REPRODUCIBILITY CHALLENGE : DEEP SEMI-SUPERVISED ANOMALY DETECTION

Jiajun Bie
jb4u20@soton.ac.uk
31863744

Lichen Deng
ld4n20@soton.ac.uk
31255892

Mao Yang
my4n20@soton.ac.uk
32142765

ABSTRACT

On large and complex data sets, the Deep approaches to anomaly detection have recently shown promising results over shallow methods. We chose to reproduce a paper named 'Deep Semi-supervised Anomaly Detection' in ICLR 2020 Conference with the theme of Deep Anomaly Detection. This report introduced Deep SAD which is an end-to-end deep methodology for general semi-supervised anomaly detection, and it was also compared with other methods. In the reproduce process, we use MNIST, FASI-MNIST and CIFAR-10 data sets with three scenarios to train the models in multiple methods. In extensive experiments, we reach a similar conclusion as the original paper, but there are still some differences between the performance of Deep SAD and our expectation. We also demonstrate that the authors' methods are the same or superior to the shallow ones. At the end of the paper, we give our reflection on reproduction.

1 INTRODUCTION

Anomaly detection (AD) is the task of identifying anomaly samples in data. (Pimentel et al., 2014) Traditional shallow unsupervised AD approaches typically require manual feature engineering to efficiently process high-dimensional data and are limited in the scalability of large data sets. For large and complex data sets, the deep anomaly detection method is more effective. This report introduces the reimplementation process of ICLR 2020 conference paper 'Deep Semi-supervised Anomaly Detection' in detail, and we will analysis the method mentioned in the paper. (Ruff et al., 2019) In the report, we will introduce the algorithm involved in this paper in detail, as well as our experiments and the analysis of the results. Finally, we will compare the results we have reproduced with the original paper and draw our conclusion.

2 METHOD

In this section we introduce the theory of Deep SAD compare with Deep SVDD, and also introduce novel baselines as the comparison algorithm.

2.1 DEEP SAD

Deep SAD is an end-to-end deep method for general semi-supervised AD. It is a generalization of unsupervised Deep SVDD method.

Deep SVDD is to train the neural network ϕ to let it learn a transformation to minimize the volume of the enclosed data hypersphere in the output space $\mathcal{Z}$ centered on the predetermined point c, and optimized via SGD using backpropagation. Deep SVDD can be interpreted as a minimum volume estimation geometrically, or as a probability minimization of the entropy of the potential distribution. Deep SVDD just use unlabeled samples and labled normal samples.

The Deep SAD method takes advantages of all training data: n unlabeled samples $x_1, ..., x_n \in \mathcal{X}$ with $\mathcal{X} \subseteq \mathbb{R}^D$, and m labeled normal/anomalies samples $(\tilde{x_1}, \tilde{y_1}), ..., (\tilde{x_m}, \tilde{y_m}) \in \mathcal{X} \times \mathcal{Y}$ with $\mathcal{Y}$={-1,+1}, when $\tilde{y}$=+1 denoted normal and $\tilde{y}$=-1 denoted anomalous samples (Ruff et al., 2019). As the paper illustrate, we have the follow Deep SAD objective function:

$$\min_{\mathcal{W}} \frac{1}{n+m} \sum || \, \phi(x_i; \mathcal{W}) - c \, ||^2 + \frac{\eta}{n+m} \sum_{j=1}^{m} (|| \, \phi(\tilde{x}_j; \mathcal{W} - c \, ||^2)^{\tilde{y_j}} + \frac{\lambda}{2} \sum_{l=1}^{L} || \, W^l \, ||_F^2 \tag{1}$$

The input space $\mathcal{X} \subseteq \mathbb{R}^D$ and output space $\mathcal{Z} \subseteq \mathbb{R}^d$, $\phi(.; \mathcal{W}) : \mathcal{X} \rightarrow \mathcal{Z}$ is a neural network, L is the hidden layers, $\mathcal{W}$ is the corresponding weight, c is the hypersphere center. In this formula, if m=0, the formula is the same as Deep SVDD.

The first and third items are the same as Deep SVDD. And the second item is to introduce a loss item for the marked m labeled data. A loss term is introduced for the labeled m data, and the hyperparameter η is used to control the balance between labeled and unlabeled items. $\eta > 1$, pay more attention to labeled data; otherwise, pay more attention to unlabeled data.

