SSD: A UNIFIED FRAMEWORK FOR SELF-SUPERVISED OUTLIER DETECTION

Hossein Rezaei
School of Computer Engineering
Iran University of Science and Technology (IUST)
hossein_rezaei@comp.iust.ac.ir

ABSTRACT

Having outliers, those samples that do not follow the main distribution can hurt the performance of deep models. Therefore, detecting outliers in the training set of deep models is a crucial step toward making robust deep neural networks. In this regard, we study a recently proposed approach called *SSD: A unified framework for self-supervised outlier detection*. By using a pre-trained feature extractor, the method calculates an outlier score based on Mahalanobis distance. According to the proposed results, SSD can achieve notable performance in comparison with the previous unsupervised methods. We extend the existing study and evaluate the effect of different normalization strategies on the SSD performance in detecting outliers. Our experimental results demonstrate that making a normalized representation space can notably enhance the outlier detector performance.

1 INTRODUCTION

Recent promising advances in training deep models demonstrate their impressive performance on various real-world tasks Devlin et al. (2019); Krizhevsky et al. (2012); Rao et al. (2017). However, sensitivity to training data distribution is a severe challenge for neural networks. Existing studies have shown that such models perform poorly on the so-called out-of-distribution (OOD) evaluation sets. Therefore, detecting the OOD sample named outliers is a crucial step toward robustifying deep models.

Trying to overcome this challenge, several outlier detectors have been proposed Kingma & Dhariwal (2018); Mohseni et al. (2020); Zisselman & Tamar (2020); Bergman & Hoshen (2020). While some methods have taken advantage of labeled outlier samples, other approaches have designed detectors without accessing labeled examples (CITE). There are standard main components in both categories, including a feature extractor, usually a pre-trained model, to provide the representations, a detector (e.g., an MLP) that jointly is trained with the feature extractor, and a score calculator function to distinguish outliers.

Among various proposed outlier detectors, in this work, we study *SSD: A Unified Framework For Self-supervised Outlier Detection* paper Sehwag et al. (2021). The primary authors' contribution is proposing an unsupervised outlier detector employing in-distribution examples. Based on the reported results, SSD outperforms the previous unsupervised approaches by a significantly large margin. In this research, we validate the reported results by re-implementing SSD. We also investigate the effect of different normalization methods on the proposed approach. Our results show that adding a simple normalization step can boost SSD performance more, highlighting the critical role of normalization in deep models.

2 METHODOLOGY

In this section, first, we explain our experimental setups, and then briefly introduce *SSD* approach. In the last part, we describe different normalization approaches and their impacts on the outlier detector performance.

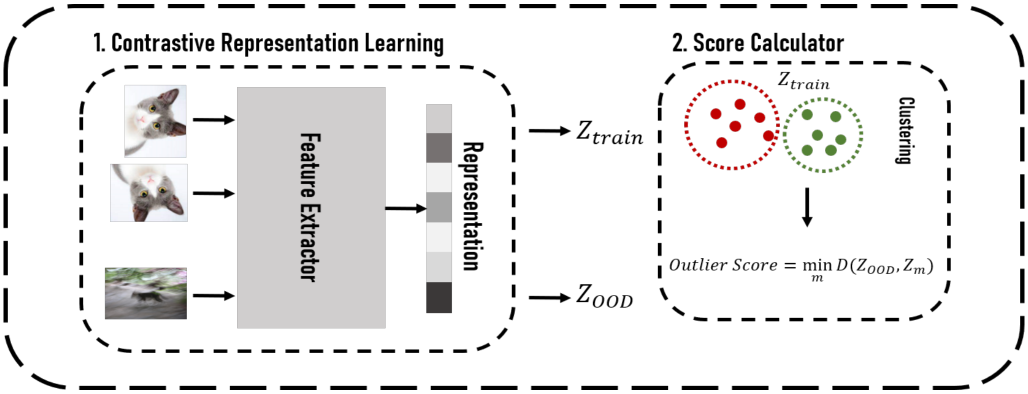

Figure 1: The visualization of the SSD framework. It mainly has two phases: 1)Learning high-quality representations using pre-trained neural networks, and 2)calculating outlier scores by clustering representations and using Mahalanobis distance.

2.1 Experimental Setups

In contrast to the original work where the pre-trained ResNet-50 model is used, we employ ResNet-18 as our primary feature extractor He et al. (2016). Following the author's suggestions, we train the SSD framework using a two-layer fully connected network for 500 epochs and a batch size of 512. We limit our study to the CIFAR-10 Krizhevsky (2009) and STL Coates et al. (2011) as the in-distribution datasets and CIFAR-100 Krizhevsky (2009), SVHN Netzer et al. (2011),and Texture Cimpoi et al. (2013) as the out-of-distribution ones.

