R2-AD2: Detecting Anomalies by Analysing the Raw Gradient

Jan-Philipp Schulze[1,3][0000−0003−1787−4102], Philip Sperl[1,3][0000−0002−7901−7168] (✉), Ana Răduțoiu[1,⋆][0000−0002−3139−2954], Carla Sagebiel[2,⋆], and Konstantin Böttinger[3][0000−0002−9337−7506]

[1] Technical University of Munich, Germany
[2] Heidelberg University, Germany
[3] Fraunhofer Institute for Applied and Integrated Security, Germany
{jan-philipp.schulze, philip.sperl, konstantin.boettinger}@aisec.fraunhofer.de

Abstract. Neural networks follow a gradient-based learning scheme, adapting their mapping parameters by back-propagating the output loss. Samples unlike the ones seen during training cause a different gradient distribution. Based on this intuition, we design a novel semi-supervised anomaly detection method called R2-AD2. By analysing the temporal distribution of the gradient over multiple training steps, we reliably detect point anomalies in strict semi-supervised settings. Instead of domain dependent features, we input the raw gradient caused by the sample under test to an end-to-end recurrent neural network architecture. R2-AD2 works in a purely data-driven way, thus is readily applicable in a variety of important use cases of anomaly detection.

Keywords: Anomaly Detection · Semi-supervised Learning · Deep Learning · Data Mining · IT Security

1 Introduction

Anomalies are inputs that significantly deviate from the given notion of normal. Depending on the use case, anomalies may lead to attacks on the infrastructure, fraudulent transactions or points of interest in general. In recent years, research on semi-supervised anomaly detection (AD) gained traction (e.g. [31, 28, 37]), where we leverage prior knowledge about the anomalous distribution to boost the overall detection performance. This setting is often found in real-world settings, where a few anomalies have already been detected manually while the rest are unknown. Unlike classification tasks, a semi-supervised AD method should not just differentiate between normal inputs and known anomalies, but also reveal yet unseen types of anomalies.

The lack of absolute training data modelling all types of anomalies complicates the use of machine learning algorithms with an automatic feature selection,

⋆ The research was done while working at Fraunhofer AISEC

e.g. deep learning (DL) methods. In our research, we alleviate this problem by analysing an abstract representation of the input: its temporal gradient distribution. Intuitively, a neural network (NN) trained only on the known normal data will fail to process anomalies in the same manner. We analyse this discrepancy with the help of an auxiliary NN. To reduce the manual work and domain expert knowledge required, we designed our AD method to be purely data-driven. Instead of hand-crafted features, we analyse the raw gradient caused by individual inputs for anomalous patterns. In our thorough empirical study, we show that our method generalises to several use cases and data types. Based on this principle, we call our novel AD method R2-AD2, raw gradient anomaly detection. In summary, our contributions to AD research are:

- We introduce a novel data-driven end-to-end neural architecture to analyse the temporal distribution of the gradient to detect point anomalies.
- To the best of our knowledge, R2-AD2 is the first semi-supervised AD method based on the analysis of gradients.
- We thoroughly analyse the performance gain by R2-AD2 on ten data sets against five baseline methods.
- To support future research, we open-sourced[4] our code.

1.1 Related Work

R2-AD2 is a DL-based, semi-supervised AD method building on the analysis of the input's gradient space. In the following, we discuss related work from all of the three categories. For a broader overview on AD, we recommend the surveys of Pang et al. [27] and Ruff et al. [30].

Anomaly Detection based on Deep Learning Methods DL methods deliver high performance even on complex inputs, but are data-demanding. Due to the inherent class imbalance of AD, it is challenging to apply DL methods. Over the past years, a variety of solutions arose, which we loosely group in three categories: methods based on 1) the reconstruction error, 2) the distance to the training data and 3) end-to-end architectures. Reconstruction-based methods use a representation or distribution estimation method, e.g. autoencoders (AEs) [5, 40, 2] or generative adversarial nets (GANs) [33, 22, 1]. Intuitively, when the network is fitted on the normal data, there is a measurable difference between the reconstructed and the input sample when an anomaly is processed. The main problems are noisy data sets, causing a low reconstruction error for some anomalies, and anomalies close to normal samples, which are easy to reconstruct. Distance-based methods, e.g. one-class classifiers [6, 31, 36], introduce a transformer network. Using a suitable metric, the transformer network maps normal samples close to each other, but anomalies far away. Problems may arise when the data set contains multiple notions of normal, which cannot be mapped to the very same centre of normality. R2-AD2 uses an end-to-end neural architecture, directly mapping

[4] https://github.com/Fraunhofer-AISEC/R2-AD2

Table 1: Work across different detection domains analysing the gradient space.

	Unsupervised	**Semi-supervised**	**Supervised**
Anomaly Det.	[18, 17]	**R2-AD2**	–
Out-of-Distr. Det.	[15, 38]	[20]	–
Adversarial Det.	–	–	[7, 23, 34, 21]

the input to an anomaly score. Usually, end-to-end architectures [26, 28, 37] require normal as well as anomalous training samples. However, manually finding anomalies is a time-consuming and error-prone process. Research on substituting real anomalies by artificially created ones, e.g. geometric transformations [8, 3] or out-of-distribution (OOD) samples [12], tries to solve this issue. These methods need careful adaptations to the respective data set. In R2-AD2, we mitigate the problem by using a simple source for trivial anomalies: a Gaussian distribution as done in A^3 [37]. Our evaluation motivates that our analysis in the gradient space of NNs allows to find a suitable boundary between real normal and real anomalous samples even with this simple source for counterexamples.

Semi-supervised Anomaly Detection In the past years, research about semi-supervised AD has gained traction. In real-world scenarios, a few known anomalies – much less than the normal samples – may already be available. These known anomalies may have been found manually or by an unsupervised AD method. Semi-supervised AD methods use this kind of prior knowledge to boost the overall detection performance. DeepSAD [31] is a semi-supervised extension of one-class classifiers. Deviation Networks (DevNet) [28] is based on distance metrics. The authors of A^3 [37] analyse the hidden activations of NNs for anomalous patterns. Reconstruction errors are evaluated in ABC [44] and ESAD [14]. Expanding the view to OOD detection, DROCC [9] uses generated counterexamples based on the prior knowledge about real anomalies. For semi-supervised AD methods, the distribution of the known anomalies may severely impact the generalisation performance [45]. Thus, a main challenge is the detection of unknown anomalies, i.e. anomalies, which have not yet been detected manually.