2.2 Baseline

- **SSAD Raw**: Here in addition to normal samples, a small number of abnormal samples with labels are also provided. (Görnitz et al., 2013)
- **Isolation Forest (IF)**: The isolation forest divides the data based on the binary tree, and the depth of the node in the tree reflects the degree of alienation of the data. Similar to random forest, the final result is generated by voting. It is no longer to describe normal sample points, but to isolate anomaly points. (Liu et al., 2008)
- **OC-SVM Raw**: One-Class SVM is not an outlier detection method, but a novelty detection method: its training set should not be adulterated with anomaly points, because the model may match these anomaly points. (Schölkopf et al., 2001)

3 Experiment

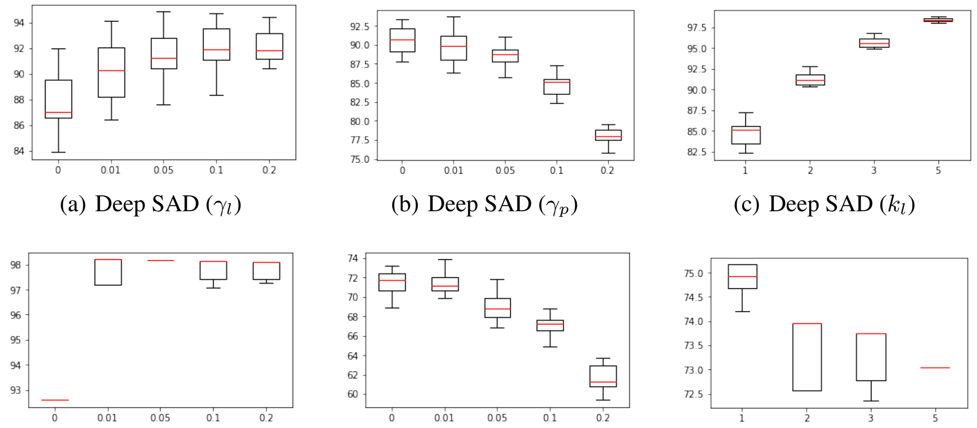
We tried to replicate the three sub-experiments using the code implemented in the original paper. We forked the original repository and added some bash code, which can be accessed at Github repository, and the address is in Appendix A.

Based on the MNIST, FMNIST and CIFAR-10 data sets, we tested them in three different scenarios. Starting from increasing the ratio of labeled outlier data, increasing the ratio of polluted training data, and changing the number of known anomaly classes, test the performance differences between different models and the baseline model. We modified the training structure in the original code, and performed 25 independent repetitive training with 150 epochs on each scenario. At the same time, for the baseline model, we modified the computing resources used and migrated the data to the GPU for training. The training was carried out on Google Colab, using a total of 3 Tesla-P100 accelerators. The training time of the test model is 90 hours in total, and the training time of the baseline model is 50 hours. Figures 1, 2, and 3 show the comparison results of the model's test results with different baseline models on different data sets.

As the ratio of labeled anomalies increased, the AUC of deep SAD gradually increases while the variance decreases. However, as the ratio of pulluted unlabeled training data increases, the AUC of the Deep SAD algorithm gradually decreases.

Unfortunately, we did not get the same AUC value as the original paper. In scenario one and scenario two, the average value of AUC differs from the original result by 0.03-0.06. The baseline model is not affected.

In the first column of Figure 2 and Figure 3, it can be clearly seen that as the ratio increases, the performance of the SSAD model gradually becomes better and exceeds the performance of DeepSAD, which is inconsistent with the description in the original paper. At the same time, we observe that as a traditional method, Isolation forest has similar or better performance than DeepSAD in many scenarios, as shown in Figure 2 b and e. Similarly, Figure 2 a, d also shows that SSAD has amazing stability and high AUC on the FMNIST data set, which has better performance than Deep SAD.



(a) Deep SAD (γ_l)

(b) Deep SAD (γ_p)

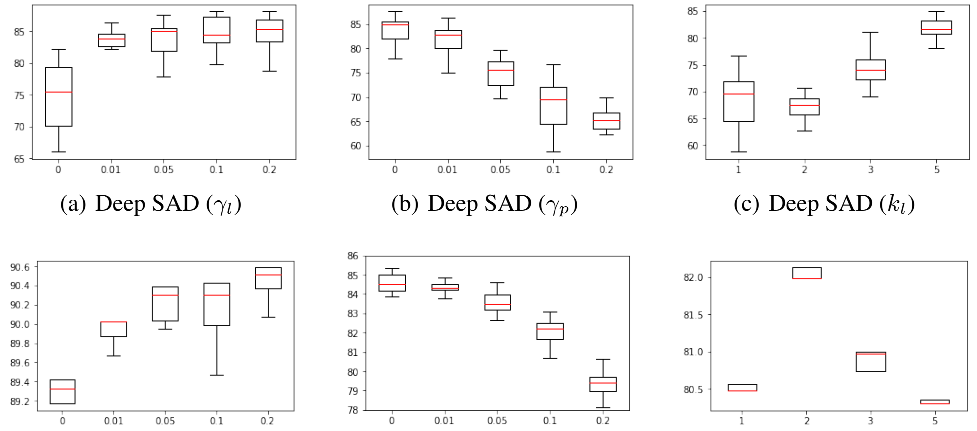
(c) Deep SAD (k_l)

