Generative Methods for Detection of Out-of-Distribution (OOD) Data

Jacob Rast
jrast@andrew.cmu.edu

Abstract

Machine learning and deep learning methods have rapidly advanced in the past decade, achieving superhuman performance on many common datasets and benchmarks. However, there is a growing consensus that the performance of these models degrades when deployed. A critical issue is that of out-of-distribution inputs. Once deployed, models are exposed to data that may differ significantly from that upon which it was trained. Models exposed to OOD inputs may produce meaningless outputs with high confidence. This poses a significant safety risk in domains such as robotics or healthcare where erroneous behavior may endanger human life. In this work, several methods for the detection of out-of-distribution inputs are discussed. Three different generative models are analyzed for use as standalone out-of-distribution detectors. Specifically, energy-based models, denoising diffusion models, and generative adversarial models are analyzed for use as an auxiliary complement OOD detection network working in conjunction with an image classifier.

1 Introduction

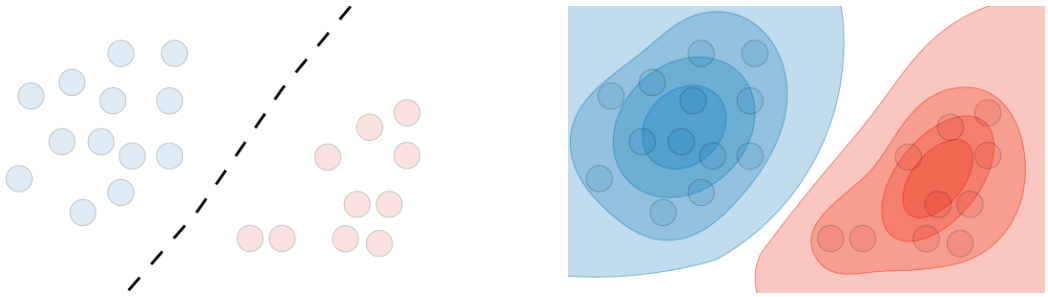
Generative models are a major class of machine learning algorithms. Given a dataset x and a set of labels or values to be predicted y, generative models learn to model the distribution from which the data was drawn $P(x|y)$. This is in contrast to alternative paradigms (eg discriminative models) which seek to directly model the output quantity $P(y|x)$ given dataset. A visual representation of these classifiers can be found in figure 1.

The ability to model the distribution from which a dataset was drawn makes generative models an ideal fit for the problem of out-of-distribution (OOD) data detection. An emerging field of research, OOD detection is the problem of detecting inputs to a machine learning model that differ significantly from those upon which the model was trained. OOD inputs can cause issues in highly brittle systems. It has been observed that machine learning models receiving OOD input can output a nonsense prediction at very high confidence levels. The addition of an OOD detector to these brittle systems can ameliorate the issue and act as an important safety feature for machine learning models deployed in safety-critical applications.

Out-of-distribution detection is related to problems such as distributional shift detection and outlier detection.

Formally, OOD detection is a classification problem. An OOD detector is a binary classifier trained on only positive examples. The classifier can discriminate between data points drawn from the same distribution as the training set and data points drawn from an alternative distribution.

10707 Final Report (Spring 2023).



(a) A discriminative classifier

(b) A generative classifier

Figure 1: Example of models trained for classification *CS229* 2023

2 Background

2.1 Terminology

The methodology of this report is to select one dataset to train and validate a model and OOD detector and evaluate the performance of that trained OOD detector using validation inputs from the same dataset (class label 1) and validation inputs from all other datasets (class label 0). The terminology ID is used to refer to the dataset used for training, and OOD for datasets not used for training. For example, to determine whether an input is a hand-written digit or not we would train on MNIST ID and evaluate on FMNIST OOD, CIFAR10 OOD, CIFAR100 OOD, ImageWoof OOD, and Imagenette OOD. To determine whether an input belongs to ImageWoof classes or not, we train a model and OOD detector on ImageWoof ID and evaluate on MNIST OOD, FMNIST OOD, CIFAR10 OOD, CIFAR100 OOD, and Imagenette OOD.

ID or OOD is used only to distinguish the use of this dataset on the benchmark task and does not denote different datasets. MNSIT ID and MNIST OOD are the same dataset used for either training and class 1 labels or class 0 labels. Train splits are used for training and test splits are used for testing.

2.2 Metrics and datasets

One common misunderstanding related to OOD detector development is the selection of a classifier threshold. In the literature, it is common to report several metrics such as AUROC, AUPR, and FPR95. For the AUROC and AUPR metrics, all possible threshold candidates are considered and the effect on select items from the confusion matrix is summarized in a single statistic. FPR95 will select a threshold such that 5% of training data is misclassified as out of distribution.

A full discussion of the metrics and datasets used in this work can be found in the appendix section.

2.3 Baseline Method

A simple baseline found in Hendrycks and Gimpel 2017 was proposed in the midway report. These authors use the model confidence (maximum value of softmax of network output) as an OOD score. The results of this method can be seen in tables 1 through 4 below. In short, this method works well for inputs which the model performs well on. However, for difficult-to-classify examples and datasets, this methodology falls short. For this reason, performance is poor across all metrics when distinguishing between CIFAR10, CIFAR100, ImageWoof, and Imagenette in-distribution inputs from out-of-distribution inputs, both near-OOD and far-OOD. The performance is directly related to classifier performance, as can be seen by the worse performance of ImageWoof (hard to classify examples) ID vs Imagenette (hard to classify examples) ID.

On the other hand, the performance of the baseline does relatively well in distinguishing ID from near-OOD such as CIFAR10 ID from CIFAR100 OOD (0.756 AUC) or MNIST ID from FashionMnist OOD (0.929 AUC).

