SHRINKAGE-BASED WEAKLY-SUPERVISED FEATURE LEARNING TO ENHANCE IoT ANOMALY DETECTION

Huu Noi Nguyen[1], *Nguyen Ngoc Tran*[1], *Van Loi Cao*[1,*]

Abstract

IoT anomaly detection faces challenges due to the rarity of IoT anomalies and the limited availability of labels. Recent weakly-supervised approaches, like Feature Encoding with AutoEncoder and Weakly-supervised Anomaly Detection (FeaWAD) and an improvement on FeaWAD (iFWAD), address this scarcity by constructing detectors from a combination of unlabeled data and a small labeled anomalous set. While effective, these methods lack constraints during the feature learning stage to delineate normal regions from anomalies. Notably, the Shrink AutoEncoder promotes clustering of normal data around the origin while preserving space for anomalies. Drawing inspiration from the Shrink AutoEncoder, the study aims to introduce Shrink iFWAD (called sFWAD), embedding a shrink regularizer into iFWAD. This term compels the feature encoder of sFWAD to learn penalizing normal data that is close to zero, while simultaneously pushing IoT anomalies further away from zero. This process facilitates the anomalous score generator of sFWAD in efficiently identifying IoT anomalies. The proposed method is evaluated against state-of-the-art weakly-supervised techniques and other common anomaly detection methods using the N-BaIoT dataset. Experimental results indicate that sFWAD often surpasses recent weakly-supervised methods as well as the common techniques in IoT anomaly detection performance. For identifying unknown/new IoT anomalies, Missed Detection Rate from sFWAD (0.008) is much lower than those from iFWAD (0.026) and RoSAS (0.015).

Index terms

Weakly-supervised, latent representation, IoT anomaly detection, IoT botnet detection.

1. Introduction

Anomaly detection has found many applications across various domains, such as cyberattack detection, IoT anomaly detection, fraud detection, and healthcare. One of the biggest challenges of this field is the rarity of anomalies and their corresponding labels. In cybersecurity, particularly IoT anomaly detection, anomalies are inherently scarce and

[1]Institute of Information and Communication Technology, Le Quy Don Technical University
[*]Corresponding author, email: loi.cao@lqdtu.edu.com, DOI: 10.56651/lqdtu.jst.v13.n1.822.ict

difficult to collect, contrasting to the abundance of accessible normal data [1]. Machine learning (ML), particularly deep learning (DL), is extensively applied in diverse security contexts, including the detection of malicious codes, network anomaly detection and IoT botnet detection [2]–[6]. These methodologies consist of both supervised [2], [7], [8], unsupervised [9]–[11] as well as weakly-supervised learning [12], [13] approaches.

Given unlabeled data along with a small number of labeled IoT anomalies, the weakly-supervised learning approach has proven to be the most viable strategy [14], [15]. Broadly, weakly-supervised anomaly detection methods amalgamate elements from both supervised and unsupervised learning paradigms to identify anomalous patterns within data. The study carried out under the assumption that a small proportion of anomalies may be interspersed within the normal data during training, albeit typically in significantly lesser quantities compared to normal data [12], [13]. Thus, weakly-supervised learning can offer a promising avenue for mitigating the challenge of scarcity in anomaly detection, leveraging both labeled and unlabeled data to enhance detection accuracy.

Recent studies, such as [6], [12], [16], have introduced end-to-end weakly supervised methods for dealing with the lack of labeled anomaly data. These methods construct models capable of generating anomaly scores for query points using a combination of a few labeled anomalies and a larger volume of unlabeled data. The unlabeled dataset predominantly consists of normal instances, with a minor fraction of anomalies. One notable method, introduced by Zhou et al. [6], is the FeaWAD method. FeaWAD comprises two main components: a Feature Encoding Network (FEN) and an Anomaly Score Generator (ASG). This method employs a two-stage learning process, where the FEN is pre-trained for feature representation, followed by training the entire network to generate anomaly scores. However, the FEN, which utilizes a standard AutoEncoder, learns to represent features without any regularizers that could distinguish anomalies from normal data. This limitation persists even in the enhanced version of FeaWAD presented by Nguyen et al. [16].

Interestingly, Cao et al. [9] introduced the Shrink AutoEncoder (SAE) as a means to acquire a latent representation tailored for anomaly detection tasks. Through the integration of a shrink regularizer, this representation encourages normal data to cluster near the origin while preserving the remaining feature space for potential anomalies. Notably, SAE is constructed exclusively using normal data. Drawing inspiration from SAE [9], our objective is to incorporate a shrink regularizer into FeaWAD, thereby introducing our method, Shrink FeaWAD (sFWAD). In essence, the shrink term is embedded within the loss function of FEN, enhancing the feature representation during the pre-training stage and facilitating ASG during fine-tuning. To examine the contribution of the shrink regularizer to sFWAD, this study operates under the assumption that unlabeled IoT data remains uncontaminated by IoT anomalies. This is also followed the same strategy used in [6], [12]. The scenario involving unlabeled IoT data contaminated with IoT anomalies is deferred to future research. In this study, the N-BaIoT dataset [17] is employed for evaluation of our proposed method in which two types of botnet attacks (i.e., Gafgyt and Mirai) are considered as IoT anomalies. Thus, two concepts such as IoT anomalies and IoT attacks can be used interchangeably in this study. Detailed descriptions of our proposed method are provided in Section 4.

The main contribution of this study can be listed as follows:

1) Propose a novel sFWAD method with a shrinkage regularizer to improve the performance of recent weakly-supervised methods for IoT anomaly detection.
2) Design a series of experiments to assess the effectiveness of our proposed method, sFWAD, in contrast to the most recent weakly-supervised methods and conventional anomaly detection techniques using the N-BaIoT dataset. Our thorough analysis and discussion of the experimental outcomes highlight the strengths and limitations of sFWAD, along with proposing future research directions.

The structure of this paper is shown as follows. Section 2 and 3 introduce two end-to-end weakly-supervised methods as well as brief discussion on recent weakly-supervised approaches for anomaly detection. Section 4 presents our proposed method, sFWAD that tackles the challenges identified in this study. Section 5 describes experimental analysis and compares with relevant studies. Finally, Section 7 concludes the paper by discussing the findings and future works.

