GAD: A Generalized Framework for Anomaly Detection at Different Risk Levels

Rulan Wei*
Zewei He*
East China Normal University
Shanghai, China
{rulanwei,zwhe}@stu.ecnu.edu.cn

Martin Pavlovski
Temple University
Philadelphia, PA, United States
martin.pavlovski@temple.edu

Fang Zhou†
East China Normal University
Shanghai, China
fzhou@dase.ecnu.edu.cn

ABSTRACT

Anomaly detection is a crucial data mining problem due to its extensive range of applications. In real-world scenarios, anomalies often exhibit different levels of priority. Unfortunately, existing methods tend to overlook this phenomenon and identify all types of anomalies into a single class. In this paper, we propose a generalized formulation of the anomaly detection problem, which covers not only the conventional anomaly detection task, but also the partial anomaly detection task that is focused on identifying target anomalies of primary interest while intentionally disregarding non-target (low-risk) anomalies. One of the challenges in addressing this problem is the overlap among normal instances and anomalies of different levels of priority, which may cause high false positive rates. Additionally, acquiring a sufficient quantity of all types of labeled non-target anomalies is not always feasible. For this purpose, we present a generalized anomaly detection framework flexible in addressing a broader range of anomaly detection scenarios. Employing a dual-center mechanism to handle relationships among normal instances, non-target anomalies, and target anomalies, the proposed framework significantly reduces the number of false positives caused by class overlap and tackles the challenge of limited amount of labeled data. Extensive experiments conducted on two publicly available datasets from different domains demonstrate the effectiveness, robustness and superior labeled data utilization of the proposed framework. When applied to a real-world application, it exhibits a lift of at least 7.08% in AUPRC compared to the alternatives, showcasing its remarkable practicality.

CCS CONCEPTS

• **Computing methodologies** → **Anomaly detection**; *Semi-supervised learning settings.*

KEYWORDS

Generalized anomaly detection; Semi-supervised method; Class overlap; Labeled data utilization

*Both authors contributed equally to this research.
†Corresponding author.

CIKM '24, October 21–25, 2024, Boise, ID, USA

ACM ISBN 979-8-4007-0436-9/24/10
https://doi.org/10.1145/3627673.3679634

Table 1: Advantages of the proposed framework GAD vs. Semi-supervised methods. The symbol "○" indicates that some of the methods consider the corresponding factors.

Advantages	GAD	Semi-super. methods
Capable of conventional anomaly detection	✓	✓
Accounting for priority among anomalies	✓	–
Significant reduction of FPs due to class overlap	✓	○
High utilization of labeled data	✓	○
Robust to anomaly contamination	✓	○

ACM Reference Format:
Rulan Wei, Zewei He, Martin Pavlovski, and Fang Zhou. 2024. GAD: A Generalized Framework for Anomaly Detection at Different Risk Levels . In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24), October 21–25, 2024, Boise, ID, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3627673.3679634

1 INTRODUCTION

Anomaly detection is a task that aims at identifying data objects significantly deviating from the majority of the data [18]. It has important applications across various domains, such as fraud detection in finance [3, 29], risks management in banking [11], safeguarding against network intrusions in cybersecurity [40], disease detection in healthcare [13, 23], among others. Since obtaining a sufficient number of accurately labeled instances is challenging in anomaly detection tasks, unsupervised methods [15, 17, 24, 27] have dominated this research area for decades [1]. However, in practical applications, a small number of labeled anomalies is easily accessible. Hence, numerous semi-supervised anomaly detection approaches [20, 25, 33, 43] have emerged in recent years, demonstrating noticeable improvements by leveraging prior knowledge derived from labeled anomalies.

The existing anomaly detection approaches are designed to address the conventional anomaly detection task, which focuses on the identification of all types of anomalies into a single class. In other words, these approaches aim to identify anomalies uniformly while *ignoring any priority that may exist among the anomalies.* However, in real-world scenarios, anomalies often exhibit distinct levels of priority. For instance, in a risk control scenario of an aggregated payment platform where millions of merchants and billions of transactions are involved every day (particularly on days such as Black Friday or China's Double 11), various anomalies with different urgency levels pose complex demands. *High-risk anomalies*, exemplified by activities like gambling and money laundering, pose significant threats, but they occur infrequently—perhaps less than *twenty* times per day. In contrast, *low-risk anomalies*, like cash-out and fake orders, are comparatively less harmful, but they manifest in much greater numbers, reaching *thousands* daily, which is **fifty**

times that of the high-risk anomalies. **Overseeing all anomaly risk levels would overly burden security teams and potentially lead to delays in promptly addressing high-risk anomalies and cause substantial economic losses.** This phenomenon is also observed in other fields.

In this study, we designate anomalies of interest and of higher risk as *target anomalies*, and those of lower interest and lower risk as *non-target anomalies*. Given the efficiency constraints of risk management as well as the scarcity and high cost of human resources, the precise identification of target anomalies becomes imperative, while adopting a more permissive stance toward non-target anomalies is deemed a preferable strategy.

Given the mentioned requirement, prior approaches fail to accurately identify target anomalies. Assuming labels for both target and non-target anomalies are available, two straightforward solutions arise: (i) The first solution is to utilize a semi-supervised anomaly detection method to separate all types of anomalies from normal instances and subsequently employ a supervised binary classifier to differentiate target from non-target anomalies. However, this approach is constrained by the performance of semi-supervised methods. (ii) The second solution is to apply a supervised three-class classification approach to distinguish between normal instances, non-target anomalies, and target anomalies. Yet due to limited amounts of labeled anomaly instances, this strategy often yields sub-optimal results. However, when labels of non-target anomalies are not available, it is more challenging for the existing semi-supervised approaches to accurately identify target anomalies, as non-target anomalies exhibit different characteristics compared to normal instances.

Three major challenges hinder the effective addressing of the aforementioned requirement: (i) Labels for all types of non-target anomalies are challenging to obtain as they are not of primary interest and hence are rarely labeled. Some types of non-target anomalies are "unknown" or even "unseen" in the training dataset. (ii) Class overlap commonly occurs among target anomalies, non-target anomalies, and marginal normal instances. Effectively differentiating them to minimize the number of false positives is the key to the precise identification of target anomalies. (iii) Non-target anomalies may be positioned further away from both target anomalies and normal instances, which poses a challenge for the existing approaches to clearly identify target anomalies.

Thus, we introduce a generalized formulation of the anomaly detection problem and present a framework for **G**eneralized **A**nomaly **D**etection (GAD) that covers the aforementioned anomaly detection scenarios. GAD addresses not only the *conventional anomaly detection*[1] task but also the *partial anomaly detection* task that is focused on identifying target anomalies of interest while intentionally disregarding low-risk non-target anomalies.

The proposed framework is built upon the idea of Deep SAD [25] and significantly extends its flexibility and capacity to broader scenarios. Deep SAD and other existing methods that only consider the normal-anomaly relationship fail to account for scenarios where not all anomalies are of interest; in contrast, our approach introduces a dual-center mechanism to maximize the difference between target anomalies and both normal and non-target anomalies. To further reduce the impact of overlap among classes, we also differentiate target anomalies into hard-to-identify and easy-to-identify sets to improve the representation learning process in GAD. Note that GAD does not assume specific positions for target and non-target anomalies relative to normal instances. Nevertheless, GAD is even more effective when non-target anomalies are more distinct from normal instances (see Sections 5.3 and 5.4).

Further, we derive three variants of the GAD framework. The first two, GAD$^{f-partial}$ and GAD$^{s-partial}$, are specialized in addressing the partial anomaly detection task in fully-supervised and semi-supervised settings, respectively, while the third variant GAD^{con} is designed for the conventional anomaly detection task.

Experimental results on two publicly available datasets from different domains and a real-world dataset show that (i) GAD$^{f-partial}$ and GAD$^{s-partial}$ outperform eleven state-of-the-art methods in partial anomaly detection tasks, achieving an average improvement of 14.72% and 17.46% in AUPRC, respectively, which suggests a significant false positive rate reduction. (ii) In the conventional anomaly detection tasks, GAD^{con} achieves an average AUPRC improvement of 39.59%. (iii) All GAD variants exhibit superior utilization of labeled data, requiring as little as 1% of labeled anomalies while achieving comparable performance to the best-performing method, respectively.

