Addressing Randomness in Evaluation Protocols for Out-of-Distribution Detection

Konstantin Kirchheim¹, Tim Gonschorek¹, Frank Ortmeier¹
Otto-von-Guericke-University Magdeburg
{konstantin.kirchheim, tim.gonschorek, frank.ortmeier}@ovgu.de

Abstract

Deep Neural Networks for classification behave unpredictably when confronted with inputs not stemming from the training distribution. This motivates out-of-distribution detection (OOD) mechanisms. The usual lack of prior information on out-of-distribution data renders the performance estimation of detection approaches on unseen data difficult. Several contemporary evaluation protocols are based on open set simulations, which average the performance over up to five synthetic random splits of a dataset into in- and out-of-distribution samples. However, the number of possible splits may be much larger, and the performance of Deep Neural Networks is known to fluctuate significantly depending on different sources of random variation. We empirically demonstrate that current protocols may fail to provide reliable estimates of the expected performance of OOD methods. By casting this evaluation as a random process, we generalize the concept of open set simulations and propose to estimate the performance of OOD methods using a Monte Carlo approach that addresses the randomness.

1 Introduction

Machine Learning driven classification systems are increasingly deployed in open, real-world environments. Traditionally, they operate under certain assumptions, among others, the i.i.d assumption, which asserts that samples from test and training set are mutually independent and generated by identical stationary distributions [Goodfellow et al., 2016], and the closed set (or closed world [Boult et al., 2019]) assumption, which states that the data generating distribution draws from a fixed, finite set of categories [Geng et al., 2020]. In more realistic scenarios, however, drifts in the data generating distribution are likely to occur between - as well as during - training and test time, and for classification tasks, the number of observable categories is (for all practical purposes) unlimited. Contemplating this problem, [Scheirer et al., 2012] argue that labeling something as new, unknown or other should always be considered a valid option. Existing literature addresses the task of recognizing inputs that do not fit into a known category (and therefore violate these assumptions) under the name of Open Set Recognition [Geng et al., 2020], which can be considered a special case of out-of-distribution (OOD) detection, where the distribution from which OOD samples are drawn is assumed to be conditioned on class categories [Ruff et al., 2021]. In closed-set settings where the true data generating distribution is unknown, classification models are usually selected according to the principles of empirical risk minimization [Vapnik, 1992]. In this framework, the performance on unseen data is estimated on a subset of the available samples, which is justified by the above assumptions. Evaluating OOD methods turns out to be difficult because there are usually no representative OOD samples available, and the number of OOD classes is possibly infinite. Usually, existing datasets are adapted for this task. Several recent publications evaluate the performance of OOD methods by artificially splitting an existing dataset into subsets of IN and OOD classes [Scheirer et al., 2012; Scheirer et al., 2014; Miller et al., 2021; Geng et al., 2020; Hassen and Chan, 2020; Oza and Patel, 2019; Zhang et al., 2020]. The model is trained only on IN samples and tested to distinguish both IN and OOD samples - which is called open set simulation. This evaluation protocol of splitting, training, and testing is repeated up to five times, and results are averaged. However, to our knowledge, unlike empirical risk minimization, this practice lacks a clear theoretical justification.