3 Related Work

Generative models were first proposed for OOD detection in Bishop 1994. A large number of approaches for OOD detection have been proposed since. In Hendrycks and Gimpel 2017 the use of model confidence as the baseline for OOD detection was proposed. Liang, Li, and Srikant 2017 proposed ODIN, which adds perturbation to images in order to make softmax score a more reliable OOD detection method. Continuing this line of research, in 2021 DICE (Sun and Li 2021) sought to use not just the softmax output of the network for OOD detection but any arbitrary activation output in a neural network. Research into generative models discovered some surprising results. Nalisnick et al. 2018 showed that generative models may even assign a higher probability to OOD data points than the training set, despite never producing samples from these distributions. Since this work, a flurry of research has been undertaken seeking to understand and correct the behavior of generative models. Notably, Morningstar et al. 2021 created an approach from statistical physics using multiple summary statistics, rather than simple likelihood, from models in order to perform OOD detection. Another promising approach is from Ren et al. 2019 who observed that the high likelihood of OOD image data is dominated by similar backgrounds. To adjust for this effect, they proposed a likelihood ratio test decomposing model probabilities into foreground and background pixels.

4 Methods

4.1 Energy Model

The computational method forming the backbone of the energy model is the Restricted Boltzmann Machine (RBM) or Deep Belief Network (DBN). As these methods were discussed in detail in 10707 lecture and homework they will not be explained here. In short, the non-normalized free energy of an input given trained hidden units is used as an OOD score. A longer discussion of this concept can be found in the appendix.

4.2 GAN Discriminator

A Generative Adversarial Model or GAN is the coupling of two machine learning models for unsupervised training. A generator is used to create plausible image outputs while a discriminator model seeks to differentiate generated images from images training data.

Once a GAN is trained to convergence, the generator can create data that is plausibly drawn from the same distribution as the training data, while the discriminator has learned to differentiate between in-distribution and out-of-distribution data.

This quality makes the use of the discriminator as a standalone model for OOD detection attractive. A standard implementation of a DCGAN (Radford, Metz, and Chintala 2015) was used with no additional modification as an OOD detector.

4.3 Diffusion Model

Diffusion models have recently seen great progress in the generation of images, including photo-realistic and prompt-based image generation (Song et al. 2020). Given both the strong performance of the model and the well-documented fact that diffusion models can perfectly reproduce (or leak) images on which they were trained (Carlini et al. 2023), this method seems an ideal choice for the detection of very near OOD samples, for example CIFAR10 vs CIFAR100 or Imagenette from ImageWoof.

For this method, the use of a denoising diffusion probabilistic model (DDPM) was used as the backbone for a reconstruction-based OOD detector.

Justification for this approach can be seen in figure 2. Investigation into a reconstruction-based classification score was undertaken. An input image was first corrupted using random noise, following the noising process used in the training of diffusion models. The noisy or corrupted image is then reconstructed back to its original form. However, it was hypothesized that the success of the reconstruction would depend upon whether or not the DDPM had been trained on a similar dataset including the input image class. For example, a DDPM trained exclusively on celebrity faces may be

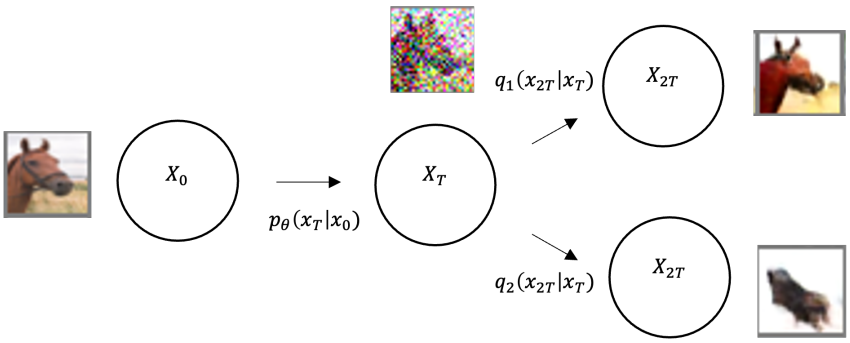


Figure 2: Theoretical justification for use of a diffusion model for OOD detection. $q1$ is a denoising diffusion model trained on a dataset including images of the same class as the input, whereas $q2$ is a diffusion model trained on a dataset not including the input class.

able to fully reconstruct any arbitrary face from a heavily noised image, while it would be unable to reconstruct an image of a cow.

This theory of reconstruction loss is shown in figure 2. We hypothesized that a DDPM trained exclusively on images of horses would be able to reconstruct a horse image, but a DDPM trained exclusively on cows would be unable to do so, and output an image of a horse given a significantly noised input image.

5 Results

Full results for all methods are shown in the tables 1 through 4 below.

5.1 Energy Model

The use of energy models in discriminating ID from OOD inputs was verified on simple tasks such as MNIST ID or Fashion-MNIST ID. On MNIST, energy models perform nearly perfectly. They outperform the baseline, achieve a perfect 1 AUROC, AUPR, and F1 score and misclassify OOD inputs as ID at a rate of only 0.04%.

The energy-based OOD detector trained on Fashion MNIST ID could reliably detect far-OOD inputs (CIFAR10, CIFAR100, ImageWoof, Imagenette) but struggled to detect near-OOD inputs (MNIST). This may reflect the higher complexity of Fashion MNIST images than MNIST images.

The performance of the energy model OOD detector is directly related to the ability of the model to capture the dataset. For complex image tasks, the model did not have sufficient complexity to accurately capture input images. In this case, the model loss remained high, and OOD detection ability was poor. For example, CIFAR10 ID, CIFAR100 ID, ImageWoof ID, and ImageNette ID energy models all performed no better than chance at detecting near-OOD inputs (AUROC 0.5). They did retain some ability to detect far-OOD inputs (AUROC 0.6 to 0.8).