2. Background on weakly-supervised approaches

This section briefly provides a general problem statement for weakly-supervised learning approach. Following this, two weakly-supervised anomaly detection methods, namely FeaWAD [6] and iFWAD [16], are presented.

Generally, weakly-supervised methods for anomaly detection are methodologies that integrate aspects of both supervised and unsupervised learning to detect anomalous patterns within data. This technique proves especially beneficial in situations where labeled data is scarce [12], [13]. Let's consider a training dataset X consisting of $N + K$ elements denoted as $X = \{x_1, ..., x_N, x_{N+1}, ..., x_{N+K}\}$, where each x_i belongs to the d-dimensional space. Here, $X_U = \{x_1, x_2, ..., x_N\}$ represents the unlabeled data (U), while $X_K = \{x_{N+1}, x_{N+2}, ..., x_{N+K}\}$ is a small set of labeled anomalies (P), where $K \ll N$. Note that X_U may contain both unlabelled normal data and a contamination of anomalies.

Weakly-supervised methods aim to construct a scoring function $\phi : X \mapsto \mathbb{R}$ from the training data of U and P. The function ϕ can assign anomalous scores for querying data points. This function should ensure that for any given anomalous data object x_i, its score will be higher ($\phi(x_i) > \phi(x_j)$) compared to a normal data object x_j [12].

2.1. The FeaWAD method

Zhou et al. [6] introduced a method, called FeaWAD, for working on unlabelled data with a small set of anomalies. FeaWAD consists of two components, namely FEN using an AutoEncoder (AE) and ASG using Multilayer Perceptron Network (MLP), as depicted in fig. 1. In overview, FeaWAD is trained in a two-stage approach: (1) training FEN on the U set to obtain a new feature representation, and (2) co-training FEN and ASG on U and P for generating anomaly score.

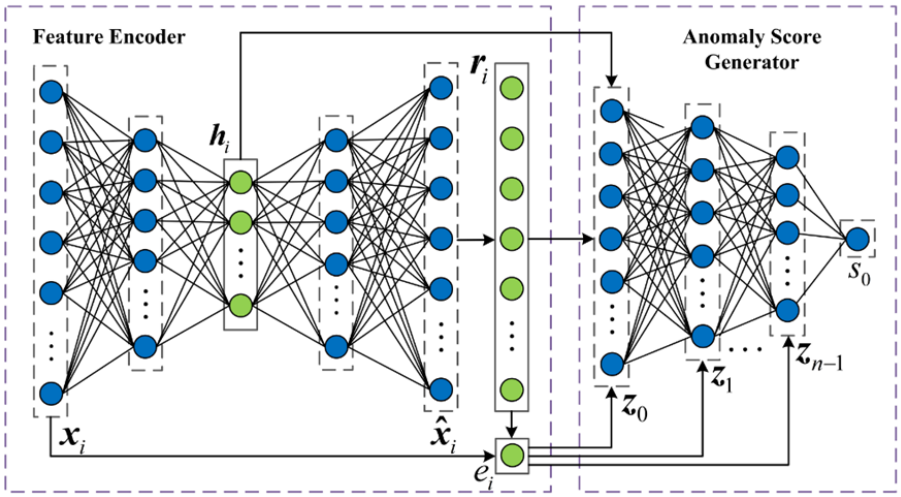

Fig. 1. The network structure of FeaWAD [6].

Initially, FEN (an AE) decomposes the input data (U) into three distinct elements: hidden representation denoted as h, reconstruction error symbolized by e, and residual vector represented as r. These components are integral to the model's data representation strategy. The calculation formulas are respectively:

$$h = f_{en}(x_i; W_{en}), \tag{1a}$$

$$e = \|\hat{x}_i - x_i\|_2, \tag{1b}$$

$$r = \frac{\hat{x}_i - x_i}{\|\hat{x}_i - x_i\|_2}, \tag{1c}$$

where W_{en} and $f_{en}(\cdot, W_{en})$ refer to the encoder's parameters, and the mapping function for the encoding process, respectively. W_{de} and $f_{de}(\cdot, W_{de})$ are also the decoder's parameters and the mapping function for the decoding process. $\hat{x}_i$ is the reconstruction value of $x_i \in U$, and calculated by $\hat{x}_i = f_{de}(h; W_{de})$. $\|\cdot\|$ is the Euclidean norm.

To estimate anomaly scores, the ASG computes each layer's output, z_k, follows a specific formula as follows,

$$z_k = f(W_m^k z_{k-1} + b_k + w_e^k e), \quad k \in \{1..n\}, \tag{2}$$

where W_m^k is the weight matrix, b_k is the bias parameters for the k-layer. At the first layer $(k-1)$, z_{k-1} is assigned by the combination vector $[r, h]$, while at the last layer, z_k determines the score s_0 for anomaly decision.

Thus, the FeaWAD network is composed of two components: a feature encoding network (FEN) $\psi(\cdot, \Theta_f)$ and anomaly score generator (ASG) $\varphi(\cdot, \Theta_a)$. The Θ_f and Θ_a are parameters of models FEN and ASG, respectively. Given an input sample x_i, FEN extracts three factors h_i, r_i, e_i as described Eq. (1a), (1b) and (1c). The entire network is represented as the function $\phi(x_i; \Theta_f; \Theta_a)$, or $\phi(x_i)$ in short.

The following formula shows how the loss function of the FeaWAD is calculated:

$$L(\Theta_f, \Theta_a) = L_d(\Theta_f, \Theta_a) + \lambda L_e(\Theta_f), \tag{3}$$

where, $L_e = \sum_i (1 - y_i)e_i + y_i \max(0, a_0 - e_i)$ is the reconstruction error on both U and P in which encourages the reconstruct of U while prevents the learning from P, resulting the e to be larger than a_0; a_0 is a predefined margin of e; $L_d = \sum_i (1-y_i)|\phi(x_i)| + y_i \max(0, a_0 - \phi(x_i))$ is the loss function of ASG that helps ASG produce the anomalous cores of anomalies that are higher than a_0; and $y_i \in \{0, 1\}$ is the label of samples ($y_i = 1$ if the sample is an anomaly), and λ is the trade-off parameter between the two components. Thus, FeaWAD can learn to encourage FEN to reconstruct well for U and reproduce poorly for P, while helping ASG to generate anomaly scores for distinguishing between anomalies and normal data.

