sensors

MDPI

Article

Contrastive Learning with Prototype-Based Negative Mixing for Satellite Telemetry Anomaly Detection

Guohang Guo [1,2], Tai Hu [1,*], Taichun Zhou [1,2], Hu Li [1] and Yurong Liu [1]

[1] National Space Science Center, Chinese Academy of Sciences, Beijing 101499, China
[2] Department of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing 101408, China
* Correspondence: hutai@nssc.ac.cn

Abstract: Telemetry data are the most important basis for ground operators to assess the status of satellites in orbit, and telemetry data-based anomaly detection has become a key tool to improve the reliability and safety of spacecrafts. Recent research on anomaly detection focuses on constructing a normal profile of telemetry data using deep learning methods. However, these methods cannot effectively capture the complex correlations between the various dimensions of telemetry data, and thus cannot accurately model the normal profile of telemetry data, resulting in poor anomaly detection performance. This paper presents CLPNM-AD, contrastive learning with prototype-based negative mixing for correlation anomaly detection. The CLPNM-AD framework first employs an augmentation process with random feature corruption to generate augmented samples. Following that, a consistency strategy is employed to capture the prototype of samples, and then prototype-based negative mixing contrastive learning is used to build a normal profile. Finally, a prototype-based anomaly score function is proposed for anomaly decision-making. Experimental results on public datasets and datasets from the actual scientific satellite mission show that CLPNM-AD outperforms the baseline methods, achieves up to 11.5% improvement based on the standard F_1 score and is more robust against noise.

Keywords: telemetry data; anomaly detection; contrastive learning; negative mixing

Citation: Guo, G.; Hu, T.; Zhou, T.; Li, H.; Liu, Y. Contrastive Learning with Prototype-Based Negative Mixing for Satellite Telemetry Anomaly Detection. *Sensors* **2023**, *23*, 4723. https://doi.org/10.3390/s23104723

Academic Editors: Chunxu Qu, Jinhe Gao, Rui Zhang, Ziguang Jia and Jiaxiang Li

Received: 7 April 2023
Revised: 8 May 2023
Accepted: 10 May 2023
Published: 13 May 2023

1. Introduction

Satellites are currently one of the most sophisticated technological systems, performing their mission in harsh conditions [1]. It is unlikely to be able to entirely dismiss the possibility of faults due to factors such as inadequate design verification, extreme space environment, damage accumulation effect and dynamic change of on-board state [2,3], and serious anomalies will lead to mission degradation or even failure. Telemetry data reflect the health condition of the corresponding device and are the most important basis for ground operators to determine the status of the satellite in orbit [4]. As a result, techniques for mining telemetry data and detecting anomalies have been developed to lessen the monitoring burden on ground operators while also improving the reliability and safety of satellites in orbit.

Anomaly detection is a technique for identifying unexpected patterns of deviation from normal behavior and has attracted a significant amount of research attention in recent years [5–7]. Satellites are usually monitored by numerous sensors, each of which measures a distinct variable. Monitoring the devices or subsystems with correlations between multiple monitoring indicators is critical to ensuring the normal operation of the satellite [8,9]. In satellite anomaly detection settings, anomaly data are relatively rare and other new types of anomalies are continually being discovered over time, owing to changing environmental factors and command sequences, so a supervised approach is not applicable in this case. In contrast, normal data are easily obtained, so we concentrate on identifying correlation

anomalies using a semi-supervised approach, which means that all of the training data are normal data.

Semi-supervised anomaly detection methods use a significant amount of normal data to create a normal profile of correlation between variables. Samples with behaviors deviate from the normal pattern are labeled as anomalies. Recently, deep anomaly detection methods have received a significant amount of attention, as they primarily employ well-designed encoders to capture complex correlations between variables, such as the autoencoder-based method [10–13], generative adversarial networks-based method [14–16], self-supervised anomaly detection method [17,18], and so on. The reconstruction error or sample features obtained from the learned encoder are then used to detect anomalies. However, due to the complex and frequently dynamically changing patterns of correlations between the variables of the telemetry data, the encoder maps normal and abnormal samples into the feature space, making them overlapping and difficult to distinguish, failing to accurately construct normal profiles of normal data, and resulting in unsatisfactory abnormality detection results.

In this paper, we propose an anomaly detection method called contrastive learning with prototype-based negative mixing for anomaly detection (CLPNM-AD). Contrastive learning aims to learn a transformation-invariant feature representation and evenly distributes the sample on the hypersphere in representation space [19,20]. Combining encoders with contrastive loss to model the correlations of normal data, we find that the learned representation can distinguish between normal and abnormal patterns. However, with hardness-aware property [21], the contrastive loss imposes larger gradient magnitudes on samples closer to the anchor, leading the semantically similar samples (false negative samples) to separate, disrupting the local semantic structure between samples. Therefore, we devise a prototype-based hard negative mixing strategy to guide network learning to ensure that the local semantic structure between samples is not disrupted, thereby constructing a more distinguishing profile for normal data.

Going beyond previous anomaly detection methods, CLPNM-AD can capture complex correlations between telemetry data variables, maintaining a local semantic structure while pushing samples with different semantics far apart, making normal and anomaly samples as separable as possible. Specifically, we first employ an augmentation process with random feature corruption to obtain augmented samples. Following that, a prototype consistency strategy is used to capture the semantic category information of the samples. After that, a prototype-based negative mixing contrast loss is used to obtain a normal profile. Unlike the vanilla contrast loss, which maps all samples evenly onto the hypersphere, the proposed method will make the true negative samples from all directions around the anchor point (excluding the false negative samples) have an effective gradient such that these samples are separable from the anchor, making normal and anomaly samples distinguishable. Finally, anomaly decisions are made using a prototype-based anomaly score function. Extensive experiments are conducted to verify the effectiveness of the proposed method.

The contributions of this paper are summarized below:

- We propose CLPNM-AD, a correlation anomaly detection method, which combines prototype consistency and prototype-based contrastive loss to guide the encoder to construct an accurate normal profile, making normal and anomaly samples more distinguishable and thus facilitating the detection of anomalies.
- We apply a sample augmentation process with random feature corruption to generate positive samples for contrast learning and to help capture the complex correlations between variables.
- We combine prototype consistency and prototype-based contrast loss to learn features that facilitate the distinguishing of normal and anomalous samples. First, prototype consistency is proposed to capture the semantic categories of samples. Then, a prototype-based hard negative mixing strategy is applied to preserve local semantic information and push samples of different prototypes farther apart.

- We propose an anomaly score function that indicates the degree of the anomaly of a sample by calculating the Mahalanobis distance between the sample and the prototype conditional Gaussian distribution.
- We conduct extensive experiments to evaluate the performance of CLPNM-AD on three public datasets and one satellite telemetry dataset from our actual mission. The experiment results show the excellent performance of the proposed method.

The remainder of the paper is organized as follows: Section 2 describes related work. Section 3 introduces the anomaly categories as well as the fundamental concepts of contrastive learning. Section 4 describes the proposed method CLPNM-AD in detail. Section 5 describes the experiment findings and analysis. Section 6 provides a summary of the paper.

2. Related Work

2.1. Anomaly Detection

Anomaly detection of satellite telemetry data has recently become a popular research topic, with a large body of research literature emerging. Traditional anomaly detection methods have mainly used statistical or machine learning methods to detect anomalies. In [22], a kernel density estimates-based model was proposed, and samples with low density were labeled as anomalies. In [23], the support vector machine was used to build a hyperplane to keep normal data away from the origin where samples close to the origin were detected as anomalies. In [24], Lishuai Li et al. proposed a clustering method based on a Gaussian mixture model to detect unusual data patterns in flight data, where samples with low probability were detected as anomalous. In [25], the authors labeled samples from low-density regions as anomalous. Although these traditional anomaly detection methods are efficient in most cases, they do not work well when the data dimension is too high and the relationships between variables are too complex.