Recent works demonstrated that the performance measurement of Deep Neural Networks (DNN) fluctuates significantly with several sources of random variation, among others the parameter initialization and the order in which the training data is presented to the model [Bouthillier et al., 2019], or implementation details [Musgrave et al., 2020]. While one could try to remedy this by setting the random seed in experiments to a fixed value, this would also limit the conclusions drawn from this experiment to this particular random seed. [Bouthillier et al., 2019] affirm that such conclusions are brittle and likely to be falsified by experiments with a different random seed. Instead, they recommend that deep learning experiments be replicated several times, results should be tested for statistical significance, and confidence intervals should be reported. The effects of randomness and the difficulty of reproducing experiments have been studied for Image Classification [Bouthillier et al., 2019], Metric Learning [Musgrave et al., 2020], Image Synthesis [Lucic et al., 2018], and Re-
In this work, we aim to demonstrate and address the possible issues that emerge from the inherent randomness of different OOD evaluation protocols. While we acknowledge the existence of protocols that draw OOD samples from unrelated datasets or synthetic distributions like Gaussian or uniform noise, this study is primarily concerned with the open set simulation framework, which, to our knowledge, the prevalent protocol in the Open Set Recognition domain. In Section 2, we provide a formal description of the concept of open set simulations that generalizes several evaluation protocols and identify several sources of randomness. Due to this randomness, we suspect that contemporary protocols fail to provide reliable performance estimates when comparing different OOD methods. In Section 3, we provide a brief overview of several OOD methods and datasets that are commonly used for comparison. In Section 4, we extensively evaluate the presented methods, running three orders of magnitude more open set simulations than previous publications, and empirically demonstrate that due to significant performance fluctuations, in experiments with five-fold open set simulations, several methods have a high chance of obtaining the highest score. Furthermore, we provide evidence that other protocols not based on open set simulations might be subject to the same phenomenon. Based on this observation, we argue that the evidence provided by present evaluation protocols constitutes a brittle foundation for conclusions. In Section 5, we propose to treat open set simulations as a fundamentally probabilistic process and cast it as a Monte Carlo approach to estimate the expected performance on unseen data. To the best of our knowledge, this approach is the first that systematically accounts for the performance fluctuations that stem from different sources of random variation in OOD experiments.

2 Open Set Simulation

Several existing evaluation protocols for OOD in classification tasks are based on open set simulations, meaning that they evaluate the performance of methods on a dataset by using a synthetic split of classes into in- and out-of-distribution. There are several variations of this protocol, some of them tailored to specific requirements of a method, which impedes comparability. In the following, we will formally describe the open set simulation framework, with the goal of generalizing as much as possible. The open set-simulations used in [Geng et al., 2020; Hassen and Chan, 2020; Scheirer et al., 2014; Miller et al., 2021; Neal et al., 2018; Oza and Patel, 2019] can be considered a special cases of this protocol.

OOD for classification tasks can be formulated as follows: Let \( C \subseteq Y \) be a subset of all possible classes, \( S \subseteq X \) a set of samples as a subset of the input domain, \( \Psi : X \rightarrow Y \) a supervisor that assigns class labels, and \( D = \{(x, y) | x \in S \wedge y \in C \wedge y = \Psi(x)\} \) a dataset, where the \( x \) are drawn from a data generating distribution. The goal of a method is to find a function \( f : X \rightarrow C \cup \emptyset \) that approximates the supervisor for the codomain such that

\[
f(x) = \begin{cases} \Psi(x) & \text{if } \Psi(x) \in C \\ \emptyset & \text{else} \end{cases}.
\]

The first case of this function describes closed-set classification, while the second case introduces the OOD task. The open set simulation framework aims to provide a means to estimate the fidelity of a model to function (1) without access to samples where \( \Psi(x) \notin C \).

2.1 Dataset Construction

Constructing a dataset for an open set simulation comprises two steps, which divide an existing dataset \( D \) into six subsets in total. An overview is provided in Figure 1. Note that some of these subsets may be empty.

Class Split

In the first step, the classes \( C \) are split into four different sets. One set of classes \( C_{\text{in}} \) whose samples are considered to be in-distribution, and three sets of out-of-distribution classes, \( C_{\text{out}}^{\text{train}}, C_{\text{out}}^{\text{val}} \) and \( C_{\text{out}}^{\text{test}} \). Models will be trained on samples from \( C_{\text{in}}^{\text{train}} \) and potentially \( C_{\text{in}}^{\text{val}}\).

Sample Split

In a second step, the samples from \( S \) are divided into 3 different subsets, \( S_{\text{train}}, S_{\text{val}} \) and \( S_{\text{test}} \) for training, validation and testing respectively. This approach follows the principle of Empirical Risk Minimization for estimating the performance of supervised models on unseen data empirically. However, in our case, it results in a total of 6 distinct subsets of \( D \), such that \( D_{\text{out}} = \{(x, y) | x \in S_{\text{train}} \wedge y \in C_{\text{out}} \} \subseteq D \), and correspondingly for the other five combinations.