(d) SSAD Raw (γ_l)

(e) Isolation Forest (γ_p)

(f) OCSVM Raw (k_l)

Figure 1: The impact of parameter changes under the MNIST data set on AUC



(a) Deep SAD (γ_l)

(b) Deep SAD (γ_p)

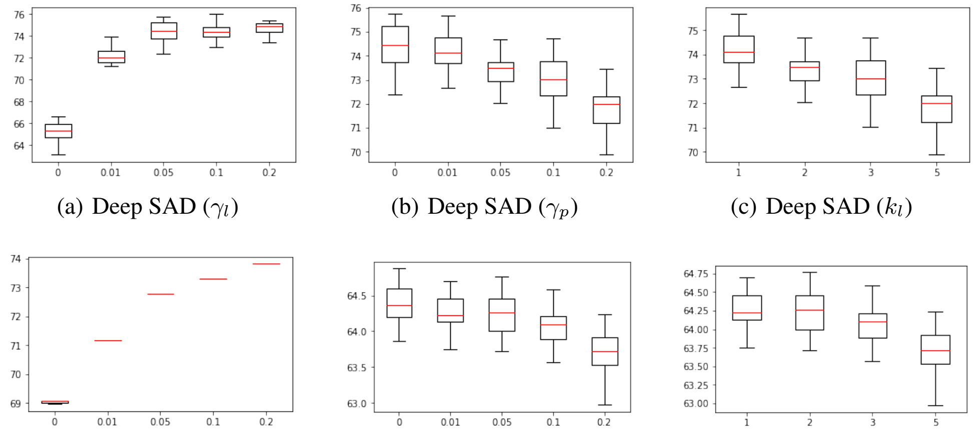
(c) Deep SAD (k_l)

(d) SSAD Raw (γ_l)

(e) Isolation Forest (γ_p)

(f) OCSVM Raw (k_l)

Figure 2: The impact of parameter changes under the FMNIST data set on AUC



(a) Deep SAD (γ_l)

(b) Deep SAD (γ_p)

(c) Deep SAD (k_l)

(d) SSAD Raw (γ_l)

(e) Isolation Forest (γ_p)

(f) Isolation Forest (k_l)

Figure 3: The impact of parameter changes under the CIFAR-10 data set on AUC

4 Discussion

We noticed by reading the training log that in the training process of DeepSAD, if the data is not contaminated, the training AUC and error of the autoencoder are higher than the results of the optimizer training. The optimizer loads the pre-trained model trained by the autoencoder, and its training loss is still converging, and it converges to a small value or even much smaller than the autoencoder loss. Since the optimizer uses the Adam algorithm, it can quickly converge to the local optimal solution but cannot cross this value.

At the same time, compared to deepSVDD, this method only considers the improvement of unsupervised learning to semi-supervised learning. By adding a mutual information mechanism, according to the principle of information bottleneck, the mutual information between features and hidden variables is minimized and the gap between hidden variables and labels is maximized. In order to achieve a balance of expression between compression and prediction. This is not a big change in itself but it has brought very good results, and greatly surpasses the traditional methods in many scenarios.

However, we have also drawn some conclusions that are different from the original paper through experiments. Due to the limitation of computing resources and experimental time, the AUC value will not be completely consistent with the original paper. Since the traditional method has approached the State-of-the-art model, the limitation on the recurrence will directly affect the conclusion of the recurrence and at the same time affect the pros and cons of the evaluation model. The results of the comparison with the SSAD baseline method in the paper are different from the results we obtained, which may inspire us not to blindly believe the experimental results. But in most scenarios, deepSAD still captures the anomaly points very well. We also tried the Isolation Forest method that was not shown in the original paper. We can see that many traditional methods can also achieve good results on the basic data set, even better than the deep method. But as the task difficulty increases, the ability of deep method to capture features gradually became prominent.

5 Conclusion

This paper mainly aims to the reproduce the experiments of ICLR 2020 conference Paper 'Deep Semi-supervised Anomaly Detection'. Starting with the theory involved in the original paper, the author's implementation method and our own reproduce process are also introduced step by step. We find a little difference from the original paper, but the major results are the same as the original paper.

References

Nico Görnitz, Marius Kloft, Konrad Rieck, and Ulf Brefeld. Toward supervised anomaly detection. *Journal of Artificial Intelligence Research*, 46:235–262, 2013.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE, 2008.

Marco AF Pimentel, David A Clifton, Lei Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Processing*, 99:215–249, 2014.

Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. *arXiv preprint arXiv:1906.02694*, 2019.

Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

A Github repository

https://github.com/COMP6248-Reproducability-Challenge/DeepSAD_Unknown