Self-supervised representation learning methods use the data themselves to construct supervised information to achieve data representation learning. In recent years, self-supervised deep anomaly detection methods have made significant progress, improving anomaly detection performance and gaining much attention. In [26], an encoder was used to map the normal samples into a hypersphere to model the representation of the normal pattern. In [11], the authors used an auto-encoder to learn the features of the samples, which were subsequently combined with reconstruction errors to be fed into a Gaussian mixture model to generate anomaly scores. In [17], the data samples were first augmented with random affine transformation, followed by a classification model for anomaly detection. In [27], the authors used contrastive loss to obtain an encoder and then fed the features generated by the encoder into a one-class classifier for anomaly detection. In [28], the authors first created distributionally shifted augmentations, then trained the encoder with contrastive loss to obtain sample features, and finally detected anomalies based on the cosine distance between the learned features. In [18], trainable augmentation processes were used to obtain augmented samples, and then an end-to-end anomaly detection pipeline was built by contrastive loss.

2.2. Contrastive Learning

Contrastive learning as a type of self-supervised method aims to learn a transformation-invariant feature representation such that multiple views (also known as augmentations) of a sample keep attracting while repelling to other samples [20]. In [29], the authors proposed a feature learning method using contrastive predictive coding, which uses an autoregressive model to predict future values in the hidden space, and an InfoNCE loss function for model training. Ref. [30] proposed an unsupervised feature learning method that constructed an instance discrimination task via a memory bank and learned sample features using a noise contrastive estimation loss function. In [31], the authors further considered the negative samples selection strategy and constructed a contrast task by obtaining negative samples from a dynamic dictionary instead of a memory bank. In [19], the authors discussed the role of sample augmentation and added a non-linear layer in the middle

of the representations and contrastive loss to improve the quality of the learned features. In [32], the authors used prototype vectors as the subject of contrasting rather than the instances themselves, learned features through classification consistency between views, and explored augmentation methods to improve the quality of the learned representation.

2.3. Debiased Negatives Sampling

Contrastive learning learns features by comparing similar or dissimilar pairs of samples, so the quality of positive and negative samples determines the quality of the learned features. Contrastive learning is a self-supervised representation learning method that does not have access to sample labels. When anchor samples and negative samples form negative pairs, these samples may belong to the same semantic class, and the hardness-aware property of contrastive learning will cause these samples to be repelled, destroying the local semantic structure and affecting representation learning. These false negative samples continue to be a major issue in contrastive learning, but relatively little research work has been conducted in this area.

In [33], the authors proposed a method for synthesizing hard negative samples in the feature space by selecting the K closest negative samples to the anchor and randomly mixing them. Ref. [34] established the distribution of difficult negative samples and proposed an importance sampling strategy, based on which the authors designed a new contrastive loss with adjustable hyperparameters to allow users to control the hardness. To lessen the impact of false negatives, ref. [35] offered a debiased contrastive loss, and [36] proposed a negative elimination and false attraction loss. In [37], the authors proposed a modified contrastive loss that eliminates false negative samples and removes negative samples that are similar to the anchors. To address the sampling bias problem, ref. [38] obtained the semantic structure of graph data by clustering and then weighted the negative samples according to the distance between semantic categories.

3. Preliminaries

3.1. Categories of Anomalies in Telemetry Data

Anomalies in satellite telemetry data are classified as point anomalies, contextual anomalies, collective anomalies and correlation anomalies, as illustrated schematically in Figure 1.

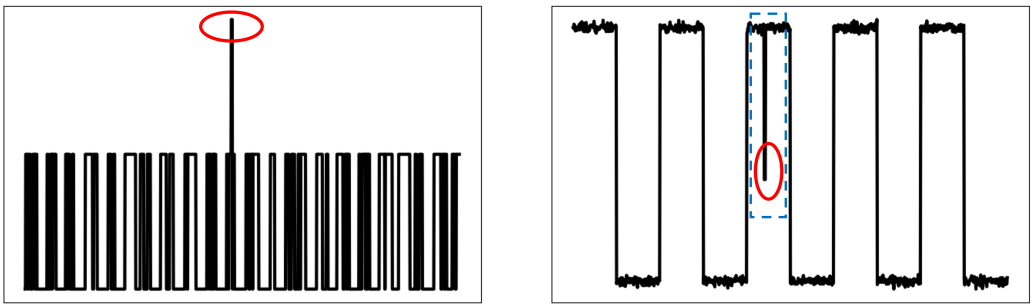
Point anomalies refer to a single data point or several consecutive data points that deviate significantly from other data points. Figure 1a depicts an example of a point anomaly in which the point in the red circle extends beyond the other points.

Contextual anomalies mean that the data points deviate a local threshold determined by the temporal context, although not deviating the global threshold. Figure 1b shows an example of a contextual anomaly where the point in the red circle deviates from the other samples in the time window identified by the blue box.

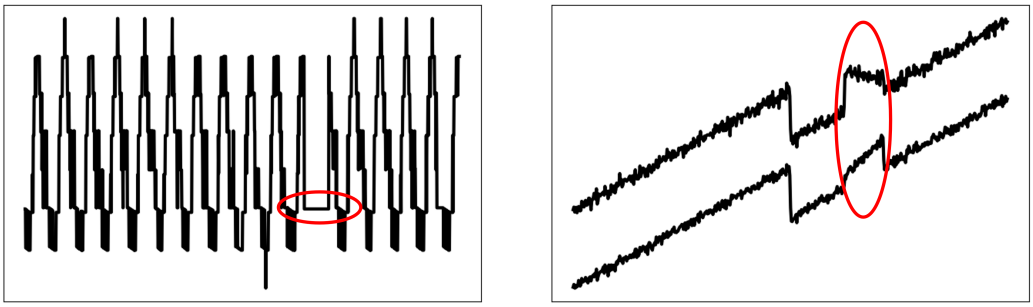
Collective anomalies are consecutive points that do not exceed a threshold, but the subsequence formed by these points violates the pattern of the original telemetry time series. As shown in Figure 1c, the sequence of consecutive points in the red circle is a collective anomaly, which has a different shape to the rest of the sequence.

Correlation anomalies occur when the stable correlation between multiple telemetry parameters is broken and the correlation between parameters exhibit different patterns. Correlations between satellite telemetry parameters include physical correlations, logical correlations and operational mode correlations. As shown in Figure 1d, there is a positive correlation between the two parameters in most cases, but in the red circle, the correlation becomes negative, indicating a correlation anomaly.

The correlations between satellite telemetry parameters in real-world scenarios can be very complex, necessitating anomaly detection methods capable of modeling such complex correlations. The objective of this paper is primarily to propose a correlation anomaly detection method that is sensitive to abnormal changes in the correlation between satellite telemetry parameters.



(**a**) Point anomaly. (**b**) Contextual anomaly.



(**c**) Collective anomaly. (**d**) Correlation anomaly.

Figure 1. Examples of anomalies in satellite telemetry data (anomalies highlighted in red circles and time window in the blue box).