Gradient-based Detection of Anomalous Instances R2-AD2 analyses the gradient space of NNs. Despite the variety of AD research, this idea has barely been covered by previous work. We give an overview in Table 1. Kwon et al. [18] propose using the l_2-norm of an AE's gradient. The same authors refine the idea in their AD method GradCon [17]. Here, they measure the cosine similarity between past normal gradients and the current input. Expanding the view to research topics related to AD, we see applications in OOD and adversarial detection. In OOD detection, multi-class data and thus known class labels are assumed, which is not applicable to AD, where we merely distinguish between monolithic sets of normal and anomalous data. Sun et al. detect OOD samples by measuring the Mahalanobis distance of the gradient. In GradNorm [15], the authors used the Kullback-Leibler divergence on the l_1-norm of the gradient. Similarly,

Lee et al. [20] use the l_1-norm, but also incorporate some known OOD samples. In adversarial detection, samples, which have been specifically generated to alter the decision of a NN, are detected. In contrast to AD and OOD detection, adversarial detection is usually considered a supervised problem because counterexamples can be easily generated. In GraN [23], the authors used the l_1-norm of the gradient, whereas in Gradient Similarity [7] the authors took the l_2-norm of the gradient along the cosine similarity to distinguish between benign and adversarial samples. Lee et al. [21] train a classifier on the layer-wise l_2-norm. In DA3G [34], the authors analyse the raw gradient of the last two layers of classifiers. In R2-AD2, we refrain from using hand-crafted features or manually selecting points of interest as each choice incorporates prior knowledge from the algorithm designer, which may not be backed by the training data. Instead, we analyse the temporal distribution of the entire raw gradient by our end-to-end DL-based architecture. Our evaluation shows that R2-AD2 outperforms past AD methods on a variety of use cases and data types.

2 Prerequisites

In AD, we discover samples that deviate from the training data set $\mathcal{X}_{\text{norm}}$. Implicitly, we assume all samples in $\mathcal{X}_{\text{norm}}$ to be normal, even when polluted by unknown anomalies. In literature, there is some ambiguity in the definition of semi-supervised AD, which is sometimes referred to as supervised AD. In this regard, we follow the notation of Ruff et al. [31]. In our semi-supervised scenario, further we have access to a small data set $\mathcal{X}_{\text{anom}}$, containing a few known anomalies, i.e. $|\mathcal{X}_{\text{norm}}| \gg |\mathcal{X}_{\text{anom}}|$. Note that AD differs from related topics as OOD detection. In OOD detection, we do have access to an underlying classifier and its multi-class training data set. Instead, in AD, we consider the entire normal data set as one class and detect deviations from it. We refer to the survey of Salehi et al. [32] for an in-depth discussion of AD and its related research topics.

2.1 Activation Anomaly Analysis

Parts of R2-AD2 are inspired by the semi-supervised AD method A^3 [37]. Sperl et al. introduced their so-called target-alarm architecture. The target network, e.g. an AE, learns the distribution of the normal data. An auxiliary NN, called the alarm network, analyses the hidden activations of the target network while processing normal as well as anomalous inputs. As additional source of anomalous patterns, they input synthetic anomalies generated from a Gaussian prior.

In R2-AD2, we extend the target-alarm architecture to analyse the temporal gradient distribution of AEs. We use a recurrent alarm network to concurrently analyse the gradient of multiple AEs for anomalous patterns. Each AE reflects a different training state of the very same architecture. Our evaluation shows that the temporal gradient distribution allows a more reliable anomaly detection performance even under severe data pollution and unknown anomalies.

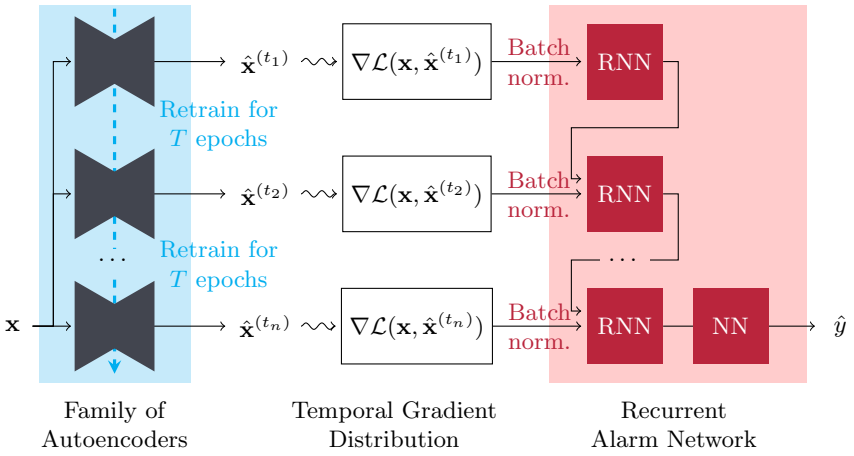

Fig. 1: Data flow of R2-AD2: we map the input sample $\mathbf{x}$ to an anomaly score $\hat{y} \in [0, 1]$, where 1 is highly anomalous. The input is processed by a family of AEs, each yielded by successive training on the normal samples. We measure the discrepancy between the predicted and the original input by calculating the respective gradient. An auxiliary network, called the alarm network, analyses this sequence of gradients for anomalous patterns.

3 R2-AD2

R2-AD2 builds upon our main intuition:

> Let $f_{\text{AE}}(\mathbf{x}; \boldsymbol{\theta})$ be an AE trained on the data set $\mathcal{X}_{\text{norm}}$ containing normal samples. The evolution of the gradient $\nabla f_{\text{AE}}(\mathbf{x}) = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{x}, f_{\text{AE}}(\mathbf{x}; \boldsymbol{\theta}))$ is useful to decide if the current input $\mathbf{x}$ is normal or anomalous.