In summary, this paper makes the following major contributions:

- To the best of our knowledge, this work is the first to emphasize the concept of priority among anomalies and precisely identify anomalies of primary interest to meet real-world requirements.
- We propose to address a generalized anomaly detection problem which covers a broader and more practical range of real-world scenarios. The main novelty of this work is that we propose an 'umbrella' (all-encompassing) framework GAD (our code is available at the repository[2]) that addresses different AD scenarios.
- The implementations of the GAD framework showcase strong adaptability across real-world anomaly detection tasks. GAD^{con} precisely identifies all types of anomalies in a conventional setting. When not all anomalies are of interest, GAD$^{f-partial}$ and GAD$^{s-partial}$ separate target anomalies from other instances, corresponding to scenarios with available and unavailable labeled non-target anomalies, respectively.
- GAD significantly reduces the number of false positives caused by class overlaps, insufficient and incomplete labeled data, and positions of non-target anomalies, demonstrating notable detection performance as well as high utilization of labeled data.

2 RELATED WORK

Unsupervised Anomaly Detection. Due to the unavailability of labeled data, most conventional anomaly detection methods are designed in an unsupervised fashion. Despite their simplicity and efficiency, classical methods [15, 30] often face challenges in dealing with high-dimensional data due to the need of feature engineering. Deep learning-based methods automatically learn feature representations of high-dimensional data. For example, Deep SVDD [24] and DeepIF [35] enhance traditional SVDD [30] and iForest [15] through deep neural networks, enhancing their performance on

[1]For clarification, *conventional anomaly detection* refers to semi-supervised anomaly detection, where only labels of all types of anomalies are accessible.

[2]https://github.com/ZhouF-ECNU/GAD

high-dimensional data. Deep autoencoder based approaches [4, 9, 42] and Generative Adversarial Networks (GAN) [26, 38] enhance the learning of the underlying distribution of normal instances. Though recent deep unsupervised methods [14, 22, 36] exhibit proficiency in handling high-dimensional data, they suffer from limited inclusion of prior knowledge, leading to high false positive rates.

Semi/Weakly-supervised Anomaly Detection. In practical applications, a small number of labeled anomalies are usually accessible. Thus, several semi-supervised approaches have been proposed to leverage labeled anomalies to enhance their performance. Some of the most well-known approaches include REPEN [17], DevNet [20], Deep SAD [25], Elite [39], the kernel-based method [33], ADMoE [41], and PIA-WAL [43]. The above semi-supervised methods demonstrate notable improvements in performance by incorporating labeled anomalies. The aforementioned semi-supervised methods treat all anomalies equally and identify all types of anomalies into a single class. Consequently, this may result in a high false positive rate when only a specific subset of anomalies requires accurate identification.

In recent years, it has been recognized that anomalies can be of various types, making it impractical to obtain labeled instances encompassing all anomaly types [8]. Several **weakly-supervised** methods [5, 19, 21, 31, 32] have been developed to detect both seen and unseen anomalies given partially labeled anomaly classes from different perspectives. These methods excel at exploring all possible anomalies based on known instances from a subset of classes. In other words, weakly-supervised methods aim at identifying target and non-target anomalies uniformly. As non-target anomalies exhibit abnormal characteristics, it is naturally easier to classify them together with target anomalies. In contrast, it is more challenging to draw a clear borderline between target anomalies and both non-target anomalies and normal instances when labels for non-target anomalies are not accessible. Therefore, weakly-supervised AD methods are not applicable to identifying target anomalies.

As opposed to the above approaches, our focus lies in proposing a generalized framework that not only enables more precise identification in the conventional anomaly detection scenarios but also excels in accurately identifying anomalies of primary interest, while disregarding those considered not of interest.

Anomaly Diagnosis. Anomaly diagnosis [6, 12, 28, 37], also known as anomaly interpretation, is focused on pinpointing specific channels (or variables, or features) that cause abnormalities. However, since anomaly diagnosis takes failures or anomalies as inputs and is a subsequent task to anomaly detection, it is beyond the scope of this work.

3 PROBLEM STATEMENT

A more generalized formulation of the *Anomaly Detection problem* is defined as follows:

Assume a large unlabeled dataset $\mathcal{D}^u = \{\boldsymbol{x}_1^u, ..., \boldsymbol{x}_{n_0}^u\} \in \mathcal{X}$ with $\mathcal{X} \subseteq \mathbb{R}^d$ which primarily consists of normal instances but may also be tainted with a mixture of anomalies. On the other hand, let $\mathcal{D}^M$ be a dataset consisted of anomalies that may fall into M different anomaly classes (depending on the scenario, the classes may represent different types of anomalies, risk levels, or other criteria by which the anomalies can be differentiated). Assuming that the anomaly classes are ordered by some measure of 'importance' or 'priority', in certain applications one may be solely interested in detecting the first k anomaly classes, where $k \in [1, M]$. In that regard, we present two major special cases:

(1) $k = M$: This special case boils down to the *conventional anomaly detection problem* where all classes of anomalies need to be detected assuming that labels are always provided for all M classes in $\mathcal{D}^M$. In such a case, the task is to develop a model capable of accurately predicting y for an instance $\boldsymbol{x}$, where $y = +1$ denotes an *anomaly* of any class, while $y = -1$ represents a normal instance.

(2) $k < M$: This special case reduces to the *partial anomaly detection problem*, the objective of which is to develop a model capable of accurately predicting y for an instance $\boldsymbol{x}$, where $y = +1$ denotes an *anomaly* which belongs to the first k anomaly classes, while $y = -1$ represents either a normal instance or an anomaly that is not among first k classes. Note that the labels for the first k anomaly classes are assumed to be always available in $\mathcal{D}^k \subseteq \mathcal{D}^M$. As for the availability of labels for the anomaly classes beyond k, we consider two supervision scenarios:

(2.1) *Fully-supervised partial AD*: The labels for the additional $M-k$ anomaly classes are also available.

(2.2) *Semi-supervised partial AD*: The labels for the remaining $M-k$ classes are not available.

Note that here the supervision refers exclusively to the availability of labels for the anomalies, not the normal instances.

For the sake of clarity, we refer to the anomalies from the first k classes as **target anomalies** and to the anomalies from classes beyond k as **non-target anomalies**.

4 PROPOSED METHOD

We present a framework, GAD, for the generalized AD problem, and describe three variants designed for different scenarios of the generalized anomaly detection problem.

4.1 GAD Framework

The primary objective of GAD is to learn latent representations to better distinguish target anomalies of interest, regardless of whether $k < M$ or $k = M$, from the remaining instances, through training a neural network $\Phi(\cdot; \mathcal{W}) : \mathcal{X} \mapsto \mathcal{F} \subseteq \mathbb{R}^{d_h}$ to effectively manage the relationships among normal instances and different levels (classes) of anomalies.

Aiming for instances other than target anomalies to cluster tightly, we adopt the same loss term as in Deep SVDD [24] for the unlabeled dataset $\mathcal{D}^u$. After initializing a neural network Φ using the encoder's weights of a pre-trained autoencoder on $\mathcal{D}^u$, $\boldsymbol{c}$ is obtained by averaging the outputs of the first forward pass of Φ and remains constant: $\boldsymbol{c} = \frac{\sum_{i=1}^{|\mathcal{D}^u|} \Phi(\boldsymbol{x}_i; \mathcal{W})}{|\mathcal{D}^u|}, \boldsymbol{x}_i \in \mathcal{D}^u$. Here, $\boldsymbol{c}$ serves as a fixed normal center. Since $\mathcal{D}^u$ primarily consists of normal instances, penalizing the distance from the learned representation of any instance in $\mathcal{D}^u$ to $\boldsymbol{c} \in \mathcal{F}$ forces Φ to map normal instances close to $\boldsymbol{c}$.