3.2. Contrastive Learning

Contrastive learning learns representations by grouping differently augmented views of the same data sample together [19]. During the training process, a mini-batch with size N is randomly selected, and the contrastive loss is defined on augmented views generated from the mini-batch. Let $x_i^{(1)} = \mathcal{A}^{(1)}(x_i)$ and $x_i^{(2)} = \mathcal{A}^{(2)}(x_i)$, where $\mathcal{A}^{(1)}$ and $\mathcal{A}^{(2)}$ are independent stochastic augmentation functions. Following [19,27], we define contrastive loss as follows:

$$\mathcal{L}_{infonce}(x_i^{(1)}, x_i^{(2)}) = \\ -\frac{1}{N}\sum_{i=1}^{N}\log\frac{exp(dist(z_i^{(1)}, z_i^{(2)})/\nu)}{\sum_{j=1}^{N}\mathbb{1}_{[j\neq i]}exp(dist(z_i^{(1)}, z_j^{(1)})/\nu) + \sum_{j=1}^{N}exp(dist(z_i^{(1)}, z_j^{(2)})/\nu)}, \quad (1)$$

where $dist(u,v) = \frac{u^T v}{||u||||v||}$, $z_i^{(1)} = h \circ f(x_i^{(1)})$, $z_i^{(2)} = h \circ f(x_i^{(2)})$, $f(\cdot)$ is an encoder, $h(\cdot)$ is a neural network layer that transforms feature vectors to the space where contrastive loss is used, $\mathbb{1}_{[j\neq i]}$ represents an indicator function that outputs 1 when $j \neq i$ holds and 0 otherwise and ν is a temperature parameter.

4. Proposed Method

We first present the problem definition of satellite telemetry data correlation anomaly detection. Then, we give an overview of CLPNM-AD and briefly introduce its modules. Finally, we describe the proposed method in detail.

4.1. Problem Definition

This paper is based on the assumption that, in routine operation, correlations between satellite telemetry variables exhibit some common patterns, and that ground operators should focus on the few outliers that deviate from these common patterns.

Each sample of satellite telemetry data from a subsystem or device at time t is represented by a vector, as in the form

$$x_t = [d_t^1, d_t^2, \ldots, d_t^M], \tag{2}$$

where d_t^j is the value of the j-th dimension of telemetry data at time t and M is the dimension of sample x_t.

The semi-supervised scenario is addressed in this paper, in which the training dataset $\mathcal{X}_{train} = \{x_1, x_2, \ldots, x_N\}$ contains only normal data samples and the test dataset $\mathcal{X}_{test}$ contains both normal and abnormal data samples. The goal of the anomaly detection method is first to create an anomaly scoring model $s(\cdot)$ from $\mathcal{X}_{train}$:

$$s : x \to b, \tag{3}$$

where b is the anomaly score that predicts how anomalous the sample x is, and then to create an anomaly decision function $\Delta(\cdot)$ to emit a judgment of whether x is normal or abnormal:

$$\Delta : b \to \{0, 1\}, \tag{4}$$

where 0 denotes normal, 1 denotes anomalous and the anomaly decision function $\Delta(\cdot)$ is often implemented by a conditional judgment, which means that the sample x is determined to be anomalous if its anomaly score, $s(x)$, exceeds a specified threshold.

4.2. Overview of CLPNM-AD Framework

The CLPNM-AD framework is shown in Figure 2, which contains a data augmentation module, representation learning module and anomaly scoring module.

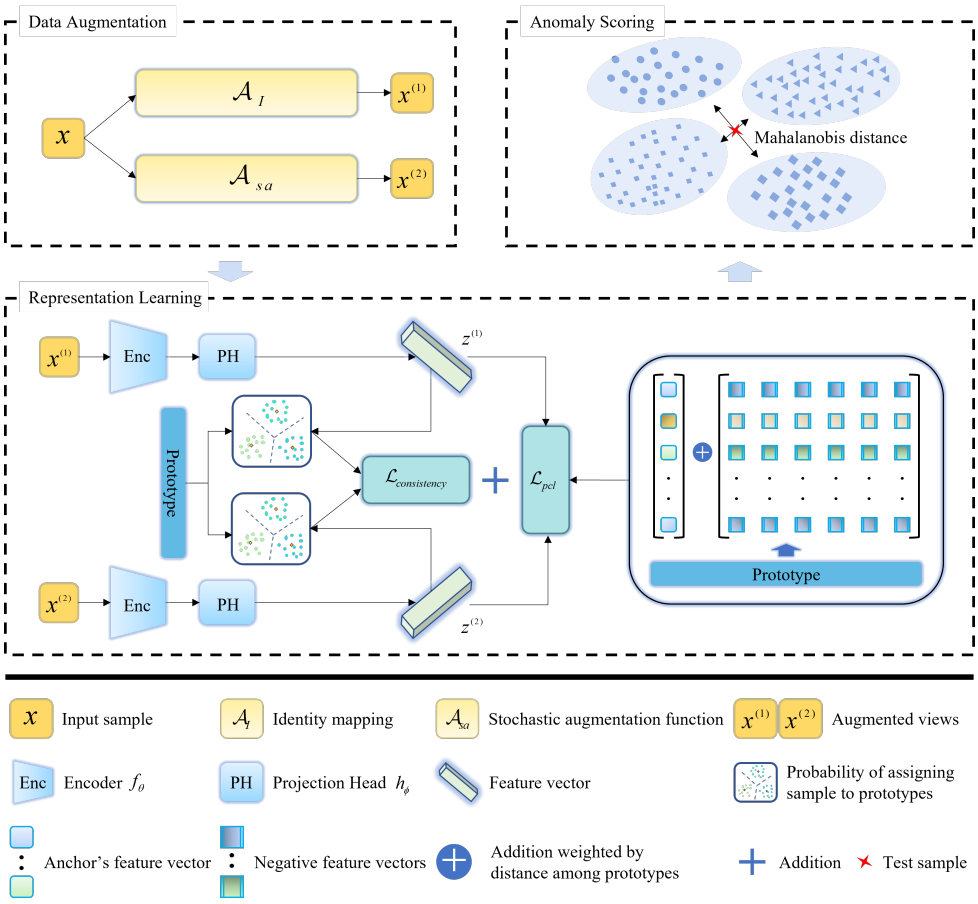


Figure 2. The framework of CLPNM-AD.

First, we use the augmentation functions $\mathcal{A}_I$ and $\mathcal{A}_{sa}$ to generate two augmented samples of sample x. We then propose a contrastive learning with the prototype-based negative mixing (CLPNM) method to learn the representation of the training data $\mathcal{X}_{train}$. As depicted in the representation learning module in Figure 2, the encoder and projection head map the augmented samples to the feature space, then the probability values of the feature vectors assigned to the prototypes are obtained to learn the semantic consistency of the augmented samples and finally the hard negative samples are generated to guide the contrastive representation learning. In this way, a representation specific to anomaly detection is obtained. Finally, the anomaly score is calculated for each sample based on the Mahalanobis distance between the sample feature and the closest prototype.

4.3. Data Augmentation

As a vital part of contrastive learning, techniques to generate views are domain-specific (e.g., color distortion [19] and geometric transformation [32] in computer vision). Unlike image data, telemetry data cannot be augmented using color and geometric transformations. In this paper, two augmentation strategies, $\mathcal{A}_I$ and $\mathcal{A}_{sa}$, are used as augmentation functions, where $\mathcal{A}_I$ is a identity mapping and $\mathcal{A}_{sa}$ is a stochastic process with random feature corruption.

Inspired by the literature [39], the stochastic augmentation generation method is shown in Algorithm 1. In this algorithm, we denote a process $\mathcal{A}_{sa}$ to obtain the augmentation view of a sample x. It first computes the empirical marginal distribution for each feature's values throughout the whole training dataset, and next randomly selects a subset of features and draws a random value from the empirical marginal distribution of each feature in that subset to replace the value of that feature in the sample.

Algorithm 1: Stochastic augmentation function $\mathcal{A}_{sa}$

Input: training dataset $\mathcal{X}_{train} \subseteq \mathbb{R}^M$, data sample $x \in \mathcal{X}_{train}$, the number of corruption dimensions k.
Output: the augmented view $\tilde{x}$

let $UD_{(i)}$ be the uniform distribution defined on $\mathcal{X}_{(i)} = \{x_{(i)} : x \in \mathcal{X}_{train}\}$, where $x_{(i)}$ is the value of i-th dimension of the sample x;
let $\mathcal{V}$ be uniformly sampled subset from $\{1, 2, \cdots, M\}$ of size k;
let $\tilde{x} = x$;
for $i \in \{1, 2, \cdots, M\}$ **do**
    if $i \in \mathcal{V}$ **then**
        $\tilde{x}_{(i)} = v$, where $v \sim UD_{(i)}$;
    end
end

4.4. Representation Learning

In the CLPNM-AD method, a one-dimensional convolution encoder maps the samples into the feature space. The prototype and pseudo-label of the sample are then calculated by self-labeling based on the features. Finally, hard negatives are synthesized based on the prototypes and pseudo-labels to guide the network parameters learning. Prototype learning and network parameters updating are performed alternately.