Our intuition is a natural extension of the manual analysis of the gradient as done in past research [18, 17]. Instead of considering certain features, e.g. magnitudes or directions, we analyse the gradient in its entirety. Let $f_{\text{AE}}^{(i)}(\mathbf{x}) = f_{\text{AE}}(\mathbf{x}; \boldsymbol{\theta}^{(i)})$ be the target AE after the i-th training epoch. With each training step, the mapping parameters $\boldsymbol{\theta}$ adapt more to the training data and hence to the normal samples. Thus, we embed the temporal distribution of the gradient in R2-AD2. Let $g_{f_{\text{AE}}}(\mathbf{x})$ denote the function that extracts the gradients over time given the target AE $f_{\text{AE}}(\cdot)$:

$$g_{f_{\text{AE}}}(\mathbf{x}) = [\nabla f_{\text{AE}}^{(i)}(\mathbf{x})]_{i=T_0+jT, j\in\mathbb{N}} = [\nabla f_{\text{AE}}^{(T_0)}(\mathbf{x}), \nabla f_{\text{AE}}^{(T_0+T)}(\mathbf{x}), \ldots], \tag{1}$$

where T is some sampling frequency and T_0 an offset.

Given the temporal gradient distribution, an auxiliary NN, called the *alarm network* $f_{\text{alarm}}(\cdot)$, analyses it for anomalous patterns. The alarm network is a binary classifier outputting an anomaly score, where 1 is highly anomalous. Both networks are combined to the overall end-to-end architecture of R2-AD2 depicted

Input: $f_{\text{AE}}(\mathbf{x}; \boldsymbol{\theta}^{T_0}), \mathcal{D}_{\text{train}} = \mathcal{X}_{\text{train}} \times \mathcal{Y}_{\text{train}}$
Result: $f_{\text{R2-AD2}}$
// Retrain the autoencoder on $\mathcal{X}_{\texttt{norm}} \subset \mathcal{X}_{\texttt{train}}$
$f_{\text{AE},0} \leftarrow f_{\text{AE}}(\mathbf{x}; \boldsymbol{\theta}^{T_0})$, $f_{\text{AE},1} \leftarrow f_{\text{AE}}(\mathbf{x}; \boldsymbol{\theta}^{T_0+T})$, ...;
for $(\mathbf{x}, y) \in \mathcal{X}_{train}$ **do**
  // Sample synthetic anomalies
  $\tilde{\mathbf{x}} \leftarrow \mathcal{N}(\mathbf{0.5}, \mathbf{1.0})$;
  // Extract the gradients from the retrained autoencoders
  $\mathbf{g} \leftarrow [\nabla f_{\text{AE},0}(\mathbf{x}), \nabla f_{\text{AE},1}(\mathbf{x}), \ldots]$, $\tilde{\mathbf{g}} \leftarrow [\nabla f_{\text{AE},0}(\tilde{\mathbf{x}}), \nabla f_{\text{AE},1}(\tilde{\mathbf{x}}), \ldots]$, eq. (1);
  // Train R2-AD2's components
  $\operatorname{argmin} \boldsymbol{\theta}_{\text{batchnorm.}} : f_{\text{alarm}} \leftarrow (\mathbf{g}, y)$, eq. (3);
  $\operatorname{argmin} \boldsymbol{\theta}_{\text{alarm}} : f_{\text{alarm}} \leftarrow (\mathbf{g}, y), (\tilde{\mathbf{g}}, 1)$, eq. (2);
end

Algorithm 1: High-level overview about R2-AD2's training objectives.

in Figure 1 and formally defined as: $f_{\text{R2-AD2}}(\mathbf{x}) = f_{\text{alarm}}(g_{f_{\text{AE}}}(\mathbf{x})) \in [0, 1]$. Due to the sequential nature of the gradient, the alarm network is a recurrent neural network (RNN). We combine the RNN with a time-distributed batch normalisation [16] layer and fully-connected output layers. In our research, we found the batch normalisation layer to be essential to scale small gradients, especially after several training epochs of the target network.

Training Objectives AD is characterised by its inherent class imbalance, where known anomalies are rare and might not cover the entire anomaly distribution. In R2-AD2, we solve this problem by sampling trivial counterexamples from a Gaussian prior, i.e. $\tilde{\mathbf{x}} \sim \mathcal{N}(\mu, \sigma^2)$. Even though these synthetic anomalies do not resemble real ones, our analysis in the gradient space results in a meaningful decision barrier between real normal and real anomalous inputs. As result, the training objective of R2-AD2 becomes a simple classification using the binary cross entropy (BXE) as loss:

$$\underset{\theta_{\text{alarm}}}{\operatorname{argmin}} \mathbb{E}[\mathcal{L}_{\text{BXE}}(y, f_{\text{R2-AD2}}(\mathbf{x})) + \mathcal{L}_{\text{BXE}}(1, f_{\text{R2-AD2}}(\tilde{\mathbf{x}}))], \tag{2}$$

where $(\mathbf{x}, y) \sim P_{\mathcal{D}}, \tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0.5}, \mathbf{1.0})$. Our input data is scaled to $\mathbf{x} \in [0, 1]^N$, thus the synthetic anomalies are likely outside this interval, i.e. clearly anomalous. Due to the random nature of the counterexamples, we adapt the batch normalisation layer on the training data only, i.e.:

$$\underset{\theta_{\text{batchnorm.}}}{\operatorname{argmin}} \mathbb{E}[\mathcal{L}_{\text{BXE}}(y, f_{\text{R2-AD2}}(\mathbf{x}))]. \tag{3}$$

In Algorithm 1, we summarise R2-AD2's training process.

Table 2: Data sets under evaluation. If multiple anomaly types were available, we tested R2-AD2 on classes unknown during training in our transfer experiments.

Data	**Normal**	**Train Ano.** $\subseteq$	**Test Ano.**	**Encoder**
CC [29]	Normal	Anomalous	Anomalous	20, 10, 5
CoverType [4]	1-3	4-5	4-7	40, 20, 10
DarkNet [10]	Non-Tor/-VPN	Tor	Tor, VPN	60, 30, 15
DoH [25]	Benign	Mal.	Mal.	20, 10, 5
FMNIST [43]	0-3	4-6	4-9	8C3-8C3-8
IDS [35]	Benign	Bot, BF	Bot, BF, Infil., Web	60, 40, 20
KDD [39]	Normal	DoS, Probe	DoS, Probe, R2L, U2R	40, 20, 10
MNIST [19]	0-3	4-6	4-9	8C3-8C3-8
Mam. [42]	Normal	Malignant	Malignant	5, 3, 2
URL [24]	Benign	Def., Mal.	Def., Mal., Phi., Spam	60, 30, 15

4 Experimental Setup

We evaluated R2-AD2 in challenging experiments mimicking real-world scenarios. In Table 2, we show the ten data sets under evaluation, ranging from commonly used baseline data sets to important applications of AD, e.g. intrusion or fraud detection. We scaled all numerical values to $[0, 1]$ and 1-Hot encoded categorical entries. If not given by the data set, 75% were used for the training split, 5% for validation and 20% for testing. While training R2-AD2's AE, 25% of the training data were held back to evaluate the gradient distribution of some fresh normal samples while training the alarm network.