Notice that non-target anomalies, in partial anomaly detection tasks, display patterns that deviate from normality. Similarly, **marginal instances**, located at the "margins" of the normal instance

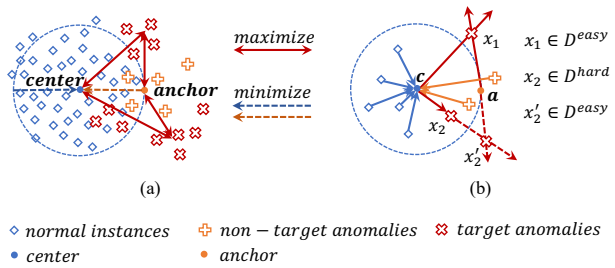


Figure 1: (a) Illustration of GAD's dual-center mechanism. (b) Illustration of $L_{compact}$ and L_{target} as a functions of D^{hard} and D^{easy} sets. Here, $x_1 \in D^{easy}$ is outside of the sphere surrounding c and thus is considered an *easy-to-identify* instance; whereas $x_2 \in \mathcal{D}^{hard}$ at the beginning, and transitions to $x_2' \in D^{easy}$ after a few iterations.

space, are also distinct from the majority of normal instances. In conventional AD scenarios, marginal instances may have similar characteristics to normal instances and yet be challenging to differentiate from anomalies, leading to high false positive rates. On the other hand, in semi-supervised partial AD scenarios, marginal instances are located on the surface of the hypersphere encapsulating the normal instances, in which case they may consist of normal instances as well as non-target anomalies.

Furthermore, non-target anomalies or marginal instances may overlap with target anomalies, thereby leading to a high rate of false positives. Therefore, we propose a dual center mechanism to minimize the difference between the center $\boldsymbol{c}$ and both normal instances and non-target anomalies (or marginal instances), while maximizing the difference between target anomalies and both normal instances and non-target anomalies, as illustrated in Fig. 1(a). Specifically, in addition to the center $\boldsymbol{c}$, we introduce an *anchor* $\boldsymbol{a}$ to intentionally segregate target anomalies of interest (regardless of whether $k < M$ or $k = M$) from not only normal instances but also from non-target anomalies or marginal instances. To achieve this goal, we construct an auxiliary dataset $\mathcal{D}^a$, which consists of either non-target anomalies or marginal instances, to obtain an anchor $\boldsymbol{a}$. The anchor $\boldsymbol{a}$ represents the average position of $\mathcal{D}^a$ and is initially set to the average output of the first forward pass of Φ on $\mathcal{D}^a$:

$$\boldsymbol{a} = \frac{\sum_{i=1}^{|\mathcal{D}^a|} \Phi(\boldsymbol{x}_i; \mathcal{W})}{|\mathcal{D}^a|}, \quad \boldsymbol{x}_i \in \mathcal{D}^a. \tag{1}$$

Although the auxiliary instances in $\mathcal{D}^a$ exhibit distinct characteristics compared to the majority of normal instances, in order to capture their contrast to target anomalies, we reduce the difference in the learned representations for the instances in $\mathcal{D}^a$ and $\mathcal{D}^u$. We apply the same technique as for $\mathcal{D}^u$, by penalizing the inverse of the distance from the instances in $\mathcal{D}^a$ to $\boldsymbol{c}$, and use η to balance the trade-off between the influence of unlabeled and auxiliary data. Since the instances in $\mathcal{D}^a$ tend to be located at a certain distance from the center $\boldsymbol{c}$ in the latent space, by appropriately increasing η, Φ is able to emphasize the auxiliary data, leading to the learning of a more compact hypersphere. The loss $L_{compact}$ designed for $\mathcal{D}^u$ and $\mathcal{D}^a$ is as follows:

$$L_{compact} = \sum_{i=1}^{|\mathcal{D}^u|} \|\Phi(\boldsymbol{x}_i; \mathcal{W}) - \boldsymbol{c}\|^2 + \eta \sum_{j=1}^{|\mathcal{D}^a|} \|\Phi(\boldsymbol{x}_j; \mathcal{W}) - \boldsymbol{c}\|^2. \tag{2}$$

Algorithm 1 The GAD Framework

Input: Unlabeled dataset $\mathcal{D}^u$, labeled anomaly dataset $\mathcal{D}^k$, training epochs $\mathcal{T}$
Output: Anomaly scoring function $\boldsymbol{s}$

```
Pre-train an autoencoder on D^u until convergence;
Initialize a neural network Φ using the encoder's weights;
Initialize the center c, anchor a, and an auxiliary dataset D^a;
Initialize the easy-to-identify set D^easy ← ∅, and hard-to-identify set D^hard ← ∅;
for epoch = 1, ..., T do
    Update D^a if need;
    Update anchor a using Eq. (1);
    Update D^easy and D^hard using Eq. (3) and (4), respectively;
    Update Φ through optimizing Eq. (6);
end for
Return scoring function s(·) = ‖Φ(·; W) − c‖².
```

$L_{compact}$ guides Φ to minimize the volume of the hypersphere that encloses normal and auxiliary instances in the latent space, thereby indirectly maximizing the differences between target anomalies and both normal and auxiliary instances.

To ensure separation of target anomalies from both normal and auxiliary instances, a simple solution would be to just penalize the inverse distance to both $\boldsymbol{c}$ and $\boldsymbol{a}$ for the labeled anomalies in $\mathcal{D}^k (k \in [1, M])$. However, for certain target anomalies located very close to the center $\boldsymbol{c}$, such solution may incur conflicting impact and hinder their separation from $\boldsymbol{c}$. To address this issue, we propose a refined loss that takes into account the positions of instances with respect to $\boldsymbol{c}$ and $\boldsymbol{a}$. Assume a hypersphere centered at $\boldsymbol{c}$ with a radius equal to the distance between $\boldsymbol{c}$ and $\boldsymbol{a}$. The set of labeled target anomalies $\mathcal{D}^k$ is partitioned into *easy-to-identify set* $\mathcal{D}^{easy}$ and *hard-to-identify set* $\mathcal{D}^{hard}$ based on their respective positions in the latent space:

$$\mathcal{D}^{easy} = \{\boldsymbol{x} | dist(\Phi(\boldsymbol{x}; \mathcal{W}), \boldsymbol{c}) \geq dist(\boldsymbol{a}, \boldsymbol{c}), \boldsymbol{x} \in \mathcal{D}^k\}, \tag{3}$$

$$\mathcal{D}^{hard} = \{\boldsymbol{x} | dist(\Phi(\boldsymbol{x}; \mathcal{W}), \boldsymbol{c}) < dist(\boldsymbol{a}, \boldsymbol{c}), \boldsymbol{x} \in \mathcal{D}^k\}, \tag{4}$$

where $\mathcal{D}^{easy} \cup \mathcal{D}^{hard} = \mathcal{D}^t, \mathcal{D}^{easy} \cap \mathcal{D}^{hard} = \emptyset$ and $dist(\cdot, \cdot)$ represents the distance between two instances in the latent space.

In the case of target instances belonging to $\mathcal{D}^{easy}$, we apply a penalty to the inverse of the distance to both $\boldsymbol{c}$ and $\boldsymbol{a}$; while for target instances within $\mathcal{D}^{hard}$, in order to avoid conflicting impact, we relax the penalty and only penalize the inverse of the distance to $\boldsymbol{c}$. The loss function for the target instances is defined as follows:

$$\begin{aligned} L_{target} = &\sum_{k=1}^{|\mathcal{D}^{easy}|} \left(\frac{1}{\|\Phi(\boldsymbol{x}_k; \mathcal{W}) - \boldsymbol{c}\|^2} + \frac{1}{\|\Phi(\boldsymbol{x}_k; \mathcal{W}) - \boldsymbol{a}\|^2} \right) \\ &+ \sum_{l=1}^{|\mathcal{D}^{hard}|} \frac{1}{\|\Phi(\boldsymbol{x}_l; \mathcal{W}) - \boldsymbol{c}\|^2}. \end{aligned} \tag{5}$$

The role of L_{target} is to directly maximize the difference between target anomalies and both normal and auxiliary instances.

