouflage, knocker, bumper_guard, toga, cesspool, synagogue, mulligatawny, dashiki, bok_choy, plantain, funeral_home, pedal_pusher, mousse, talus, bap, chowder, brussels_sprouts, hearth, kazoo, grapefruit, tamale, blower, clothesbrush, tricorn, sable, borsch, token, shirred_egg, shredder, shaving_cream, sashimi, golf_bag, propeller, ferry, nailbrush, Zamboni, watch_case, landfill, knee_brace, shawl, eggdrop_soup, router, white_sauce, plane_seat, pillion, dump_truck, sunflower_seed, volcanic_crater, bell_tower, mouthpiece, trimmer, pepper_pot, papaya, eyeliner, celery, Statehouse, aviary, beef_stew, pavilion, straightener, skateboard, eyeshadow, khukuri, deodorant, fish_stew, antenna, blowtorch, bird_feeder, stud_finder, tomahawk, hamburger_bun, ackee, bathrobe, polo_shirt, lobster_pot, masjid, sewing_kit, vane, baluster, biryani, pea_soup, fez, carboy, Sloppy_Joe, toothpick, ocean_floor, potage, bucket_seat, saute, wheel, litterbin, duffel_coat, ice_mass, comb, cigar_box, cake_mix, cabana, rissole, damper, cloth_cap, antiperspirant, tudung, toilet_bag, bait, Welsh_rarebit, Venn_diagram, French_toast, veloute, chocolate_pudding, catacomb, rope_bridge, orphanage, convector, vertical_tail, curler, hand_glass, gauge, weekender, lever_lock, kumquat, footbridge, trampoline, Reuben, barrette, English_muffin, bath_salts, bollard, bobbin, asparagus, gatepost, Ferris_wheel, prosthesis, knit, corned_beef_hash, cartridge_holder, openface_sandwich, toast, grindstone, dishtowel, steak_tartare, Spanish_rice, pulley, hand_calculator, towel_rack, courthouse, cheval_glass, porridge, frock_coat, plum_pudding, plum, weatherman, koto, bicycle_seat, minaret, shot_tower, sweet_roll, pot-au-feu, bilberry, sweat_pants, beer_garden, neckerchief, crystal, press, pocketknife, cylinder_lock, cenotaph, repair_shop, surgical_instrument, deflector, prickly_pear, coq_au_vin, drawbridge, hall_of_residence, supercomputer, plotter, nut_and_bolt, backscratcher,

butty, Frisbee, beef_Wellington, gearshift, shotgun, mechanical_piano, snuffbox, doll, shift_key, luggage_rack, compressor, hearing_aid, dildo, blueberry, shaving_brush, centrifuge, drawstring_bag, hard_roll, galantine, morion, fennel, franking_machine, clarinet, rotunda, vitamin_pill, blinker, spur, scraper, newspaper, shoebox, power_shovel, cotton_flannel, cowpea, sandwich_plate, stuffed_tomato, truss_bridge, food_processor, frigate, pesto, sash_fastener, presbytery, lavaliere, okra, gauntlet, turret, goulash, petite_marmite, nest_egg, speculum, headstock, brass_knucks, gun_case, bowling_ball, depth_finder, centrifugal_pump, steering_wheel, carburetor, eggs_Benedict, hot_sauce, cot, autoclave, sprinkler, lamppost, aerosol, parts_bin, opera, Tesla_coil, windowsill, Jordan_almond, wafer, call_center, shoehorn, face_guard, regulator, rocker, currant, pet_shop, X-ray_machine, bowling_alley, church_tower, club_sandwich, ski_boot, salad, control_key, inkwell, seeder, finder, dugout, crouton, coffin, backlighting, inkle, cable, concertina, music_hall, cap_opener, greatcoat, golf_club, retainer, settlement_house, rain_stick, raisin_bread, jambalaya, gravy, trowel, veal_parmesan, capacitor, fish_stick, heat_pump, sack_coat, poplin, bisque, bandbox, newel_post, zither, succotash, silencer, compass, dowel, musket, broth, medicine_ball, vibraphone, destroyer, paper_fastener, wild_cherry, serving_cart, organ_pipe, shelf_bracket, bicycle_pump, pouch, abattoir, control_center, scouring_pad, counter, quiver, splint, Walkman, radio-phonograph, churn, moussaka, coil_spring, hasp, astronaut, toothbrush, pepper_steak, pipe_cutter, clock_tower, sonogram, jujube, drumhead, eggbeater, cattle_guard, pieplant, valve, nailfile, cinnamon_bread, cave, nylon, plughole, sump_pump, kitchen_table, noisemaker, stator, hollandaise, breadfruit, snow_thrower, crock, barbecue, spectator_pump, first-aid_kit, marmite, armet, Irish_stew, tea_bag, plug, regalia, gyro, ramekin, berth, wicket, amphora, Murphy_bed, runner, boiled_egg, adding_machine, stun_gun, sour_bread, Bailey_bridge, tin, cherry, stuffed_peppers, cleat, mascara, sachet, caliper, roaster, brioche, Sharpie, bunk_bed, stabilizer, kale, brake_shoe, plush, cappuccino, megaphone, silo, beer_mug, coupling, dais, gunnysack, reporter, planchet, saddle, rivet, stirrup, aerator, hula-hoop, enchilada, hunting_knife, air_filter, tread, apricot, record_changer, western, sport_kite, radar, catapult, miter_box, imprint, mess_kit, frankfurter_bun, tongue_depressor, box_spring, brake_pad, denim, riveting_machine, silk, supporting_tower, high-rise, twenty-two, tam, ruin, lanyard, jodhpurs, mill, futon, carafe, snare, flintlock, breathalyzer, stob, canopy, breadbasket, optical_telescope, ankle_brace, doorsill, mulberry, case_knife, overpass, floss, Segway, souffle, adapter, chow_mein, screw_thread, acerola, tank_engine, chickpea, brace, rudder, respirator, grease-gun, nut, duff, tool_bag, gourd, dropper, spear, bouquet, ottoman, cruise_control, fried_egg, bayonet, goalpost, trophy_case, vibrator, harness, finger-painting, fishing_gear, watch_cap, shiv, cigarette_case, date, tee, elevator, tidal_basin, shutter, armored_vehicle, town_hall, bourguignon, meatball, ski_binding, toby, sitar, julienne, squeegee, headlight, titrator, graham_cracker, backspace_key, shunter, split-pea_soup, webbing, harbor, pump, pannier, cottage_pie, spray_gun, bowling_pin, circuit, hairbrush, rumble, capote, bell_jar, pilot_boat, veal_cordon_bleu, cereal_box, piece_de_resistance, bicycle_rack, canopic_jar, bottle, step-down_transformer, gefilte_fish, bracelet, cheerleader, control_panel, crescent_roll, saucepot, hotel, Chinese_lantern, oil_lamp, briefcase, concrete_mixer, scrambled_eggs, winder, distributor_cap, serving_dish, rotor_blade, theodolite, golf_glove, barge, jack, samisen, cardcase, eyepatch, frittata, turnover, record_sleeve, backseat, chapterhouse, lightning_rod, home_fries, fishbowl, camise, chop_suey, rattan, wrap, Bowie_knife, sugar_bowl, carpet_sweeper, transporter, outrigger, lingonberry, florist, broadcaster, gasket, garlic_bread, solar_cell, backboard, radiator_cap, solar_heater, shaper, customhouse, button, volleyball_net, capitol, snowshoe, bunk, ski_cap, signal_box, music_stand, paella, gas_oven, arrester, pea, backbench, sword, rundle, electric_hammer, tempura, textile_machine, staple_gun, wineglass, croquette, doily, flatbread, buskin, challah, eggplant, gazpacho, fencing_mask, equalizer, bomber, record_player, adobo, thermometer, marble, spice_rack, rigging, bearing, pumpkin, onion, dressing_gown, mosaic, school, bread_knife, nan, ferrule, hammock, audiometer, toastrack, tuning_fork, bolt, backstay, gazebo, police_boat, carving_knife, embassy, chicken_sandwich, surge_suppressor, drain_basket, bootjack, accelerator, cold_cathode, kepi, leash, nurse, parer, receiver, melon, observatory, viand, bedpan, green_pea_soup, slicer, octant, central_processing_unit, buffalo_wing, dagger