4.4.1. One-Dimensional Convolution Encoder

The main goal of the encoder is to map the samples to the feature space. As an encoder in this paper, a one-dimensional convolutional network is designed, as shown in Figure 3. To begin, a linear layer is used to map the samples to a high-dimensional vector, which is then reshaped to obtain the features of multiple channels, after which one-dimensional convolutional layers are used to perform convolutional operations, and finally the features of multiple channels are flattened to obtain the final feature vector of the samples.

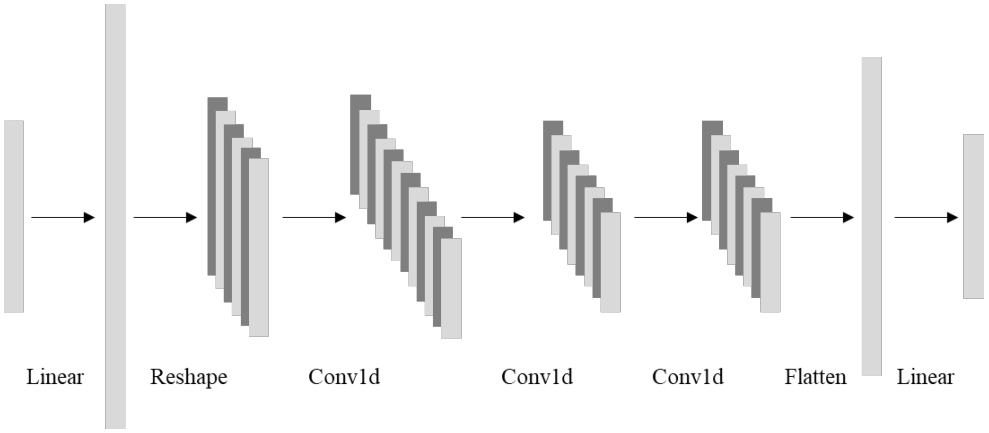


Figure 3. One-dimensional convolutional network structure.

4.4.2. Self-Labeling by Clustering Consistency

As shown in Figure 2, deep neural networks $h \circ f(\cdot)$ map the augmented view $x_i^{(a)}$ to the feature vector $z_i^{(a)} \in \mathbb{R}^D$ by $z_i^{(a)} = h \circ f(x_i^{(a)})$, where $a \in \{1, 2\}$ is the index of a view. After that, the feature vector $z_i^{(a)}$ is mapped to K clusters, the centroids of which are implemented by K trainable prototype vectors $\{c_1, c_2, \ldots, c_K\}$. We define matrix C, whose columns are composed of the vectors $c_1, c_2, \ldots, c_K$, and which is implemented by a single linear neural network. The matrix C converts the feature vector $z_i^{(a)}$ to class scores. The softmax operator is then used to map the class scores to the class probabilities:

$$p(y = \cdot | x_i^{(a)}) = softmax(C \circ h \circ f(x_i^{(a)})). \tag{5}$$

We represent the pseudo-label of the sample $x_i^{(a)}$ by posterior distribution $q(y = \cdot | x_i^{(a)})$. The cross-entropy loss of $p(y = \cdot | x_i^{(a)})$ and $q(y = \cdot | x_i^{(a)})$ defines as

$$E(p(x^{(a)}), q(x^{(a)})) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{y=1}^{K} q(y | x_i^{(a)}) \log p(y | x_i^{(a)}). \tag{6}$$

In the anomaly detection setting, the semantic label of the sample is unknown. Optimizing $E(p(x^{(a)}), q(x^{(a)}))$ will lead q to a degenerate solution, namely assign all samples to a randomly single pseudo-label. To prevent degenerate solutions, we introduce a constraint that the assignment of pseudo-labels must divide the data samples from a mini-batch into clusters with equal size [40]. Formally, the optimization objective is thus

$$\min_{p,q} E(p(x^{(a)}), q(x^{(a)})) \, s.t. \, \forall y : q(y | x_i^{(a)}) \in [0, 1] \, and \, \sum_{i=1}^{N} q(y | x_i^{(a)}) = \frac{N}{K}. \tag{7}$$

The objective of Equation (7) can be viewed as an *optimal transport problem*. To solve this problem, we define P and Q as two $K \times N$ matrices of joint probability. The elements in P and Q denote as

$$P_{yi} = p(y | x_i^{(a)}) \frac{1}{N}; \; Q_{yi} = q(y | x_i^{(a)}) \frac{1}{N}. \tag{8}$$

To meet the equal partition constraint, we make the matrix Q a *transportation polytope* [40]:

$$U = \{Q \in \mathbb{R}_+^{K \times N} \mid Q\mathbf{1}_N = \frac{1}{K}\mathbf{1}_K, \; Q^T\mathbf{1}_K = \frac{1}{N}\mathbf{1}_N\}, \tag{9}$$

where $\mathbf{1}_N$ and $\mathbf{1}_K$ are vectors of all ones of corresponding dimensions.

Then, the optimization objective in Equation (7) can thus be expressed as

$$\min_{Q \in U} \langle Q, -\log P \rangle, \tag{10}$$

where $\langle \cdot \rangle$ denotes the Frobenius dot-product of two matrices. The Sinkhorn–Knopp [41] algorithm is then used to solve the above transport problem, which amounts to introducing a regularization term

$$\min_{Q \in U} \langle Q, -\log P \rangle + \frac{1}{N} KL(Q||rc^T), \tag{11}$$

where KL denotes the Kullback–Leibler divergence, $r = \frac{1}{K}\mathbf{1}_K$, and $c = \frac{1}{N}\mathbf{1}_N$. The advantage of bringing in the regularization term $KL(Q||rc^T)$ is that the minimizer of Equation (11) can be transformed to

$$Q = diag(\Gamma) P^{\lambda} diag(\Lambda), \tag{12}$$

where Γ and Λ are two scaling coefficient vectors and λ is used to balance the convergence speed and proximity level to the original transport problem. In this paper, we specify a fixed value for λ.

The pseudo-label matrix Q is then used to guide the learning of the neural network parameters. To encourage the two augmented views $x_i^{(1)}$ and $x_i^{(2)}$ to be assigned to the same cluster, we assign $x_i^{(1)}$ the pseudo-label $q(y = \cdot|x_i^{(2)})$ (not $q(y = \cdot|x_i^{(1)})$) and vice versa. The consistency loss denotes as

$$\mathcal{L}_{consistency} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{y=1}^{K} [\, q(y|x_i^{(1)}) \log p(y|x_i^{(2)}) + q(y|x_i^{(2)}) \log p(y|x_i^{(1)}) \,], \tag{13}$$

where N denotes the mini-batch size and K denotes the number of prototypes.

4.4.3. Negative Mixing Contrastive Objective

The pseudo-label matrix Q, on one hand, is used to learn clustering consistency, and on the other hand, is used to mitigate the sampling bias and obtain hard negatives to guide the contrastive loss. The soft label is represented by Q in the preceding section, while we require the hard label in this part to reflect the semantic clusters of the data. To acquire the hard labels, we simply apply the *argmax* function to the soft labels.