As the model training process iterates, the hard and easy-to-identify sets will be updated along with the update of $\boldsymbol{a}$. Namely, the second component in Eq. (5) will push the target anomalies

in $\mathcal{D}^{hard}$ away from the center $\boldsymbol{c}$. In the meantime, as $\boldsymbol{a}$ undergoes dynamic updates during each epoch according to Eq. (1), it moves towards $\boldsymbol{c}$, which shrinks the hypersphere and thus aids the gradual movement of the target anomalies in $\mathcal{D}^{hard}$ outside the hypersphere. Once target anomalies in $\mathcal{D}^{hard}$ are located outside the hypersphere and are transferred to $\mathcal{D}^{easy}$, the first term in Eq. (5) would push them away from both $\boldsymbol{c}$ and $\boldsymbol{a}$ (see the example x_2 in Fig. 1(b)). This mechanism allows our model to effectively identify target anomalies without assuming specific positions for target and non-target anomalies relative to normal instances (see Sections 5.3 and 5.4).

The overall objective function of GAD is defined as follows:

$$\min_{\mathcal{W}} \frac{L_{compact} + L_{target}}{|\mathcal{D}^u| + |\mathcal{D}^a| + |\mathcal{D}^k|}. \tag{6}$$

The effect of both $L_{compact}$ and L_{target} are illustrated in Fig. 1(b). We simultaneously optimize the GAD objective and update $\mathcal{W}$ using the Adam optimizer. After the model converges, the anomaly scores are calculated as follows: $\boldsymbol{s}(\boldsymbol{x}) = \|\Phi(\boldsymbol{x};\mathcal{W}) - \boldsymbol{c}\|^2$, where $\boldsymbol{s}(\boldsymbol{x})$ represents the distance from a mapped instance $\boldsymbol{x}$ to the center $\boldsymbol{c}$ in the latent space. A higher score indicates a greater probability of $\boldsymbol{x}$ being a target anomaly. Algorithm 1 presents the training procedure of GAD.

4.2 GAD Variants

We first introduce GAD$^{f-partial}$, a variant for partial anomaly detection under full supervision of both target and non-target anomalies. Next, we continue by describing GAD$^{s-partial}$, a variant for partial anomaly detection in a semi-supervised setting where labels for non-target anomalies are not available. Finally, we also provide a variant GAD^{con} designed for addressing the conventional AD task. The main difference among the three variants lies in the selection of D^a in order to generate an $\boldsymbol{a}$.

GAD$^{f-partial}$. The goal of GAD$^{f-partial}$ is to prioritize the identification of target anomalies of interest, rather than aiming to uniformly detect abnormal instances of all risk levels. Since non-target anomalies are available, $\mathcal{D}^a$ consists of labeled non-target anomalies.

GAD$^{s-partial}$. In GAD$^{s-partial}$, labeled non-target anomalies are not available. Consider that, apart from the extremely rare target anomalies, the unlabeled dataset $\mathcal{D}^u$ may also contain a minor fraction of non-target anomalies. Owing to their intrinsic dissimilarity from normal instances, they are distributed at the margin of the hypersphere. Thus, $\mathcal{D}^a$ is composed of marginal instances selected from $\mathcal{D}^u$.

It is worth noting that $\mathcal{D}^a$ is updated in each iteration. During each epoch, all instances in $\mathcal{D}^u$ are sorted based on their distances to the center $\boldsymbol{c}$ in the latent space $\mathcal{F}$. The instances with the largest distances are selected to reconstruct $\mathcal{D}^a$ and are utilized to obtain an anchor $\boldsymbol{a}$ according to Eq. (1). The sorting operation for constructing $\mathcal{D}^a$ introduces an additional time complexity of $O(NlogN)$ per epoch. Section 4.3 provides a theoretical analysis of this complexity, while Section 5.7 presents experimental evidence on time efficiency, showcasing that this design enhances detection effectiveness with minimal impact on runtime efficiency.

GAD^{con}. In conventional AD tasks, where all types of anomalies need to be detected, some normal instances that are challenging to classify are positioned on the margin of the hypersphere. Their latent representations may overlap with anomalies, leading to an increase in false positives. To alleviate this challenge, GAD^{con} adopts an approach akin to GAD$^{s-partial}$ and selects marginal instances from $\mathcal{D}^u$ to determine an anchor $\boldsymbol{a}$. This strategy effectively widens the gap between normal instances and anomalies, leading to a significant reduction in false positives.

4.3 Complexity Analysis

We first analyze the complexity of pre-training using an autoencoder (Line 1 in Algorithm 1). For a tabular dataset $\mathcal{D}$ with N instances (including normal instances, non-target anomalies and target anomalies) each having d dimensions, the complexity of the pre-training process can be expressed as $O(Nd)$, which is linear w.r.t. both the input data size and input data dimension. This computational cost is the same as that of the original Deep SAD.

In terms of the training procedure, the network architecture remains the same as that of the encoder used for pre-training. In the initial forward propagation, the computation of the center $\boldsymbol{c}$ of the hidden representations of all data points induces a complexity of $O(Nd_h)$, where d_h is the representation dimension. In each forward propagation, both the hidden representations and the anchor $\boldsymbol{a}$ need to be updated (Line 7 in Algorithm 1). Therefore, for GAD$^{f-partial}$, each epoch has a complexity of $O(N\times d+|\mathcal{D}^a|\times d_h)$; for GAD$^{s-partial}$ and GAD^{con}, an additional complexity arises from selecting the farthest $|\mathcal{D}^a|$ marginal instances, resulting in each epoch having a complexity of $O(N\times d+|\mathcal{D}^a|\times d_h+(N-|\mathcal{D}^a|-|\mathcal{D}^k|)\log(N-|\mathcal{D}^a|-|\mathcal{D}^k|))$. Due to the extremely small sizes of $\mathcal{D}^a$ and $\mathcal{D}^k$ compared to N, the asymptotic complexity of a single epoch of GAD$^{f-partial}$'s training procedure becomes $O(Nd)$, or $O(Nd + N\log N)$ in the case of GAD$^{s-partial}$ and GAD^{con}. After training, during the inference phase (Line 11 in Algorithm 1), both proposed GAD variants and Deep SAD have a complexity of $O(Nd)$.

Overall, the training complexity of GAD$^{f-partial}$ is consistent with that of the original Deep SAD, while GAD$^{s-partial}$ and GAD^{con} introduce additional complexity due to the sorting of unlabeled data to identify the most suitable $\mathcal{D}^a$. Considering that the proposed GAD variants effectively address the partial anomaly detection problem and significantly improve detection performance (as demonstrated in Section 5), the extra complexity of $O(N\log N)$ is deemed acceptable.

5 EXPERIMENTS

5.1 Experimental Setup

5.1.1 Datasets. To evaluate the effectiveness of the proposed framework, we constructed 14 datasets from two publicly available datasets (*UNSW_NB15* [16] and *FMNIST* [34]), stemming from diverse domains and having different numbers of target anomaly classes. The *UNSW_NB15* [16] dataset, used in the field of network intrusion, is a tabular dataset that comprises 7 different types of real anomalies. We selected 3 of them as target anomalies, while the remaining 4 types were designated as non-target anomalies. The *FMNIST* [34]

Table 2: Statistics of datasets. d denotes the dimension of a dataset.

dataset		training set			validation set			testing set		
dataset name	d	unlabeled ($\mathcal{D}^u$)	target ($\mathcal{D}^k$)	non-target ($\mathcal{D}^a$)	normal	target	non-target	normal	target	non-target
UNSW_NB15	196	57,318	300(3)	400(4)	18,600	1,666(3)	2,335(4)	18,600	1,666(3)	2,335(4)
FMNIST[1], FMNIST[2]	28×28	5,100	100(1)	100(1)	1,000	100(1)	100(1)	1,000	100(1)	100(1)
FMNIST[3]	28×28	5,100	100(1)	0	1,000	100(1)	0	1,000	100(1)	0
SQB	182	134,299*	205(3)	205(5)	33,575*	41(3)	41(5)	148,323*	129(3)	463(5)

The number of distinct categories present in a dataset is surrounded with "()".
* Since normal instances are not available in the SQB dataset, we consider the unlabeled instances as normal for validation and testing.

dataset is a collection of images consisting of 10 fashion categories. We selected instances from these different categories as normal, target and non-target anomalies to assess the detection performances of models in various settings.