n02811468, n04154938, n04387095, n07618119, n03542605, n02995345, n02893608, n04318787, n04314914, n04615226, n03343354, n07648997, n07682197, n04228215, n02943964, n02783994, n09238926, n02696165, n02927764, n04112252, n02705944, n04589325, n03061674, n04294426, n03122073, n04211219, n03680858, n02925666, n07743902, n04181561, n03005033, n04364160, n03820318, n02683558, n07696977, n04258859, n03955489, n04071263, n03610682, n03238586, n03253796, n12246232, n04108822, n03027250, n04206790, n07695652, n02835829, n04606574, n04249882, n03996416, n03814817, n03936466, n04530283, n07879072, n02890662, n02682569, n07920349, n03345837, n02952237, n03839671, n07588574, n04385536, n04461437, n04269270, n04211528, n03198500, n03229244, n04238321, n03844045, n02936570, n03840823, n03887330,

n04532831, n03721047, n02944579, n03494537, n02715229, n07682808, n02865931, n02882190,
n03255899, n07877849, n07648913, n04200258, n04594489, n07618432, n03161450, n04482177,
n03805280, n03614782, n07880751, n02960690, n03342127, n04459773, n02977330, n03698360,
n12158031, n12713063, n03333711, n07866409, n03659809, n04389854, n12560282, n03785721,
n07654148, n09454153, n02723165, n07714078, n03903733, n07586604, n03973628, n03938401,
n03805180, n03309110, n04497570, n02886321, n04095342, n07696403, n02811618, n03207835,
n04082562, n02986160, n04228693, n07861813, n04024862, n04258333, n07682316, n03853924,
n03156767, n04592099, n07691954, n07937461, n07591961, n04186051, n07762740, n03075097,
n09451237, n03442756, n12744387, n06275095, n02864593, n07835457, n03774327, n04134008,
n04252653, n03350204, n07713074, n02911332, n07881404, n04167346, n04015908, n03087069,
n04412416, n04132603, n02920259, n04117464, n04119091, n03609397, n02725872, n03423479,
n02771004, n03503997, n07929351, n04176190, n03603594, n07714287, n02757337, n04287747,
n04050933, n12709688, n03678558, n03469903, n03626760, n03941231, n03704549, n09818022,
n04432662, n07580359, n12636224, n03295012, n07750736, n07579688, n02890940, n04093625,
n03329663, n04483925, n04185946, n03571625, n04233124, n04119751, n07690152, n09913593,
n04373894, n03420801, n03506727, n07592094, n02998003, n04177931, n02742468, n03448590,
n07713763, n04526964, n02973017, n07750872, n03547054, n07865196, n03219010, n03431745,
n03158885, n03367410, n04495698, n07866868, n02910145, n03970546, n02678897, n03363749,
n04488202, n03176386, n02768226, n10542888, n12301445, n04354182, n03329302, n03235796,
n04016846, n04350581, n02973904, n03232543, n03589513, n03261776, n11877283, n07585208,
n03641569, n07832416, n03941417, n03293741, n03499354, n03266371, n02931294, n03993180,
n07723039, n03067093, n04075291, n04300643, n04198722, n03525074, n03897943, n07579917,
n03141702, n07875436, n12911673, n04438897, n03034405, n03834040, n03082807, n02824058,
n07843464, n03349469, n04205318, n04354487, n07862244, n03788047, n03176594, n03937931,
n07835921, n04450749, n03696301, n04306592, n03359566, n09376526, n13136316, n12441183,
n03519387, n03034663, n03033362, n10091651, n02775897, n03490884, n03114504, n02919148,
n03172038, n07684517, n03356982, n04579986, n03920641, n04364545, n07879953, n04381587,
n02689434, n03235327, n03475823, n02905036, n03398228, n04409128, n04043411, n07617932,
n12648045, n03861271, n07837362, n04414675, n07869522, n03142679, n02957008, n07613815,
n07758680, n04097760, n04080833, n07840027, n04492749, n07692614, n03875955, n03795269,
n03495039, n03425325, n07733394, n02852173, n03140652, n02930080, n04358117, n03856012,
n07743224, n09309292, n07919572, n04176068, n07715221, n03952576, n02771286, n07762114,
n03565288, n03446070, n03254046, n07593004, n03797182, n03378174, n07611991, n03050655,
n07691758, n07680761, n03938037, n10521662, n03402941, n02810782, n02826068, n07745046,