5.2 GAN Discriminator

The GAN discriminator had the best overall performance of any method attempted. It showed a strong ability to discriminate simple data from both near and far-OOD inputs, achieving at or near best-in-class performance for Fashion MNIST ID (all AUROC above 0.9) and strong performance for MNIST ID (perfect AUROC of 1.0 on many tasks, all above 0.9).

Where the GAN model truly outperformed other metrics were in distinguishing complex image ID datasets from near-OOD inputs. For example, the GAN model can perfectly classify CIFAR 10 ID from near-OOD Imagnette and ImageWoof datasets and far-OOD MNIST and Fashion MNIST datasets while the baseline and energy models perform little better than random chance.

The GAN could do well at distinguishing ID from very-near OOD inputs. This can be seen on Imagenette ID from ImageWoof OOD (AUROC 0.811) and Imagenette ID from CIFAR10 OOD and CIFAR100 OOD (AUROC 0.862, AUROC 0.617).

One area where the GAN fell short is when the model fails to converge. For example, the ImageWoof ID GAN performs much worse than random chance at nearly all OOD tasks. The difficulty of training a GAN is well discussed, and the result is therefore not surprising. Additionally, ImageWoof is a dataset curated to be difficult for a model to learn so the failure of the GAN to learn this distribution is reasonable.

5.3 Diffusion Model

Despite best efforts, diffusion model reconstruction-based OOD detection performed no better than chance on all datasets. Full results are therefore not reported. An analysis of this model can be found in the discussion and analysis section.

6 Discussion and Analysis

6.1 Energy models

Energy models proved to be a compelling solution to OOD detector development which achieved near-perfect performance on simple tasks. It stands to reason that the only difficulty in scaling up energy-based OOD detection to more complex datasets is increasing the complexity of the model to allow it to capture more complex distributions.

Results in the literature have long shown that deeper energy-based networks can better capture more complex distributions. Lee et al. 2009 demonstrated that unsupervised training of deep belief networks allows for networks to capture higher-order features of the data distribution. This gives theoretical justification for the use of a Deep Belief Network for OOD detection, which was not explored in this project.

Another potential direction for future investigation is the use of deep belief networks in an autoencoder configuration for reconstruction-based OOD detection. Salakhutdinv and Hinton showed in 2009 that DBN autoencoders can capture richer features for reconstruction than a corresponding PCA approach (Hinton and Salakhutdinov 2007).

Although these approaches are not new, generative models are well understood and come with strong theoretical guarantees. The ability of deep belief networks to train on unlabeled data makes them further attractive for re-purposing as OOD detectors.

6.2 GAN Discriminator

The surprisingly effective performance of a GAN model begs for further investigation. This model surpassed the baseline on challenging tasks without the need for any modifications. There are several attractive and straightforward avenues for future works, such as WGANS (Arjovsky, Chintala, and Bottou 2017) or the use of contrastive learning (Yu et al. 2021) (Kang and Park 2020).

6.3 Diffusion Model

In this project, diffusion models were used in a reconstruction configuration. It was hypothesized that OOD inputs would be reconstructed to appear similar to ID classes and therefore suffer heavy distortion, whereas ID inputs would be well reconstructed.

It was discovered that for moderately distorted images, both ID and OOD inputs could be denoised effectively by the same network. If the images were heavily distorted, conditional reconstruction would cause both ID and OOD inputs to regenerate into the class being conditioned on, yet both would create a new image from the class. In this case, MSE was high for both ID and OD images.

While it is quite surprising that denoising diffusion models are able to successfully denoise both ID and OOD images, there are some reports in the literature showing similar properties for other generative models. For example, the PixelCNN++ model will assign a high likelihood to OOD inputs despite never generating these images in practice (Nalisnick et al. 2018).

For future works, we plan to investigate alternative diffusion models which have shown the ability to regenerate the input from distorted images such as cold diffusion models (Bansal et al. 2022).

Full examples of diffusion model reconstruction for OD and ID images can be found in the appendix.

7 Acknowledgements

Thank you to Joao Monteiro for introducing me to the interesting problem. Thank you to Prof. Russ Salakhutdinov, his student Brandon Trabucco, and TA mentor Albert Liang for their help on this project.

References

Arjovsky, Martin, Soumith Chintala, and Léon Bottou (2017). "Wasserstein GAN". In: eprint: arXiv:1701.07875.

Bansal, Arpit et al. (2022). "Cold diffusion: Inverting arbitrary image transforms without noise". In: eprint: arXiv:2208.09392.

Bishop, Christopher M (1994). "Novelty detection and neural network validation". In: *IEE Proceedings-Vision, Image and Signal processing* 141.4, pp. 217–222.

Carlini, Nicholas et al. (2023). "Extracting training data from diffusion models". In: eprint: arXiv:2301.13188.

CS229 (2023). URL: https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-supervised-learning.

Hendrycks, Dan and Kevin Gimpel (2017). "A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks". In: eprint: arXiv:1610.02136.

Hinton, Geoffrey E and Russ R Salakhutdinov (2007). "Using Deep Belief Nets to Learn Covariance Kernels for Gaussian Processes". In: *Advances in Neural Information Processing Systems*. Ed. by J. Platt et al. Vol. 20. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2007/file/4b6538a44a1dfdc2b83477cd76dee98e-Paper.pdf.

Kang, Minguk and Jaesik Park (2020). "ContraGAN: Contrastive learning for conditional image generation". In: eprint: arXiv:2006.12681.

Lee, Honglak et al. (June 2009). "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations". In: *Proceedings of the 26th Annual International Conference on Machine Learning*. Montreal Quebec Canada: ACM.

Liang, Shiyu, Yixuan Li, and R Srikant (2017). "Enhancing the reliability of out-of-distribution image detection in neural networks". In: eprint: arXiv:1706.02690.

Morningstar, Warren et al. (2021). "Density of states estimation for out of distribution detection". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 3232–3240.