To simulate real-world anomaly detection scenarios, for the UNSW_NB15 and FMNIST datasets, we randomly sampled a small number of target and non-target anomalies as labeled datasets. Next, we integrated both target and non-target anomalies into the set of the normal instances at a default contamination ratio of 2% to generate unlabeled datasets. This process allowed us to construct the datasets UNSW_NB15, FMNIST[1] and FMNIST[2] (the difference between FMNIST[1] and FMNIST[2] mainly lies in the distance between normal instances and target anomalies). Additionally, to simulate the conventional anomaly detection scenario, we generated another dataset, FMNIST[3], where non-target anomalies were excluded from both the unlabeled and labeled datasets.

Real-world application: SQB is a real-world fraud detection dataset derived from actual merchants' daily transactions on the ShouQianBa aggregated payment platform[3]. The task is to predict if a merchant is engaged in fraudulent activities based on their daily transactions. Based on practical demands, specific anomalies such as gambling and money laundering are categorized as target anomalies due to their significant risk and potential harm, while anomalies such as cash-out and fake orders, which pose a relatively lower risk, are categorized as non-target anomalies. We collected transaction data for 165,478 merchants from April 2021 to April 2022, and extracted 182 features such as transaction frequency and payment amount. In total, 316,197 unlabeled instances were obtained, along with 375 target and 709 non-target anomalies. It is worth noting that, the SQB dataset's unlabeled data includes a significant yet unknown proportion of hidden target and non-target anomalies, the statistics of which are presented in Table 2.

5.1.2 Competing Methods & Evaluation Metrics. We present a total of 11 baselines for comparison, consisting of four unsupervised methods (**DeepIF** [35], **iForest** [15], **OC-SVM** [27], **Deep SVDD** [24]), five semi-supervised methods (**PReNet** [19],**Deep SAD** [25], **DevNet** [20], **PIA-WAL** [43], **Kernel-Based Method** [33]), and two fully-supervised methods (**Random Forest (RF)** [2], **Deep SAD + Random Forest (RF)**). For all unsupervised and semi-supervised methods, only the labels of target anomalies are used for training. For fully-supervised methods, labels of non-target anomalies are also used.

[3]https://www.shouqianba.com/

When running GAD, Deep SAD, Deep SVDD, and the kernel-based method on the high-dimensional tabular datasets (UNSW_NB15 and SQB), we utilized an MLP with three hidden layers having 168, 64, and 32 hidden nodes. DevNet, PIA-WAL, PReNet, DeepIF, iForest, OC-SVM and Random Forest were run with their default settings recommended in their respective papers.

As for the FMNIST image dataset, we used a variant of LeNet [10] to obtain the image representations needed for GAD, Deep SAD, Deep SVDD, and the kernel-based method. For DevNet, we used its official implementation for ingesting image data. For PIA-WAL, PReNet, DeepIF, iForest, OC-SVM and Random Forest, which are not specifically designed for image data, we applied an unsupervised autoencoder (based on a variant of LeNet) to map the FMNIST images into a 64-dimensional space before running these methods.

The Leaky ReLU function $g(z) = \max(0, z) + 0.01 * \min(0, z)$ was used for gradient propagation, and an L2-norm regularizer was applied to each hidden layer to mitigate overfitting. All models' parameters were fine-tuned using a grid search on the validation set of each dataset. Regarding the kernel-based method, we utilized an autoencoder to calculate the reconstruction error and modified its loss function based on the formula provided in [33]. For the remaining baselines, we used the publicly available implementations provided by the authors of their respective papers. All methods were run on a workstation equipped with an Intel(R) Xeon(R) Gold 6240R CPU, a Tesla V100-SXM2-32GB GPU, and 256 GB of RAM.

Area Under the Receiver Operating Characteristic Curve (AUROC) and Area Under the Precision-Recall Curve (AUPRC) were used to evaluate the performance of the models. Since AUPRC holds greater significance as an evaluation metric for anomaly detection problems, our focus primarily lies on AUPRC in the subsequent analysis. All reported AUROC and AUPRC values are averages (along with their standard deviation) over 10 independent experiment runs.

5.2 Effectiveness on Real-world Datasets

We evaluated the models' effectiveness on five datasets: UNSW_NB15, SQB, FMNIST[1], FMNIST[2], and FMNIST[3]. FMNIST[3] illustrates a conventional AD scenario without non-target anomalies, while the other four datasets represent partial AD scenarios.

The AUPRC and AUROC performances of three GAD variants and 11 competing methods are shown in Table 3. Overall, the GAD variants demonstrate superior performance in terms of AUPRC and AUROC across the five datasets, respectively. For example, in terms of AUPRC, GAD$^{f-partial}$ yields improvements of 0.8%-61.35% over

Table 3: AUPRC and AUROC performance (with ± standard deviation) of three GAD variants ($GAD^{f-partial}$, $GAD^{s-partial}$ and GAD^{con}) and eleven competing methods. The best performance is boldfaced; the runner-up is underlined.

Model	use of labeled non-targ.	AUPRC					AUROC				
		Partial AD				Conventional AD	Partial AD				Conventional AD
		UNSW_NB15	SQB	$FMNIST^1$	$FMNIST^2$	$FMNIST^3$	UNSW_NB15	SQB	$FMNIST^1$	$FMNIST^2$	$FMNIST^3$
DeepIF	×	56.06±3.57	1.34±0.43	19.9±0.55	10.66±0.38	22.85±1.34	93.94±0.18	86.21±0.34	78.63±1.32	60.17±1.26	63.72±1.09
DeepSVDD	×	47.7±2.76	0.37±0.18	21.71±1.02	18.7±1.93	22.61±3.08	93.3±0.51	66.25±15.46	78.17±1.8	61.49±2.2	62.59±3.33
iForest	×	36.24±7.49	1.63±0.37	25.5±2.53	10.67±0.51	15.79±0.98	83.97±1.7	90.92±0.61	86.39±1.61	58.48±1.02	63.77±0.98
OC-SVM	×	30.93±0.0	1.03±0.0	14.69±0.0	11.14±0.0	15.63±0.0	88.82±0.0	84.98±0.0	74.42±0.0	59.11±0.0	63.27±0.0
DeepSAD	×	72.24±1.04	23.0±0.98	94.78±1.32	64.58±4.68	70.68±4.03	96.35±0.11	97.57±0.48	98.29±0.75	89.0±1.69	90.46±2.12
DevNet	×	65.71±1.42	14.89±0.89	94.38±1.52	44.08±6.45	57.87±3.96	94.95±0.3	97.36±0.84	98.56±0.5	81.98±2.26	85.94±1.95
Kernel-Based	×	68.58±5.39	2.53±0.52	88.97±2.47	44.83±8.5	44.69±4.61	94.57±0.34	85.11±0.61	97.12±0.72	75.06±5.26	75.0±3.1
PIA-WAL	×	72.2±2.08	18.58±1.02	64.31±10.8	19.46±8.57	35.12±7.42	95.65±0.14	96.14±1.0	88.85±3.2	68.32±8.69	75.58±2.94
PReNet	×	63.28±0.61	23.53±1.72	93.72±0.18	49.5±3.27	58.34±0.6	94.2±0.21	90.0±1.62	99.24±0.12	79.58±2.83	82.68±2.16
$GAD^{s-partial}$	×	**74.74±1.1**	**30.23±1.12**	**97.1±0.3**	**72.6±1.04**	-	**97.3±0.07**	**98.74±0.72**	**99.31±0.2**	**92.12±0.62**	-
RF (3 classes)	✓	78.33±0.18	19.21±1.02	91.43±0.51	52.28±1.77	-	97.52±0.05	97.15±0.52	98.58±0.11	86.14±0.19	-
Deep SAD+RF (2 classes)	✓	54.31±3.1	1.73±0.85	76.82±6.64	14.74±2.73	-	13.25±0.64	0.27±0.01	15.51±2.5	11.84±1.41	-
$GAD^{f-partial}$	✓	**79.13±0.21**	**33.84±2.9**	**96.57±0.35**	**76.09±1.0**	-	**97.62±0.05**	**98.77±0.78**	**99.31±0.15**	**92.71±0.4**	-
RF (2 classes)	-	-	-	-	-	43.38±2.11	-	-	-	-	84.23±0.69
GAD^{con}	-	-	-	-	-	**78.29±1.3**	-	-	-	-	**93.24±0.57**