We obtain hard negatives by mixing the anchor feature vector with sample vectors from different prototypes in the mini-batch. The weight of sample mixing is determined by the distance between prototypes. We denote the distance between two prototypes as

$$d(c_i, c_j) = 1 - \frac{c_i^T c_j}{||c_i||_2 ||c_j||_2}, \tag{14}$$

where c_i and c_j are two prototypes from prototypes C in Figure 2.

For the anchor feature vector z_i and sample feature vector z_j, the mixed sample can be calculated by

$$h_j = \frac{\tilde{h}_j}{||\tilde{h}_j||_2}; \tilde{h}_j = z_i \cdot (1 - d(c_i, c_j)) + z_j \cdot d(c_i, c_j). \tag{15}$$

The prototype-based contrastive loss is defined as

$$\mathcal{L}_{pcl} = -\frac{1}{N} \sum_{i=1}^{N} \log \left\{ \frac{exp(dist(z_i^{(1)}, z_i^{(2)})/\nu)}{\sum_{j=1}^{2N} \mathbb{1}_{c_i \neq c_j} exp(dist(z_i^{(1)}, h_j)/\nu)} + \frac{exp(dist(z_i^{(2)}, z_i^{(1)})/\nu)}{\sum_{j=1}^{2N} \mathbb{1}_{c_i \neq c_j} exp(dist(z_i^{(2)}, h_j)/\nu)} \right\}. \tag{16}$$

The final training objective couples $\mathcal{L}_{consistency}$ and $\mathcal{L}_{pcl}$ as

$$\mathcal{L} = \mathcal{L}_{pcl} + \eta \, \mathcal{L}_{consistency}, \tag{17}$$

where η is used to balance $\mathcal{L}_{pcl}$ and $\mathcal{L}_{consistency}$.

The core of the proposed method is to learn a model $C \circ g \circ f$ and a pseudo-label matrix Q. This is accomplished by alternately performing the following two steps:

Step 1: representation learning. Given the pseudo-label assignments matrix Q, the parameters of model $C \circ g \circ f$ are optimized by minimizing Equation (17).

Step 2: self-labeling. Given the model $C \circ g \circ f$ obtained from step 1, we first calculate the class probability P by using Equation (5), and then calculate Q using Equation (12):

$$\forall y : \Gamma_y = [P^\lambda \Lambda]_y^{-1}; \quad \forall i : \Lambda_i = [\Gamma^T P^\lambda]_i^{-1}. \tag{18}$$

4.5. Score Functions for Detecting Anomaly

To determine whether a sample is normal or anomalous, we use representations and prototypes to assess the degree of anomaly of the sample. Refs. [27,42] show that the projection head focuses too much on the proxy task, and the output features lose useful information for the downstream task. Therefore, we construct the anomaly score function based on the representations of encoder f_θ.

Let x_i be an input and $y_i \in \{1, \cdots, K\}$ be its label, which is obtained by executing *argmax* operation on the pseudo-label matrix Q in Section 4.4.2. Following [43,44], we assume that the feature vectors with the same prototype follow a prototype-conditional multivariate Gaussian distribution. Specifically, we define a conditional Gaussian distribution for each of the K prototypes, and all K conditional Gaussian distributions share a common covariance Σ: $P(f(x_i)|y_i = c) = \mathcal{N}(f(x_i)|\mu_c, \Sigma)$, where μ_c is the mean of feature vectors calculated by $f(\cdot)$ with the prototype $c \in \{1, \cdots, K\}$. To estimate the parameters μ_c and Σ in the conditional Gaussian distribution, we use N data samples from the entire training set for the calculation (not the samples in a batch):

$$\tilde{\mu}_c = \frac{1}{N_c} \sum_{i=1}^{N} \mathbb{1}_{[y_i=c]} f(x_i); \tilde{\Sigma} = \frac{1}{N} \sum_{c=1}^{K} \sum_{i=1}^{N} \mathbb{1}_{[y_i=c]} (f(x_i) - \tilde{\mu}_c)(f(x_i) - \tilde{\mu}_c)^T, \tag{19}$$

where N denotes the number of samples contained in the training set, N_c denotes the number of samples with prototype c and $\mathbb{1}_{[y_i=c]}$ outputs 1 when $y_i = c$ holds, otherwise 0.

Based on the empirical class mean and the covariance of prototype conditional Gaussian distribution obtained above, we define the anomaly score $s(x)$ for sample x by the following equation:

$$s(x) = \min_c (f(x) - \tilde{\mu}_c)^T \tilde{\Sigma}^{-1} (f(x) - \tilde{\mu}_c). \tag{20}$$

The anomaly score indicates the degree of abnormality of each sample. The higher the anomaly score, the more abnormal the sample is. To achieve an anomaly decision for the samples, the protocol in the literature [11,17] is used, in which the decision threshold is based on the percentage of anomalies in the test set, e.g., if the percentage of anomalies in the test set is ρ, the ρ of samples with the highest anomaly scores are considered to be anomalous.

5. Experiments

In this section, we describe the selected datasets and the baseline methods, and perform a series of experiments to validate the proposed method. Firstly, the proposed method is compared with the state-of-the-art methods and validated by statistical hypothesis testing. Then, the robustness of the method is verified on data with different contamination ratios. Furthermore, the effectiveness of each module of the method is verified by ablation experiments. Finally, sensitivity analyses are also implemented to explore the response of the model to different hyperparameters. The code is available at https://github.com/guoguohang/CLPNM_AD, (accessed on 15 March 2023).

5.1. Datasets

We adopt three public datasets: Thyroid, Satellite, Satimage and one actual telemetry data source from the Quantum Science Experiment Satellite, also known as Micius. The statistical information of the datasets is shown in Table 1.

Table 1. Statistical information of the datasets.

Dataset	Dimensions	Instances	Anomaly Ratio (ρ)
Thyroid	6	3772	0.025
Micius	19	11250	0.200
Satellite	36	6435	0.316
Satimage	36	5803	0.012

- **Thyroid**. The dataset is a thyroid disease classification dataset from the outlier detection datasets (OODS) repository (http://odds.cs.stonybrook.edu/thyroid-disease-dataset/, (accessed on 29 March 2023)), which contains six continuous attributes. The original dataset contains three classes. As hyperfunction is a minority, researchers in the field of anomaly detection treat hyperfunction as anomaly class.
- **Micius**. The dataset is from the telemetry data of China's Quantum Science Experiment Satellite, also known as Micius (https://doi.org/10.57760/sciencedb.o00009.00042, (accessed on 29 March 2023)), which contains 19 attributes, and the time span is from January 2017 to February 2019. Micius has four operation patterns. As pattern 4 is rare, we treat pattern 4 as anomaly class and the rest as normal class.
- **Satellite**. The Satellite dataset is integrated by Landsat Satellite from the OODS repository (http://odds.cs.stonybrook.edu/satellite-dataset/, (accessed on 29 March 2023)). Data from three minority categories, 2, 4 and 5, are combined to form the anomaly class, while the remaining classes are combined to form the normal class.
- **Satimage**. The Satimage-2 dataset is from the OODS repository (http://odds.cs.stonybrook.edu/satimage-2-dataset/, (accessed on 29 March 2023)) and is also integrated from Landsat Satellite. Combining the training and test data in the Landsat Satellite dataset, there are 71 outliers in class 2 and all other classes are merged into a normal class.