2.2 Training

After dataset construction, a method is used to create a model \( f \). This model is trained on \( D_{\text{train}} = D_{\text{train}}^{\text{in}} \cup D_{\text{train}}^{\text{out}} \). While there are approaches that require training or finetuning on out-of-distribution data (e.g. [Dhamija et al., 2018; Hendrycks et al., 2018]), others do not (e.g. [Hendrycks and Gimpel, 2017]). In the latter case, \( C_{\text{in}}^{\text{val}} \) can be empty. Similarly, in some settings, validating a model before testing is not required, and \( C_{\text{in}}^{\text{val}} \) can be the empty set. However, certain techniques require a separate validation set, for example, early stopping - a method to prevent overfitting by interrupting the training once the generalization performance
decreases [Prechelt, 1998]. Having a separate validation set also allows tuning hyperparameters.

2.3 Testing

After the model is trained (and possibly validated), it is tested on $\mathcal{D}_{\text{test}} = \mathcal{D}_{\text{train}}^\text{in} \cup \mathcal{D}_{\text{test}}^\text{out}$. Different performance metrics are used to assign a performance score $P(f, \mathcal{D}_{\text{test}})$ to the model. While the classification performance is commonly measured with the accuracy, there are several metrics for the OOD performance. The following two are, to our knowledge, the most frequently used:

AUROC The Area under Receiver Operating Characteristic, which characterizes the tradeoff between the false positive rate and the true positive rate, provides a threshold-independent metric for binary classification tasks. It ranges from zero to one, with larger values indicating better OOD performance and 0.5 corresponding to random guessing.

AUPR The Area Under Precision-Recall characterizes the tradeoff between Precision and Recall for varying thresholds. It ranges from zero to one, where larger values indicate better performance. In contrast to the AUROC, this metric is not symmetric, which means that there are two versions: one where the $\mathcal{D}_{\text{test}}^\text{in}$ data is treated as positive (AUPR-IN), and one where samples from $\mathcal{D}_{\text{test}}^\text{out}$ are treated as positive (AUPR-OUT).

Further information regarding these metrics can be taken from [Ruff et al., 2021].

3 Methods and Datasets

In the following, we provide a brief overview of DNNs, OOD methods, and datasets that will be used in subsequent experiments.

3.1 Architecture & Training

In our experiments, we employ the ResNet-18 DNN architecture [He et al., 2016], as it is widely used, and training is comparably cheap. This model comprises a 17-layered convolutional feature encoder with residual connections, followed by a global average pooling layer that produces a feature vector. We apply dropout [Srivastava et al., 2014] of 0.2 to the features and propagate the result through the final fully connected layer that outputs a logit vector $z \in \mathbb{R}^K$ where $K$ is the number of IN classes. This vector is passed through the softmax activation function

$$\sigma_i(z) = \frac{\exp(z_i)}{\sum_k \exp(z_k)} \quad (2)$$

which normalizes the output for each class $i$, so that the scores can be interpreted as posterior probabilities of class membership. We train our models using stochastic gradient descent with a mini-batch size of 128 and weight decay of $5 \times 10^{-4}$ minimizing the categorical cross-entropy between the predicted class membership distribution and the one-hot encoded target label. We use an initial learning rate of 0.01 that we gradually vary with a cosine annealing schedule [Loshchilov and Hutter, 2017]. We apply standard pre-processing and augmentation techniques: input normalization over $\mathcal{D}_{\text{train}}^\text{in}$, horizontal flipping, and random rotation and scaling. During training, we monitor the validation loss on $\mathcal{D}_{\text{val}}^\text{in}$ and interrupt the training once the loss stops decreasing.