2.2 SSD approach

In the unsupervised extension of the SSD, we train a feature extractor using a contrastive learning loss function whose aim is to learn expressive representations1. In the following equation, N is the number of images in a batch, $h(.)$ is the projector head, and η is the temperature.

$$L_{batch} = \frac{1}{2N} \sum_{i=1}^{2N} -\log \frac{e^{u_i^T u_j/\eta}}{\sum_{k=1}^{2N} 1(k \neq i) e^{u_i^T u_j/\eta}}; u_i = \frac{h(f(x_i))}{\|h(f(x_i))\|} \tag{1}$$

Intuitively, the loss function minimizes the distance between every image and its transformation while simultaneously maximizing their distances to other samples. Employing the provided representations, an effective OOD detector is defined consecutively. We cluster the provided representations(features) to m clusters using the k-means clustering algorithm and represent features for each cluster as Z_m. In the last step, the outlier score is calculated as below:

$$outlierscore(s_x) = min_m D(x, Z_m) \tag{2}$$

where $D(.,.)$ is the distance metric. The authors suggest using Mahalanobis distance. However, we also evaluate the SSD framework using two other popular distance metrics: Cosine distance and Euclidean distance. Figure 1 visualizes the SSD framework flow.

2.3 Normalization Effects

In this part, we study the effect of different normalization methods on detecting outliers using the explained score in 2. We consider several linear and non-linear normalization approaches that we describe the detail below.

Zero-mean . This is a simple linear normalization method in which we push the representations toward the center of the space by subtracting the mean representation from all representations.

Linear scaling . In this approach, we scale the representations' features into the -1 and $+1$ range. We use the following equation for the normalization:

	CIFAR-10			*STL*		
	CIFAR-100	SVHN	Texture	CIFAR-100	SVHN	Texture
Baseline	56.89	90.50	99.42	99.58	99.96	79.61
Zero-mean	56.89	90.50	99.42	99.58	99.96	79.61
Linear scaling	56.89	90.50	99.42	99.58	99.96	79.61
Standardization*	80.14	99.30	99.94	99.99	99.99	79.67
Log-scaling	58.42	93.80	98.57	73.80	94.22	84.81
Isotropy	55.30	57.37	98.84	99.58	99.96	79.61
Zero-mean	56.61	90.66	99.42	99.72	99.96	85.46
Linear scaling	56.82	90.56	99.42	99.74	99.97	84.26
Standardization*	80.14	99.31	98.56	99.99	99.99	83.47
Log-scaling	58.41	93.49	98.56	62.51	90.27	84.36
Isotropy	94.04	96.70	99.99	99.43	98.38	98.97

Table 1: Numerical results of applying different normalization strategies based on AUROC score without (the first part) and with (the second part) Shrunk covariance estimators. We consider the baseline when no normalization has been done. * denotes the approach that has been employed in the original paper. The results demonstrate the remarkable effect of applying isotropy on the outliers detection.

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3}$$

Standardization . Or **Z-Score Normalization** is the other method we employ to normalize representations which is summarized below:

$$x' = \frac{x - mean}{std} \tag{4}$$

Log-scaling . This is a non-linear normalization method that gets every feature's logarithm.

Isotropy . In any vector space, isotropy is a desirable property that can enhance the expressiveness of representations Huang et al. (2018). We utilize a famous method to increase isotropy Mu & Viswanath (2018). In this approach, we first make the representations zero-mean, and then, using the PCA algorithm, we find the most dominant feature and discard it.

For all normalization methods, we also calculate the covariance matrix used in 2 employing *Shrunk covariance estimators* Ledoit & Wolf (2003).

2.3.1 Results

Table 1 presents our experimental results. We take having no normalization as our baseline. As can be seen, applying any normalization methods, including linear and non-linear ones, can improve the outlier detector performance. However, among all, using the isotropy approach and shrunk estimator achieve the best AUROC scores. The other interesting observation is that linear normalization methods (Zero-mean and linear-scaling) do not affect the Mahalanobis-based metric when we use the exact covariance matrix. This is because Mahalanobis distance computation uses the inverse of the covariance matrix while computing the distance, which normalizes the original data by the variance and covariance present in the original data.

3 Conclusion

In this work, we study an unsupervised approach named *SSD: a unified framework for self-supervised outlier detection*. We validate the proposed results by re-implementing the framework. We also investigate the effect of several linear and non-linear normalization methods. Our results suggest that having a normalized feature space can improve the outlier detector's ability to find out-of-distribution examples.

REFERENCES

Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *ArXiv*, abs/2005.02359, 2020.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. *CoRR*, abs/1311.3618, 2013. URL http://dblp.uni-trier.de/db/journals/corr/corr1311.html#CimpoiMKMV13.

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 215–223, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL https://proceedings.mlr.press/v15/coates11a.html.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Lei Huang, Dawei Yang, Bo Lang, and Jia Deng. Decorrelated batch normalization. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 791–800, 2018.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/d139db6a236200b21cc7f752979132d0-Paper.pdf.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Olivier Ledoit and Michael Wolf. Honey, i shrunk the sample covariance matrix. 2003.

Sina Mohseni, Mandar Pitale, Jbs Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. In *AAAI*, 2020.

Jiaqi Mu and Pramod Viswanath. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HkuGJ3kCb.

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. URL http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Kanishka Rao, Hasim Sak, and Rohit Prabhavalkar. Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer. *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 193–199, 2017.

Vikash Sehwag, Mung Chiang, and Prateek Mittal. Ssd: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=v5gjXpmR8J.

Ev Zisselman and Aviv Tamar. Deep residual flow for out of distribution detection, 2020. URL https://arxiv.org/abs/2001.05419.