5.2. Baseline Methods

The following six methods are selected to compare with CLPNM-AD:

- OC-SVM [23]. One-Class Support Vector Machine (OC-SVM) is a classical kernel-based anomaly detection method that uses normal data to learn a decision boundary in order to distinguish normal from anomalous data.
- LOF [25]. Local Outlier Factor (LOF) uses the degree of isolation of a sample relative to its surrounding neighbors as the anomaly score to achieve the distinction between normal and abnormal data samples.
- DAGMM [11]. Deep Autoencoding Gaussian Mixture Model (DAGMM) consists of a compression network and an estimation network. The autoencoder acts as a compression network to map the samples into the feature space. The obtained feature vectors and the reconstruction error of the compression network are fed to the estimation network to obtain the energy score/anomaly score.
- Deep SVDD [26]. Deep Support Vector Data Description (Deep SVDD) is the deep variant of SVDD, which aims to leverage the feature extraction capabilities of deep learning to learn a hypersphere only using normal data.
- GOAD [17]. GOAD is a classification-based method for detecting anomalies for general data, which obtains anomaly scores by training a classifier on a set of random auxiliary tasks.
- NeuTraL AD [18]. Neural Transformation Learning for Deep Anomaly Detection (NeuTraL AD) uses a set of learnable transform to replace the fixed random affine transform

in the previous literature and provides an end-to-end anomaly detection method with loss function in a form similar to the contrastive loss function.

5.3. Evaluation Metrics

Following the settings in [11,17], the models are trained using a randomly selected subset of 50% of the normal data and are evaluated on the remaining normal data as well as all the anomaly data. We use the mean and standard deviation (σ) of F_1 score after 20 random splits to compare the anomaly detection performance. The threshold for identifying anomalous samples is determined by the anomaly ratio ρ in Table 1, which indicates that samples with an anomaly score higher than $100 * \rho$ quantiles will be identified as anomaly. We consider the anomaly class to be positive class and define the F_1 score accordingly. F_1 score is defined as follows: $F_1 = \frac{2*Precision*Recall}{Precision+Recall}$, where Precision=$\frac{|G \cap R|}{|R|}$, Recall=$\frac{|G \cap R|}{|G|}$, G denotes the set of real anomalous samples and R denotes the set of anomalous samples identified by our method.

5.4. Model Configuration

This section describes the network structure as well as the details of model training. In our experiments, all the CLPNM-AD instances are implemented based on the Pytorch framework. For each dataset, the network structure of the encoder, projection head and prototype in CLPNM-AD are shown in Table 2. For the other hyperparameters, the corruption dimension k in the stochastic augmentation process $\mathcal{A}_{sa}$ is set to 4, 16, 34 and 32 for the datasets Thyroid, Micius, Satellite and Satimage, respectively, the temperature ν in Equation (16) is set to 0.07 and the balance coefficient η in Equation (17) is set to 0.4 for each dataset. All model instances are trained by using mini-batch SGD with a momentum of 0.9. The initial learning rate LR_0 is set as 0.001 and updated by $LR_t = LR_{t-1} * 0.5 * (1 + cosine(\pi + t/N_D))$ where N_D is the number of the total epochs.

Table 2. Network structure of CLPNM-AD for each dataset.

	Operation	Units				Activation Function
		Thyroid	Micius	Satellite	Satimage	
Encoder	Linear	(6, 128)	(19, 128)	(36, 128)	(36, 128)	LeakyReLU(0.2)
	Reshape	(*, 16, 8)	(*, 16, 8)	(*, 16, 8)	(*, 16, 8)	
	Conv1d	(16, 32, 1)	(16, 32, 1)	(16, 32, 1)	(16, 32, 1)	
	BatchNorm1d	(32)	(32)	(32)	(32)	LeakyReLU(0.2)
	Conv1d	(32, 6, 1)	(32, 32, 1)	(32, 36, 1)	(32, 256, 1)	
	BatchNorm1d	(6)	(32)	(36)	(256)	LeakyReLU(0.2)
	Conv1d	(6, 6, 1)	(32, 32, 1)	(36, 36, 1)	(256, 256, 1)	
	BatchNorm1d	(6)	(32)	(36)	(256)	
	Flatten					LeakyReLU(0.2)
	Linear	(48, 6)	(256, 32)	(288, 36)	(2048, 256)	LeakyReLU(0.2)
Projection Head	Linear	(6, 6)	(32, 32)	(36, 36)	(256, 256)	
Prototypes (C)	Linear	(6, 4)	(32, 8)	(36, 4)	(256, 12)	

The symbol * in this table denotes a positive integer determined by the number of samples in the batch.

5.5. Effectiveness Evaluation

On four datasets, we examine the effectiveness of CLPNM-AD compared with that of six baseline methods. We use bold font to indicate the best F_1 score and underlined font to indicate the second-best F_1 score.

Table 3 shows the average F_1 scores and standard deviations of CLPNM-AD as well as the baseline methods. The F_1 score of CLPNM-AD surpasses all baseline methods on all datasets. Specifically, CLPNM-AD performs significantly better than the baseline methods on Thyroid and Micius, which achieves 11.5% and 11.0% improvement compared to the sub-optimal approaches DAGMM and LOF. For Satellite and Satimage datasets, CLPNM-AD still surpasses the next best method DAGMM and GOAD by 3.5% and 0.8%, respectively.

Table 3. Mean F_1 and σ of CLPNM-AD and all baseline methods(%). The highest F_1 scores are shown in bold, and the next highest F_1 scores are underlined.

Method	Dataset			
	Thyroid $F_1 \pm \sigma$	Micius $F_1 \pm \sigma$	Satellite $F_1 \pm \sigma$	Satimage $F_1 \pm \sigma$
OC-SVM [23]	57.6 ± 4.5	67.0 ± 0.7	75.0 ± 0.4	77.7 ± 3.1
LOF [25]	60.4 ± 3.3	<u>75.4</u> ± 1.4	75.0 ± 0.7	84.6 ± 2.6
DAGMM [11]	<u>74.1</u> ± 8.9	56.4 ± 7.2	<u>75.1</u> ± 3.6	86.6 ± 3.1
DSVDD [26]	67.9 ± 6.0	55.5 ± 0.7	74.7 ± 0.4	89.2 ± 2.3
GOAD [17]	74.0 ± 3.7	70.6 ± 6.0	72.7 ± 1.8	<u>91.9</u> ± 2.2
NeuTraL AD [18]	70.4 ± 2.6	69.1 ± 5.9	74.4 ± 0.3	85.3 ± 4.9
CLPNM-AD (Ours)	**82.6** ± 3.0	**83.7** ± 1.1	**77.7** ± 0.5	**92.7** ± 1.3

OC-SVM maps samples to one side of the hyperplane using a kernel function and does not take into account the semantic information shared by different samples. Simply mapping these samples to one side of the hyperplane tends to result in the overlapping of samples and thus makes it difficult to distinguish between normal and abnormal samples. Therefore, the results in Table 3 show that OC-SVM performs poorly among all methods.

LOF uses the distance between a sample and its surrounding points for anomaly detection, which takes into account the local similarity of the samples and to some extent the semantic similarity information of the samples. As a result, LOF outperforms OC-SVM, particularly on the Micius dataset, where OC-SVM performs suboptimally. However, because LOF is based on the original samples and lacks feature extraction capability, it cannot effectively model the data's characteristics.

DAGMM uses the reconstruction error and intermediate layer vectors as feature vectors, followed by a Gaussian mixture model for anomaly detection, achieving good performance on most datasets because the mixture component of the Gaussian mixture model models the semantic category information of the samples. However, DAGMM performs poorly on the Micius dataset because the distinction between normal and abnormal samples was not very clear, and it cannot map different semantic category samples apart, resulting in poor performance.

DSVDD achieves anomaly detection by mapping samples into a hypersphere, but it lacks the ability to capture the semantic information of the samples and maps all samples into the same hypersphere, which results in samples with different semantic information overlapping in the feature space and does not facilitate the distinguishing of normal and abnormal samples. As can be seen from Table 3, DSVDD achieves poor performance on the Micius dataset, as does DAGMM.

GOAD augments the samples with a random affine transformation and then maps the augmented samples to the feature space for anomaly detection by predicting the transform used. The affine transform is used as the semantic class in this approach, and the augmented samples obtained using the same affine transform are clustered into one class in the feature space. Such an approach lacks sample semantic information modeling and achieves good results when the anomalous samples are of a single semantic category and the anomaly rate is low, e.g., as shown in Table 3, the model performs well on the Satimage dataset but performs poorly on the Satellite dataset where the anomalous samples contain more semantic categories and the anomaly rate is high.

