DEEP POSITIVE-UNLABELED ANOMALY DETECTION FOR CONTAMINATED UNLABELED DATA

Anonymous authors
Paper under double-blind review

ABSTRACT

Semi-supervised anomaly detection has attracted attention, which aims to improve the anomaly detection performance by using a small amount of labeled anomaly data in addition to unlabeled data. Existing semi-supervised approaches assume that most unlabeled data are normal, and train anomaly detectors by minimizing the anomaly scores for the unlabeled data while maximizing those for the labeled anomaly data. However, in practice, the unlabeled data are often contaminated with anomalies. This weakens the effect of maximizing the anomaly scores for anomalies, and prevents us from improving the detection performance. To solve this, we propose the deep positive-unlabeled anomaly detection framework, which integrates positive-unlabeled learning with deep anomaly detection models such as autoencoders and deep support vector data descriptions. Our approach enables the approximation of anomaly scores for normal data using the unlabeled data and the labeled anomaly data. Therefore, without labeled normal data, our approach can train anomaly detectors by minimizing the anomaly scores for normal data while maximizing those for the labeled anomaly data. Our approach achieves better detection performance than existing approaches on various datasets.

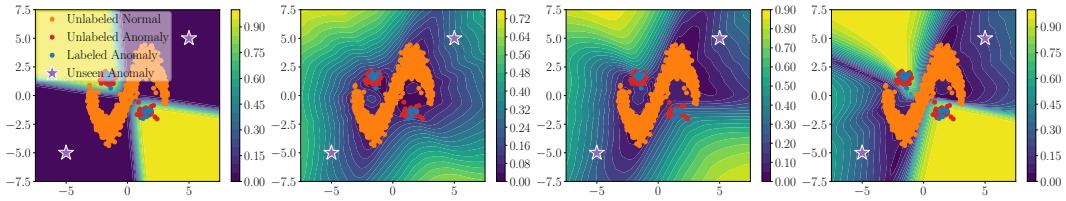
1 INTRODUCTION

Anomaly detection, which aims to identify unusual data points, is an important task in machine learning (Ruff et al., 2021). It has been performed in various fields such as cyber-security (Kwon et al., 2019), infrastructure monitoring (Borghesi et al., 2019), novelty detection (Marchi et al., 2015), medical diagnosis (Litjens et al., 2017), and natural sciences (Min et al., 2017; Cerri et al., 2019; Pracht et al., 2020).

In general, anomaly detection is performed by unsupervised learning because it does not require expensive and time-consuming labeling. Unsupervised approaches assume that most unlabeled data are normal, and try to detect anomalies by using an anomaly score, which represents the difference from normal data (Hinton & Salakhutdinov, 2006; Ruff et al., 2018). Although these approaches are easy to handle, their detection performance is limited because they cannot use information about anomalies. To improve the detection performance, semi-supervised anomaly detection uses a small amount of labeled anomaly data in addition to unlabeled data. Existing semi-supervised approaches train anomaly detectors to minimize the anomaly scores for the unlabeled data, and to maximize those for the labeled anomaly data (Hendrycks et al., 2018; Ruff et al., 2019; Yamanaka et al., 2019). However, in practice, the unlabeled data are often contaminated with anomalies. This weakens the effect of maximizing the anomaly scores for anomalies, and prevents us from improving the anomaly detection performance. This frequently occurs because it is difficult to label all anomalies.

To handle contaminated unlabeled data, we propose the deep positive-unlabeled anomaly detection framework, which integrates positive-unlabeled (PU) learning (Du Plessis et al., 2014; 2015; Kiryo et al., 2017) with deep anomaly detectors such as the autoencoder (AE) (Hinton & Salakhutdinov, 2006) and the deep support vector data description (DeepSVDD) (Ruff et al., 2018). PU learning assumes that an unlabeled data distribution is a mixture of normal and anomaly data distributions[1].

[1]Note that the anomaly data in the training dataset follow the anomaly data distribution, but new types of anomalies, unseen during training, may NOT follow this distribution. In general, no distribution can fully represent all possible anomalies.



(a) PU learning (b) Unsupervised (c) Semi-supervised (d) Proposed

Figure 1: The comparison of the PU learning, the unsupervised anomaly detector (DAE), the semi-supervised anomaly detector (ABC), and the proposed method on the toy dataset. The unlabeled data in this dataset include both normal and anomaly data points. The yellow and blue in the contour maps represent abnormality and normality, respectively. The purple stars represent the examples of unseen anomalies, which are new types of anomalies unseen during training.

Accordingly, the normal data distribution is approximated by using the unlabeled and anomaly data distributions. With this assumption, we approximate the anomaly scores for normal data using the unlabeled data and the labeled anomaly data. Therefore, without labeled normal data, we can train anomaly detectors to minimize the anomaly scores for normal data, and to maximize those for the labeled anomaly data. Compared to existing semi-supervised approaches designed to handle contaminated unlabeled data (Zhang et al., 2018; Ju et al., 2020; Zhang et al., 2021; Pang et al., 2023; Li et al., 2023; Perini et al., 2023), our approach is theoretically justified from the perspective of unbiased PU learning (Du Plessis et al., 2014; 2015; Kiryo et al., 2017).

Figure 1 compares the PU learning (Kiryo et al., 2017), the unsupervised detector, the semi-supervised detector, and our approach on the toy dataset. We used the denoising AE (DAE) (Vincent et al., 2008) for the unsupervised detector, and the autoencoding binary classifier (ABC) (Yamanaka et al., 2019) for the semi-supervised detector. Our approach is based on the DAE. The toy dataset consists of unlabeled and anomaly data, where the unlabeled data include both normal and anomaly data points.

We first focus on the PU learning, which aims to train the binary classifier from the unlabeled data and the labeled anomaly data. This can detect *seen anomalies*, which are similar to anomalies included in the training dataset. However, since its decision boundary is between normal data points and seen anomalies, it cannot detect *unseen anomalies*, which are new types of anomalies unseen during training, such as novel anomalies and zero-day attacks (Wang et al., 2013; Pang et al., 2021; Ding et al., 2022). We next focus on unsupervised and semi-supervised detectors. They can detect unseen anomalies to some extent since they try to detect anomalies by using the difference from normal data. The unsupervised detector cannot detect seen anomalies since it cannot use information about anomalies. The semi-supervised detector can detect seen anomalies to some extent since it can use the labeled anomaly data. However, the contaminated dataset weakens the effect of maximizing the anomaly scores for anomalies in the semi-supervised detector. Finally, we focus on our approach. Our approach can detect seen anomalies according to the effectiveness of PU learning, and can detect unseen anomalies to some extent according to the effectiveness of the deep anomaly detector.

Our framework is applicable to various anomaly detectors. When selecting a detector, we require that its loss function be non-negative and differentiable. In this paper, we apply our framework to the AE and the DeepSVDD. We refer to the former as the positive-unlabeled autoencoder (PUAE), and the latter as the positive-unlabeled support vector data description (PUSVDD).

Our contributions can be summarized as follows:

- To handle contaminated unlabeled data, we propose the deep positive-unlabeled anomaly detection framework, which integrates unbiased PU learning with deep anomaly detectors such as the AE and the DeepSVDD.
- We demonstrated that our approach outperformed existing approaches including the current state-of-the-art approach (Li et al., 2023) on various datasets.

2 PRELIMINARIES

In this section, we first explain our problem setup. Next, we review the AE (Hinton & Salakhutdinov, 2006) and the ABC (Yamanaka et al., 2019). They are typical unsupervised and semi-supervised anomaly detection approaches, and our framework can be applied to them.