n04217546, n03549732, n03643253, n07698672, n03784270, n13879320, n07585758, n03098140,
n04252331, n04064747, n04289027, n07697699, n12515925, n04248851, n03719743, n12638218,
n03720163, n02786837, n04072193, n03150232, n03572321, n07753743, n03650551, n07586894,
n07617708, n03256166, n04193377, n03007444, n03015851, n04590263, n07696839, n02786331,
n11851578, n03351434, n03041114, n03223686, n07685730, n04218564, n12172364, n07868508,
n07710952, n02835724, n03923379, n04313503, n03064758, n04575723, n03789946, n04582869,
n03066359, n02981024, n03858418, n03505133, n02816656, n02806379, n07868955, n12433081,
n04130907, n04392526, n02918831, n09217230, n03175189, n04065789, n07870313, n02970685,
n03556679, n03408444, n03099274, n04442441, n04013729, n02887970, n04022332, n11879054,
n03460297, n04315342, n03080633, n03553019, n03152303, n03097362, n04041243, n04516214,
n07682624, n03178000, n03219135, n02682922, n03272125, n07866277, n02951703, n04489817,
n04064401, n02978055, n03770316, n03116767, n03751458, n03537241, n07842753, n02826886,
n02770721, n02869249, n02983189, n03986949, n04344003, n07869611, n07591049, n03175457,
n03505504, n07830593, n03842012, n03340723, n07588299, n03047052, n12400489, n04282494,
n09846755, n04060647, n02688273, n07838073, n03629231, n03731483, n04549629, n07842308,
n03668067, n07775197, n03357716, n04220250, n02713003, n07654298, n07586318, n07868340,
n04051549, n04445040, n02882301, n03645011, n03451711, n03548086, n04186455, n07765999,
n02874442, n03011355, n04392113, n04113406, n04370288, n02976249, n02993368, n03432129,
n04284869, n04257790, n04118635, n07879350, n07865484, n03483823, n07823951, n02953197,

n03479397, n04469514, n03890093, n03267468, n04502502, n02808185, n03094159, n12578916,
n02887489, n02812949, n02776205, n12136392, n04190376, n03163381, n04417180, n04346157,
n04160847, n02767956, n02904803, n03941684, n04402449, n07697825, n02670683, n03475581,
n07687053, n04525821, n07585557, n07867021, n03919430, n03121431, n03407369, n02855089,
n02947660, n03904909, n07621618, n03423719, n03392741, n04156140, n07869775, n04312432,
n09335809, n03157348, n07880458, n04172107, n04200537, n03237992, n03725600, n07734017,
n04477387, n02731629, n07842605, n07765073, n03105467, n03802007, n04182152, n03086868,
n07594066, n03738066, n02776825, n07686873, n02982232, n02903204, n12544539, n03490119,
n03507241, n03456299, n03281145, n03612965, n03329536, n02726681, n02666943, n07770763,
n04586581, n02807616, n03374649, n02822220, n03946076, n03765561, n07877299, n03836906,
n03326795, n03718212, n02831724, n07746334, n03724066, n02998563, n04075916, n04122349,
n07933154, n04303357, n03801760, n07591473, n04570815, n04219424, n07880213, n02799323,
n03115400, n04054670, n03005285, n03894677, n07587111, n10514429, n07588193, n07843636,
n02708711, n02705429, n03327553, n02775483, n07872593, n07866151, n04171831, n07878647,
n07696728, n03885904, n12805146, n09834699, n03968581, n07744057, n03030353, n07730406,
n04479939, n07584332, n04303497, n03308481, n06267145, n03600285, n03031012, n04452528,
n04497801, n03393017, n04585745, n07876651, n03050453, n10772092, n03767112, n03240892,
n02823586, n07877675, n04270891, n04501947, n02769669, n03799876, n07691539, n02940385,
n07862611, n07864934, n04326896, n09457979, n03497352, n03883524, n04290259, n03018712,
n07842202, n04540255, n03119510, n02962200, n03447075, n13136556, n04474035, n04519153,
n03492250, n03739518, n07590320, n02868975, n03699591, n03445617, n03247083, n03549473,
n07849336, n03978966, n03099147, n04206356, n04063154, n03430418, n04496872, n03637181,