The network undergoes a two-phase gradient descent training. The first phase involves pre-training the FEN $\psi(\cdot; \Theta_f)$ using reconstruction loss $L_{fen}(\Theta_f) = \sum_i e_i$, where e_i is defined in Eq. (1b). FEN is trained on unlabeled dataset (U) to establish a foundational encoding strategy. Following this, the entire network $\phi(\cdot; \Theta_f; \Theta_a)$ is fine-tuned with both unlabeled dataset (U) and labeled anomalous data (P) for generating anomalous score. Note that, a balanced mini-batch training approach is adopted for the fine-tuned stage. This means that equal numbers of unlabeled and anomalous samples are used, leading to a more frequent selection of anomalies.

2.2. The iFWAD method

This section briefly presents a recent improvement on FeaWAD, iFeaWAD (iFWAD for short) from the study [16]. This refinement is the basis for us to evaluate and develop our proposed method in the following parts.

iFWAD is focused on adjusting the value of r within the triple values h, e, and r of the FeaWAD model. Instead of employing the original formula $(\hat{x}_i - x_i)$ as shown in Eq. (1c), the authors use the absolute value $|\hat{x}_i - x_i|$. This enhancement arises from the utilization of the ReLU activation function in the original method. If the component e goes with the absolute, all elements of the input vector $[r, h]$ are non-negative. This can make the training process of ASG become easier. This adjustment can streamline the training process of ASG. Here, the authors prioritize the actual values of the input data for ASG over the directional aspects of the input vector.

Furthermore, the authors introduce an additional hyperparameter, denoted as σ, which serves to scale r for input into ASG. The aim is to guarantee that the input features obtained from r for ASG are minimized, potentially smaller when compared to those derived from h. Empirically, the value of σ was set to 10^{-5}. Then, the formula of r can be rewritten as follows:

$$r^* = \frac{\sigma \cdot |\hat{x}_i - x_i|}{\|\hat{x}_i - x_i\|_2}. \tag{4}$$

The latent representation h and the RE e are followed as the same as in the original study [6]. Other hyper-parameters and the two-stage training process are also based on FeaWAD.

3. Related work

In a weakly-supervised manner, some recent works have explored anomaly detection with extremely unbalanced data, using the scarce but useful anomalous samples [12], [13], [15]. A feedback-guided anomaly detection framework was proposed by Siddiqui et al. [18]. The model can potentially enhance unsupervised anomaly detection by leveraging the analyst's existing knowledge to allocate elevated scores to instances more likely to be anomalous. Ruff et al. [13] presents Deep SAD (an extended version of Deep SVDD [19]), an end-to-end methodology for anomaly detection that utilizes both labeled and unlabeled data. It introduces an information-theoretic perspective, suggesting that the entropy of normal data's latent distribution should be lower than that of anomalous data. Deep SAD introduced a novel loss based on the information-theoretic analysis to pull the normal data towards a fixed centroid and push the anomalies away. In [12], Pang et al. propose an anomaly detection model with a scenario for limited labeled anomaly data and unlabeled data. The model transforms the anomaly detection task into an ordinal regression problem of pairwise relationships, using the full number of labeled anomaly samples to form sample pairs for the next anomaly detection process.

In [6], Zhou et al. propose an AE-based method to extract three components, including hidden representation, reconstruction error, and reconstructed residual vector, to characterize each input data. These features are then used to generate the anomaly generator, which is used to classify the data as normal or out of the ordinary. Our research is based on this research idea, with improvements in specific components to improve the efficiency of anomaly detection for IoT networks. Pang et al. [14] present PReNet, a novel method that can detect known and unknown anomalies by learning and predicting the relationships between data pairs, significantly outperforming other methods. In [20], Xu et al. introduce a weakly-supervised anomaly detection method that's robust against unlabeled anomalies, using continuous supervision to improve detection accuracy, especially in IoT networks.

Despite progress, IoT anomaly detection research faces limitations such as data scarcity, model generalization challenges, adaptability issues, computational intensity, and high false alarm rates. Future efforts should focus on creating more efficient, adaptable detection methods suitable for the dynamic IoT environment.

4. Proposed method

Beginning with the assumption that a significant portion of IoT data comprises benign samples with a limited number of IoT labeled anomalies, it's observed that these benign samples often exhibit common traits, rendering them more likely to cluster in localized regions. Conversely, IoT anomalies tend to diverge from each other, manifesting in low-density regions. In a study conducted by Cao et al. [9], the authors introduced the concept of a SAE incorporating a shrink regularizer within its loss function. By employing this regularizer to compress normal data towards the origin, the SAE can effectively generate a "favorable" latent representation conducive to anomaly detection. Consequently, in the latent

space, normal samples are compelled to congregate near the origin, while anomalies are anticipated to manifest at a considerable distance from it.

Drawing inspiration from this concept, the research objective is to integrate a shrinkage term into the iFWAD method, thus giving rise to sFWAD. Essentially, this involves penalizing the h values of benign data to approximate zero, while encouraging those of IoT anomalies to be far from zero. This can enhance the differentiation between benign data and IoT anomalies within the latent feature space of FEN. This enhancement in discrimination aids the subsequent component, ASG, in learning anomalous scores crucial for identifying IoT anomalies. Concretely, the shrink regularizer applied to the value of h is incorporated into two key training stages of FEN. Firstly, during the pre-training of FEN, the shrink regularizer (the 2^{nd} term in Eq. (5)) will penalize the latent values of benign data to approximately zero. Note that this study is carried out under the assumption that unlabeled IoT data remains uncontaminated by IoT anomalies. Secondly, during the end-to-end training sFWAD, the shrink regularizer forces FEN to reprensent benign data to be close to zero, pushing IoT anomalies far away from zero. This is illustrated in the Eq. (8). This integration is manifested by augmenting the reconstruction error term, as denoted by Eq. (1b), with the shrinkage term to yield e^* as delineated in Eq. (5) below,

$$e^* = e + \gamma\|h\|_2 = \|\hat{x}_i - x_i\|_2 + \gamma\|h_i\|_2, \tag{5}$$

where the second term in (5) is the shrinkage with a trade-off parameter γ.