2.1 PROBLEM SETUP

We can use unlabeled dataset $\mathcal{U} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ and anomaly dataset $\mathcal{A} = \{\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_M\}$ for training. $\mathcal{U}$ contains both normal data points and seen anomalies that are similar to those in $\mathcal{A}$. The test dataset contains normal data points, seen anomalies, and unseen anomalies that are new types of anomalies unseen during training. Our goal is to obtain a high-performance detector with $\mathcal{U}$ and $\mathcal{A}$.

2.2 AUTOENCODER

For unsupervised anomaly detectors, we train them only using the unlabeled dataset $\mathcal{U}$. As an example, we focus on the AE, which has been successfully applied to anomaly detection (Sakurada & Yairi, 2014). The AE is presented for representation learning, which learns the representation of data points through data reconstruction. Let $\mathbf{x}$ be a data point and $\mathbf{z}$ be its low-dimensional latent representation. The AE consists of two neural networks: encoder $E_\theta(\mathbf{x})$ and decoder $D_\theta(\mathbf{z})$, where θ is the parameter of these neural networks. $E_\theta(\mathbf{x})$ maps a data point $\mathbf{x}$ into a low-dimensional latent representation $\mathbf{z}$, and $D_\theta(\mathbf{z})$ reconstructs the original data point $\mathbf{x}$ from the latent representation $\mathbf{z}$. The reconstruction error for each data point $\mathbf{x}$ in the AE is defined as follows:

$$\ell(\mathbf{x}; \theta) = \left\| D_\theta(E_\theta(\mathbf{x})) - \mathbf{x} \right\|, \tag{1}$$

where $\|\cdot\|$ represents ℓ_2 norm. When the AE is used for anomaly detection, all unlabeled data points are assumed to be normal. We train the AE by minimizing the following objective function:

$$\mathcal{L}_{\mathrm{AE}}(\theta) = \frac{1}{N} \sum_{n=1}^{N} \ell(\mathbf{x}_n; \theta). \tag{2}$$

After training, the AE is expected to successfully reconstruct normal data and fail to reconstruct anomaly data because the training dataset is assumed to contain only normal data and no anomaly data. Hence, the reconstruction error can be used for the anomaly score.

Although unsupervised approaches are widely used, their detection performance is limited because they cannot use information about anomalies.

2.3 AUTOENCODING BINARY CLASSIFIER

Semi-supervised anomaly detection aims to improve the anomaly detection performance using the unlabeled dataset $\mathcal{U}$ and the anomaly dataset $\mathcal{A}$. A number of studies have been presented such as the ABC (Yamanaka et al., 2019), the deep semi-supervised anomaly detection (DeepSAD) (Ruff et al., 2019), and the outlier exposure (Hendrycks et al., 2018). Here, we focus on the ABC, which is based on the AE.

Let $y = 0$ be normal and $y = 1$ be anomaly. The ABC models the conditional probability of y given $\mathbf{x}$ by using the reconstruction error $\ell(\mathbf{x}; \theta)$ as follows:

$$p_\theta(y|\mathbf{x}) = \begin{cases} \exp(-\ell(\mathbf{x}; \theta)) & (y = 0) \\ 1 - \exp(-\ell(\mathbf{x}; \theta)) & (y = 1) \end{cases}. \tag{3}$$

A small reconstruction error results in a higher probability of normality $p_\theta(y = 0|\mathbf{x})$, while a large reconstruction error results in a higher probability of abnormality $p_\theta(y = 1|\mathbf{x})$. With this conditional probability, the ABC introduces the binary cross entropy as the loss function for each data point as follows:

$$\ell_{\mathrm{BCE}}(\mathbf{x}, y; \theta) = -\log p_\theta(y|\mathbf{x}) = (1 - y)\ell(\mathbf{x}; \theta) - y \log(1 - \exp(-\ell(\mathbf{x}; \theta))). \tag{4}$$

Like the AE, the ABC assumes all unlabeled data points to be normal. The ABC is trained by minimizing the following objective function:

$$\mathcal{L}_{\mathrm{ABC}}(\theta) = \frac{1}{N}\sum_{n=1}^{N} \ell_{\mathrm{BCE}}(\mathbf{x}_n, 0; \theta) + \frac{1}{M}\sum_{m=1}^{M} \ell_{\mathrm{BCE}}(\tilde{\mathbf{x}}_m, 1; \theta). \tag{5}$$

This minimizes the reconstruction errors for the unlabeled data and maximizes those for the anomaly data. Hence, after training, the AE becomes to reconstruct the unlabeled data assumed to be normal, and fail to reconstruct anomalies. Other semi-supervised anomaly detection approaches such as the DeepSAD (Ruff et al., 2019) and the outlier exposure (Hendrycks et al., 2018) also minimize the anomaly scores for the unlabeled data and maximize those for the anomaly data.

However, the unlabeled dataset $\mathcal{U}$ is often contaminated with anomalies in practice. This weakens the effect of maximizing the anomaly scores for anomalies, and prevents us from improving the detection performance. This frequently occurs because it is difficult to label all anomalies in the unlabeled dataset.

3 Proposed Method

We aim to improve the detection performance even if the unlabeled dataset $\mathcal{U}$ contains anomalies. To handle contaminated unlabeled data, we propose the deep positive-unlabeled anomaly detection framework, which integrates PU learning (Du Plessis et al., 2014; 2015; Kiryo et al., 2017) with deep anomaly detection models such as the AE and the DeepSVDD. We refer to the former as the positive-unlabeled autoencoder (PUAE), and the latter as the positive-unlabeled support vector data description (PUSVDD). We also refer anomalies as positive (+) samples, and normal data points as negative (-) samples.

3.1 Positive-Unlabeled Autoencoder

At first, we explain the PUAE. Let $p_{\mathcal{N}}$ be the normal data distribution, $p_{\mathcal{A}}$ be the seen anomaly distribution, and $p_{\mathcal{U}}$ be the unlabeled data distribution. We assume that the datasets $\mathcal{U}$ and $\mathcal{A}$ are drawn from $p_{\mathcal{U}}$ and $p_{\mathcal{A}}$, respectively. We also assume that $p_{\mathcal{U}}$ can be rewritten as follows:

$$p_{\mathcal{U}}(\mathbf{x}) = \alpha p_{\mathcal{A}}(\mathbf{x}) + (1-\alpha) p_{\mathcal{N}}(\mathbf{x}), \tag{6}$$

where $\alpha \in [0, 1]$ is the probability of anomaly occurrence in the unlabeled data. Hence, $p_{\mathcal{N}}$ can be rewritten as follows:

$$(1-\alpha) p_{\mathcal{N}}(\mathbf{x}) = p_{\mathcal{U}}(\mathbf{x}) - \alpha p_{\mathcal{A}}(\mathbf{x}). \tag{7}$$

Although α is the hyperparameter and is assumed to be known throughout this paper, it can be estimated from the datasets $\mathcal{U}$ and $\mathcal{A}$ in conventional PU learning approaches (Menon et al., 2015; Ramaswamy et al., 2016; Jain et al., 2016; Christoffel et al., 2016).