Baseline Methods R2-AD2 is a deep semi-supervised AD method based on the analysis of the gradient space of AEs. AEs themselves can be used as AD method by measuring the reconstruction error, when only trained on the normal data. We used the mean squared error as anomaly score, i.e. $\hat{y} = \|f_{\text{AE}}(\mathbf{x}) - \mathbf{x}\|_2^2$. GradCon [17] is a AD method based on the analysis of the gradient space of NNs. We favoured GradCon over the authors' initial AD method based on l_2-norms [18] as it generally performed better according to their evaluation. Both aforementioned baseline methods are unsupervised, thus do not profit from known anomalies. Expanding our view to deep semi-supervised AD, DeepSAD [31] is a commonly used baseline. In the same category, DevNet [28] and A^3 [37] are currently the best performing methods.

Parameter Choices We designed R2-AD2 as a data-driven method, which readily applies to a diverse set of use cases and data types. Thus, we chose one common set of hyperparameters for the entire evaluation. Across all data sets, we analysed a target network trained for $T_0 = 10$ epochs across 2 retraining steps, each with $T = 5$ epochs resulting in three models. The alarm network had the dimensions $1000, 500, 200, 75$ except for the small Mammography data set, where we used $100, 50, 25, 10$. LSTM [13] elements were used for the first

two dimensions, ReLU-activated dense layers else. R2-AD2 was trained for 100 epochs at a learning rate of 0.001 using Adam as optimiser. For a fair comparison, we chose the same hyperparameters for the other baseline methods if applicable.

5 Evaluation

We carefully followed the best practices introduced by Hendrycks & Gimpel [11] and report the performance as area under the ROC curve (AUC) and average precision (AP). Both metrics measure the performance independently of a chosen detection threshold. An ideal AD method scores an AUC and AP of 1. To measure the significance of our results, we report the p-value of the Wilcoxon signed-rank test [41]. It evaluates the null hypothesis that a ranked list of measurements was derived from the same distribution.

5.1 Known Anomalies

In our first experiment, we evaluated the performance gain in an ideal semi-supervised AD setting. We limited the number of known anomalies to 100 randomly chosen samples, i.e. far less than normal samples available. In Table 3, we summarise the results. R2-AD2 took the lead across all baseline methods, scoring the best on 7 out of 10 data sets.

As expected, the unsupervised baseline methods could not match the performance of the semi-supervised methods as they do not profit from the known anomalies. Looking at the AUC, R2-AD2 was 24 % better than the other gradient-based AD method, GradCon. KDD was the only data set, where the unsupervised methods took the lead. Here, some unknown anomalies are within the test data set. Similar to the discussion of Ye et al. [45], we believe the semi-supervised methods overfitted to the known anomalies. Comparing our performance to GradCon, we see strong evidence that the analysis of the raw gradient is favourable over a hand-crafted feature set: GradCon's analysis of the cosine similarity works well on some data sets (e.g. MNIST and DarkNet), but does not generalise to all ten data sets. R2-AD2 had the more consistent performance.

Considering the semi-supervised baselines, the largest margin was on DoH, where R2-AD2 performed 8 % better than DevNet, and 6 % better on IDS compared to A^3. A^3 has a similar architecture as R2-AD2, but analyses the hidden activations of a single AE instead of the temporal gradient distribution. Overall R2-AD2 performed 8 % better than A^3. Only on the image data sets, A^3 was the preferable method. Summarising this section, R2-AD2 clearly profited from the prior knowledge available in semi-supervised AD and allowed a more reliable detection performance compared to other state-of-the-art methods.

5.2 Noise Resistance

In real-world settings, it is usually infeasible to guarantee a clean training data set. We evaluated this scenario by polluting the data with anomalous training

Table 3: Detection of known anomalies, i.e. the training and test data set contained the same anomaly classes. We limited the number of known anomalies to 100 and show the results after five detection runs.

	Ours		Unsupervised Baselines				Semi-supervised Baselines					
	R2-AD2		AE		GradCon		DeepSAD		DevNet		A^3	
	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP
CC	**.98 ± .01**	**.81 ± .03**	.95 ± .00	.43 ± .00	.80 ± .16	.34 ± .08	.88 ± .03	.34 ± .26	**.98 ± .00**	.74 ± .00	.88 ± .06	.51 ± .21
CT	**.84 ± .02**	**.43 ± .05**	.76 ± .02	.25 ± .03	.70 ± .05	.21 ± .06	.57 ± .07	.16 ± .05	.83 ± .01	.33 ± .02	.46 ± .06	.08 ± .01
DN	**.92 ± .01**	**.75 ± .02**	.54 ± .01	.23 ± .01	.62 ± .09	.24 ± .04	.68 ± .15	.36 ± .12	.90 ± .01	.69 ± .02	.84 ± .02	.54 ± .05
DoH	**.98 ± .00**	**1.00 ± .00**	.85 ± .01	.98 ± .00	.74 ± .06	.97 ± .01	.73 ± .08	.97 ± .01	.91 ± .01	.99 ± .00	.83 ± .04	.98 ± .01
FMN.	.92 ± .00	.95 ± .00	.86 ± .00	.92 ± .00	.82 ± .02	.88 ± .03	.69 ± .02	.77 ± .02	.93 ± .02	.96 ± .01	**.95 ± .01**	**.97 ± .00**
IDS	**.93 ± .01**	**.89 ± .01**	.84 ± .01	.52 ± .04	.45 ± .10	.19 ± .06	.67 ± .08	.38 ± .12	.87 ± .01	.67 ± .07	.88 ± .02	.67 ± .11
KDD	.88 ± .02	.90 ± .03	**.95 ± .00**	**.95 ± .00**	.74 ± .04	.83 ± .01	.85 ± .07	.88 ± .06	.92 ± .03	.94 ± .01	.93 ± .04	**.95 ± .02**
MN.	.97 ± .01	**.98 ± .00**	.75 ± .01	.79 ± .01	.82 ± .04	.84 ± .04	.70 ± .02	.75 ± .02	.97 ± .00	**.98 ± .00**	**.98 ± .00**	**.98 ± .01**
Mam.	**.94 ± .01**	**.69 ± .03**	.90 ± .01	.27 ± .01	.89 ± .01	.26 ± .02	.69 ± .23	.16 ± .12	**.94 ± .01**	.65 ± .04	.88 ± .04	.43 ± .06
URL	**.95 ± .01**	**.99 ± .00**	.92 ± .00	.98 ± .00	.90 ± .01	.97 ± .00	.94 ± .01	**.99 ± .00**	**.95 ± .01**	**.99 ± .00**	.94 ± .01	**.99 ± .00**
mean	.93	.84	.83	.63	.75	.57	.74	.58	.92	.79	.86	.71
p-val	-	-	.02	.01	.00	.00	.00	.00	.38	.05	.06	.05