Hidden representation h and residual vector r^* are keep as in Eq. (1a) and (4). Based on the modification in Eq. (5), when rewriting the formula for ASG, z_k will be recalculated according to the following formula:

$$\begin{aligned} z_k^* &= f(f_m^k z_{k-1} + b_k + w_{e^*}^k e^*) \\ &= f(f_m^k z_{k-1} + b_k + w_{e^*}^k(\|\hat{x}_i - x_i\|_2 + \gamma\|h_i\|_2)). \end{aligned} \tag{6}$$

Hence, our proposed method, sFWAD, comprises two components akin to FeaWAD and iFWAD, albeit with alterations concerning the input of ASG and the loss functions of FEN across two distinct training stages. Consequently, the FEN component within sFWAD is symbolized as $\psi(\cdot, \Theta_f^*)$, and, while the subsequent ASG component is also represented as $\varphi(\cdot, \Theta_a^*)$, where Θ_f^* and Θ_a^* denote the respective parameters of FEN and ASG. The entire network is denoted as $\phi^*(x_i, \Theta_f^*, \Theta_a^*)$ or more concisely as $\phi^*(x_i)$.

The formula in Eq. (3) can be rewritten for the loss function of sFWAD as follows:

$$L^*(\Theta_f^*, \Theta_a^*) = L_d^*(\Theta_f^*, \Theta_a^*) + \lambda L_e^*(\Theta_f^*). \tag{7}$$

The second component of the loss Eq. (7), L_e^* contains the shrink regularizer, $\|h\|_2$. The

specific formula for L_e^* is rewritten as in Eq. (8) below:

$$\begin{aligned} L_e^* &= \sum_i (1-y_i)e_i^* + y_i \max(0, a_0 - e_i^*) \\ &= \sum_i (1-y_i)(e_i + \gamma\|h_i\|_2) + y_i \max(0, a_0 - (e_i + \gamma\|h_i\|_2)) \\ &= \sum_i (1-y_i)(\|\hat{x}_i - x_i\|_2 + \gamma\|h_i\|_2) + y_i \max(0, a_0 - (\|\hat{x}_i - x_i\|_2 + \gamma\|h_i\|_2)), \end{aligned} \tag{8}$$

where h_i, e_i are extracted from the input sample x_i by Eq. (1a), (1b), respectively. L_e^* encourages FEN reconstruct well on unlabeled data as well as penalize the latent vector h, while preventing FEN from learning on anomalies.

First term L_d^* in the loss function of sFWAD can be obtained by replacing ϕ^* from L_d as follows,

$$L_d^* = \sum_i (1-y_i)|\phi^*(x_i)| + y_i \max(0, a_0 - \phi^*(x_i)). \tag{9}$$

Again, this term aims to encourage the anomalous cores of anomalies being higher than the threshold a_0, while that of normal data being close to zero.

Similarly to FeaWAD, our proposed method uses a two-stage training process. In the first phase, FEN $\psi(\cdot;\Theta_f^*)$ is trained with the loss function $L_{fen^*}(\Theta_f^*) = \sum_i e_i^*$, where e_i^* is described in Eq. (5). The unlabeled dataset (U) is involved in the process to establish the pre-trained FEN model for the second training phase. Subsequently, sFWAD $\phi^*(\cdot;\Theta_f^*;\Theta_a^*)$ is fine-tuned with both U and P for generating anomalous score.

In real-world scenarios, the unlabeled dataset (U) may contain normal examples along with a small proportion of IoT anomalies. Despite this, the majority of the unlabeled dataset still consists of normal data. Consequently, the shrink regularizer can assist FEN in learning a "good" representation from both U and P, ultimately leading to efficient performance of sFWAD. Additionally, a study by Nguyen et al. [21] suggested that SAE [9] with a shrink regularizer can perform efficiently with up to 5% contamination of anomalies.

5. Experiments

Our experiments are designed to investigate our proposed method in two scenarios: identifying known IoT anomalies and identifying unseen IoT anomalies. The first scenario aims to test its performance in identifying IoT anomalies from the same category used for training. In the second scenario, we aim to explore its ability to detect unseen/new types of IoT anomalies. The performance of sFWAD is evaluated in comparison to that of FeaWAD, as originated in [6], and its development version iFWAD [16], as well as two recent weakly-supervised learning methods: Prenet [14] and RoSAS [20], and well-known anomaly detection methods. The well-known methods consist of Isolation Forest (IF), Local Outlier Factor (LOF), and One-class Support Vector Machine (OCSVM). The N-BaIoT dataset [17] is utilized for evaluation of the above methods. The two types of botnet attacks

(i.e. Gafgyt and Mirai) from N-BaIoT are considered as IoT anomalies, thus, the concepts of IoT anomalies and IoT attacks are used interchangeably in this study. The performance of these methods is measured using the metrics outlined in Subsection 5.3. For experimental settings, the dataset description, parameter settings, and evaluation metrics are presented in the subsections below.

5.1. Datasets

The N-BaIoT dataset[1], which was originally introduced by the authors from the Ben-Gurion University of the Negev (BGU) [17], is employed for evaluated our proposed model. N-BaIoT encompasses data from nine different IoT devices from four categories such as doorbells, thermostats, monitors, and cameras/webcams. Each traffic connection is characterized by 115 features, providing a ready-to-used data for analysis and machine learning/deep learning-based methods. The details of N-BaIoT are shown in table 1.

The dataset consists of two well-known botnet types, namely Gafgyt and Mirai as well as benign traffic. These botnets are used to infect IoT devices and launch DDoS attacks. Gafgyt mainly uses SYN, UDP, and ACK Flooding attacks, while Mirai is often highly sophisticated and dangerous to various IoT devices with different kinds of DDoS attacks (i.e. based on TCP, UDP and HTTP protocols). Thus, these botnets can yields multiple DDoS attacks resulting in the variety of network traffic patterns from the infected devices.