If we have access to the normal data distribution $p_{\mathcal{N}}$, we can train the AE by minimizing the ideal objective function as follows:

$$\mathcal{L}_{\mathrm{PN}}(\theta) = \alpha \mathbb{E}_{p_{\mathcal{A}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 1; \theta)] + (1-\alpha)\mathbb{E}_{p_{\mathcal{N}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 0; \theta)], \tag{8}$$

where $\mathbb{E}[\cdot]$ is the expectation. Since we cannot access $p_{\mathcal{N}}$ in practice, we have to approximate the second term in Eq. (8). According to Eq. (7), this can be rewritten as follows:

$$(1-\alpha)\mathbb{E}_{p_{\mathcal{N}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 0; \theta)] = \mathbb{E}_{p_{\mathcal{U}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 0; \theta)] - \alpha \mathbb{E}_{p_{\mathcal{A}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 0; \theta)]. \tag{9}$$

Hence, by using the seen anomaly distribution $p_{\mathcal{A}}$ and the unlabeled data distribution $p_{\mathcal{U}}$, $\mathcal{L}_{\mathrm{PN}}(\theta)$ can be rewritten as follows:

$$\mathcal{L}_{\mathrm{PN}}(\theta) = \alpha \mathbb{E}_{p_{\mathcal{A}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 1; \theta)] + \mathbb{E}_{p_{\mathcal{U}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 0; \theta)] - \alpha \mathbb{E}_{p_{\mathcal{A}}}[\ell_{\mathrm{BCE}}(\mathbf{x}, 0; \theta)]. \tag{10}$$

With the datasets $\mathcal{U}$ and $\mathcal{A}$, we can approximate $\mathcal{L}_{\mathrm{PN}}(\theta)$ by the empirical distribution as follows:

$$\mathcal{L}_{\mathrm{PN}}(\theta) \simeq \underbrace{\alpha \frac{1}{M}\sum_{m=1}^{M} \ell_{\mathrm{BCE}}(\tilde{\mathbf{x}}_m, 1; \theta)}_{\mathcal{L}^{+}_{\mathcal{A}}(\theta)} + \underbrace{\frac{1}{N}\sum_{n=1}^{N} \ell_{\mathrm{BCE}}(\mathbf{x}_n, 0; \theta)}_{\mathcal{L}^{-}_{\mathcal{U}}(\theta)} \underbrace{- \alpha \frac{1}{M}\sum_{m=1}^{M} \ell_{\mathrm{BCE}}(\tilde{\mathbf{x}}_m, 0; \theta)}_{\mathcal{L}^{-}_{\mathcal{A}}(\theta)}. \tag{11}$$

In this equation, the sum of the second and third terms is the approximation of the anomaly scores for normal data:

$$(1-\alpha)\mathbb{E}_{p_{\mathcal{N}}}[\ell_{\mathrm{BCE}}(\mathbf{x},0;\theta)] \simeq \mathcal{L}_{\mathcal{U}}^{-}(\theta) - \alpha\mathcal{L}_{\mathcal{A}}^{-}(\theta). \tag{12}$$

The left-hand side in Eq. (12) is always greater than or equal to zero, but the right-hand side can be negative. In experiments, it often converges towards negative infinity, resulting in the meaningless solution. To avoid this, based on (Hammoudeh & Lowd, 2020), our training objective function to be minimized ensures that $\mathcal{L}_{\mathcal{U}}^{-}(\theta) - \alpha\mathcal{L}_{\mathcal{A}}^{-}(\theta)$ is not negative as follows:

$$\mathcal{L}_{\mathrm{Proposed}}(\theta) = \alpha\mathcal{L}_{\mathcal{A}}^{+}(\theta) + \left|\mathcal{L}_{\mathcal{U}}^{-}(\theta) - \alpha\mathcal{L}_{\mathcal{A}}^{-}(\theta)\right|. \tag{13}$$

We can optimize this training objective function by using the stochastic gradient descent (SGD) such as Adam (Kingma & Ba, 2015). We refer to this approach as the PUAE. Algorithm 1 in Appendix A shows the pseudo code of the PUAE.

3.2 Positive-Unlabeled Support Vector Data Description

We next apply our framework to the DeepSVDD. The DeepSVDD aims to pull the representation of the normal data towards the pre-defined center, and push those of the anomaly data away from the center. Let $f_\theta(\mathbf{x})$ be the feature extractor, like the encoder in the AE. The loss function for each data point of the DeepSVDD is defined as follows:

$$\tilde{\ell}(\mathbf{x};\theta) = \left\|f_\theta(\mathbf{x}) - \mathbf{c}\right\|^2, \tag{14}$$

where $\mathbf{c} \neq \mathbf{0}$ is the pre-defined center vector.

The DeepSAD (Ruff et al., 2019) is a semi-supervised extension of the DeepSVDD. The DeepSAD trains the DeepSVDD model to minimize Eq. (14) for the unlabeled data, and to maximize it for the anomaly data. The loss function for each data point of the DeepSAD is defined as follows:

$$\tilde{\ell}_{\mathrm{SAD}}(\mathbf{x},y;\theta) = (1-y)\tilde{\ell}(\mathbf{x};\theta) + \frac{y}{\tilde{\ell}(\mathbf{x};\theta)}. \tag{15}$$

We can apply our framework to the DeepSVDD by replacing Eq. (4) in the PUAE with Eq. (15), while keeping all other components identical to the PUAE. We refer to this approach as the PUSVDD. In this way, our framework is applicable to various anomaly detectors by substituting the loss function of the desired model into Eq. (1) in the PUAE or Eq. (14) in the PUSVDD. When selecting a detector, we require that its loss function be non-negative and differentiable.

4 Related Work

4.1 Unsupervised Anomaly Detection

Numerous unsupervised approaches have been presented, ranging from shallow approaches such as the one-class support vector machine (OCSVM) (Tax & Duin, 2004) and the isolation forest (IF) (Liu et al., 2008) to deep approaches such as the AE (Hinton & Salakhutdinov, 2006) and the DeepSVDD (Ruff et al., 2018). In addition, generative models such as the variational autoencoder (Kingma & Welling, 2014; Kingma et al., 2015) and the generative adversarial nets (Goodfellow et al., 2014) are also used for anomaly detection (Choi et al., 2018; Serrà et al., 2019; Ren et al., 2019; Perera et al., 2019; Xiao et al., 2020; Havtorn et al., 2021; Yoon et al., 2021). Although they are often used in anomaly detection, their detection performance is limited because they cannot use information about anomalies. For example, generative models may fail to detect anomalies that are obvious to the human eye (Nalisnick et al., 2018). Furthermore, these approaches assume that unlabeled data are mostly normal. However, they are contaminated with anomalies in practice, degrading the detection performance. Several unsupervised approaches have been presented to handle such contaminated unlabeled data (Zhou & Paffenroth, 2017; Qiu et al., 2022; Shang et al., 2023). Among them, the latent outlier exposure (LOE) (Qiu et al., 2022) is a representative approach. The LOE introduces the label for each data point as the latent variable, and alternates between inferring the latent label and optimizing the parameter of the base anomaly detector. Compared to these approaches, our approach can achieve better detection performance by using the unlabeled data and the labeled anomaly data, even if the unlabeled data are contaminated with anomalies. As the base detector for our approach, the AE and the DeepSVDD. In addition, our approach can also be applied to the LOE by substituting its objective function into $\mathcal{L}_{\mathcal{U}}^{-}(\theta)$ in Eq. (11).

4.2 Semi-supervised Anomaly Detection

Several semi-supervised approaches have been presented, aiming to improve the anomaly detection performance using labeled anomaly data in addition to unlabeled data (Hendrycks et al., 2018; Yamanaka et al., 2019; Ruff et al., 2019). Compared to these approaches, our approach can effectively handle the unlabeled data that are contaminated with anomalies, as described in Section 3.1.