In comparison to GOAD, NeuTraL AD employs a trainable multilayer neural network as the data augmentation function and a loss function similar to contrast loss to separate samples with different transformations for anomaly detection. The method suffers from the same limitation of a lack of sample semantic information modeling, and thus its performance is inadequate, as shown in Table 3.

Overall, CLPNM-AD outperforms all baselines on four datasets. The one-dimensional convolutional network in CLPNM-AD helps to capture the features of the samples, which can be adapted to anomaly detection tasks on different datasets. In addition, the consistency loss in CLPNM-AD helps the network to capture the semantic information of the samples,

while the hard negative mixing strategy provides stronger guidance to help the contrast loss to separate and disperse the semantically different samples onto the hypersphere as much as possible, which allows the model to accurately profile the normal samples and thus make the normal and abnormal samples more distinguishable.

To quantitatively assess the effectiveness of CLPNM-AD, we statistically evaluate the results of 20 runs of CLPNM-AD with six baseline methods on four datasets using the Wilcoxon rank sum test. We test which of the null hypothesis H0 or the alternative hypothesis H1 holds for each pair of methods: H0: A $\approx$ B, H1: A > B, where A represents the F_1 score of CLPNM-AD on a specific dataset and B represents the F_1 score of the baseline method on the same dataset, A $\approx$ B means that the results of the two methods compared are not significantly different and A > B means that the results of A are better than those of B. For each test, we compute the p-value, and the hypothesis is tested at a significance level of $\alpha_s = 0.01$.

As shown in Table 4, except for GOAD on Satimage, all Wilcoxon test results meet $p < \alpha_s$, allowing us to reject the null hypothesis H0 and accept the alternative hypothesis H1. This means that CLPNM-AD outperforms the baseline methods in all cases except for the Satimage dataset, where CLPNM-AD has nearly the same excellent results as GOAD.

In general, on all datasets, the CLPNM-AD method outperforms all baseline methods, and the advantage is statistically significant.

Table 4. p-values of Wilcoxon rank sum test for F_1.

Dataset	Method					
	OC-SVM	**LOF**	**DAGMM**	**DSVDD**	**GOAD**	**NeuTraL AD**
Thyroid	3.26×10^{-8}	3.25×10^{-8}	1.69×10^{-6}	3.79×10^{-8}	3.71×10^{-7}	3.52×10^{-8}
Micius	3.39×10^{-8}	3.38×10^{-8}	3.39×10^{-8}	3.39×10^{-8}	4.57×10^{-8}	3.39×10^{-8}
Satellite	3.36×10^{-8}	3.37×10^{-8}	6.89×10^{-3}	3.36×10^{-8}	3.37×10^{-8}	3.34×10^{-8}
Satimage	2.60×10^{-8}	2.60×10^{-8}	3.52×10^{-8}	1.85×10^{-8}	1.33×10^{-1}	8.20×10^{-8}

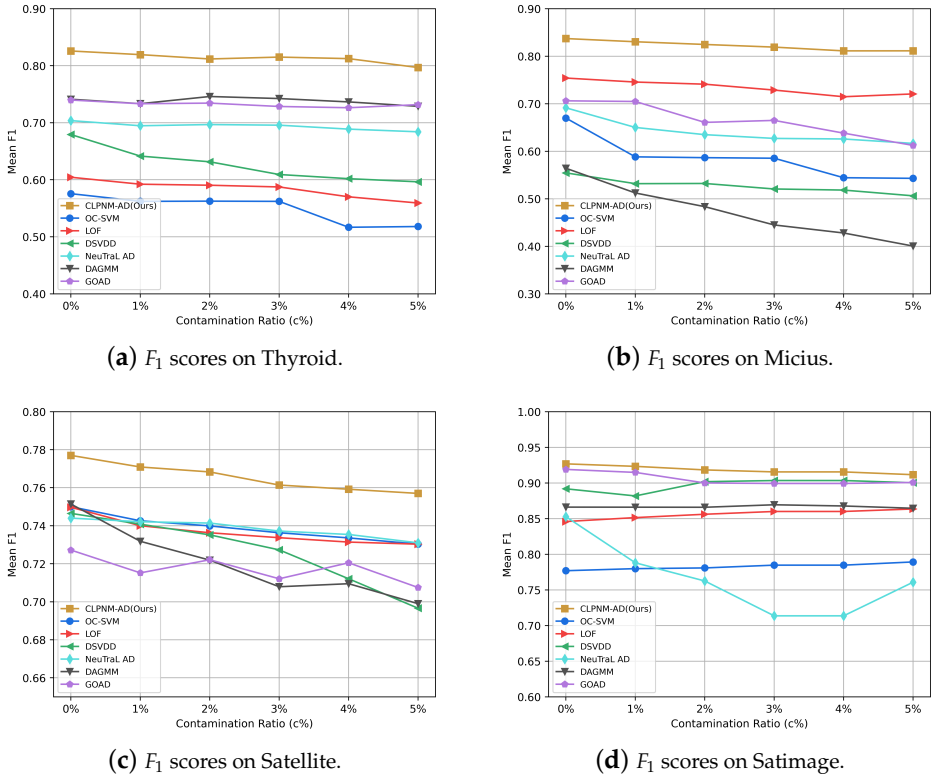
5.6. Robustness Evaluation

In satellite telemetry anomaly detection applications, the training data are frequently mixed with noisy data, or the anomalous samples are incorrectly labeled as normal during the training dataset construction, so we need to investigate the model's sensitivity to contamination. In this experiment, we investigate how CLPNM-AD and all baselines respond to contaminated training data. We adopt the experiment setup in the literature [11] according to which, in each run, all of the anomaly data are combined with 50% of the randomly selected normal data to form the test set, and the remaining 50% of the normal data are mixed with c% of the anomaly data for model training. We examine the model's average F_1 value when the contamination ratio is $c\% = [1\%, 2\%, 3\%, 4\%, 5\%]$.

Figure 4 illustrates the F_1 score of CLPNM-AD and all baselines after 20 runs. As expected, contaminated training data have a detrimental impact on detection accuracy in the majority of the situations. The mean F_1 score of CLPNM-AD declines somewhat across all datasets when the contamination ratio c% grows from 1% to 5%. When the contamination ratio is increased from 0 to 5%, the performance of CLPNM-AD decreased by 3.51%, 3.08%, 2.57% and 1.64% on the Thyroid, Micius, Satellite and Satimage datasets, respectively. Such a performance decline is somewhat tolerable. We also note that CLPNM-AD outperforms the baseline methods at different contamination ratios on all datasets.

CLPNM-AD has a certain tolerance for contaminated data, which may be because the consistency loss allocates the samples in a batch equally into each semantic category, preventing contaminated samples from being assigned to the same cluster. As a result, the contaminated samples have less impact on the probability distribution calculation of each semantic category, making the model somewhat robust to the anomalous samples mixed in the training set.

In general, CLPNM-AD is robust against contaminated data, maintaining good performance as the contamination ratio c% increases from 1% to 5% and always outperforming the performance of all baselines. In addition, to concentrate the model on the anomaly detection task and obtain greater anomaly detection accuracy, the model must be trained with a high-quality dataset, i.e., a clean dataset or a dataset with a low contamination ratio.



(**a**) F_1 scores on Thyroid. (**b**) F_1 scores on Micius.

(**c**) F_1 scores on Satellite. (**d**) F_1 scores on Satimage.

Figure 4. Anomaly detection results of CLPNM-AD and baseline methods on training datasets with contamination ratio c% = [0%, 1%, 2%, 3%, 4%, 5%].