n04448361, n07698401, n07841495, n03775388, n03159640, n03934311, n07690019, n04502851,
n02831237, n03019685, n03521544, n07584423, n03397266, n02880842, n03865557, n03760944,
n03485198, n12158443, n04434207, n04253057, n07817871, n12352990, n02860415, n07709172,
n03429682, n04150980, n02993194, n02882647, n03314227, n09472413, n03029445, n04095577,
n03309356, n04275175, n04283255, n03572107, n02838345, n04361260, n03954393, n07879174,
n03364599, n03767745, n03999160, n04179824, n03827536, n07711232, n02956795, n04146050,
n07592481, n02995871, n02962061, n07641928, n02893941, n04035836, n03977592, n04091693,
n07607138, n02960903, n02881757, n04579056, n03821518, n04590746, n12642090, n03037709,
n02967782, n03900979, n03901229, n04112752, n02698634, n04119230, n02988066, n04443766,
n04123448, n04150153, n04405540, n04476972, n02939866, n04260364, n07592768, n07938149,
n02962843, n03631177, n04554406, n03652932, n04049753, n04321453, n03983612, n10345015,
n04139140, n04039848, n03628215, n03029197, n03254189, n03849814, n07708124, n04112654,
n02955247, n03509608, n12641413, n04225987, n04421872, n07666176, n03367545, n03884926,
n03536122, n07588817, n07586718, n02892948, n04320973, n03571280, n03743279, n12771192,
n03468821, n03625355, n04020298, n07680517, n04533199, n04273285, n03436182, n07698543,
n04072960, n07867421, n02710044, n03443149, n12711984, n04453156, n03887185, n03923918,
n03724756, n04093775, n07842433, n12761284, n07588419, n04292921, n03063199, n03249342,
n04538552, n03659950, n04098513, n02821202, n07642361, n04442741, n04613939, n02902687,
n02817031, n04449966, n02851939, n04348359, n03592773, n03309687, n02900705, n11706761,
n03801533, n03140431, n07665438, n03254862, n10366966, n12709103, n04386051, n07655263,
n03233744, n03542333, n03482252, n03331077, n12373100, n02779435, n03783430, n03973839,
n04441790, n03948950, n07866015, n04305210, n04559166, n03105088, n04335886, n07867164,
n03466493, n07698782, n03460040, n02895438, n04568069, n02776978, n02807523, n04210120,
n02920083, n03282295, n15086247, n04228581, n04011827, n02763604, n03615790, n03852688,
n07866723, n03021228, n02811204, n07587023, n07680313, n07713267, n04556408, n07874780,
n03968293, n07867324, n07696625, n03456665, n04224842, n02792552, n04184435, n04227900,
n02740533, n03397947, n02877266, n07862461, n02956699, n02758960, n04453390, n04399537,
n03622839, n03742019, n02879087, n02928608, n03180504, n04185529, n07880325, n07868200,
n12088223, n03097673, n07681691, n02738741, n02925009, n03213538, n04257986, n03939178,
n04123740, n03051249, n04546340, n03456024, n04609651, n03620967, n04272389, n03089753,
n02843553, n07587441, n02768655, n07711080, n07698250, n03133415, n03819448, n07871436,

n04028764, n07681450, n03379828, n03014440,
n03429288, n04079933, n03101664, n03506184,
n02855390, n02770830, n04483073, n03087245,
n07862348, n04331639, n07806221, n04446844,
n02835915, n07586099, n03440682, n04495843,
n07755411, n03092883, n04568841, n04374735,
n03548402, n04251791, n04122685, n04081699,
n07863374, n04221823, n03854815, n07682477,
n03649161, n07878926, n09259219, n07686720,
n04451318, n04168199, n07684164, n12501202,
n03716966, n03250279, n07593471, n04136800,
n07879450, n03967396, n07864756, n03428349,
n07696527, n12399132, n04148703, n04116098,
n07868830, n07691650, n03006626, n02796318,
n02679257, n04419073, n04237423, n03103563,
n07879659, n12333771, n03890514, n04590553,
n07873464, n03836451, n07877961, n02995998,
n03414676, n02841187, n03287351, n07861557,
n04556533, n07587331, n04114844, n03049924,
n04000311, n07624466, n04520784, n03296081