To handle contaminated unlabeled data, several semi-supervised approaches have been presented, including PU learning approaches (Zhang et al., 2018; Ju et al., 2020; Zhang et al., 2021; Pang et al., 2023; Li et al., 2023; Perini et al., 2023). Among them, the semi-supervised outlier exposure with a limited labeling budget (SOEL) (Li et al., 2023) is the current state-of-the-art approach. The SOEL is a semi-supervised extension of the LOE (Qiu et al., 2022), and presents the query strategy for the LOE, deciding which data should be labeled. Compared to these approaches, our approach is theoretically justified from the perspective of unbiased PU learning (Du Plessis et al., 2014; 2015; Kiryo et al., 2017). In addition, our approach achieved equal to or better performance than the SOEL on various datasets, as described in Section 5.

4.3 Positive-Unlabeled Learning

A lot of PU learning approaches have been presented for binary classification (Elkan & Noto, 2008; Du Plessis et al., 2014; 2015; Kiryo et al., 2017; Bekker & Davis, 2020; Nakajima & Sugiyama, 2023). Among them, our approach is based on the unbiased PU learning (Du Plessis et al., 2014; 2015; Kiryo et al., 2017). The empirical risk estimators in Eq. (11) is unbiased and consistent with respect to all popular loss function. This means for fixed θ, Eq. (11), which is the approximation of Eq. (8), converges to Eq. (8) as the dataset sizes $N, M \to \infty$. It is known that if the model is linear with respect to θ, a particular loss function will result in convex optimization, and the globally optimal solution can be obtained (Natarajan et al., 2013; Patrini et al., 2016; Niu et al., 2016). Despite this ideal property, when the model is complex such as neural networks, Eq. (11) can become negative, potentially leading to the meaningless solution. To address this issue, following (Hammoudeh & Lowd, 2020), we take its absolute value, as described in Section 3.1. Compared to conventional PU learning that is based on the binary classifier, our approach is based on deep anomaly detection models such as the AE and the DeepSVDD. Although conventional PU learning cannot detect unseen anomalies since its decision boundary is between normal data points and seen anomalies, our approach can detect both seen and unseen anomalies.

5 Experiments

5.1 Data

To evaluate our approach, we used following eight image datasets: MNIST (Salakhutdinov & Murray, 2008), FashionMNIST (Xiao et al., 2017), SVHN (Netzer et al., 2011), CIFAR10 (Krizhevsky et al., 2009), CIFAR100 (Krizhevsky et al., 2009), Path, OCT, and Tissue (Yang et al., 2021; 2023).

First, we explain the first four datasets. MNIST is the handwritten digits, FashionMNIST is the fashion product images, SVHN is the house number digits, and CIFAR10 is the animal and vehicle images. We resized all datasets to 32×32 resolution. These datasets consist of 10 class images. For MNIST and SVHN, we used the digits as class indices. For FashionMNIST, we indexed labels as: $\{$T-shirt/top: 0, Trouser: 1, Pullover: 2, Dress: 3, Coat: 4, Sandal: 5, Shirt: 6, Sneaker: 7, Bag: 8, and Ankle boot: 9$\}$. For CIFAR10, we indexed labels as: $\{$airplane: 0, automobile: 1, bird: 2, cat: 3, deer: 4, dog: 5, frog: 6, horse: 7, ship: 8, and truck: 9$\}$. We extend the experiments in (Ruff et al., 2018) to semi-supervised anomaly detection with contaminated unlabeled data. Of the 10 classes, we used one class as normal, another class as unseen anomaly, and the remaining classes as seen anomaly. For example with MNIST, if we use the digit 1 as normal and the digit 0 as unseen anomaly, seen anomaly corresponds to the digits 2, 3, 4, 5, 6, 7, 8, and 9. For all datasets, we used class 0 as unseen anomaly, and select one normal class from the remaining 9 classes. The training dataset consists of 5,000 samples, of which 4,500 samples are unlabeled normal data points, 250 samples are labeled seen anomalies, and 250 samples are unlabeled seen anomalies. That is, the unlabeled data points in this dataset are contaminated with seen anomalies. We used 10% of the training dataset as the validation dataset. The test dataset consists of 2,000 samples, about half of which are normal

and the rest are anomalies, including both seen and unseen anomalies. More specifically, normal data points are sampled from the normal class in the test dataset, with a maximum of 1,000 samples, while both seen and unseen anomalies are sampled from their respective classes, with a maximum of 500 samples each. The example of the MNIST dataset is provided in Appendix B.

Next, we explain the last four datasets. CIFAR100 is just like CIFAR10 but consists of 100 classes. These classes are grouped into 20 superclasses, from which we used nature-related classes as normal and human-related classes as anomalies. We used the people class as unseen anomaly. Path, OCT, and Tissue are real medical image datasets, which are also used in (Li et al., 2023). Path is colorectal cancer histology dataset with 9 tissue types. OCT is retinal optical coherence tomography dataset with 4 diagnostic categories. Tissue is kidney cortex cell dataset with 8 categories. For Path and Tissue, we used the first class as unseen anomaly, selected one class from the remaining ones as normal, and treated the rest as seen anomaly. For OCT, since a pre-defined normal class exists, we used it as normal, used the first class as unseen anomaly, and treated the remaining classes as seen anomaly.

5.2 METHODS

For comparison with our PUAE and PUSVDD, we used the following unsupervised and semi-supervised approaches.

Unsupervised approaches: We used the IF (Liu et al., 2008) as the shallow approach, and used the AE (Hinton & Salakhutdinov, 2006) and the DeepSVDD (Ruff et al., 2018) as the deep approaches. We also used the LOE (Qiu et al., 2022), which is robust to the anomalies in the unlabeled data. We chose the DeepSVDD as the base detector for the LOE.

Semi-supervised approaches: We used the ABC (Yamanaka et al., 2019), the DeepSAD (Ruff et al., 2019), and the SOEL (Li et al., 2023) that is a semi-supervised extension of the LOE. We chose the DeepSVDD as the base detector for the SOEL. We also used the PU learning binary classifier (PU) (Kiryo et al., 2017) for reference.

5.3 SETUP

First, we outline the setups for all approaches except the IF. We used convolutional neural networks for the AE, the DeepSVDD, and the PU. The network architecture follows (Ruff et al., 2018). For the AE-based and DeepSVDD-based approaches, we set the dimension of the latent variable to 128. For the DeepSVDD-based approaches, we used no bias terms in each layer, pre-trained these feature extractors as the AE, and set the center $\mathbf{c}$ in Eq. (14) to the mean of the outputs of the encoder. We trained all methods by using Adam (Kingma & Ba, 2015) with a mini-batch size of 128. We set the learning rate to 10^{-4} and the maximum number of epochs to 200. We also used the weight decay (Goodfellow et al., 2016) with 10^{-3} and used early-stopping (Goodfellow et al., 2016) based on the validation dataset. We set the hyperparameter $\alpha = 0.1$ for the PUAE, the PUSVDD, the PU, the LOE, and the SOEL, which is the probability of anomaly occurrence. Next, we outline the setup for the IF. We used the scikit-learn implementation (Pedregosa et al., 2011) and kept all hyperparameters at their default values in our experiments.

We trained unsupervised approaches using the unlabeled data[2], while we trained semi-supervised approaches using both the unlabeled data and the labeled anomaly data. To measure the detection performance, we calculated the AUROC scores for all datasets. We ran all experiments five times while changing the random seeds. The machine specifications used in the experiments are as follows: the CPU is AMD EPYC 9124 16-Core Processor, the memory size is 512GB, and the GPU is NVIDIA RTX 6000 Ada.

5.4 RESULTS

Tables 1 and 2 compare the detection performance on each dataset. We showed the average of the AUROC scores for all normal classes. We used bold to highlight the best results and statistically non-different results according to a pair-wise t-test. We used 5% as the p-value.