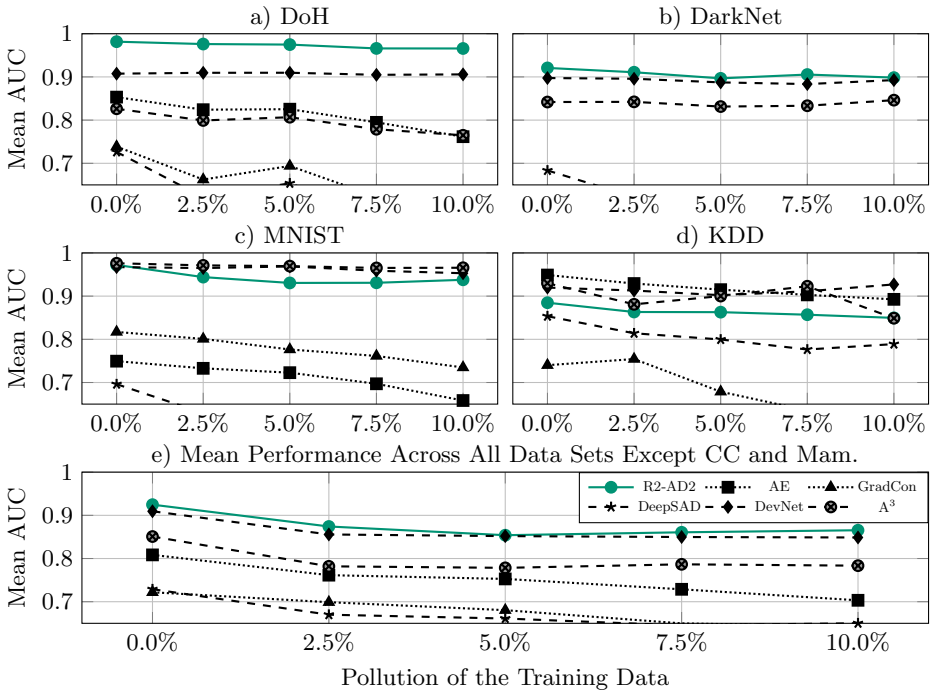

Fig. 2: Detection performance depending on the training data pollution. All semi-supervised methods had access to 100 known anomalies. Note that CC and Mammography did not contain enough anomalies, thus were excluded.

samples labelled as normal. All semi-supervised methods still had access to 100 known anomalies. We summarise the performance in Figure 2 for DoH and DarkNet, where R2-AD2 took the lead in our first experiment, and MNIST and KDD, where the baseline methods performed better. Additionally, we show the mean performance across all data sets, which scaled to this experiment.

Looking at the mean performance, R2-AD2 took the lead across all pollution levels. The performance dropped only by $-6\,\%$, when every tenth training sample was an anomaly labelled as normal. For the unsupervised baselines, the performance drop was considerably larger, e.g. $-13\,\%$ for the AE. The known anomalies seemed to stabilise the performance. Across the data sets, the general ranking between the baseline methods did not change: AD methods that performed well on cleaner data sets also performed well on polluted data sets. Our evaluation showed that R2-AD2 is resistant to noisy training data sets as often found in real-world settings.

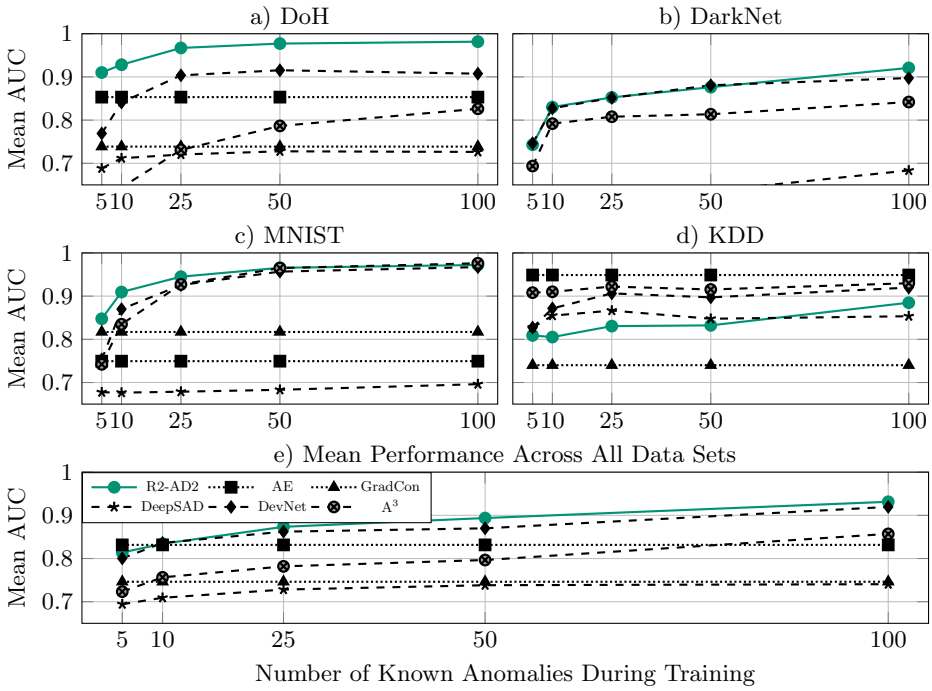

Fig. 3: Detection performance depending on the number of known anomalies during training. The training data was not polluted by unknown anomalies.

5.3 Number of Known Anomalies

In our next experiment, we evaluated the impact of the number of known anomalies available during training. We gradually decreased the amount towards unsupervised regimes shown in Figure 3. As the known anomalies were randomly selected among all anomaly classes, some classes might have been excluded during training.