Nalisnick, Eric et al. (2018). "Do deep generative models know what they don't know?" In: *arXiv preprint arXiv:1810.09136*.

Radford, Alec, Luke Metz, and Soumith Chintala (2015). "Unsupervised representation learning with deep convolutional generative adversarial networks". In: eprint: arXiv:1511.06434.

Ren, Jie et al. (2019). *Likelihood Ratios for Out-of-Distribution Detection*. eprint: arXiv:1906.02845.

Song, Yang et al. (2020). "Score-based generative modeling through stochastic differential equations". In: eprint: arXiv:2011.13456.

Sun, Yiyou and Yixuan Li (2021). "DICE: Leveraging sparsification for out-of-distribution detection". In: eprint: arXiv:2111.09805.

Yu, Ning et al. (2021). "Dual Contrastive Loss and Attention for GANs". In: eprint: arXiv:2103.16748.

	Method	MNIST	Fashion MNIST	CIFAR10	CIFAR100	ImageWoof	Imagenette
MNIST OOD	Energy	NA	0.568	0.713	0.835	0.710	0.564
	GAN	NA	0.925	1.000	1.000	0.138	0.811
	Baseline	NA	0.929	0.803	0.751	0.573	0.490
FMNIST OOD	Energy	1.000	NA	0.652	0.708	0.674	0.638
	GAN	0.993	NA	1.000	0.986	0.371	0.500
	Baseline	0.980	NA	0.804	0.660	0.574	0.473
CIFAR10 OOD	Energy	1.000	0.983	NA	0.503	0.500	0.504
	GAN	1.000	0.922	NA	0.646	0.651	0.862
	Baseline	0.984	0.918	NA	0.623	0.504	0.490
CIFAR100 OOD	Energy	1.000	0.961	0.499	NA	0.493	0.500
	GAN	1.000	0.921	0.503	NA	0.594	0.617
	Baseline	0.984	0.912	0.756	NA	0.498	0.503
ImageWoof OOD	Energy	1.000	0.983	0.500	0.503	NA	0.504
	GAN	1.000	0.922	1.000	0.974	NA	0.821
	Baseline	0.983	0.905	0.792	0.682	NA	0.510
Imagenette OOD	Energy	1.000	0.961	0.499	0.500	0.493	NA
	GAN	1.000	0.913	1.000	0.932	0.494	NA
	Baseline	0.983	0.906	0.788	0.668	0.505	NA

Table 1: AUROC (higher is better)

	Method	MNIST	Fashion MNIST	CIFAR10	CIFAR100	ImageWoof	Imagenette
MNIST OOD	Energy	NA	0.689	0.759	0.876	0.768	0.671
	GAN	NA	0.987	1.000	1.000	0.167	0.643
	Baseline	NA	0.940	0.483	0.782	0.358	0.272
FMNIST OOD	Energy	1.000	NA	0.689	0.749	0.711	0.712
	GAN	0.989	NA	1.000	0.984	0.224	0.274
	Baseline	0.935	NA	0.486	0.323	0.357	0.270
CIFAR10 OOD	Energy	1.000	0.981	NA	0.533	0.500	0.533
	GAN	1.000	0.931	NA	0.620	0.420	0.729
	Baseline	0.986	0.927	NA	0.654	0.285	0.272
CIFAR100 OOD	Energy	1.000	0.952	0.480	NA	0.480	0.500
	GAN	1.000	0.926	0.500	NA	0.357	0.745
	Baseline	0.986	0.920	0.671	NA	0.282	0.280
ImageWoof OOD	Energy	1.000	0.981	0.500	0.533	NA	0.533
	GAN	1.000	0.968	1.000	0.985	NA	0.808
	Baseline	0.994	0.961	0.854	0.852	NA	0.501
Imagenette OOD	Energy	1.000	0.952	0.480	0.500	0.480	NA
	GAN	1.000	0.964	1.000	0.959	0.486	NA
	Baseline	0.994	0.962	0.846	0.839	0.506	NA

Table 2: AUPR (higher is better)

ID	Method	MNIST	Fashion MNIST	CIFAR10	CIFAR100	ImageWoof	Imagenette
MNIST OOD	Energy	NA	0.667	0.676	0.771	0.669	0.667
	GAN	NA	0.937	1.000	0.997	0.440	0.608
	Baseline	NA	0.847	0.468	0.698	0.509	0.440
FMNIST OOD	Energy	0.996	NA	0.685	0.689	0.681	0.667
	GAN	0.968	NA	1.000	0.945	0.440	0.443
	Baseline	0.870	NA	0.470	0.329	0.511	0.440
CIFAR10 OOD	Energy	1.000	0.939	NA	0.667	0.667	0.667
	GAN	1.000	0.845	NA	0.680	0.481	0.680
	Baseline	0.941	0.834	NA	0.667	0.441	0.440
CIFAR100 OOD	Energy	1.000	0.912	0.667	NA	0.667	0.457
	GAN	1.000	0.844	0.667	NA	0.448	0.692
	Baseline	0.943	0.827	0.594	NA	0.440	0.440
ImageWoof OOD	Energy	1.000	0.939	0.667	0.667	NA	0.667
	GAN	1.000	0.903	1.000	0.962	NA	0.761
	Baseline	0.959	0.889	0.751	0.836	NA	0.666
Imagenette OOD	Energy	1.000	0.912	0.667	0.667	0.667	NA
	GAN	1.000	0.897	1.000	0.933	0.668	NA
	Baseline	0.960	0.891	0.749	0.836	0.667	NA

Table 3: F1 Score (higher is better)