3.2 Methods

Softmax Thresholding [Hendrycks and Gimpel, 2017] is a baseline method for OOD. It is based on the observation that the maximum class score $\max_i \sigma_i(z)$ tends to be lower for out-of-distribution samples. A simple threshold can be applied to this value to separate in- and out-of-distribution samples.

Temperature Scaling [Guo et al., 2017] (TS) is a method that recalibrates the outputs of a DNN by dividing the logits $z$ by a constant temperature $T$ before passing them through the softmax, which leads to more uniformly distributed class probabilities. As a result, the number of overconfident predictions is reduced.

ODIN [Liang et al., 2018] is a preprocessing method based on the observation that the score for the predicted class $\max_i \sigma_i(z)$ changes more for in-distribution data than for out-of-distribution data when taking a single gradient step in the input space. Formally, it can be described as $\hat{x} = x - \epsilon \text{sgn}(\nabla_x \sigma_i(f_T(x)))$, where $\hat{x}$ is the preprocessed input, $\epsilon$ is the step size, and $f_T$ is the temperature scaled DNN.

OpenMax [Bendale and Boult, 2016] is a post-training replacement for the softmax layer which adds an additional other class. For each known class, the method estimates a cluster center in the logit space and uses a probabilistic model to calculate a pseudo-activation for the other class. Like [Oza and Patel, 2019], we use the score of this class for OOD.

Monte Carlo Dropout [Gal and Ghahramani, 2016] (MCD) is a method that makes use of an interpretation of dropout as approximate Bayesian inference. While dropout is usually only applied during training of a DNN, the authors claim that when using it during inference, the average scores over multiple forward passes can be seen accurate uncertainty estimates.

3.3 Datasets

During experiments, we used both test and training set to create open set simulations from the following datasets:

- **MNIST** 70,000 $28 \times 28$ grayscale images of 10 handwritten numbers.
- **SVHN** 99,289 $32 \times 32$ color images of house numbering signs. The task is to determine the number on the sign; thus, there are 10 class labels in total.
- **CIFAR-10** 60,000 $32 \times 32$ color images of 10 different animal species or objects.
- **CIFAR-100** a larger version of the CIFAR-10 with 100 classes. As it still features 60,000 images, the number of per class samples is smaller than for the CIFAR-10.
- **Tiny-ImageNet** a downscaled subset of the ImageNet dataset that with 120,000 $64 \times 64$ color images from 200 diverse classes.
4 Randomness in Open Set Simulations

Considering the evaluation protocol described in Section 2, we see that several steps, from the construction of an open set simulation, over the initialization of the model parameters to the training itself, are subject to randomness. In the case of MCD, the evaluation also involves randomness. As publications only recently demonstrated significant variation in the performance of DNNs based on the random seed, to the best of our knowledge, this effect has never been studied systematically in the OOD domain. We suspect that the effects of randomness are even more severe in this field since the nature of the evaluation introduces an additional source of random variation.

4.1 Impact of Class Split on Performance

It is usually assumed that the performance of a method varies depending on the class split since there might be configurations for which detecting OOD samples is simpler, for example, if the visual similarity between the selected known and unknown classes is large. Therefore, the performance is usually evaluated over several splits. Generally, selecting \(k\) classes as \(C^m\) from a dataset with \(N\) classes in total results in \(\binom{N}{k}\) possible splits. If we consider the ImageNet with 1000 classes and assume 600 classes to be known, there are \(\approx 4.96 \times 10^{299}\) such splits. To the best of our knowledge, the common practice of using three to five splits is arbitrary and not chosen based on theoretical considerations or empirical evidence.

First, we aim to determine if the class split, as usually assumed, does indeed impact the performance of methods, and if so, how this relates to other sources of randomness. For the five class splits of the MNIST dataset used in [Miller et al., 2021], we conducted 100 trials with different random seeds. The seed determined, among others, the sample split, the network parameter initialization, and the ordering of the training data. Figure 2 depicts the distribution of the AUROC for the Softmax Thresholding baseline approach [Hendrycks and Gimpel, 2017]. As expected, the results indicate performance differences between different class splits. However, even for a single split, the performance varies significantly. Considering the observed performance variance and the vast number of possible splits, it is questionable that experiments based on five random seeds can provide reliable performance estimates.