[2]Note that we did NOT use the labeled anomaly data since unsupervised approaches cannot effectively use them.

Table 1: Comparison of anomaly detection performance on MNIST, FashionMNIST, SVHN, and CIFAR10.

	MNIST	FashionMNIST	SVHN	CIFAR10
IF	0.885 ± 0.062	0.916 ± 0.077	0.501 ± 0.014	0.610 ± 0.095
AE	0.912 ± 0.042	0.841 ± 0.102	0.562 ± 0.040	0.535 ± 0.120
DeepSVDD	0.937 ± 0.045	0.921 ± 0.088	0.582 ± 0.035	0.709 ± 0.054
LOE	0.945 ± 0.033	0.916 ± 0.094	0.624 ± 0.054	0.718 ± 0.062
ABC	0.916 ± 0.042	0.841 ± 0.104	0.562 ± 0.041	0.535 ± 0.119
DeepSAD	0.942 ± 0.041	0.928 ± 0.089	0.652 ± 0.034	0.726 ± 0.051
SOEL	0.965 ± 0.026	0.936 ± 0.079	0.727 ± 0.043	0.775 ± 0.057
PU	0.962 ± 0.034	0.922 ± 0.092	0.681 ± 0.096	0.693 ± 0.120
PUAE	0.983 ± 0.015	0.918 ± 0.085	0.689 ± 0.060	0.667 ± 0.066
PUSVDD	$\mathbf{0.989 \pm 0.012}$	$\mathbf{0.948 \pm 0.079}$	$\mathbf{0.747 \pm 0.080}$	$\mathbf{0.803 \pm 0.046}$

Table 2: Comparison of anomaly detection performance on CIFAR100, Path, OCT and Tissue.

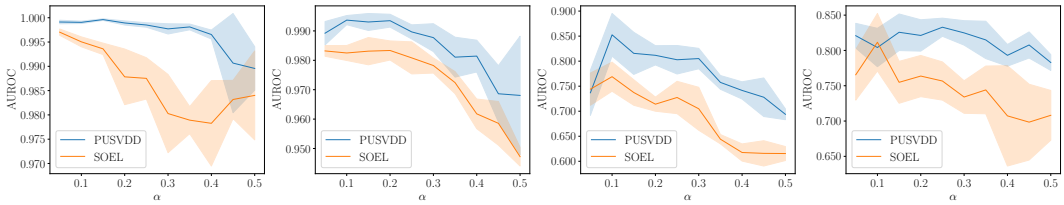
	CIFAR100	Path	OCT	Tissue
IF	0.604 ± 0.004	$\mathbf{0.809 \pm 0.118}$	0.714 ± 0.004	0.472 ± 0.187
AE	0.589 ± 0.010	0.605 ± 0.240	$\mathbf{0.860 \pm 0.005}$	0.468 ± 0.178
DeepSVDD	0.587 ± 0.026	0.759 ± 0.148	0.726 ± 0.052	0.661 ± 0.055
LOE	0.576 ± 0.035	0.721 ± 0.160	0.783 ± 0.030	0.635 ± 0.086
ABC	0.590 ± 0.010	0.604 ± 0.241	$\mathbf{0.857 \pm 0.001}$	0.472 ± 0.177
DeepSAD	0.594 ± 0.012	0.763 ± 0.187	$\mathbf{0.823 \pm 0.038}$	0.683 ± 0.053
SOEL	$\mathbf{0.633 \pm 0.013}$	0.791 ± 0.145	$\mathbf{0.856 \pm 0.016}$	0.703 ± 0.062
PU	0.541 ± 0.025	$\mathbf{0.807 \pm 0.132}$	0.614 ± 0.109	0.633 ± 0.082
PUAE	0.623 ± 0.014	$\mathbf{0.776 \pm 0.168}$	$\mathbf{0.847 \pm 0.011}$	0.594 ± 0.104
PUSVDD	$\mathbf{0.637 \pm 0.017}$	$\mathbf{0.831 \pm 0.152}$	$\mathbf{0.857 \pm 0.017}$	$\mathbf{0.731 \pm 0.077}$

At first, we focus on unsupervised approaches. The IF and the AE show significant performance variations across different datasets. Although the IF performed well on Path and the AE performed well on OCT, their performance became poor on other datasets. The DeepSVDD outperformed the IF and the AE, and the LOE, which is based on the DeepSVDD, often performed better than the DeepSVDD. However, since the LOE estimates anomalies within the contaminated unlabeled data in an unsupervised manner, incorrect estimations can lead to degraded performance.

Next, we focus on semi-supervised approaches. The ABC and the DeepSAD performed equal to or slightly better than the AE and the DeepSVDD, respectively. The reason for this is that ABC and DeepSAD assume that the unlabeled data are not contaminated with anomalies, which weakens the effect of maximizing anomaly scores for the labeled anomaly data. On the other hand, the SOEL outperformed DeepSAD in all cases. This is because the SOEL is capable of handling anomalies present in the unlabeled data.

Finally, we focus on the proposed methods. In most cases, the PUAE and the PUSVDD performed better than the AE and the ABC, the DeepSVDD and the DeepSAD, respectively. Especially, the PUSVDD achieved the best performance across all datasets. These results strongly indicate the effectiveness of our framework, which integrates PU learning with deep anomaly detection models. The detection performance under varying numbers of unlabeled anomalies are shown in Appendix C, where our PUSVDD achieves performance equal to or better than that of SOEL.

We also focus on the difference between the proposed methods and PU. The proposed methods performed equal to or better than the PU in all datasets. The reason is as follows. Since the PU is for binary classification, it sets the decision boundary between normal data points and seen anomalies. This prevents us from detecting unseen anomalies. On the other hand, our approach can detect



(a) MNIST (b) FashionMNIST (c) SVHN (d) CIFAR10

Figure 2: Relationship between the anomaly detection performance and the hyperparameter α of the PUSVDD and the SOEL on each dataset. We used class 0 as unseen anomaly, class 1 as normal, and the remaining classes as seen anomalies. The semi-transparent area represents standard deviations.

unseen anomalies since it can model normal data points by the anomaly detector. The detection performance for seen and unseen anomalies are shown in Appendix D. We also show the additional experiments using the MVTec Anomaly Detection dataset (MVTec AD) (Bergmann et al., 2021; 2019) in Appendix E.

5.5 Hyperparameter Sensitivity

Our approach and the SOEL have the hyperparameter α, which represents the probability of anomaly occurrence. In the above experiments, we set it to $\alpha = 0.1$ since the 10% of the training dataset is anomalies. Finally, we evaluate the sensitivity of α. Figure 2 shows the relationship between the detection performance and the hyperparameter α of the PUSVDD and the SOEL on each dataset.

In most datasets, the PUSVDD and the SOEL achieve the best performance around $\alpha = 0.1$. This indicates that, as with conventional PU learning, it is important to set α accurately. Note that α can be estimated from the unlabeled and anomaly training data using conventional PU learning approaches (Menon et al., 2015; Ramaswamy et al., 2016; Jain et al., 2016; Christoffel et al., 2016).

Compared to the SOEL, the PUSVDD is more robust to variations in α. In other words, even if α deviates from the true value, the PUSVDD maintains relatively stable performance. This is because α in the SOEL is closely related to the number of anomalies within the unlabeled data. If α deviates from the true value, normal data may be incorrectly treated as anomalies, or vice versa. On the other hand, since α in the PUSVDD only adjusts the weight of the loss function, it is expected to be relatively robust to deviations in α.