	Method	MNIST	Fashion MNIST	CIFAR10	CIFAR100	ImageWoof	Imagenette
MNIST OOD	Energy	NA	0.500	0.500	0.465	0.500	0.500
	GAN	NA	0.268	0.000	0.003	0.500	0.337
	Baseline	NA	0.230	0.411	0.433	0.500	0.500
FMNIST OOD	Energy	0.004	NA	0.438	0.434	0.445	0.497
	GAN	0.033	NA	0.000	0.056	0.500	0.485
	Baseline	0.074	NA	0.410	0.471	0.500	0.499
CIFAR10 OOD	Energy	0.000	0.062	NA	0.500	0.500	0.500
	GAN	0.000	0.240	NA	0.451	0.467	0.302
	Baseline	0.063	0.243	NA	0.483	0.496	0.500
CIFAR100 OOD	Energy	0.000	0.091	0.489	NA	0.497	0.500
	GAN	0.000	0.233	0.499	NA	0.481	0.292
	Baseline	0.062	0.250	0.442	NA	0.494	0.500
ImageWoof OOD	Energy	0.000	0.062	0.500	0.500	NA	0.500
	GAN	1.000	0.231	0.000	0.070	NA	0.334
	Baseline	0.064	0.266	0.426	0.462	NA	0.498
Imagenette OOD	Energy	0.000	0.091	0.489	0.500	0.494	NA
	GAN	0.000	0.253	0.000	0.143	0.495	NA
	Baseline	0.065	0.261	0.420	0.464	0.498	NA

Table 4: False Positive Rate @ 95% TPR (lower is better)

8 Appendix

8.1 Model theory and background

8.1.1 Confronting the partition function

A bottleneck to the direct application of energy-based models to OOD detectection is the calculation of the partition function. For initial investigations into this line of research a toy problem for which the energy function can be brute-force calculated was investigated. A 300 hidden unit RBM was trained on a dataset of 8 possible 16 pixel images representing 0 or 1 for 10,000 epochs. From this, partition function can be calculated by summing the energy of the learned hidden units for all 2^{16} possible binary configurations. The top most probable configurations for this problem can be seen in figure 3. It can be observed that highly plausible 0 and 1 images dominate the top results.

Noting that the partition function is independent of the visible units in RBMs and DBMs, the energy of a visible configuration given learned hidden units can be directly used as a scoring function for discrimination of OOD inputs. For an RBM the log energy can be calculated as given in equation 1, which can be immediately recognized as the non-normalized free energy.

$$\textit{score}(x) = \exp(c^T x + \sum_{j=1}^{H} \log(1 + exp(b_j + W_j x))) \tag{1}$$

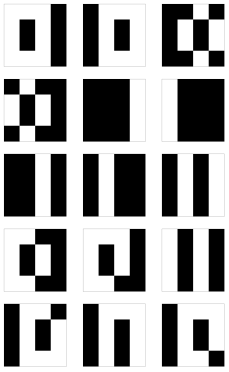


Figure 3: Most likely visible unit configurations learned from a 300 hidden unit Restricted Boltzmann Machine trained on 4x4 images of zero and one

8.2 Metrics

Several methods are used in this work to analyze the performance of a binary OOD classifier. These are briefly discussed below.

8.2.1 Area Under ROC Curve

The ROC or receiver operating characteristic curve is a graph which predicates true positive rate (TPR) vs false positive rate (FPR) of a binary classifier. Each point in the positive and negative class is given a score or probability. The TPR and FPR of the labled at every possible score threshold is computed and plotted on a curve. Note that a trivial classifier would have a TPR of 0 when the FPR is 0 (corresponding to always predicting the negative class) and a TPR of 1 when the FPR is 1 (corresponding to always predicting the positive class). A perfect classifier would have a TPR of 1 when the FPR is 0 (always predicting the correct class) and likewise a TPR of 1 when the FPR is 1 (if the threshold is set such that all points are given a positive label). This means that the AUROC score can range from 0.5 (random guessing) to 1 (perfect classification). Any score below 0.5 is worse than random and could theoretically be improved by reversing the classification decision.

An attractive property of the AUROC metric is that it handles class imbalance well. A trivial classifier (always predicting positive) would have 99% accuracy on an imbalanced dataset composed of 99% positive class, but would still receive an AUROC of 0.5.

8.2.2 Area Under Precision-Recall Curve

The area under precision-recall curve (AUPR) is similar to the AUROC curve yet is more informative when positive class prediction is more important than negative class prediction. Precision is the accuracy of positive predictions, ie how many predicted positive classes are truly positive. Recall measures how comprehensive a positive prediction is, ie how many of the true positives were predicted to be positive. Much like AUROC, these quantities are measured at all possible threshold values. Again, similar to AUROC trivial classifiers (always predicting 0 or always predicting 1) define the bounds of the AUPR curve. However, note that the baseline for a trivial classifier is no longer 0.5, but is now the fraction of positive items in the dataset.

AUPR can be harder to interpret than AUROC as a given score may correspond good or bad performance depending on the dataset. However, it has the property of only measuring the ability of the model to predict a positive class. Both precision and recall are defined in terms of the positive class alone. This makes it desirable for problems such as OOD detection where performance on the positive class may be more important than that on the negative class.

8.2.3 F1 Score

The F1 score is the harmonic mean of the precision and recall. It is mathematically defined as follows:

$$F_1 = \frac{precision * recall}{precision + recall}$$

The F1 scores ranges from 0 (worst) to 1 (best). It requires the selection of a classification threshold, which can be performed automatically by software packages.

F1 scores also have the desirable property of detecting class imbalance due to the harmonic mean. A trivial classifier would be given a F1 score of 0 on an imbalanced dataset.

8.2.4 False Positive Rate at 95% True Positive Rate

For this metric, a classification threshold is set such that 5% of the training data is labeled as positive (OOD) and 95% labeled as negative (ID). This threshold is then used to detrmine the false positive rate on the test data.