4.2 Randomness in Experimental Outcomes

Second, we investigate if the outcomes of experiments with five open set-simulations are stable and thus able to provide a solid foundation for conclusions. We evaluate the OOD methods presented in Section 3 on datasets with color images in 1000 open set-simulations each, which took approximately 113 hours on two Nvidia A100 GPUs. This number of simulations is about three orders of magnitude larger than current evaluation protocols and provides us with a large pool of possible experimental outcomes. For each method, we can then sample five trials from this pool and compare the average scores - which will give us a “winning” method. By iterating this process (in our case 10,000 times), we can estimate the probability that a method will win in such experiments. If the outcomes of such evaluations were stable, the probability of winning should be concentrated on a single method.

The results are depicted in Figure 3. The bars represent the approximate probability that a particular method will have the highest average score for the given performance metric in five open set simulations if the experiment is replicated several times with different random seeds. As we can see, while the newest method we tested (ODIN) usually has the highest chance of winning, the probabilities are distributed between TScaling, OpenMax, and ODIN, which means that each of these methods has a substantial chance of appearing to be the best method in such an evaluation. This observation holds for all datasets and across all metrics we used. We conclude that (with the used hyperparameters), these three methods usually outperform the baseline, as well as MCD. However, which of these three methods “outperforms” the others largely depends on random chance. We note that the probabilities seem to be more evenly distributed for the AUPR metrics. For the Tiny-ImageNet dataset and the AUPR-OUT, all methods, including the baseline, have a chance > 15% of winning the comparison.

4.3 Other Evaluation Protocols

Apart from open set simulations, there are evaluation protocols that use samples from different datasets or synthetic data for \(D_{test}\) [Hendrycks et al., 2018]. Intuitively, this task seems easier since samples from the same dataset might share more statistical properties than samples from unrelated datasets, which would make it easier to detect the latter.

Third, we conducted the following experiment: we train
the baseline method on $D^{\text{train}}_{\text{CIFAR-10}}$ and calculated the AUROC on $D^{\text{test}}_{\text{CIFAR-10}}$ and samples from different datasets. These other datasets include the entire SVHN dataset, 1000 samples from Uniform $U(0, 255)$ or Gaussian Noise $N(128, 128)$ (clipped to $[0, 255]$). Figure 4 depicts the distribution of the AUROC for 1000 different random seeds. The baseline achieves higher AUROC scores for the SVHN, which indicates that discriminating these unrelated OOD samples easier than discriminating OOD samples from the original dataset. For samples from Uniform and Gaussian Noise, the AUROC tends to be lower, implying that, in this setting, it can be more challenging to distinguish synthetic images than out-of-distribution samples from the same dataset. Further, we note that in these experiments, the performance also fluctuates significantly. These results show that open set simulations do not provide a lower bound for the OOD performance. In tests with different datasets as $D^{\text{test}}_{\text{out}}$, the model might be less performant.

5 Probabilistic Open Set Simulation

The results presented in Section 4 call the common practice of drawing conclusions based on three to five open set simulations into question. Just like for other fields, randomness significantly impacts the performance of OOD methods, arguably even more. Since all sources of random variation can be controlled in a computing environment, the execution of experiments is deterministic in theory. However, given the previous finding, we argue that the evaluation process should be treated as fundamentally probabilistic in practice. In the following, we propose a generalized version of the open set simulation framework that addresses this probabilistic behavior. It is conceptually simple: repeatedly sample open set simulation and evaluate methods until a sufficient confidence level is reached.

The different sources of randomness, from the construction of an open set simulation, over the initialization of the model parameters to the training itself, can be seen as random variables and described by probability distributions. The score of a method $M$ then becomes a function $O_M$ of this randomness. All randomness can be reduced to the seed that the random number generator is initialized in a deterministic computing environment. To compare two methods, we would compare the expected value of their score over the random seeds.