6 Conclusion and Limitations

Although most unlabeled data are assumed to be normal in semi-supervised anomaly detection, they are often contaminated with anomalies in practice, which degrades the detection performance. To solve this, we propose the deep positive-unlabeled anomaly detection framework, which integrates PU learning with deep anomaly detection models such as the AE and the DeepSVDD. Our approach enables us to approximate the anomaly scores for normal data with the unlabeled data and labeled anomaly data. Therefore, without the labeled normal data, we can train the anomaly detector to minimize the anomaly scores for normal data, and to maximize those for the anomaly data. Our approach achieves better detection performance than existing approaches on various image datasets.

A limitation of our approach lies in the assumption that the unlabeled data do not contain unseen anomalies, which implies that our approach lacks performance guarantees when such anomalies are present. Since our approach is based on the anomaly detector, it may still detect a small number of unseen anomalies in the unlabeled data due to the inherent properties of the detector. In addition, robust anomaly detectors such as the LOE can be employed as the base detector for our framework. Providing theoretical guarantees in the presence of unseen anomalies in unlabeled data remains an important direction for our future work.

REFERENCES

Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: A survey. *Machine Learning*, 109:719–760, 2020.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9592–9600, 2019.

Paul Bergmann, Kilian Batzner, Michael Fauser, David Sattlegger, and Carsten Steger. The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129(4):1038–1059, 2021.

Andrea Borghesi, Andrea Bartolini, Michele Lombardi, Michela Milano, and Luca Benini. Anomaly detection using autoencoders in high performance computing systems. In *Proceedings of the AAAI Conference on artificial intelligence*, volume 33, pp. 9428–9433, 2019.

Olmo Cerri, Thong Q Nguyen, Maurizio Pierini, Maria Spiropulu, and Jean-Roch Vlimant. Variational autoencoders for new physics mining at the large hadron collider. *Journal of High Energy Physics*, 2019(5):1–29, 2019.

Hyunsun Choi, Eric Jang, and Alexander A Alemi. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.

Marthinus Christoffel, Gang Niu, and Masashi Sugiyama. Class-prior estimation for learning from positive and unlabeled data. In *Asian Conference on Machine Learning*, pp. 221–236. PMLR, 2016.

Choubo Ding, Guansong Pang, and Chunhua Shen. Catching both gray and black swans: Open-set supervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7388–7398, 2022.

Marthinus Du Plessis, Gang Niu, and Masashi Sugiyama. Convex formulation for learning from positive and unlabeled data. In *International conference on machine learning*, pp. 1386–1394. PMLR, 2015.

Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. *Advances in neural information processing systems*, 27, 2014.

Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 213–220, 2008.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Zayd Hammoudeh and Daniel Lowd. Learning from positive and unlabeled data with arbitrary positive shift. *Advances in Neural Information Processing Systems*, 33:13088–13099, 2020.

Jakob D Havtorn, Jes Frellsen, Søren Hauberg, and Lars Maaløe. Hierarchical vaes know what they don't know. In *International Conference on Machine Learning*, pp. 4117–4128. PMLR, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2018.

Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

Shantanu Jain, Martha White, and Predrag Radivojac. Estimating the class prior and posterior from noisy positives and unlabeled data. *Advances in neural information processing systems*, 29, 2016.

Hyunjun Ju, Dongha Lee, Junyoung Hwang, Junghyun Namkung, and Hwanjo Yu. Pumad: Pu metric learning for anomaly detection. *Information Sciences*, 523:167–183, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, 2015.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Durk P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. *Advances in neural information processing systems*, 28, 2015.

Ryuichi Kiryo, Gang Niu, Marthinus C Du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. *Advances in neural information processing systems*, 30, 2017.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Donghwoon Kwon, Hyunjoo Kim, Jinoh Kim, Sang C Suh, Ikkyun Kim, and Kuinam J Kim. A survey of deep learning-based network anomaly detection. *Cluster Computing*, 22:949–961, 2019.

Aodong Li, Chen Qiu, Marius Kloft, Padhraic Smyth, Stephan Mandt, and Maja Rudolph. Deep anomaly detection under labeling budget constraints. In *International Conference on Machine Learning*, pp. 19882–19910. PMLR, 2023.

Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez. A survey on deep learning in medical image analysis. *Medical image analysis*, 42: 60–88, 2017.

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.

Erik Marchi, Fabio Vesperini, Florian Eyben, Stefano Squartini, and Björn Schuller. A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional lstm neural networks. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 1996–2000. IEEE, 2015.

Aditya Menon, Brendan Van Rooyen, Cheng Soon Ong, and Bob Williamson. Learning from corrupted binary labels via class-probability estimation. In *International conference on machine learning*, pp. 125–134. PMLR, 2015.

Seonwoo Min, Byunghan Lee, and Sungroh Yoon. Deep learning in bioinformatics. *Briefings in bioinformatics*, 18(5):851–869, 2017.

Shota Nakajima and Masashi Sugiyama. Positive-unlabeled classification under class-prior shift: a prior-invariant approach based on density ratio estimation. *Machine Learning*, 112(3):889–919, 2023.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? *arXiv preprint arXiv:1810.09136*, 2018.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

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.

Gang Niu, Marthinus Christoffel Du Plessis, Tomoya Sakai, Yao Ma, and Masashi Sugiyama. Theoretical comparisons of positive-unlabeled learning against positive-negative learning. *Advances in neural information processing systems*, 29, 2016.

Guansong Pang, Anton van den Hengel, Chunhua Shen, and Longbing Cao. Toward deep supervised anomaly detection: Reinforcement learning from partially labeled anomaly data. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 1298–1308, 2021.

Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. Deep weakly-supervised anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1795–1807, 2023.

Giorgio Patrini, Frank Nielsen, Richard Nock, and Marcello Carioni. Loss factorization, weakly supervised learning and label noise robustness. In *International conference on machine learning*, pp. 708–717. PMLR, 2016.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2898–2906, 2019.

Lorenzo Perini, Vincent Vercruyssen, and Jesse Davis. Learning from positive and unlabeled multi-instance bags in anomaly detection. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1897–1906, 2023.

Philipp Pracht, Fabian Bohle, and Stefan Grimme. Automated exploration of the low-energy chemical space with fast quantum chemical methods. *Physical Chemistry Chemical Physics*, 22(14): 7169–7192, 2020.

Chen Qiu, Aodong Li, Marius Kloft, Maja Rudolph, and Stephan Mandt. Latent outlier exposure for anomaly detection with contaminated data. In *International Conference on Machine Learning*, pp. 18153–18167. PMLR, 2022.

Harish Ramaswamy, Clayton Scott, and Ambuj Tewari. Mixture proportion estimation via kernel embeddings of distributions. In *International conference on machine learning*, pp. 2052–2060. PMLR, 2016.

Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. *Advances in neural information processing systems*, 32, 2019.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pp. 4393–4402. PMLR, 2018.

Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations*, 2019.

Lukas Ruff, Jacob R Kauffmann, Robert A Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, 2021.

Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis*, pp. 4–11, 2014.

Ruslan Salakhutdinov and Iain Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the 25th international conference on Machine learning*, pp. 872–879. ACM, 2008.

Joan Serrà, David Álvarez, Vicenç Gómez, Olga Slizovskaia, José F Núñez, and Jordi Luque. Input complexity and out-of-distribution detection with likelihood-based generative models. In *International Conference on Learning Representations*, 2019.

Zuogang Shang, Zhibin Zhao, Ruqiang Yan, and Xuefeng Chen. Core loss: Mining core samples efficiently for robust machine anomaly detection against data pollution. *Mechanical Systems and Signal Processing*, 189:110046, 2023.

David MJ Tax and Robert PW Duin. Support vector data description. *Machine learning*, 54:45–66, 2004.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pp. 1096–1103, 2008.