Table 1. The description of the NBaIoT dataset

ID	Device Name	Type	Benign	Gafgyt	Mirai
D1	Danmini	Doorbell	49548	652100	316650
D2	Ecobee	Thermostat	13113	512133	310630
D3	Ennio	Doorbell	39100	316400	
D4	Philips B120N10	Monitor	175240	312273	610714
D5	Provision PT 737E	Camera	62154	330096	436010
D6	Provision PT 838	Camera	98514	309040	429337
D7	Samsung SNH 1011 N	Webcam	52150	323072	
D8	SimpleHome XCS7 1002 WHT	Camera	46585	303223	513248
D9	SimpleHome XCS7 1003 WHT	Camera	19528	316438	514860

The original data from table 1 is sampled to create datasets for the two scenarios mentioned above. Initially, data from each IoT device undergoes random sampling, allocating 80% for training and 20% for evaluation purposes. In the scenario aimed at identifying known IoT attacks, Gafgyt is selected for experimentation due to the absence of Mirai on certain IoT devices (specifically, D3 and D7). Normal instances within the training dataset are considered as an unlabeled dataset (U_N), while a small portion of Gafgyt instances is randomly selected to form a labeled IoT anomaly set (P). The size of IoT anomaly set versus the unlabelled set is randomly generated within the range (2%, 20%) for each IoT device, with actual proportions displayed in the "Outlier Perc" column of table 2 from 3.85% to 18.03%. This variation in data ratios contributes to a more precise evaluation of the experimental outcomes.

[1]https://archive.ics.uci.edu/ml/datasets/detection_of_IoT_botnet_attacks_N_BaIoT

For the scenario detecting unseen or new IoT attacks, if Gafgyt is utilized for training, Mirai will be employed for evaluation, and vice versa. The proportion of labeled IoT anomalies for each IoT device remains consistent with that of the first scenario. However, IoT devices D3 and D7 are excluded from this experiment due to the absence of Mirai on these devices.

5.2. Parameter settings

Firstly, the hyper-parameters of FeaWAD, iFWAD and sFWAD are used as FeaWAD reported in the original study [6]. For these methods, FEN consists of three hidden layers $\{100, 50, 100\}$ and an input (also an output) layer of 115 neurons, while ASG has two hidden layers of $\{256, 32\}$ and an output layer of 1 neurons. Following the recommendations outlined in [6], the parameters, specifically the margin (a_0) and λ, are configured to values of 5 and 1 respectively. Additionally, the learning rate is set to a conventional value of 10^{-3}. These methods are trained on 100 epochs with the batch size of 64. In addition, the σ parameter of sFWAD is set to 10^{-5} followed iFWAD in [16]. The parameter γ for trading off two terms of the FEN loss in sFWAD is equal to 5 followed the study [9]. Furthermore, Prenet and RoSAS are configured with parameters consistent with those outlined in their original studies [14], [20]. Similarly, the IF, LOF, and OCSVM methods utilize default parameter values.

All experiments were implemented in Python using the Keras, scikit-learn, and Tensorflow frameworks. We ran the experiments on a computer with Ubuntu 22.04 LTS, an Intel(R) Core i5 11400H CPU, 24 GB of RAM, and a Geforce 3050 GPU.

5.3. Evaluation metrics

In experiments, we utilize the Area Under the Curve (AUC) as a key metric to evaluate the performance of both the proposed models and those related to our study. The AUC is the entire area below the Receiver Operating Characteristic (ROC) curve. The ROC curve shows the true positive rate (TPR) versus false positive rate (FPR) for different thresholds.

$$TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN}, \tag{10}$$

where TP, TN, FP and FN refer to True Positives, True Negatives, False Positives and False Negatives, respectively. Thus, AUC can measure the performance for all possible thresholds.

To further assess the precision of these models, we incorporate two additional measures: False Alarm Rate (FAR) and Missed Detection Rate (MDR) as in Eq. (11). FAR measures the proportion of negative samples that are incorrectly classified as positive, while MDR measures the proportion of positive samples that are incorrectly classified as negative. These metrics are instrumental in quantifying the models' ability to accurately identify true positives while minimizing the instances of false alerts and overlooked anomalies.

$$FAR = \frac{FP}{FP + TN}, \quad MDR = \frac{FN}{FN + TP}. \tag{11}$$

6. Results and discussion

This section provides analysis and discussion on the experimental results of the two scenarios. Tables 2 and 3 illustrate the results for the first scenario, while the results for the second scenario are shown in Tables 4 and 5.

6.1. Known IoT anomaly detection

This section analyzes the experimental results of the first scenario: training and evaluating on the same type of IoT anomalies (i.e. Gafgyt) with the AUC metric. Please note that FeaWAD, iFWAD, and sFWAD follow a two-stage training process: (1) pre-training their initial component (FEN) on an unlabeled dataset (U_N); (2) conducting end-to-end training on the entire network using both U and P datasets. In contrast, well-known methods such as IF, LOF, and OCSVM utilize a different approach where the unlabeled dataset (U_N) is combined with the labeled IoT anomaly dataset (P) to create a normal dataset with contamination of IoT anomalies. In addition, we further investigates the performance of these methods in terms of the FAR and MDR metrics. This is done on the device D1 using both Gafgyt and Mirai for evaluation.

Table 2. Performance on identifying known anomalies

ID	Outlier Perc	AUC							
		IF	**LOF**	**OCSVM**	**FeaWAD**	**iFWAD**	**Prenet**	**RoSAS**	**sFWAD**
D1	18.03	0.958	0.365	0.970	0.979	0.987	0.980	0.989	**0.994**
D2	14.53	0.963	0.340	0.976	0.979	0.987	0.979	0.990	**0.993**
D3	2.91	0.927	0.537	0.967	0.962	0.986	0.989	**0.990**	**0.990**
D4	7.41	0.971	0.446	0.980	0.979	**0.990**	0.979	**0.990**	**0.990**
D5	6.54	0.977	0.349	0.981	0.979	0.990	0.977	**0.995**	0.990
D6	15.97	0.950	0.345	0.955	0.979	0.990	0.969	0.984	**0.989**
D7	15.25	0.888	0.364	0.900	0.982	0.989	0.989	**0.990**	0.973
D8	18.70	0.955	0.365	0.960	0.986	0.909	0.990	0.963	**0.993**
D9	3.85	0.986	0.552	0.985	0.985	0.988	0.989	0.990	**0.995**

The results from table 2 indicate that well-known learning methods (IF, LOF, OCSVM) often yield lower AUC values compared to other weakly-supervised learning methods across all IoT devices. This can be attributed to the inherent limitations of shallow/stand-alone methods such as OCSVM, LOF, and IF, which are typically less powerful than deep/end-to-end methods, as demonstrated in [9].