8.3 Datasets

In this project, a number of datasets were used to detect OOD inputs. One dataset is used to train a classifier and the ability of the OOD detector to discriminate between the validation set of this dataset against all other datasets was evaluated in a pairwise fashion.

8.3.1 MNIST

A very common dataset of 28 x 28 pixel images of grey-scale handwritten digits. Empirically, MNIST is similar (near OOD) to Fashion MNIST and relatively dissimilar (far OOD) to CIFAR10, CIFAR100, Imagenette and Imagewoof.

8.3.2 Fashion MNIST

An alternative to MNIST, Fashion MNIST is a dataset of 28x28 pixel grey-scale images of common clothing items. Empirically, Fashion MNIST is similar (near OOD) to MNIST and relatively dissimilar (far OOD) to CIFAR10, CIFAR100, Imagenette and Imagewoof.

8.3.3 CIFAR10

CIFAR10 is a datat of 32x32 pixel color images drawn from 10 classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. This is a common dataset for image classification tasks. Empirically, CIFAR10 is very similar to CIFAR100 (very near OOD), similar to Imagenette and Imagewoof (near OOD) and relatively dissimilar to MNIST and Fashion MNIST (far OOD).

8.3.4 CIFAR100

CIFAR10 is a datat of 32x32 pixel color images drawn from 100 classes. Some research has shown [0.48550/arXiv.1902.00423] that CIFAR100 contains a large percentage of duplicates or near-duplicates. It is not know if any images are shared between CIFAR10 and CIFAR100, but there is not class overlap. Both CIFAR10 and CIFAR100 are subsets of the TinyImage database. Empirically, CIFAR100 is very similar to CIFAR10 (very near OOD), similar to Imagenette and Imagewoof (near OOD) and relatively dissimilar to MNIST and Fashion MNIST (far OOD).

8.3.5 Imagenette

Imagenette is a subset of 10 easy-to-classify classes from the Imagenet data set. It contains images of class tench, English springer, cassette player, chain saw, church, French horn, garbage truck, gas pump, golf ball, parachute. The dataset is color 160x60 pixel images. It is relatively small with 9,460 images. Imagenette and Imagewoof are both subsets of the Imagenet dataset. Empirically, Imagenette is very similar to Imagewoof (very near OOD), similar to CIFAR10 and CIFAR100 (neary OOD), and relatively dissimilar to MNIST and Fashion MNIST (far OOD).

8.3.6 Imagewoof

Imagewoof is a subset of 10 hard-to-classify classes from the Imagenet data set. It contains images of class Australian terrier, Border terrier, Samoyed, Beagle, Shih-Tzu, English foxhound, Rhodesian ridgeback, Dingo, Golden retriever, Old English sheepdog. Like Imagenette, the dataset is color 160x60 pixel images. and relatively small with 9,030 images. Empirically, Imagenette is very similar to Imagewoof (very near OOD), similar to CIFAR10 and CIFAR100 (neary OOD), and relatively dissimilar to MNIST and Fashion MNIST (far OOD).

8.4 GAN generated images

Below are images generated from the GAN generator used to assess convergence of the model.