Lingyu Wang, Sushil Jajodia, Anoop Singhal, Pengsu Cheng, and Steven Noel. k-zero day safety: A network security metric for measuring the risk of unknown vulnerabilities. *IEEE Transactions on Dependable and Secure Computing*, 11(1):30–44, 2013.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Zhisheng Xiao, Qing Yan, and Yali Amit. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. *Advances in neural information processing systems*, 33:20685–20696, 2020.

Yuki Yamanaka, Tomoharu Iwata, Hiroshi Takahashi, Masanori Yamada, and Sekitoshi Kanai. Autoencoding binary classifiers for supervised anomaly detection. In *PRICAI 2019: Trends in Artificial Intelligence: 16th Pacific Rim International Conference on Artificial Intelligence, Cuvu, Yanuca Island, Fiji, August 26–30, 2019, Proceedings, Part II 16*, pp. 647–659. Springer, 2019.

Jiancheng Yang, Rui Shi, and Bingbing Ni. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 191–195. IEEE, 2021.

Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, 10(1):41, 2023.

Sangwoong Yoon, Yung-Kyun Noh, and Frank Park. Autoencoding under normalization constraints. In *International Conference on Machine Learning*, pp. 12087–12097. PMLR, 2021.

Huayi Zhang, Lei Cao, Peter VanNostrand, Samuel Madden, and Elke A Rundensteiner. Elite: Robust deep anomaly detection with meta gradient. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2174–2182, 2021.

Jiaqi Zhang, Zhenzhen Wang, Jingjing Meng, Yap-Peng Tan, and Junsong Yuan. Boosting positive and unlabeled learning for anomaly detection with multi-features. *IEEE Transactions on Multimedia*, 21(5):1332–1344, 2018.

Chong Zhou and Randy C Paffenroth. Anomaly detection with robust deep autoencoders. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 665–674, 2017.

A Pseudo Code

Algorithm 1 Positive-Unlabeled Autoencoder

Require: Unlabeled and anomaly datasets $(\mathcal{U}, \mathcal{A})$, mini-batch size K, hyperparameter $\alpha \in [0, 1]$
Ensure: Model parameter θ

```
while not converged do
    Sample mini-batch B from datasets (U, A)
    Compute L_A^+(θ), L_U^-(θ), and L_A^-(θ) in Eq. (11) with B
    Set the gradient ∇_θ(αL_A^+(θ) + |L_U^-(θ) − αL_A^-(θ)|)
    Update θ with the gradient
end while
```

Algorithm 1 shows the pseudo code of the PUAE, where K is the mini-batch size for the SGD.

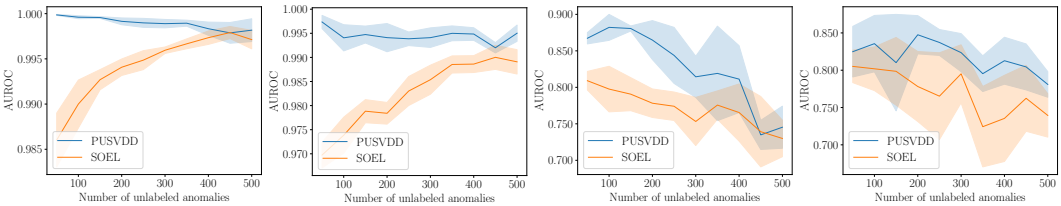
B Dataset Example

(a) Training (unlabeled) (b) Training (anomaly) (c) Test (normal) (d) Test (anomaly)

Figure 3: The example of the dataset in the case of MNIST.

Figure 3 shows the example of the dataset in the case of MNIST. In this example, the digit 1 is normal, the digit 0 is unseen anomaly, and the digits 2, 3, 4, 5, 6, 7, 8, and 9 are seen anomalies. (a) The unlabeled data points in the training dataset are contaminated with seen anomalies. (b) The anomaly data points in the training dataset contain seen anomalies but not unseen anomalies. (c) The normal data points in the test dataset are not contaminated with anomalies. (d) The anomaly data points in the test dataset contain both seen and unseen anomalies.

C Anomaly Detection Performance with Various Numbers of Unlabeled Anomalies



(a) MNIST (b) FashionMNIST (c) SVHN (d) CIFAR10

Figure 4: Comparison of anomaly detection performance between the PUSVDD and the SOEL with various numbers of unlabeled anomalies on each dataset. We used class 0 as unseen anomaly, class 1 as normal, and the remaining classes as seen anomalies. The semi-transparent area represents standard deviations.

Figure 4 shows the anomaly detection performance with various numbers of unlabeled anomalies. The hyperparameter α was set to its true value for each case. Our PUSVDD achieved equal to or better performance than the SOEL.

D Anomaly Detection Performance for Seen and Unseen Anomalies

Table 3: Seen anomaly detection performance on MNIST, FashionMNIST, SVHN, and CIFAR10.

	MNIST	FashionMNIST	SVHN	CIFAR10
IF	0.814 ± 0.096	0.915 ± 0.053	0.510 ± 0.016	0.585 ± 0.106
AE	0.843 ± 0.073	0.840 ± 0.092	0.563 ± 0.039	0.573 ± 0.131
DeepSVDD	0.925 ± 0.056	0.942 ± 0.040	0.600 ± 0.033	0.694 ± 0.069
LOE	0.942 ± 0.038	0.940 ± 0.042	0.645 ± 0.055	0.713 ± 0.077
ABC	0.852 ± 0.072	0.841 ± 0.092	0.564 ± 0.039	0.572 ± 0.131
DeepSAD	0.930 ± 0.052	0.956 ± 0.032	0.674 ± 0.031	0.716 ± 0.074
SOEL	0.967 ± 0.024	0.963 ± 0.028	0.751 ± 0.035	0.773 ± 0.073
PU	0.959 ± 0.036	0.943 ± 0.057	0.678 ± 0.099	0.703 ± 0.154
PUAE	0.980 ± 0.017	0.942 ± 0.041	0.716 ± 0.053	0.695 ± 0.083
PUSVDD	$\mathbf{0.994 \pm 0.004}$	$\mathbf{0.972 \pm 0.029}$	$\mathbf{0.787 \pm 0.069}$	$\mathbf{0.796 \pm 0.059}$

Table 4: Unseen anomaly detection performance on MNIST, FashionMNIST, SVHN, and CIFAR10.

	MNIST	FashionMNIST	SVHN	CIFAR10
IF	0.955 ± 0.031	$\mathbf{0.917 \pm 0.111}$	0.492 ± 0.018	0.635 ± 0.095
AE	0.981 ± 0.013	0.842 ± 0.130	0.560 ± 0.045	0.497 ± 0.111
DeepSVDD	0.950 ± 0.044	0.900 ± 0.143	0.564 ± 0.043	0.724 ± 0.087
LOE	0.949 ± 0.034	0.892 ± 0.152	0.602 ± 0.058	0.724 ± 0.099
ABC	0.980 ± 0.014	0.840 ± 0.133	0.559 ± 0.046	0.497 ± 0.109
DeepSAD	0.954 ± 0.040	0.900 ± 0.153	0.631 ± 0.044	0.735 ± 0.078
SOEL	0.963 ± 0.033	0.908 ± 0.135	$\mathbf{0.702 \pm 0.061}$	0.778 ± 0.087
PU	0.965 ± 0.038	0.901 ± 0.134	$\mathbf{0.684 \pm 0.102}$	0.683 ± 0.133
PUAE	$\mathbf{0.985 \pm 0.017}$	0.894 ± 0.140	0.662 ± 0.078	0.639 ± 0.090
PUSVDD	$\mathbf{0.983 \pm 0.024}$	$\mathbf{0.924 \pm 0.134}$	$\mathbf{0.708 \pm 0.103}$	$\mathbf{0.811 \pm 0.101}$

Table 5: Seen anomaly detection performance on CIFAR100, Path, OCT and Tissue.