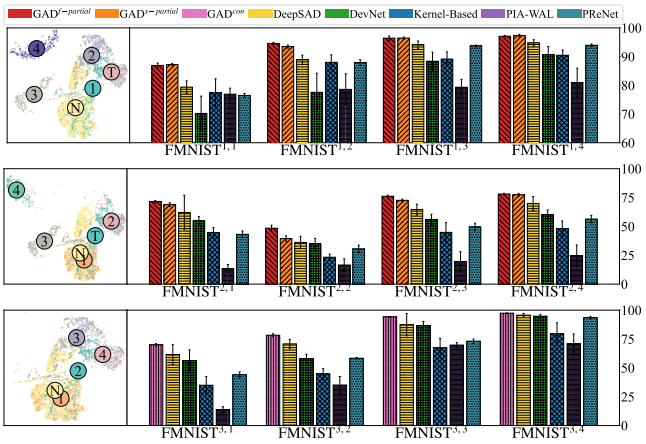


Figure 2: UMAP embeddings of the selected FMNIST datasets and AUPRC ± standard deviation w.r.t. different scenarios. For the partial AD scenarios, N denotes the normal instances, T denotes target anomalies and 1-4 denote different selections of non-target anomalies. The $FMNIST^{1,\cdot}$ and $FMNIST^{2,\cdot}$ datasets are sorted by the ascending inter-class distances between their respective non-target and target anomaly classes. In the case of conventional AD, N denotes the normal instances, and 1-4 denote different selections of anomalies. $FMNIST^{3,\cdot}$ is sorted based on the ascending inter-class distances between their respective anomaly and normal classes.

fully-supervised baselines. $GAD^{s-partial}$ attains lifts ranging from 2.5% to 53.14% relative to the semi-supervised baselines. GAD^{con} surpasses all baselines with lifts ranging from 7.61% to 62.66%.

It is worth noting that when non-target anomalies are relatively further away from both normal instances and target anomalies in the $FMNIST^2$ dataset, the average AUPRC gap between the proposed models and the semi-supervised baselines increased from 9.6% to 29.86% on the $FMNIST^2$ dataset, compared to the performances on the $FMNIST^1$ dataset where target anomalies are located far away from the normal and non-target anomalies. This shows the superiority of the proposed framework in addressing the complex partial anomaly detection problem.

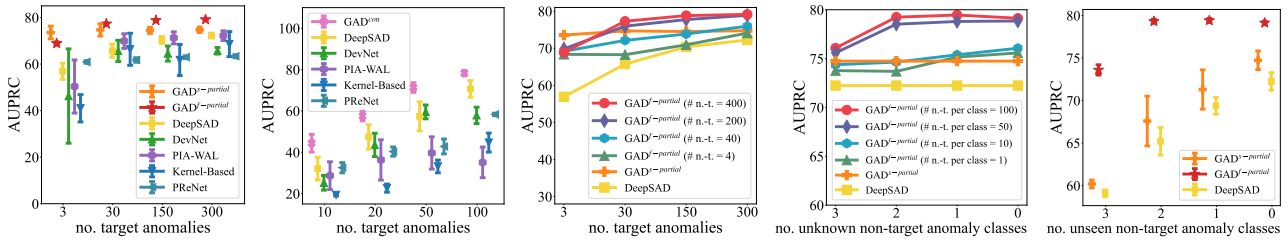
The two fully-supervised models exhibit poor performance with limited labeled instances, rendering them inapplicable to real-world scenarios. Their testing is omitted in subsequent experiments.

5.3 Effect of Overlap Degree and Positions of Non-Target Anomalies on Partial AD

To explore the effect of different positions for target and non-target anomalies relative to normal instances in partial anomaly detection tasks, we conducted experiments using various class combinations of the FMNIST dataset. $FMNIST^{1,\cdot}$ corresponds to scenarios in which target anomalies are clearly distinguishable from normal instances. We then chose non-target anomalies based on their degree of overlap with the target anomalies in a descending order, resulting in the creation of four datasets: $FMNIST^{1,1}$, $FMNIST^{1,2}$, $FMNIST^{1,3}$, and $FMNIST^{1,4}$. On the other hand, $FMNIST^{2,\cdot}$ represents scenarios where target anomalies overlap significantly with normal instances. Non-target anomalies were chosen based on their degree of overlap with the target anomalies in a descending order to construct four datasets: $FMNIST^{2,1}$, $FMNIST^{2,2}$, $FMNIST^{2,3}$, and $FMNIST^{2,4}$.

The AUPRC values along with their standard deviations w.r.t. different scenarios of FMNIST are presented in Fig. 2. We omitted three unsupervised baselines on the FMNIST dataset due to their poor AUPRC values. $GAD^{f-partial}$ and $GAD^{s-partial}$ consistently outperform all competing methods in terms of AUPRC across eight datasets (first two rows in Fig. 2). In comparison to the five semi-supervised baselines, $GAD^{f-partial}$ and $GAD^{s-partial}$ demonstrate average improvements of 15.36% and 17.39%, respectively, while showcasing enhanced predictive stability.

In the experiments conducted on the $FMNIST^{1,\cdot}$ group (refer to the first row in Fig. 2), $GAD^{f-partial}$ and $GAD^{s-partial}$ achieve an AUPRC improvement of 2.34%-17.1% and 2.4%-17.13% over the semi-supervised baselines, respectively. As the overlap degree between non-target and target anomalies decreases (transitioning from $FMNIST^{1,1}$ to $FMNIST^{1,4}$), we observe improvements in the performances of both GAD variants as well as the five semi-supervised baselines. This could be attributed to the fact that, when the non-target anomalies are less overlapped with the target anomalies,



(a) partial AD (b) conventional AD (c) partial AD (d) partial AD (e) partial AD

Figure 3: Detection performance of models w.r.t. a different (a,b) number of target anomalies, (c) number of target and non-target anomalies, (d) number of unknown non-target anomaly classes, and (e) number of unseen non-target anomaly classes. Note that "n.-t." in the legends refers to "non-target anomalies".

the semi-supervised methods (which leverage prior knowledge on target anomalies) can identify target anomalies more easily.

Moving on to the more challenging group FMNIST$^{2,\cdot}$ (refer to the second row in Fig. 2), GAD$^{f-partial}$ and GAD$^{s-partial}$ exhibit significant AUPRC improvements of 8.21%-58.01% and 3.32%-55.38% compared to the semi-supervised baselines, respectively. From the results on the FMNIST$^{2,\cdot}$ group, we make the following observations: (1) Although all methods demonstrate poorer performance compared to their performances on the FMNIST$^{1,\cdot}$ datasets, GAD$^{f-partial}$ and GAD$^{s-partial}$ exhibit more substantial improvements on FMNIST$^{2,\cdot}$ than the other baselines. This can be attributed to the fact that target anomalies overlap severely with the normal instances in the FMNIST$^{2,\cdot}$ group, making targeted anomaly detection more challenging. The employed dual center mechanism proves to be effective in accurately distinguishing overlapped instances. (2) Transitioning from the FMNIST^{2,2} dataset to the FMNIST^{2,4} dataset, the performances of the semi-supervised methods improve. This is due to the decrease in the overlap degree between the non-target anomalies and the target anomalies. However, all methods obtained higher detection accuracies on the FMNIST^{2,1} dataset compared to their performances on the FMNIST^{2,2} dataset. The reason is that non-target anomalies have a severe overlap with normal instances in FMNIST^{2,1}, which makes partial anomaly detection much easier.