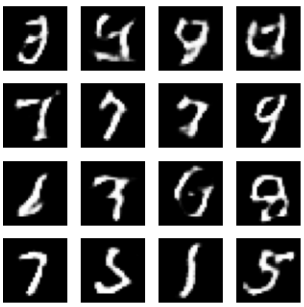


Figure 4: GAN generated images after training on MNIST

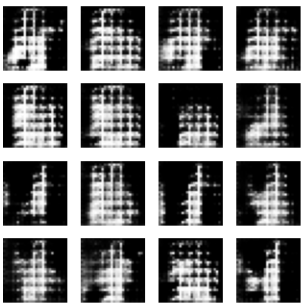


Figure 5: GAN generated images after training on Fashion MNIST

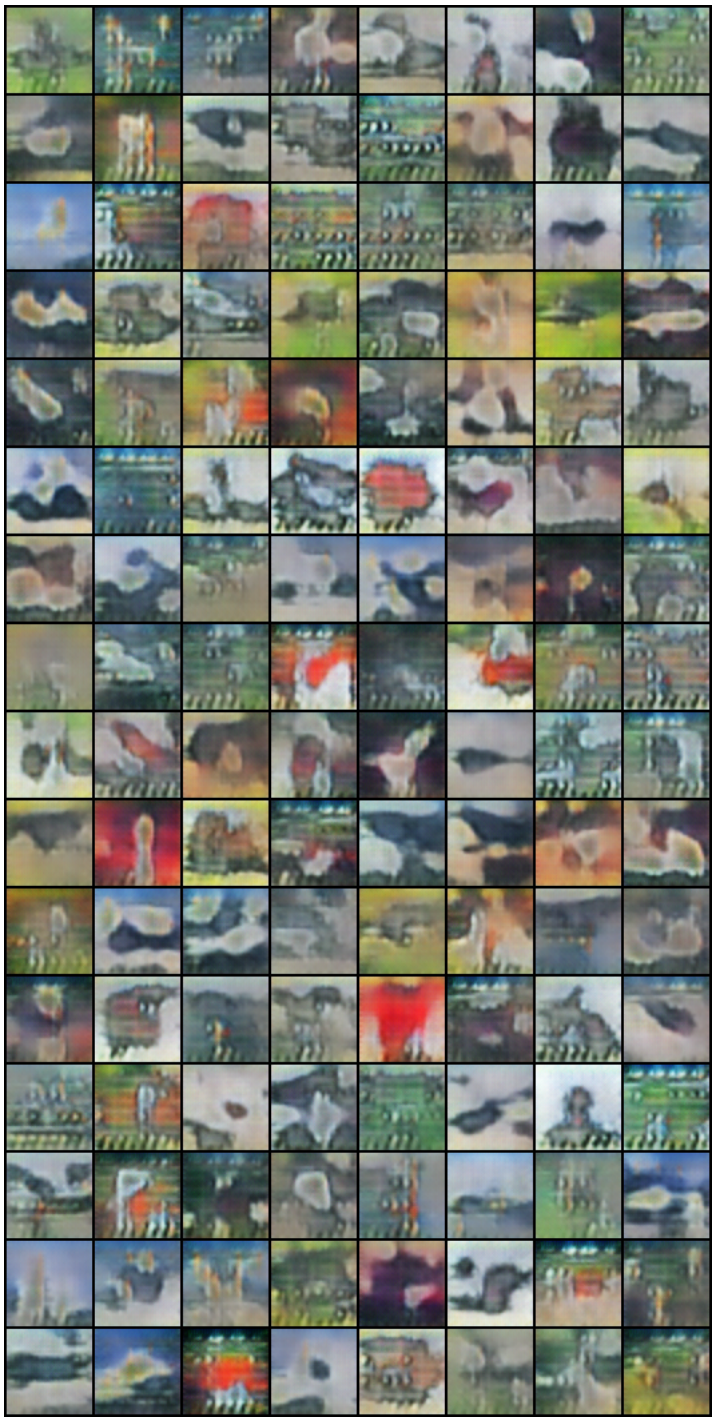


Figure 6: GAN generated images after training on CIFAR10

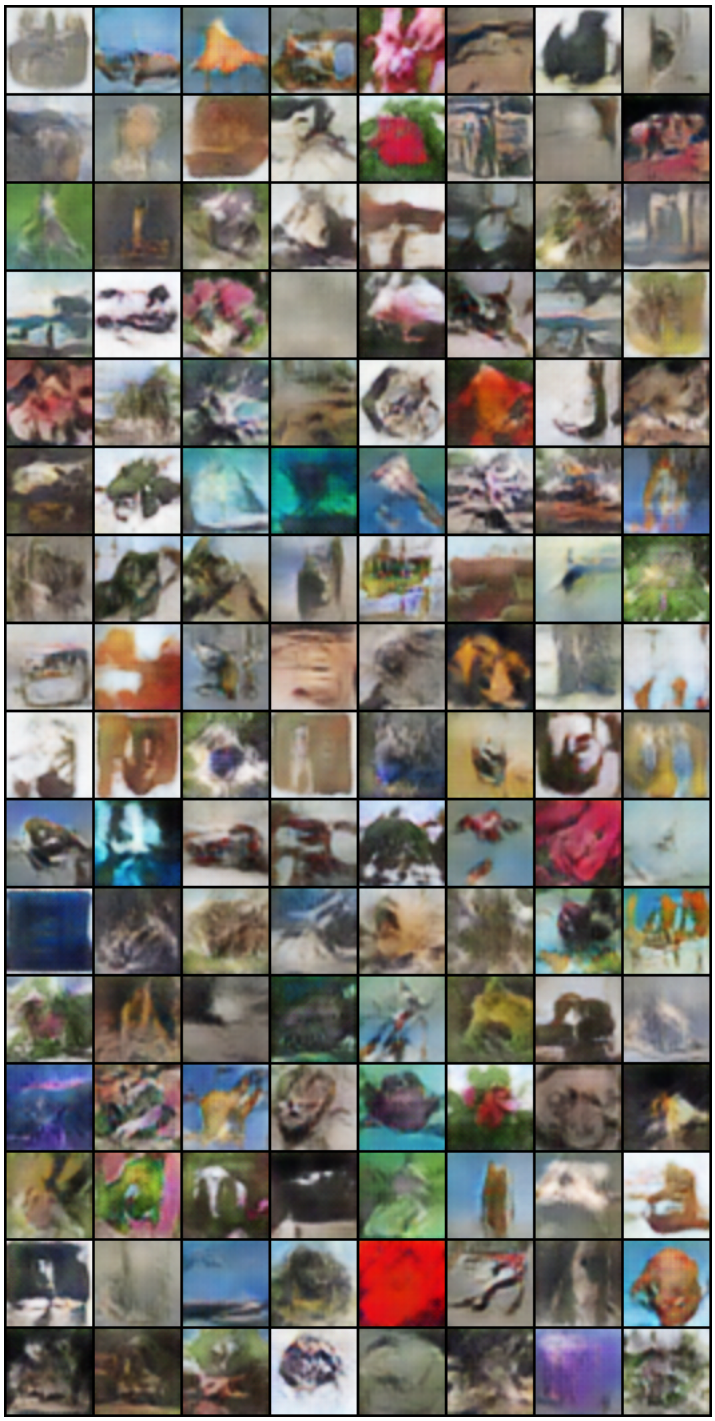


Figure 7: GAN generated images after training on CIFAR100