As expected, the unsupervised methods remained at their initial performance as they do not incorporate known anomalies. R2-AD2 exceeded the performance of the AE with as little as ten known anomalies. Looking at the mean performance, R2-AD2 was better than the semi-supervised methods across all anomaly counts. Interestingly, R2-AD2 took the lead on MNIST for small amounts of prior knowledge, i.e. less than 50 anomalies. In this experiment, we saw R2-AD2 to perform well even with little prior knowledge about known anomalies.

Table 4: Detection of unknown anomalies, i.e. there are more anomaly classes in the test set than there were in the training data. We limited the number of known anomalies to 100 and show the results after five detection runs.

	Ours		Unsupervised Baselines				Semi-supervised Baselines					
	R2-AD2		AE		GradCon		DeepSAD		DevNet		A^3	
	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP	AUC	AP
CT	.64 ± .02	.21 ± .01	**.76 ± .02**	**.25 ± .03**	.70 ± .05	.21 ± .06	.51 ± .07	.12 ± .05	.63 ± .05	.16 ± .02	.38 ± .07	.07 ± .02
DN	**.86 ± .02**	**.65 ± .02**	.54 ± .01	.23 ± .01	.62 ± .09	.24 ± .04	.51 ± .05	.29 ± .03	.62 ± .05	.28 ± .03	.56 ± .07	.23 ± .03
FMN.	.92 ± .02	.95 ± .01	.86 ± .00	.92 ± .00	.82 ± .02	.88 ± .03	.69 ± .02	.77 ± .03	**.94 ± .01**	**.96 ± .01**	.92 ± .02	.95 ± .01
IDS	**.92 ± .01**	**.89 ± .01**	.84 ± .01	.52 ± .04	.45 ± .10	.19 ± .06	.62 ± .25	.42 ± .25	.87 ± .01	.66 ± .05	.86 ± .03	.64 ± .13
KDD	.87 ± .04	.90 ± .03	**.95 ± .00**	**.95 ± .00**	.74 ± .04	.83 ± .01	.86 ± .03	.90 ± .01	.91 ± .01	.93 ± .01	.91 ± .03	.93 ± .02
MN.	.93 ± .01	**.96 ± .01**	.75 ± .01	.79 ± .01	.82 ± .04	.84 ± .04	.69 ± .02	.75 ± .01	.93 ± .01	.95 ± .00	**.94 ± .01**	**.96 ± .01**
URL	**.94 ± .01**	**.99 ± .00**	.92 ± .00	.98 ± .00	.90 ± .01	.97 ± .00	.92 ± .01	.98 ± .00	.92 ± .02	.98 ± .00	.92 ± .02	.98 ± .01
mean	.87	.79	.80	.66	.72	.59	.69	.60	.83	.70	.78	.68
p-val	-	-	.47	.30	.05	.03	.02	.03	.47	.30	.30	.30

Table 5: Mean performance depending on the number of retraining steps of the target network. We evaluated the detection of known anomalies given 100 anomalous training samples and show the results after five detection runs.

Number of AEs	1		2		3		4	
Metric	AUC	AP	AUC	AP	AUC	AP	AUC	AP
Mean Performance	.87	.77	**.93**	.83	**.93**	**.84**	.92	**.84**

5.4 Unknown Anomalies

In our final experiment, we evaluated the transfer performance, i.e. how well the knowledge about known anomalies transfers to unknown ones. In real-world setting, the known anomalies often cover only a small part of all possible anomalies. Semi-supervised AD methods are expected to use the prior knowledge about known anomalies to detect unknown ones. We simulate this setting by limiting the training anomaly classes. Also in this experiment, R2-AD2 took the lead with a mean AUC of 87% as shown in Table 4. R2-AD2 was 5 % better compared to the next best baseline, DevNet. Except for CoverType and KDD, R2-AD2 was better than the unsupervised methods. This experiment suggested that the raw gradient contains features that generalise across anomaly types.

Throughout our evaluation, we have seen R2-AD2 to deliver superior anomaly detection performance under common limitations in AD. R2-AD2 reliably detected known anomalies, yet generalised to unknown ones. Moreover, the detection performance remained at a high level even under polluted data sets and little prior knowledge about potential anomalies. With R2-AD2 we provide a reliable AD method applicable to a variety of important applications of AD.

5.5 Ablation Study

In our ablation study, we took a critical look on the temporal component of R2-AD2. We analysed the gradient of multiple training states of the target AE for anomalous patterns. Would it have been sufficient to consider a single time step only? In Table 5, we evaluated the gradient of 1, 2, 3 and 4 time steps. For the single time step case, we replaced the LSTM elements by dense layers. The mean performance considerably decreased when only evaluating a single time step. In comparison to three time steps, the AUC dropped by $-6\,\%$. A single extra time step improved the performance. Expanding the number of time steps did not result in further improvements. We conclude that the temporal gradient distribution contains features important to AD, which are not present in a static one-step analysis.

Discussion and Future Work

In R2-AD2, we expanded the analysis of the gradient space of NNs – to the best of our knowledge – the first time to semi-supervised AD. Based on our evaluation, we have seen that the temporal gradient distribution allows to reliably detect anomalous inputs under diverse extents of prior knowledge on several important fields of application. Due to the end-to-end nature of R2-AD2, it readily integrates in other application areas. We hope to spark interest in porting our framework to sequential inputs like sensor measurements or video streams. Moreover, as we have seen related work in other detection areas, e.g. OOD or adversarial detection, we see potential to apply a gradient-based analysis to other important data mining and IT security applications e.g. deepfake detection.

Summary

In this paper, we introduced R2-AD2: a semi-supervised AD method based on the analysis of the temporal gradient distribution of NNs. R2-AD2 showed superior performance in a purely data-driven way, generalising to several important applications of AD. Our evaluation motivated that R2-AD2 is less susceptible to noisy training data than other state-of-the-art AD methods and requires less known anomalies for reliable detection performance. With R2-AD2, we extend the analysis of the NN's gradient the first time to semi-supervised AD, providing a reliable AD method to researchers and practitioners.

Acknowledgement

This research was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy in the project "Cognitive Security".

Ethical Implications

Data-driven AD reveals data points that differ from the training data distribution. Underrepresented groups in the training data may cause a bias in the detection results. In example of the census data set, which we analysed during our evaluation, e.g. the origin of the citizens could be used for the anomaly decision leading to ethical implications. To this end, we encourage users of R2-AD2 and AD in general to thoroughly evaluate potential biases in the data.