5.4 Effect of Overlap Degree and Positions of Anomalies on Conventional AD

Similar to Section 5.3, we construct the dataset group FMNIST$^{3,\cdot}$ to explore the impact of the anchor $\boldsymbol{a}$ with respect to different positions of target and non-target anomalies relative to normal instances. For the FMNIST$^{3,\cdot}$ group, anomalies were chosen based on their descending overlap degree with the normal instances, resulting in four datasets: FMNIST^{3,1}, FMNIST^{3,2}, FMNIST^{3,3}, and FMNIST^{3,4}.

The AUPRC performances of GAD^{con} and five semi-supervised methods along with their standard deviations w.r.t. different scenarios of FMNIST$^{3,\cdot}$ are presented in the third row of Fig. 2. GAD^{con} exhibits an AUPRC improvement of 1.71%-61.23% compared to the semi-supervised baselines. All methods yielded improvements from the FMNIST^{3,1} dataset to the FMNIST^{3,4} dataset, as the overlap degree between anomalies and normal instances decreases. Results on the FMNIST$^{3,\cdot}$ datasets show that using marginal instances to widen the gap between normal and anomalous instances is effective in conventional anomaly detection, especially with significant class overlap.

5.5 Effectiveness under Different Quantities of Labeled Anomalies

To investigate the effectiveness of the models w.r.t. the quantity of available labeled anomalies, we designed the following experiments by adjusting: (1) the number of labeled target anomalies, (2) the number of labeled non-target anomalies, and (3) the number of labeled non-target anomaly classes in the training set. The experiments for partial and conventional anomaly detection tasks were conducted on the UNSW_NB15 and FMNIST^3 datasets, respectively.

AUPRC w.r.t. the number of labeled target anomalies for partial anomaly detection task is plotted in Fig. 3(a), while for conventional anomaly detection task is plotted in Fig. 3(b). The performances of all methods improve with more labeled target anomalies. Notably, all GAD variants significantly outperform semi-supervised baselines at all levels of labeled target anomalies. In partial anomaly detection, GAD$^{s-partial}$ achieves an average AUPRC lift of 1.31% with only 3 target anomalies compared to Deep SAD using 300 target anomalies, showcasing substantial data utilization efficiency.

We next evaluate the effectiveness of GAD$^{f-partial}$ concerning the number of both labeled target and non-target anomalies. Since Deep SAD yields the highest AUPRC among the semi-supervised baselines, we omit the results of other semi-supervised baselines. The number of labeled non-target anomalies ranges from 4 to 400, while the number of labeled target anomalies is varied between 3 and 300. From Fig. 3(c), several observations can be made: (1) GAD$^{f-partial}$ and GAD$^{s-partial}$ consistently outperform Deep SAD. (2) The performance of GAD$^{f-partial}$ improves as the number of labeled non-target anomalies increases. (3) When the number of labeled target anomalies is small (e.g. the number is 3), GAD$^{s-partial}$ obtains even better results than GAD$^{f-partial}$.

Effect of Unknown and Unseen Non-target Anomalies. Fig. 3(d) shows AUPRC under various quantities of known non-target anomalies and numbers of unknown non-target anomaly classes in the training set. Since the test set remained unchanged, the semi-supervised baselines are not sensitive to changes in the quantity and types of labeled non-target anomalies in the training set. We still chose Deep SAD as the reference baseline. Fig. 3(d) demonstrates that: (1) Even without knowledge of all non-target anomaly classes, GAD$^{f-partial}$ still exhibits significant improvements over the baseline. For example, when three classes of non-target anomalies remain unknown, GAD$^{f-partial}$ yields an AUPRC

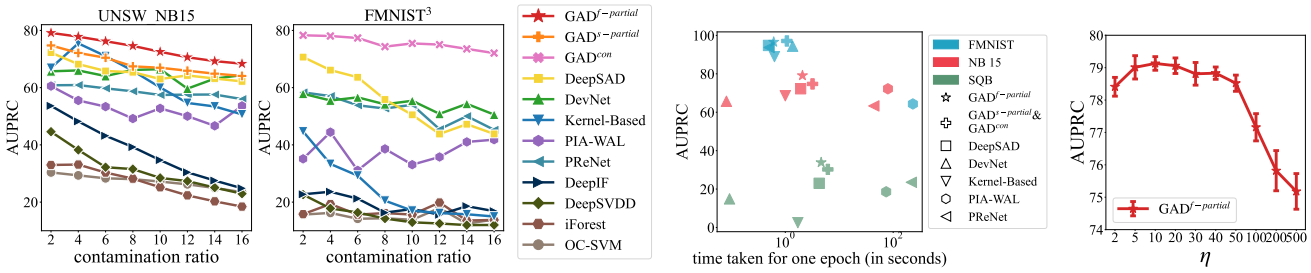


Figure 4: (a) Detection performance of models as a function of contamination ratio on the partial (left) and conventional (right) anomaly detection tasks. (b) Trade-off between the time required for each training epoch (represented on a logarithmic scale) and AUPRC performance. (c) Detection performance of GAD$^{f-partial}$ as a function of η on the UNSW_NB15 dataset.

improvement of up to 3.84%. (2) Utilizing only one labeled non-target anomaly per class still enhances the detection performance.

Unlike Fig. 3(d), which illustrates the model performance w.r.t. unknown non-target classes, Fig. 3(e) illustrates the model performance w.r.t. unseen non-target classes. The difference between unknown and unseen lies in the presence of corresponding non-target anomalies in the unlabeled set. Unseen non-target anomalies may not be part of the unlabeled set, which makes detection more challenging and requires a model with better generalization compared to the unknown scenario. Deep SAD was chosen for comparison due to its best performance among the semi-supervised methods. Fig. 3(e) demonstrates that: (1) GAD$^{f-partial}$ demonstrates strong generalization performance across unseen non-target anomalies, ranging from 0 to 4 classes. This demonstrates the generalization ability of using a small amount of labeled non-target anomalies to obtain an anchor $\boldsymbol{a}$ for assisting in target anomaly detection. (2) The performance of GAD$^{s-partial}$ declines slightly as non-target anomalies in the unlabeled data have not been encountered, causing the inaccurate positioning of anchor $\boldsymbol{a}$. (3) Both GAD variants demonstrate superior generalization performance compared to the best-performing baseline Deep SAD.

5.6 Robustness under Anomaly Contamination

We next evaluated the robustness of models against contamination ratios. The experiments for partial and conventional anomaly detection tasks were conducted on the UNSW_NB15 and FMNIST^3 datasets, respectively. We purposefully varied the contamination ratio in the training dataset from 2% to 16%, while the number of labeled target and non-target anomalies remained the same.

AUPRC w.r.t. different contamination ratios for the partial and conventional anomaly detection tasks is presented in Fig. 4(a). As the contamination ratio increases, the learning of the normal class becomes insufficient, resulting in varying degrees of decline in AUPRC across all methods. However, it is evident that all GAD variants maintain a stable and superior AUPRC, consistently outperforming all competing methods across all contamination ratios. The reason behind this is that the anomaly contamination primarily influences the $L_{compact}$ loss, the value of which is much smaller compared to that of L_{target}. Thus, the anomaly contamination has a minimal impact on the optimization of the proposed framework.

5.7 Time Efficiency and Sensitivity Analysis

We evaluated the time efficiency of three GAD variants and other semi-supervised baselines on three datasets. Fig. 4(b) illustrates the logarithmic representation of the time required for one epoch of model training and the corresponding AUPRC. The following was observed: (1) GAD$^{f-partial}$ exhibits time efficiency comparable to Deep SAD while achieving an AUPRC improvement ranging from 1.79% to 10.84% across different datasets. (2) GAD$^{s-partial}$ involves a sorting process but maintains comparable time efficiency to Deep SAD and enhances AUPRC by 0.83%-7.23% on various datasets. (3) Among the semi-supervised methods, PIA-WAL displays the poorest time efficiency due to challenges with GAN convergence. (4) DevNet shows distinct performance patterns on tabular datasets (SQB and UNSW_NB15) and the image dataset (FMNIST) due to differences in the default neural network architecture. DevNet on FMNIST uses ResNet with a parameter count of 11.69M [7], significantly larger than the variant of LeNet (0.22M) [10] used by other semi-supervised methods, including three GAD variants.