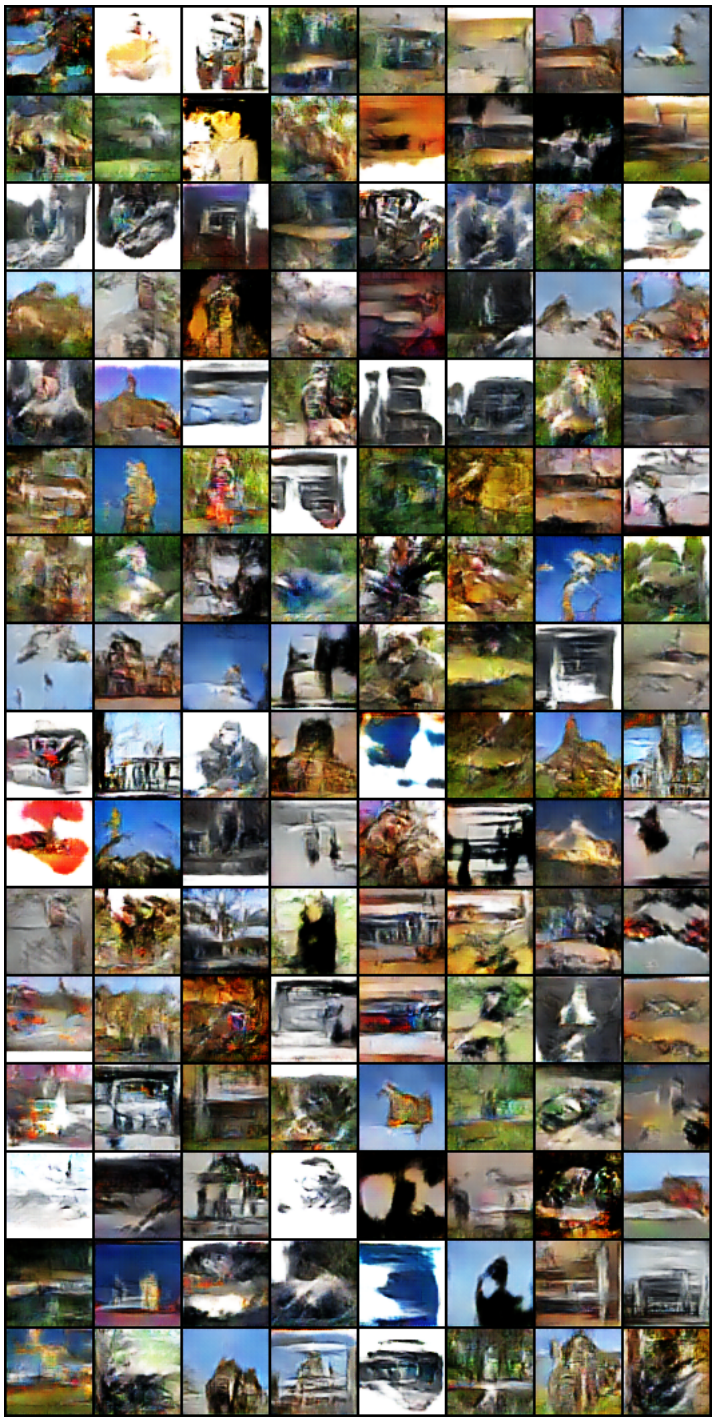


Figure 8: GAN generated images after training on Imagenette

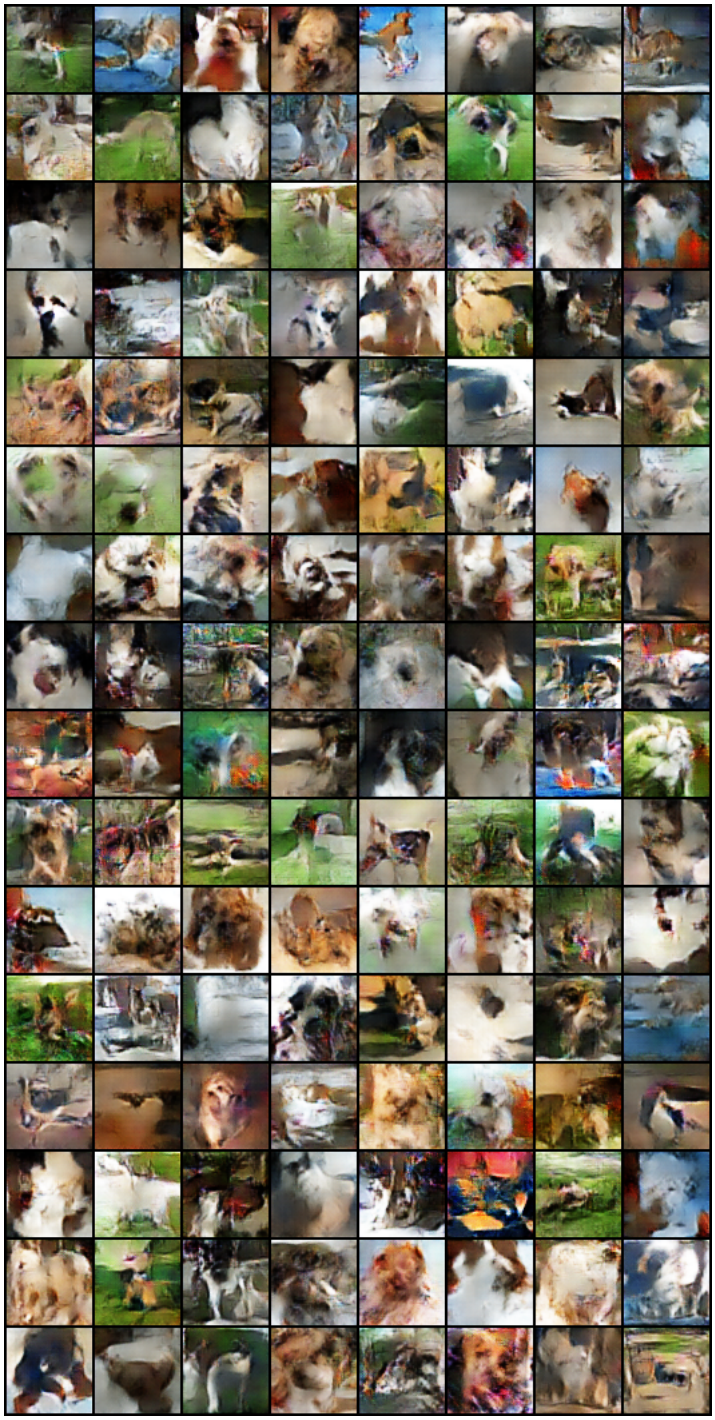


Figure 9: GAN generated images after training on Imagewoof

8.5 Diffusion model conditional generation

Figure 10: DDPM Reconstruction of OOD input with varying degree of distortion

Figure 11: DDPM Reconstruction of ID input with varying degree of distortion