When focusing exclusively on weakly-supervised learning methods, it becomes evident that our proposed method, sFWAD, consistently showcases significant performance superiority over both the original FeaWAD and the previously enhanced iFWAD. For instance, consider device D3: sFWAD demonstrates a notably higher AUC of 0.990 compared to FeaWAD's AUC of 0.962. Similarly, in comparison to iFWAD, sFWAD consistently outperforms across nearly all devices, except for datasets D4 and D7. In addition, our method, sFWAD, outperforms both Prenet and RoSAS across the majority of IoT devices. Specifically, sFWAD achieves superior performance on 7 datasets compared to RoSAS' 4 datasets. Regarding

Table 3. Evaluate FAR and MDR on D1 in identifying known anomalies

Model	Gafgyt			Mirai		
	AUC	**FAR**	**MDR**	**AUC**	**FAR**	**MDR**
IF	0.963	0.039	0.393	0.985	0.008	0.045
LOF	0.340	0.161	0.899	0.346	0.170	0.949
OCSVM	0.976	0.023	0.129	0.985	0.010	0.056
FeaWAD	0.979	0.003	0.017	0.353	0.006	0.034
iFWAD	0.987	0.003	0.017	0.989	**0.003**	0.011
PreNet	0.979	0.012	0.015	0.988	**0.003**	0.012
RoSAS	0.990	0.040	**0.006**	0.989	0.006	0.009
sFWAD	**0.993**	**0.002**	**0.006**	**0.993**	**0.003**	**0.007**

Prenet, sFWAD consistently produces higher AUC values, with the exception of the 7th device.

Apart from AUC, we also investigate the efficiency of our proposed method using the FAR and MDR metrics. For this analysis, data from the first device is selected. The experimental results are presented in table 3. These results indicate that sFWAD outperforms across all three criteria: AUC, FAR, and MDR. Notably, the false alarm rate and misidentification rate of sFWAD are considerably low. Specifically, according to the FAR criterion, iFWAD, Prenet, and sFWAD yield identical results on the Mirai data. Regarding the MDR criterion, sFWAD and RoSAS produce equivalent results on the Gafgyt dataset.

6.2. Unknown IoT anomaly detection

This section provides analysis on the performance of our proposed method on unseen or new IoT attack type. Specifically, if the Gafgyt attack is utilized for training, the testing phase will consist of the Mirai attack, and vice versa. Other training and testing processes are followed as in the first scenario in Subsection 6.1. We also carry out two investigations such as evaluating the methods with the AUC metric on 7 devices, and further analysis the FAR and MDR on the device D1. The results obtained are presented in table 4 and 5, respectively.

In the realm of well-known anomaly detection methods, Table 4 highlights the superiority of our approach, sFWAD, over established techniques like IF, LOF, and OCSVM across two cases: training on Gafgyt and testing on Mirai, and vice versa. Notably, when compared with FeaWAD and iFWAD, sFWAD consistently delivers superior outcomes across most devices, with the exception of D6 and D9 in identifying Mirai, where competitive performance is observed. The superior performance of sFWAD over iFWAD in identifying unknown attacks can be attributed to the shrink regularizer's role in facilitating FEN to learn a "good" latent representation for sFWAD. This regularization compels benign data to cluster near zero, while known IoT attacks are pushed away from this region. Since unknown IoT attacks typically exhibit distinct characteristics from benign data, they are mapped to significantly different positions than benign data within the hidden layer of FEN, appearing far from zero. Furthermore, in comparison to cutting-edge methodologies like Prenet and RoSAS, sFWAD exhibits notably better performance. A closer examination reveals that sFWAD tends to perform better in detecting Mirai anomalies, while Prenet and RoSAS excel in the

Table 4. Performance on identifying unknown/new anomalies

ID	Data	Outlier Perc	AUC							
			IF	LOF	OCSVM	FeaWAD	iFWAD	Prenet	RoSAS	sFWAD
D1	G/M	18.03	0.985	0.476	0.980	0.986	0.967	0.806	0.611	**0.987**
	M/G	18.03	0.803	0.162	0.972	0.984	0.822	**0.990**	0.987	0.987
D2	G/M	14.53	0.987	0.473	0.982	0.939	0.988	0.987	0.602	**0.990**
	M/G	14.53	0.867	0.136	0.962	0.892	0.972	**0.990**	**0.990**	**0.990**
D4	G/M	7.41	0.987	0.695	0.985	0.626	0.924	**0.987**	0.595	**0.987**
	M/G	7.41	0.966	0.070	0.981	0.987	0.987	**0.989**	0.987	0.983
D5	G/M	6.54	0.987	0.610	0.986	0.988	0.983	0.611	0.791	**0.989**
	M/G	6.54	0.983	0.055	0.981	0.984	0.987	0.986	0.980	**0.987**
D6	G/M	15.97	0.987	0.462	0.981	**0.989**	**0.989**	0.640	0.729	**0.989**
	M/G	15.97	0.516	0.162	0.971	0.636	0.988	**0.990**	**0.990**	0.989
D8	G/M	18.70	0.975	0.480	0.979	0.973	0.806	0.592	0.684	**0.985**
	M/G	18.70	0.906	0.170	0.972	0.653	0.792	**0.990**	**0.990**	0.986
D9	G/M	3.85	0.979	0.686	0.987	**0.988**	**0.988**	0.943	0.749	**0.988**
	M/G	3.85	0.970	0.034	0.983	0.984	0.982	**0.989**	0.985	**0.989**

context of Gafgyt anomalies. Overall, our proposed method demonstrates a consistently stable performance with a higher frequency of superior results.

Interestingly, training on Gafgyt and testing on Mirai lead to the better performance of sFWAD comparing to the others over all devices. The results can be explained that Gafgyt is quite similar to benign, whereas Mirai is more deviated. Hence, training on Gafgyt and testing on Mirai can be slightly easier than training on Mirai for detecting Gafgyt. This statement has been discussed and confirmed in a previous study [10]. In some instances, when trained on Mirai and tested on Gafgyt, Prenet performs better than sFWAD on D2, D8, and D9, while sFWAD yields better results on D5.