References

1. Akcay, S., Atapour-Abarghouei, A., Breckon, T.P.: GANomaly: Semi-supervised Anomaly Detection via Adversarial Training. In: Jawahar, C.V., Li, H., Mori, G., Schindler, K. (eds.) Computer Vision – ACCV 2018. pp. 622–637. Lecture Notes in Computer Science, Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-20893-6“˙39
2. Beggel, L., Pfeiffer, M., Bischl, B.: Robust Anomaly Detection in Images Using Adversarial Autoencoders. In: Brefeld, U., Fromont, E., Hotho, A., Knobbe, A., Maathuis, M., Robardet, C. (eds.) Machine Learning and Knowledge Discovery in Databases. pp. 206–222. Lecture Notes in Computer Science, Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-46150-8“˙13
3. Bergman, L., Hoshen, Y.: Classification-Based Anomaly Detection for General Data. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=H1lK_lBtvS
4. Blackard, J.A., Dean, D.J.: Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. Computers and Electronics in Agriculture **24**(3), 131–151 (Dec 1999). https://doi.org/10.1016/S0168-1699(99)00046-0
5. Borghesi, A., Bartolini, A., Lombardi, M., Milano, M., Benini, L.: Anomaly Detection Using Autoencoders in High Performance Computing Systems. Proceedings of the AAAI Conference on Artificial Intelligence **33**(01), 9428–9433 (Jul 2019). https://doi.org/10.1609/aaai.v33i01.33019428
6. Chalapathy, R., Menon, A.K., Chawla, S.: Anomaly Detection using One-Class Neural Networks. arXiv:1802.06360 [cs, stat] (Jan 2019), http://arxiv.org/abs/1802.06360, arXiv: 1802.06360
7. Dhaliwal, J., Shintre, S.: Gradient Similarity: An Explainable Approach to Detect Adversarial Attacks against Deep Learning. arXiv:1806.10707 [cs] (Jun 2018), http://arxiv.org/abs/1806.10707, arXiv: 1806.10707
8. Golan, I., El-Yaniv, R.: Deep Anomaly Detection Using Geometric Transformations. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 31. Curran Associates, Inc. (2018), https://proceedings.neurips.cc/paper/2018/file/5e62d03aec0d17facfc5355dd90d441c-Paper.pdf
9. Goyal, S., Raghunathan, A., Jain, M., Simhadri, H.V., Jain, P.: DROCC: Deep Robust One-Class Classification. In: Proceedings of the 37th International Conference on Machine Learning. pp. 3711–3721. PMLR (Nov 2020), https://proceedings.mlr.press/v119/goyal20c.html, iSSN: 2640-3498

10. Habibi Lashkari, A., Kaur, G., Rahali, A.: DIDarknet: A Contemporary Approach to Detect and Characterize the Darknet Traffic using Deep Image Learning. In: 2020 the 10th International Conference on Communication and Network Security. pp. 1–13. ICCNS 2020, Association for Computing Machinery, New York, NY, USA (Nov 2020). https://doi.org/10.1145/3442520.3442521
11. Hendrycks, D., Gimpel, K.: A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In: International Conference on Learning Representations (2017), https://openreview.net/forum?id=Hkg4TI9xl
12. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep Anomaly Detection with Outlier Exposure. In: International Conference on Learning Representations (2019), https://openreview.net/forum?id=HyxCxhRcY7
13. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Computation **9**(8), 1735–1780 (Nov 1997). https://doi.org/10.1162/neco.1997.9.8.1735, conference Name: Neural Computation
14. Huang, C., Ye, F., Zhao, P., Zhang, Y., Wang, Y.F., Tian, Q.: ESAD: End-to-end Deep Semi-supervised Anomaly Detection. In: The 32nd British Machine Vision Conference (Oct 2021), https://www.bmvc2021-virtualconference.com/conference/papers/paper_0329.html
15. Huang, R., Geng, A., Li, Y.: On the Importance of Gradients for Detecting Distributional Shifts in the Wild. arXiv:2110.00218 [cs] (Oct 2021), http://arxiv.org/abs/2110.00218, arXiv: 2110.00218
16. Ioffe, S., Szegedy, C.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In: Proceedings of the 32nd International Conference on Machine Learning. pp. 448–456. PMLR (Jun 2015), https://proceedings.mlr.press/v37/ioffe15.html, iSSN: 1938-7228
17. Kwon, G., Prabhushankar, M., Temel, D., AlRegib, G.: Backpropagated Gradient Representations for Anomaly Detection. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M. (eds.) Computer Vision – ECCV 2020. pp. 206–226. Lecture Notes in Computer Science, Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-58589-1“˙13
18. Kwon, G., Prabhushankar, M., Temel, D., AlRegib, G.: Novelty Detection Through Model-Based Characterization of Neural Networks. In: 2020 IEEE International Conference on Image Processing (ICIP). pp. 3179–3183 (Oct 2020). https://doi.org/10.1109/ICIP40778.2020.9190706, iSSN: 2381-8549
19. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (Nov 1998). https://doi.org/10.1109/5.726791
20. Lee, J., AlRegib, G.: Open-Set Recognition With Gradient-Based Representations. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 469–473 (Sep 2021). https://doi.org/10.1109/ICIP42928.2021.9506430, iSSN: 2381-8549
21. Lee, J., Prabhushankar, M., AlRegib, G.: Gradient-Based Adversarial and Out-of-Distribution Detection. In: International Conference on Machine Learning (ICML) Workshop on New Frontiers in Adversarial Machine Learning (2022)
22. Li, D., Chen, D., Jin, B., Shi, L., Goh, J., Ng, S.K.: MAD-GAN: Multivariate Anomaly Detection for Time Series Data with Generative Adversarial Networks. In: Tetko, I.V., Kůrková, V., Karpov, P., Theis, F. (eds.) Artificial Neural Networks and Machine Learning – ICANN 2019: Text and Time Series. pp. 703–716. Lecture Notes in Computer Science, Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-30490-4“˙56
23. Lust, J., Condurache, A.P.: GraN: An Efficient Gradient-Norm Based Detector for Adversarial and Misclassified Examples. In: ESANN 2020. p. 6 (2020)