5.1 Monte Carlo Estimator

The exact calculation of the expected score is intractable because it involves a sum over all possible configurations. As described in Section 4, the number of possible configurations - e.g., of class splits or parameter initializations - may be huge. However, the expected value can be approximated numerically, using a Monte Carlo estimate, calculated as

$$E_x [O_M(x)] \approx \frac{1}{N} \sum_{i=1}^{N} O_M(x_i)$$  \hspace{2cm} (3)$$

where $N$ is the number of open set simulations, and $x_i$ is the $i$th random seed. This estimate will almost surely converge to the expected score for $N \to \infty$. For $N = 5$, this approach is equivalent to the standard protocol of calculating the average
score over five open set simulations. While this change of perspective might seem trivial, we argue that this interpretation enables the utilization of extensions of Monte Carlo methods, for example, different sampling strategies.

5.2 Convergence

In the following, we aim to provide some hints for when such a “sufficient confidence level” may be reached for the performance estimate. [Bouthillier et al., 2019] proposed to run trials over a large number of random seeds and test the results for statistical significance. For each method and dataset, we determined the number of open set simulations required before a two-sided Welch’s t-test (as used for a similar purpose by [Hassen and Chan, 2020]) with a level of significance of 0.05 would indicate a significant difference between the AUROC of the two methods. The results are depicted in Figure 5.

We only observe six cases in which the evidence provided by five or fewer open set simulations was sufficient to infer a significant performance difference. For several setups, even 1000 simulations were insufficient. Comparing these results to Figure 3, we note that for experiments where several methods have a high probability of winning, the number of trials required for a significant result is increased. Unsurprisingly, this demonstrates the effectiveness of statistical tests in preventing drawing conclusions from random outcomes.

5.3 Further Countermeasures

Our results indicate significant divergence of the OOD performance between different evaluation protocols (e.g., unknown classes from the same dataset, unrelated datasets, or synthetic samples). Subjecting methods to rigorous tests covering different scenarios should increase the overall robustness of conclusions against fluctuations in quantitative results. Since we also observe significant performance variance within different test scenarios, using a Monte Carlo approach to estimate the expected performance and testing results for statistical significance seems reasonable for other evaluation protocols as well. Generally, our findings illustrate limitations regarding the comparability of OOD methods tested in different settings. Establishing standardized test suits comprising various benchmarks could facilitate reproducibility and comparability in OOD research.

6 Conclusion

In this work, we studied the effects of randomness in open set-simulation frameworks, a prevalent evaluation protocol for out-of-distribution detection. We formalized and generalized the protocol, found that it includes several sources of randomness, and examined the effects in a large-scale study. We ran three orders of magnitude more open set simulation than recent publications to create a pool of experimental outcomes, which we used to determine the probability that a method would appear to be the best based on the common practice of averaging the results of trials with five different random seeds. We found that for our setting, the outcomes of open set simulations largely depend on chance. This observation suggests that too few simulations may fail to provide a solid foundation for conclusions. Further experiments indicate that other evaluation protocols might be subject to the same phenomenon.

Based on these observations, we proposed to treat the evaluation of OOD methods in open set simulations as a fundamentally probabilistic process and to estimate the expected value of the performance using a Monte Carlo approach to draw more reliable conclusions. The hypothesis tests we conducted demonstrated that even a considerable number of simulations was in some cases insufficient to infer a statistically significant performance difference between the compared methods.

Future work should further investigate the performance variance of OOD evaluation protocols not based on open set simulations. Such experiments could also include different types of data, like natural language, sound, or video. Studying the influence of different sources of randomness in isolation might help quantify the contribution of individual factors to fluctuations and enable better design of experiments.

We want to emphasize that the goal of this work was not to evaluate the selected methods but rather to demonstrate the brittleness of a current best practice evaluation protocol. Studies with different experimental setups or hyperparameters might come to different conclusions regarding the performance of the evaluated approaches. However, we argue that these experiments might also be subject to inherent randomness and should address it accordingly.
Acknowledgement
We thank all reviewers for their extensive and helpful feedback.