We next examine the model's sensitivity w.r.t. the hyperparameter η, varied from 2 to 500, which controls the weight contributed by $\mathcal{D}^a$ in $L_{compact}$. Fig. 4(c) shows the AUPRC of GAD$^{f-partial}$ w.r.t. different values of η. Increasing η appropriately strengthens the model's attention on marginal instances or non-target anomalies, thus promoting compact representations. However, excessively large η leads to insufficient learning of normal instances, and thus to a significant performance decline. The findings demonstrate that GAD$^{f-partial}$ achieves optimal performance on the UNSW_NB15 dataset when η is set to 10.

6 CONCLUSION

This paper introduces a generalized anomaly detection problem. We present an all-encompassing framework GAD and three variants tailored for different anomaly detection scenarios. Experimental results demonstrate significant improvements in detection performance over state-of-the-art baselines for both partial and conventional anomaly detection tasks.

7 ACKNOWLEDGMENT

This work was supported by Shanghai "Science and Technology Innovation Action Plan" Project (No.23511100700).

REFERENCES

[1] Charu C Aggarwal and Charu C Aggarwal. 2017. *An introduction to outlier analysis*. Springer.

[2] Leo Breiman. 2001. Random forests. *Machine learning* 45 (2001), 5–32.

[3] Bokai Cao, Mia Mao, Siim Viidu, and Philip Yu. 2018. Collective fraud detection capturing inter-transaction dependency. In *KDD 2017 Workshop on Anomaly Detection in Finance*. PMLR, 66–75.

[4] Jinghui Chen, Saket Sathe, Charu Aggarwal, and Deepak Turaga. 2017. Outlier detection with autoencoder ensembles. In *Proceedings of the 2017 SIAM international conference on data mining*. SIAM, 90–98.

[5] Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. 2023. Addressing heterophily in graph anomaly detection: A perspective of graph spectrum. In *Proceedings of the ACM Web Conference 2023*. 1528–1538.

[6] Astha Garg, Wenyu Zhang, Jules Samaran, Ramasamy Savitha, and Chuan-Sheng Foo. 2021. An evaluation of anomaly detection and diagnosis in multivariate time series. *IEEE Transactions on Neural Networks and Learning Systems* 33, 6 (2021), 2508–2517.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. arXiv:1512.03385 [cs.CV]

[8] Minqi Jiang, Chaochuan Hou, Ao Zheng, Xiyang Hu, Songqiao Han, Hailiang Huang, Xiangnan He, Philip S Yu, and Yue Zhao. 2023. Weakly supervised anomaly detection: A survey. *arXiv preprint arXiv:2302.04549* (2023).

[9] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *ICLR* (2013).

[10] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[11] Meng-Chieh Lee, Yue Zhao, Aluna Wang, Pierre Jinghong Liang, Leman Akoglu, Vincent S Tseng, and Christos Faloutsos. 2020. Autoaudit: Mining accounting and time-evolving graphs. In *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 950–956.

[12] Guoliang Li, Xuanhe Zhou, Ji Sun, Xiang Yu, Yue Han, Lianyuan Jin, Wenbo Li, Tianqing Wang, and Shifu Li. 2021. opengauss: An autonomous database system. *Proceedings of the VLDB Endowment* 14, 12 (2021), 3028–3042.

[13] Wenyuan Li, Yunlong Wang, Yong Cai, Corey Arnold, Emily Zhao, and Yilian Yuan. 2018. Semi-supervised rare disease detection using generative adversarial network. *arXiv preprint arXiv:1812.00547* (2018).

[14] Boyang Liu, Pang-Ning Tan, and Jiayu Zhou. 2022. Unsupervised Anomaly Detection by Robust Density Estimation. *Proceedings of the AAAI Conference on Artificial Intelligence* 36, 4 (Jun. 2022), 4101–4108.

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

[16] Nour Moustafa and Jill Slay. 2015. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 military communications and information systems conference (MilCIS)*. IEEE, 1–6.

[17] Guansong Pang, Longbing Cao, Ling Chen, and Huan Liu. 2018. 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*. 2041–2050.

[18] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. 2021. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)* 54, 2 (2021), 1–38.

[19] Guansong Pang, Chunhua Shen, Huidong Jin, and Anton van den Hengel. 2023. Deep weakly-supervised anomaly detection. *Proceedings of the 29th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2023).

[20] Guansong Pang, Chunhua Shen, and Anton van den Hengel. 2019. Deep anomaly detection with deviation networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 353–362.

[21] Guansong Pang, Anton van den Hengel, Chunhua Shen, and Longbing Cao. 2021. 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*. 1298–1308.

[22] Lorenzo Perini, Vincent Vercruyssen, and Jesse Davis. 2022. Transferring the contamination factor between anomaly detection domains by shape similarity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 4128–4136.

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

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

[25] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. 2020. Deep semi-supervised anomaly detection. *ICLR* (2020).

[26] Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. 2017. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *Information Processing in Medical Imaging: 25th International Conference, IPMI 2017, Boone, NC, USA, June 25-30, 2017, Proceedings*. Springer, 146–157.

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

[28] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. 2019. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2828–2837.

[29] Jianrong Tao, Jianshi Lin, Shize Zhang, Sha Zhao, Runze Wu, Changjie Fan, and Peng Cui. 2019. Mvan: Multi-view attention networks for real money trading detection in online games. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2536–2546.

[30] David MJ Tax and Robert PW Duin. 2004. Support vector data description. *Machine learning* 54 (2004), 45–66.

[31] Bowen Tian, Qinliang Su, and Jian Yin. 2022. Anomaly detection by leveraging incomplete anomalous knowledge with anomaly-aware bidirectional gans. *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence (IJCAI-22)* (2022).

[32] Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. 2023. Alleviating structural distribution shift in graph anomaly detection. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 357–365.

[33] Shuang Wu, Jingyu Zhao, and Guangjian Tian. 2022. Understanding and Mitigating Data Contamination in Deep Anomaly Detection: A Kernel-based Approach. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, Lud De Raedt (Ed.). International Joint Conferences on Artificial Intelligence Organization, 2319–2325. Main Track.

[34] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. *Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms*. arXiv:cs.LG/1708.07747 [cs.LG]

[35] Hongzuo Xu, Guansong Pang, Yijie Wang, and Yongjun Wang. 2023. Deep isolation forest for anomaly detection. *IEEE Transactions on Knowledge and Data Engineering* (2023).

[36] Hangting Ye, Zhining Liu, Xinyi Shen, Wei Cao, Shun Zheng, Xiaofan Gui, Huishuai Zhang, Yi Chang, and Jiang Bian. 2023. UADB: Unsupervised Anomaly Detection Booster. *2023 IEEE 39th International Conference on Data Engineering (ICDE)* (2023).

[37] Dong Young Yoon, Ning Niu, and Barzan Mozafari. 2016. Dbsherlock: A performance diagnostic tool for transactional databases. In *Proceedings of the 2016 international conference on management of data*. 1599–1614.

[38] Houssam Zenati, Manon Romain, Chuan-Sheng Foo, Bruno Lecouat, and Vijay Chandrasekhar. 2018. Adversarially learned anomaly detection. In *2018 IEEE International conference on data mining (ICDM)*. IEEE, 727–736.

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

[40] Simin Zhang, Bo Li, Jianxin Li, Mingming Zhang, and Yang Chen. 2015. A novel anomaly detection approach for mitigating web-based attacks against clouds. In *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*. IEEE, 289–294.

[41] Yue Zhao, Guoqing Zheng, Subhabrata Mukherjee, Robert McCann, and Ahmed Awadallah. 2023. Admoe: Anomaly detection with mixture-of-experts from noisy labels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 4937–4945.

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

[43] Weixian Zong, Fang Zhou, Martin Pavlovski, and Weining Qian. 2022. Peripheral Instance Augmentation for End-to-End Anomaly Detection Using Weighted Adversarial Learning. In *Database Systems for Advanced Applications: 27th International Conference, DASFAA 2022, Virtual Event, April 11–14, 2022, Proceedings, Part II*. Springer, 506–522.