24. Mamun, M.S.I., Rathore, M.A., Lashkari, A.H., Stakhanova, N., Ghorbani, A.A.: Detecting Malicious URLs Using Lexical Analysis. In: Chen, J., Piuri, V., Su, C., Yung, M. (eds.) Network and System Security. pp. 467–482. Lecture Notes in Computer Science, Springer International Publishing, Cham (2016). https://doi.org/10.1007/978-3-319-46298-1“˙30
25. MontazeriShatoori, M., Davidson, L., Kaur, G., Lashkari, A.H.: Detection of DoH Tunnels using Time-series Classification of Encrypted Traffic. In: The 5th IEEE Cyber Science and Technology Congress. pp. 63–70 (Aug 2020). https://doi.org/10.1109/DASC-PICom-CBDCom-CyberSciTech49142.2020.00026
26. Pang, G., Cao, L., Chen, L., Liu, H.: Learning Representations of Ultrahigh-dimensional Data for Random Distance-based Outlier Detection. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2041–2050. KDD ’18, Association for Computing Machinery, New York, NY, USA (Jul 2018). https://doi.org/10.1145/3219819.3220042
27. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep Learning for Anomaly Detection: A Review. ACM Computing Surveys **54**(2), 38:1–38:38 (Mar 2021). https://doi.org/10.1145/3439950
28. Pang, G., Shen, C., van den Hengel, A.: Deep Anomaly Detection with Deviation Networks. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 353–362. KDD ’19, Association for Computing Machinery, New York, NY, USA (Jul 2019). https://doi.org/10.1145/3292500.3330871
29. Pozzolo, A.D., Caelen, O., Johnson, R.A., Bontempi, G.: Calibrating Probability with Undersampling for Unbalanced Classification. In: 2015 IEEE Symposium Series on Computational Intelligence. pp. 159–166 (Dec 2015). https://doi.org/10.1109/SSCI.2015.33
30. Ruff, L., Kauffmann, J.R., Vandermeulen, R.A., Montavon, G., Samek, W., Kloft, M., Dietterich, T.G., Müller, K.R.: A Unifying Review of Deep and Shallow Anomaly Detection. Proceedings of the IEEE pp. 1–40 (2021). https://doi.org/10.1109/JPROC.2021.3052449
31. Ruff, L., Vandermeulen, R.A., Görnitz, N., Binder, A., Müller, E., Müller, K.R., Kloft, M.: Deep Semi-Supervised Anomaly Detection. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=HkgH0TEYwH
32. Salehi, M., Mirzaei, H., Hendrycks, D., Li, Y., Rohban, M.H., Sabokrou, M.: A Unified Survey on Anomaly, Novelty, Open-Set, and Out-of-Distribution Detection: Solutions and Future Challenges. arXiv:2110.14051 [cs] (Oct 2021), http://arxiv.org/abs/2110.14051, arXiv: 2110.14051
33. Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., Schmidt-Erfurth, U.: f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. Medical Image Analysis **54**, 30–44 (May 2019). https://doi.org/10.1016/j.media.2019.01.010
34. Schulze, J.P., Sperl, P., Böttinger, K.: DA3G: Detecting Adversarial Attacks by Analysing Gradients. In: Bertino, E., Shulman, H., Waidner, M. (eds.) Computer Security – ESORICS 2021. pp. 563–583. Lecture Notes in Computer Science, Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-88418-5“˙27
35. Sharafaldin, I., Lashkari, A.H., Ghorbani, A.A.: Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: ICISSP. pp. 108–116 (2018)

36. Sohn, K., Li, C.L., Yoon, J., Jin, M., Pfister, T.: Learning and Evaluating Representations for Deep One-Class Classification. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=HCSgyPUfeDj
37. Sperl, P., Schulze, J.P., Böttinger, K.: Activation Anomaly Analysis. In: Hutter, F., Kersting, K., Lijffijt, J., Valera, I. (eds.) Machine Learning and Knowledge Discovery in Databases. pp. 69–84. Lecture Notes in Computer Science, Springer International Publishing (2021). https://doi.org/10.1007/978-3-030-67661-2‟5
38. Sun, J., Yang, L., Zhang, J., Liu, F., Halappanavar, M., Fan, D., Cao, Y.: Gradient-based Novelty Detection Boosted by Self-supervised Binary Classification. arXiv:2112.09815 [cs] (Dec 2021), http://arxiv.org/abs/2112.09815, arXiv: 2112.09815
39. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A.: A detailed analysis of the KDD CUP 99 data set. In: 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications. pp. 1–6 (Jul 2009). https://doi.org/10.1109/CISDA.2009.5356528, iSSN: 2329-6275
40. Vu, H.S., Ueta, D., Hashimoto, K., Maeno, K., Pranata, S., Shen, S.M.: Anomaly Detection with Adversarial Dual Autoencoders. arXiv:1902.06924 [cs] (Feb 2019), http://arxiv.org/abs/1902.06924, arXiv: 1902.06924
41. Wilcoxon, F.: Individual Comparisons by Ranking Methods. In: Kotz, S., Johnson, N.L. (eds.) Breakthroughs in Statistics: Methodology and Distribution, pp. 196–202. Springer Series in Statistics, Springer, New York, NY (1992). https://doi.org/10.1007/978-1-4612-4380-9‟16
42. Woods, K.S., Doss, C.C., Bowyer, K.W., Solka, J.L., Priebe, C.E., Kegelmeyer, W.P.: Comparative evaluation of pattern recognition techniques for detection of microcalcifications in mammography. International Journal of Pattern Recognition and Artificial Intelligence **07**(06), 1417–1436 (Dec 1993). https://doi.org/10.1142/S0218001493000698
43. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. arXiv:1708.07747 [cs, stat] (Sep 2017), http://arxiv.org/abs/1708.07747, arXiv: 1708.07747
44. Yamanaka, Y., Iwata, T., Takahashi, H., Yamada, M., Kanai, S.: Autoencoding Binary Classifiers for Supervised Anomaly Detection. In: Nayak, A.C., Sharma, A. (eds.) PRICAI 2019: Trends in Artificial Intelligence. pp. 647–659. Lecture Notes in Computer Science, Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-29911-8‟50
45. Ye, Z., Chen, Y., Zheng, H.: Understanding the Effect of Bias in Deep Anomaly Detection. In: Zhou, Z.H. (ed.) Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21. pp. 3314–3320. International Joint Conferences on Artificial Intelligence Organization (Aug 2021). https://doi.org/10.24963/ijcai.2021/456, https://doi.org/10.24963/ijcai.2021/456