In the context of FAR and MDR, table 5 provides a comprehensive evaluation of the methods concerning unseen or new IoT anomalies originating from D1. Notably, when the model is trained on Gafgyt and tested on Mirai, the sFWAD method emerges as the clear frontrunner, showcasing superior performance across all evaluation metrics. However, in the scenario where training occurs on Mirai and testing on Gafgyt, while sFWAD maintains its lead in terms of AUC and MDR, Prenet demonstrates a more favorable outcome in FAR. These findings underscore the specific strengths of sFWAD within certain training and testing contexts while also indicating areas for potential enhancement to achieve more consistent and comprehensive performance across all evaluation criteria. Overall assessment, sFWAD gives higher AUC values and lower FAR than the original FeaWAD method, as well as other competing methods.

In summary, the proposed sFWAD method generally performs better on identifying unseen and new IoT anomalies, particularly on Mirai. When trained on Mirai and tested on Gafgyt, it shows slightly competitive performance to the latest methods, Prenet and RoSAS. This will be a focal point for improvement in our future research.

Table 5. Evaluate FAR and MDR on $D1$ in identifying unknown/new anomalies

Model	Gafgyt/Mirai			Mirai/Gafgyt		
	AUC	**FAR**	**MDR**	**AUC**	**FAR**	**MDR**
IF	0.987	0.004	0.027	0.867	0.073	0.448
LOF	0.473	0.103	0.604	0.136	0.170	0.800
OCSVM	0.982	0.017	0.100	0.962	0.026	0.154
FeaWAD	0.939	0.012	0.068	0.892	0.089	0.524
iFWAD	0.988	0.067	0.397	0.972	0.004	0.026
PreNet	0.987	0.006	0.037	0.990	**0.001**	0.015
RoSAS	0.602	0.066	0.389	0.990	0.002	0.012
sFWAD	**0.990**	**0.004**	**0.022**	**0.991**	0.015	**0.008**

7. Conclusion and future work

This study introduces a novel approach called sFWAD, which integrates a shrink regularizer into the FeaWAD method. This integration empowers sFWAD to learn a more refined representation of normal IoT data during pre-training, subsequently producing precise anomalous scores during fine-tuning. By effectively distinguishing IoT anomalous data from benign points, sFWAD significantly enhances the accuracy of anomaly detection models.

The sFWAD method represents a state-of-the-art solution for identifying anomalies in IoT networks. Through comprehensive evaluations against recent weakly-supervised techniques and conventional anomaly detection methods using the N-BaIoT dataset, our proposed method consistently outperforms recent approaches, demonstrating superior performance in IoT anomaly detection.

Moving forward, we plan to enhance the sFWAD method by prioritizing improvements in computational efficiency and extending its applicability across diverse IoT environments. Additionally, our future efforts will focus on enabling real-time detection capabilities and integrating advanced feature selection techniques, deep learning models, and collaborative frameworks. These enhancements aim to further elevate the accuracy and robustness of sFWAD in IoT anomaly detection tasks.

References

[1] A. E. Omolara, A. Alabdulatif, O. I. Abiodun, M. Alawida, A. Alabdulatif, H. Arshad *et al.*, "The internet of things security: A survey encompassing unexplored areas and new insights," *Computers & Security*, vol. 112, p. 102494, 2022. doi: 10.1016/j.cose.2021.102494

[2] J. Liu, X. Song, Y. Zhou, X. Peng, Y. Zhang, P. Liu, D. Wu, and C. Zhu, "Deep anomaly detection in packet payload," *Neurocomputing*, vol. 485, pp. 205–218, 2022. doi: 10.1016/j.neucom.2021.01.146

[3] C. Qiu, T. Pfrommer, M. Kloft, S. Mandt, and M. Rudolph, "Neural transformation learning for deep anomaly detection beyond images," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8703–8714.

[4] B. N. Vi, H. N. Nguyen, N. T. Nguyen, and C. T. Tran, "Adversarial examples against image-based malware classification systems," in *2019 11th International Conference on Knowledge and Systems Engineering (KSE)*. IEEE, 2019. doi: 10.1109/KSE.2019.8919481 pp. 1–5.

[5] A. Prasad and S. Chandra, "Botdefender: A collaborative defense framework against botnet attacks using network traffic analysis and machine learning," *Arabian Journal for Science and Engineering*, vol. 49, no. 3, pp. 3313–3329, 2024. doi: 10.1007/s13369-023-08016-z