	CIFAR100	Path	OCT	Tissue
IF	0.577 ± 0.005	0.657 ± 0.211	0.679 ± 0.005	0.443 ± 0.189
AE	0.514 ± 0.013	0.562 ± 0.253	$\mathbf{0.808 \pm 0.005}$	0.443 ± 0.188
DeepSVDD	0.623 ± 0.036	0.769 ± 0.139	0.702 ± 0.043	0.692 ± 0.047
LOE	0.624 ± 0.035	0.746 ± 0.167	0.750 ± 0.031	0.657 ± 0.084
ABC	0.516 ± 0.014	0.564 ± 0.252	$\mathbf{0.805 \pm 0.002}$	0.448 ± 0.187
DeepSAD	0.628 ± 0.042	0.772 ± 0.165	$\mathbf{0.798 \pm 0.032}$	0.715 ± 0.040
SOEL	$\mathbf{0.696 \pm 0.020}$	0.806 ± 0.142	$\mathbf{0.825 \pm 0.017}$	0.739 ± 0.044
PU	0.630 ± 0.023	$\mathbf{0.847 \pm 0.097}$	0.565 ± 0.089	0.628 ± 0.080
PUAE	0.576 ± 0.025	0.745 ± 0.171	0.800 ± 0.011	0.613 ± 0.094
PUSVDD	$\mathbf{0.700 \pm 0.021}$	$\mathbf{0.826 \pm 0.137}$	$\mathbf{0.822 \pm 0.016}$	$\mathbf{0.764 \pm 0.044}$

Table 6: Unseen anomaly detection performance on CIFAR100, Path, OCT and Tissue.

	CIFAR100	Path	OCT	Tissue
IF	0.632 ± 0.005	$\mathbf{0.961 \pm 0.055}$	0.785 ± 0.004	0.500 ± 0.187
AE	0.665 ± 0.009	0.647 ± 0.239	$\mathbf{0.965 \pm 0.005}$	0.493 ± 0.170
DeepSVDD	0.552 ± 0.033	0.749 ± 0.200	0.774 ± 0.072	0.629 ± 0.069
LOE	0.528 ± 0.047	0.696 ± 0.205	0.851 ± 0.030	0.613 ± 0.093
ABC	0.665 ± 0.009	0.644 ± 0.241	$\mathbf{0.961 \pm 0.002}$	0.496 ± 0.168
DeepSAD	0.561 ± 0.024	0.755 ± 0.242	0.874 ± 0.049	0.651 ± 0.075
SOEL	0.570 ± 0.034	0.776 ± 0.190	0.918 ± 0.021	0.666 ± 0.093
PU	0.452 ± 0.039	0.767 ± 0.215	0.712 ± 0.154	0.638 ± 0.098
PUAE	$\mathbf{0.671 \pm 0.011}$	0.806 ± 0.204	0.941 ± 0.011	0.574 ± 0.116
PUSVDD	0.573 ± 0.027	0.835 ± 0.215	0.925 ± 0.022	$\mathbf{0.697 \pm 0.114}$

As the additional experiments, we evaluate the anomaly detection performance for seen and unseen anomalies on each dataset. The setups are the same as those described in experimental section.

Tables 3, 4, 5 and 6 show the detection performance for seen and unseen anomalies, respectively. We showed the average of the AUROC scores for all normal classes. We used bold to highlight the best results and statistically non-different results according to a pair-wise t-test. We used 5% as the p-value.

For seen anomalies, the PUSVDD achieved the best performance among all approaches. This shows the effectiveness of our approach, which is robust to the contaminated unlabeled data according to PU learning.

For unseen anomalies, although the detection performance is highly dataset-dependent, the PUSVDD generally performs well. This indicates that we may be able to improve the detection performance for unseen anomalies by using seen anomalies.

These results also show the difference between the conventional PU learning and our approach. The PU achieved the poor detection performance for unseen anomalies. This is because that it sets the decision boundary between normal data points and seen anomalies. On the other hand, our approach can detect unseen anomalies since it is based on the anomaly detector.

E Anomaly Detection Performance on MVTec AD

Table 7: Comparison of anomaly detection performance on MVTec AD.

	PUSVDD	SOEL
all	$\mathbf{0.793 \pm 0.016}$	0.669 ± 0.021
seen	$\mathbf{0.799 \pm 0.013}$	0.688 ± 0.021
unseen	$\mathbf{0.787 \pm 0.020}$	0.649 ± 0.027

As the additional experiments, we evaluate the anomaly detection performance of the proposed method on the MVTec Anomaly Detection dataset (MVTec AD) (Bergmann et al., 2021; 2019). MVTec AD is an industrial inspection dataset consisting of 5,354 high-resolution images. The dataset contains 15 object categories, each associated with multiple types of anomalies. In our setup, we designated the following anomaly types as unseen anomalies: broken_large for bottle, bent_wire for cable, crack for capsule, color for carpet, bent for grid, crack for hazelnut, color for leather, bent for metal_nut, color for pill, manipulated_front for screw, crack for tile, bent_lead for transistor, color for wood, and broken_teeth for zipper. We did not set unseen anomalies for the toothbrush object because it only has one anomaly category and thus no unseen anomaly could be defined.

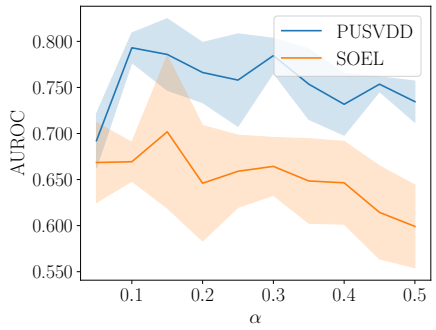


Figure 5: Relationship between the anomaly detection performance and the hyperparameter α of the PUSVDD and the SOEL on MVTec AD. The semi-transparent area represents standard deviations.

The remaining anomaly types were treated as seen anomalies. Among them, 725 images were used for training and 281 images were used for test. Out of the 725 training anomalies, 300 were used as labeled anomalies and 425 were used as unlabeled anomalies. These 425 unlabeled anomalies were combined with 3,629 normal training images to form the unlabeled training data.

In short, the training data consisted of 4,054 unlabeled samples (3,629 normal and 425 anomaly) and 300 labeled anomaly samples. The test data consisted of 467 normal images, 281 seen anomalies, and 252 unseen anomalies. All images were resized to 224×224.

For the base detector, we used the DeepSVDD that uses the pre-trained ResNet34 (He et al., 2016) as the feature extractor. The final fully connected layer of ResNet34 was replaced with the linear layer outputting a 128-dimensional embedding. All affine transformations in the batch normalization layers were disabled. We set $\alpha = 0.1$ for the PUSVDD and the SOEL. The other setups are the same as those described in experimental section.

Table 7 shows the comparison of AUROC between the PUSVDD and the SOEL. We used bold to highlight the best results and statistically non-different results according to a pair-wise t-test. We used 5% as the p-value. Figure 5 shows the relationship between the anomaly detection performance and the hyperparameter α of the PUSVDD and the SOEL.

The PUSVDD outperforms the SOEL, and the PUSVDD maintains superior performance across different values of α. These results indicate that our approach is also effective for high-resolution images such as MVTec AD.

F Large Language Models Usage

We used large language models solely for grammar and spelling checks in this paper; they were not used for drafting, analysis, or content generation.