A Replication of "Zero-Shot Anomaly Detection without Foundation Models", (Arxiv, 2023)

Serafim Alex-Mihai[1]*
Faculty of Mathematics and Computer Science
University of Bucharest
alex.serafim@s.unibuc.ro

Filipescu Radu[2]†
Faculty of Mathematics and Computer Science
University of Bucharest
radu.filipescu@s.unibuc.ro

Abstract

A Zero-Shot anomaly detection model is a machine-learning model which can detect anomalies in a data set without having been trained beforehand on training data having the same structure or data distribution as the testing data. The authors of this paper claim to provide an approach to create lightweight, easy to implement and use models, as opposed to using foundation models.

1 Methods

In this paper we will be trying to replicate the methods used and the results obtained by Aodong Li, Chen Qiu, Marius Kloft, Padhraic Smyth, Maja Rudolph and Stephan Mandt in their study **"Zero-Shot Anomaly Detection without Foundation Models"** [1], published in 2023 in Arxiv.

The core concept and result of this paper is the Adaptive Centered Representations (ACR) model, claimed to be a lightweight, adaptive, zero-shot ML model, being trained from different samples from a *meta-distribution*, namely a conceptual set from which different data sets having the same distribution can be sampled. The paper describes two implementations of ACR, one based on Deep SVDD, where the anomaly score is calculated based on the distance to the center of the hypersphere in the feature space, and one based on a binary classifier, where the anomaly score is calculated based on the distance to the center of the distribution

The authors compare the implemented model with other proposed solutions to anomaly detection and classify those alternative solutions by the approach they use. The solutions that used a few-shot approach are not expected to perform well when used in a zero-shot context, and foundation models, meaning very large models, trained on vast quantities of data, are most often proprietary and costly, and are bound to the types of data they have been trained on.

2 Implementation

We chose to implement the ACR-BCE architecture, based on the above described ACR methodology and paired with a binary classifier with cross entropy loss. The binary classifier is a CNN with four convolution layers (64 kernels each with a kernel size of 3), each layer being followed by a batch normalization layer and a ReLU activation. We use a fully-connected layer with output size of 1 as the final layer and lastly we pass the output through a sigmoid activation function.

[1]Contributed with researching models, building the training setup and benchmarking the ACR-BCE model.
[2]Contributed with researching models, building the dataset and benchmarking the ResNet+BN model.

Table 1: AUROC(%) obtained by ACR-BCE with standard deviation on MNIST with varying anomaly ratios and batch sizes.

Batch Size	$\pi = 0.05$	$\pi = 0.1$
16	85.3±3.4	83.2±2.56
32	83.2±3.11	84.9±3.56
64	85.6±4.24	84.8±3.76
128	86.8±1.83	85.1±2.16

Table 2: AUROC(%) obtained by ResNet152+BN compared to ACR-BCE with a batch size of 128.

Model	$\pi = 0.05$	$\pi = 0.1$
ACR-BCE	86.8±1.83	85.1±2.16
ResNet+BN	75.3±2.84	72.6±2.05

Benchmarking of ACR-BCE was done on the MNIST dataset by following the Meta Outlier Exposure paradigm as described in the original paper. We trained the model on distributions of type P_j^π, where j is the class and π is the ratio of anomalous data to normal data. A π of 0.8 was used, with j from 0 to 4. Although in practice the ratio of anomalous data is much smaller, π was kept high to avoid creating an imbalanced dataset. For testing we used P_5^π. Classes 6-9 were used as anomalous data points and we randomly sampled from each of them when building the training and testing set.

Training is done simultaneously on all training distributions. The parameters are updated once every 5 batches, each distribution contributing with one batch. Loss is then averaged across the 5 batches and thus one gradient descent step is performed.

To test the validity of our training setup, we calculated the distance between a training and a test distribution, using Maximum Mean Discrepancy, for different anomaly ratio. As expected, a lower anomaly ratio means greater distance between distributions since distributions use the same classes as anomalies. However, the overall distances are rather small and we believe our training setup could be improved further in this direction. We wish to do more experiments with different techniques to build MNIST distributions which could improve the performance. Full results in Table 3.

Train \ Test	$\pi = 1.0$	$\pi = 0.5$	$\pi = 0.1$
$\pi = 1.0$	0.012	0.016	0.015
$\pi = 0.5$	0.026	0.015	0.023
$\pi = 0.1$	0.033	0.035	0.037

Table 3: Distance between training (class 0) and testing (class 5) distributions, calculated using MMD, for different anomaly ratios.

3 Results

While the authors of the original paper touched very briefly on the results on the MNIST dataset, our aim was to explore this area further by performing more detailed experiments on the effect of the batch size and anomaly ratio. Results of the experiments are expressed as the AUROC averaged across five runs and with standard deviation.

The authors report good results on the MNIST dataset, with 88.7%, 87.8%, 86.5% on their 0.01, 0.05 and 0.1 anomaly ratio experiments respectively. These results slightly edge out the benchmarks on OC-MAML [2], which was the most robust model out of the other ones. While CLIP-AD [3] performs well on other experiments as the flagship for foundation models, it does poorly on non-natural images and cannot be used as comparison for the MNIST dataset.

Our trials obtain similar results to those of the authors, albeit with a bit more numerical instability. A π of 0.8 was found to provide the best results across all testing scenarios. As the main goal of our implementation was to experiment with different batch sizes and anomaly ratio, we trained our model with 4 different batch sizes and 2 anomaly ratios. Our results are highlighted in Table 1. Best score and with the smallest deviation was obtained by the batch size 128 with a π= 0.05.

To put our results into perspective, we implemented a second method based on a ResNet. A fully connected layer was added on top of the output of the penultimate layer of a pretrained ResNet152 and the final output was passed through a batch normalization layer. Training for this second method was done in the same manner as the first one. Results for this experiment are highlighted in Table 2.

References

[1] Li, Aodong, et al. "Zero-Shot Anomaly Detection without Foundation Models." arXiv preprint arXiv:2302.07849 (2023).

[2] Frikha, Ahmed, et al. "Few-shot one-class classification via meta-learning." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 35. No. 8. 2021.

[3] Liznerski, Philipp, et al. "Exposing Outlier Exposure: What Can Be Learned From Few, One, and Zero Outlier Images." arXiv preprint arXiv:2205.11474 (2022).