[6] Y. Zhou, X. Song, Y. Zhang, F. Liu, C. Zhu, and L. Liu, "Feature encoding with autoencoders for weakly supervised anomaly detection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 6, pp. 2454–2465, 2021. doi: 10.1109/TNNLS.2021.3086137
[7] T. Shenkar and L. Wolf, "Anomaly detection for tabular data with internal contrastive learning," in *International Conference on Learning Representations*, 2021.
[8] N. H. Noi, D. Van Hoa, and T. N. Ngoc, "A deep learning approach combining autoencoder with supervised classifiers for IoT anomaly detection," *Journal of Military Science and Technology*, no. CSCE7, pp. 98–110, 2023. doi: 10.54939/1859-1043.j.mst.CSCE7.2023.98-110
[9] M. Nicolau, J. McDermott, and V. L. Cao, "Learning neural representations for network anomaly detection," *IEEE transactions on cybernetics*, vol. 49, no. 8, pp. 3074–3087, 2018. doi: 10.1109/TCYB.2018.2838668
[10] H. N. Nguyen, N. N. Tran, T. H. Hoang, and V. L. Cao, "Denoising Latent Representation with SOMs for Unsupervised IoT Malware Detection," *SN Computer Science*, vol. 3, no. 6, pp. 1–15, 2022. doi: 10.1007/s42979-022-01344-1
[11] H. Xu, G. Pang, Y. Wang, and Y. Wang, "Deep isolation forest for anomaly detection," *IEEE Transactions on Knowledge and Data Engineering*, 2023. doi: 10.1109/TKDE.2023.3270293
[12] G. Pang, C. Shen, and A. van den Hengel, "Deep anomaly detection with deviation networks," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019. doi: 10.1145/3292500.3330871 pp. 353–362.
[13] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," *arXiv preprint arXiv:1906.02694*, 2019. doi: 10.48550/arXiv.1906.02694
[14] G. Pang, C. Shen, H. Jin, and A. van den Hengel, "Deep weakly-supervised anomaly detection," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023. doi: 10.1145/3580305.3599302 pp. 1795–1807.
[15] M. Jiang, C. Hou, A. Zheng, X. Hu, S. Han, H. Huang, X. He, P. S. Yu, and Y. Zhao, "Weakly supervised anomaly detection: A survey," *arXiv preprint arXiv:2302.04549*, 2023. doi: 10.48550/arXiv.2302.04549
[16] N. H. Noi and T. N. Ngoc, "Learning latent representation with limited labels for IoT anomaly detection," *Journal of Science and Technology on Information Security*, vol. 3, no. 20, pp. 14–22, 2023. doi: 10.54654/isj.v3i20.986
[17] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-BaIoT—network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, 2018. doi: 10.1109/MPRV.2018.03367731
[18] M. A. Siddiqui, A. Fern, T. G. Dietterich, R. Wright, A. Theriault, and D. W. Archer, "Feedback-guided anomaly discovery via online optimization," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018. doi: 10.1145/3219819.3220083 pp. 2200–2209.
[19] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *International conference on machine learning*. PMLR, 2018, pp. 4393–4402.
[20] H. Xu, Y. Wang, G. Pang, S. Jian, N. Liu, and Y. Wang, "RoSAS: Deep semi-supervised anomaly detection with contamination-resilient continuous supervision," *Information Processing & Management*, vol. 60, no. 5, p. 103459, 2023. doi: 10.1016/j.ipm.2023.103459
[21] T. Nguyễn, N. H. Hao, D. L. D. Trang, N. Van Tuan, and C. Van Loi, "Robust anomaly detection methods for contamination network data," *Journal of Military Science and Technology*, no. 79, pp. 41–51, 2022. doi: 10.54939/1859-1043.j.mst.79.2022.41-51

Manuscript received 12-05-2024; Accepted 25-06-2024.

Huu Noi Nguyen received the B.Sc. degree in applied mathematics and informatics from Lipetsk State University, Lipetsk, Russia. He is currently studying the Ph.D. program in Computer Science at Le Quy Don Technical University in Vietnam. His current research interests include Machine Learning, Anomaly Detection, IoT and Information Security.
Email: noi.nguyen@lqdtu.edu.vn

Nguyen Ngoc Tran is an Associate Professor and the Head of the Cyber Security Group at Le Quy Don Technical University in Vietnam. He received PhD in System analysis, control, and information processing from Don State Technical University, Russia. His research interests focus on pattern recognition, cyber security, and artificial intelligence.
Email: ngoctn@lqtdu.edu.vn

Van Loi Cao received the B.Sc. and M.Sc. degree in computer science from Le Quy Don Technical University in Vietnam, and the Ph.D degree from University College Dublin, Dublin, Ireland. He is currently the Head of the Information Security Department at the Institute of Information Technology and Communication, Le Quy Don Technical Unversity. His current research interests include Deep Learning, Machine Learning, Anomaly Detection, IoT Security, and Information Security.
Email: loi.cao@lqdtu.edu.vn

HỌC ĐẶC TRƯNG BÁN GIÁM SÁT DỰA TRÊN KỸ THUẬT NÉN ĐỂ TĂNG CƯỜNG PHÁT HIỆN BẤT THƯỜNG MẠNG IoT

Nguyễn Hữu Nội, Trần Nguyên Ngọc, Cao Văn Lợi

Tóm tắt

Phát hiện bất thường trong mạng IoT đang phải đối mặt với những thách thức do có nhiều khó khăn trong việc thu thập và gắn nhãn cho dữ liệu bất thường nói chung cũng như dữ liệu tấn công nói riêng. Các phương pháp gần đây dựa vào giám sát yếu, như FeaWAD và iFWAD, đã giải quyết vấn đề khó khăn này bằng cách xây dựng các bộ phát hiện từ sự kết hợp giữa dữ liệu không có nhãn và một số ít dữ liệu bất thường được gán nhãn. Tuy nhiên, những phương pháp này thiếu ràng buộc trong giai đoạn học đặc trưng để phân tách dữ liệu bình thường và bất thường. Mã hóa tự động dựa trên kỹ thuật nén (Shrink Autoencoder) có khả năng phân tách các lớp dữ liệu này bằng cách nén dữ liệu bình thường về xung quanh gốc tọa độ, và dành phần không gian còn lại cho bất thường có thể xuất hiện trong tương lai. Lấy cảm hứng từ Shrink Autoencoder, mục tiêu nghiên cứu này giới thiệu Shrink iFWAD (gọi là sFWAD), nhúng một bộ điều chỉnh giúp nén dữ liệu vào mô hình iFWAD. Thành phần shrink giúp bộ mã hóa đặc trưng của sFWAD học cách phạt dữ liệu bình thường gần giá trị không, đồng thời kéo các dữ liệu bất thường của IoT ra xa khỏi giá trị không. Quá trình này giúp thành phần sinh điểm bất thường của sFWAD nhận dạng hiệu quả các dữ liệu bất thường của IoT. Phương pháp đề xuất này được đánh giá so với các kỹ thuật giám sát yếu hàng đầu và các phương pháp phát hiện bất thường thông thường khác sử dụng tập dữ liệu N-BaIoT. Kết quả thực nghiệm cho thấy phương pháp này thường cho kết quả tốt hơn các phương pháp học giám sát yếu gần đây cũng như các phương pháp thông thường theo hiệu suất phát hiện bất thường mạng IoT. Trong phát hiện bất thường chưa biết trước/mới, tỉ lệ phát hiện sai của sFWAD (0.008) thấp hơn đáng kể so với các phương pháp iFWAD (0.026) và RoSAS (0.015).

Từ khóa

Giám sát yếu, biểu diễn ẩn, phát hiện bất thường IoT, phát hiện IoT botnet.