References
[Bendale and Boult, 2016] 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. 3

[Boul et al., 2019] TE Boult, S Cruz, AR Dhamija, M Gunther, J Henrydoss, and WJ Scheirer. Learning and the unknown: Surveying steps toward open world recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 9801–9807, 2019. 1

[Bouthillier et al., 2019] Xavier Bouthillier, César Laurent, and Pascal Vincent. Unreproducible research is reproducible. In International Conference on Machine Learning, pages 725–734, 2019. 1, 6

[Dhamija et al., 2018] Akshay Raj Dhamija, Manuel Günther, and Terrance Boult. Reducing network agnostophobia. In Advances in Neural Information Processing Systems, pages 9157–9168, 2018. 2

[Gal and Ghahramani, 2016] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In International conference on machine learning, pages 1050–1059. PMLR, 2016. 3

[Geng et al., 2020] Chuanxing Geng, Sheng-jun Huang, and Songcan Chen. Recent advances in open set recognition: A survey. IEEE transactions on pattern analysis and machine intelligence, 2020. 1, 2

[Goodfellow et al., 2016] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016. 1

[Guo et al., 2017] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In International Conference on Machine Learning, pages 1321–1330. PMLR, 2017. 3

[Hassen and Chan, 2020] Mehdai Hassen and Philip K Chan. Learning a neural-network-based representation for open set recognition. In Proceedings of the 2020 SIAM International Conference on Data Mining, pages 154–162. SIAM, 2020. 1, 2, 6

[He et al., 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016. 3

[Henderson et al., 2018] Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 32, 2018. 2

[Hendrycks and Gunther, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016. 3

[Hendrycks and Gimpel, 2017] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. International Conference on Learning Representations, 2017. 2, 3, 4

[Hendrycks et al., 2018] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In International Conference on Learning Representations, 2018. 2, 4

[Liang et al., 2018] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In 6th International Conference on Learning Representations, ICLR 2018, 2018. 3

[Loshchilov and Hutter, 2017] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. International Conference on Learning Representations, 2017. 3

[Lucie et al., 2018] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. In Advances in Neural Information Processing Systems, pages 700–709, 2018. 1

[Miller et al., 2021] Dimitry Miller, Niko Sunderhauf, Michael Milford, and Feras Dayoub. Class anchor clustering: A loss for distance-based open set recognition. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 3570–3578, 2021. 1, 2, 4

[Musgrave et al., 2020] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In European Conference on Computer Vision, pages 681–699. Springer, 2020. 1

[Neal et al., 2018] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, Computer Vision – ECCV 2018, pages 620–635, Cham, 2018. Springer International Publishing. 2

[Oza and Patel, 2019] Poojan Oza and Vishal M Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2307–2316, 2019. 1, 2, 3

[Prechelt, 1998] Lutz Prechelt. Early stopping—but when? In Neural Networks: Tricks of the trade, pages 55–69. Springer, 1998. 3

[Ruff et al., 2021] Lukas Ruff, Jacob R Kauffman, Robert A Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. Proceedings of the IEEE, 2021. 1, 3

[Scheirer et al., 2012] Walter J Scheirer, Anderson de Rezende Rocha, Archana Sapkota, and Terrance E Boult. Toward open set recognition. IEEE transactions on pattern analysis and machine intelligence, 35(7):1757–1772, 2012. 1
[Scheirer et al., 2014] Walter J Scheirer, Lalit P Jain, and Terrance E Boult. Probability models for open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2317–2324, 2014.

[Srivastava et al., 2014] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[Vapnik, 1992] Vladimir Vapnik. Principles of risk minimization for learning theory. In *Advances in Neural Information Processing Systems*, pages 831–838, 1992.

[Zhang et al., 2020] Hongjie Zhang, Ang Li, Jie Guo, and Yanwen Guo. Hybrid models for open set recognition. In *European Conference on Computer Vision*, pages 102–117. Springer, 2020.