5.7. Ablation Experiment

In this experiment, we investigate the impact of various modules in the model. We repeat our experiment with/without a specific key module. These key modules are primarily vanilla InfoNCE loss $\mathcal{L}_{infonce}$, clustering consistency objective loss $\mathcal{L}_{consistency}$ and the negative samples mixing strategy, where $\mathcal{L}_{infonce}$ and negative samples mixing strategy form $\mathcal{L}_{pcl}$.

As shown in Table 5, our full model achieves the best performance on all datasets. When only $\mathcal{L}_{consistency}$ is used, the method performs poorly because it makes no sense to emphasize consistency without the similarity information between samples as a semantic category guide. When only $\mathcal{L}_{infonce}$ is used, the method performs better than when only $\mathcal{L}_{consistency}$ is used because $\mathcal{L}_{infonce}$ can explore the consistency of positive samples in the feature space while also pushing negative samples away, resulting in a more conducive semantic structure to detect anomalies. Combining $\mathcal{L}_{consistency}$ and $\mathcal{L}_{infonce}$ for model training outperforms $\mathcal{L}_{consistency}$ and $\mathcal{L}_{infonce}$ alone on most datasets. This is probably because $\mathcal{L}_{consistency}$ can further exploit the semantic similarity information extracted by $\mathcal{L}_{infonce}$ to make the clusters more separable and thus facilitate anomaly detection. Our full model achieves the best performance due to the ability of hard negative samples to contribute a larger magnitude gradient to update the network parameters, while also weakening the contribution of false negative samples to the gradient, preventing semantic similarity from being broken and thus facilitating the separation of normal and abnormal

samples. In conclusion, these ablation experiments validate that each module in our model is useful and necessary.

Table 5. Ablation study on Thyroid, Micius, Satellite and Satimage datasets.

$\mathcal{L}_{consistency}$	$\mathcal{L}_{infonce}$	$\mathcal{L}_{pcl}$ Negatives Mixing	Datasets			
			Thyroid	Micius	Satellite	Satimage
✓			63.3 ± 5.0	83.2 ± 1.1	74.8 ± 0.3	91.4 ± 1.0
	✓		75.4 ± 3.5	83.1 ± 1.3	75.5 ± 8.3	92.6 ± 1.9
✓	✓		80.3 ± 1.9	83.2 ± 1.2	75.7 ± 0.6	91.5 ± 2.5
✓	✓	✓	82.6 ± 3.0	83.7 ± 1.1	77.7 ± 0.5	92.7 ± 1.3

5.8. Sensitivity Studies

In this section, we discuss the effect of the number of prototypes K and batch size N on the performance of CLPNM-AD on the Thyroid, Micius, Satellite and Satimage datasets.

Figure 5 depicts the performance of CLPNM-AD with varying numbers of prototypes $K = [4, 8, 12, 16, 20, 24]$. It can be observed that CLPNM-AD is resistant to prototype numbers on Micius, Satellite and Satimage. In the case of Thyroid, the number of prototypes has an evident effect on CLPNM-AD, with F_1 decreasing as K increases from 4 to 24. We conjecture that the model degeneration on Thyroid is due to the fact that the dataset itself has specific semantic clusters, the number of which is close to four, and increasing the number of prototypes too much will break this semantic structure.

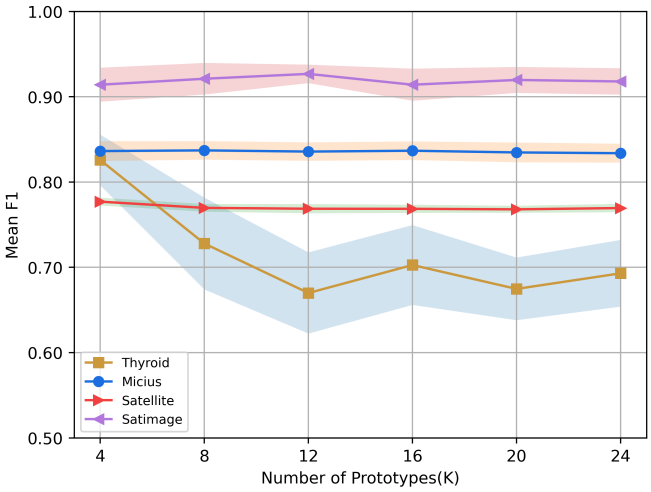


Figure 5. Sensitivity analysis for the number of prototypes.

Figure 6 illustrates the detection performance of CLPNM-AD with batch size $N = [32, 64, 128, 256, 512, 1024]$. On Micius, Satellite and Satimage, we can see that CLPNM-AD is not sensitive to batch size, and the performance does not vary significantly when the batch size N is varied. On Thyroid, however, CLPNM-AD was more clearly affected by the batch size, with the method achieving the best performance at batch size $N = 512$. This may be because too small batches contain too few samples and lack negative samples to guide network learning, while too large batches cause the network to converge more slowly and fail to fully converge for a given epoch number.

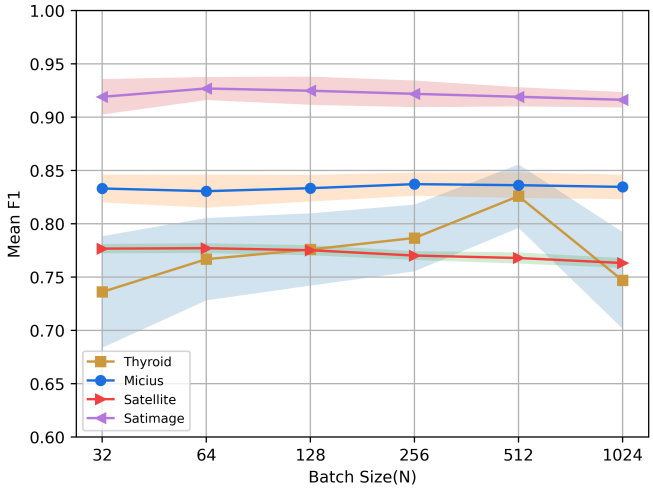


Figure 6. Sensitivity analysis for batch size.

6. Conclusions

In this paper, we propose a correlation anomaly detection method for telemetry data, namely CLPNM-AD. The CLPNM-AD framework first employs an augmentation process with random feature corruption to obtain augmented samples. Following that, a prototype consistency strategy is used to capture the semantic category information of the samples. After that, prototype-based negative mixing contrastive loss is used to pull positive samples from the same prototype closer together and to push samples from different prototypes farther apart to obtain a profile that makes normal and anomaly distinguishable. Finally, anomaly decisions are made using a prototype-based anomaly score function. Extensive experiments are carried out to demonstrate superior performance to baselines on a variety of datasets, with up to 11.5% improvements on the standard F_1 score. The experiments also show CLPNM-AD's robustness against noise, insensitivity to batch size and the number of prototypes and validate the need for each module in CLPNM-AD.

The method proposed in this paper combines contrastive learning and prototype learning to train an encoder to build a profile of normal telemetry data and then use that profile to distinguish between normal and anomalous samples, allowing for better anomaly detection. Our contribution is to use the distance between prototypes as a weight to synthesize hard negative samples and guide encoder learning, resulting in a more discriminative normal profile and a new framework for detecting correlation anomalies in telemetry parameters. In our real-world satellite operation tasks, the method complements existing limit checking and expert system approaches.

The results of the ablation experiments in Section 5.7 show that using the hard-negative samples synthesis strategy improves the algorithm's performance on different datasets to varying degrees, and that using the learned semantic category information to guide the encoder's learning is beneficial for learning normal profiles. It can also be seen that the degree of algorithm improvement varies across datasets, which may be related to the underlying semantic categories in the data, and thus exploring how to better design prototype learning algorithms may be an interesting direction for future research. Furthermore, the goal of anomaly detection is to create a profile that distinguishes between normal and anomalous samples, and in the future, adding stronger constraints to the normal profile (using synthetic anomalous samples or a small number of real anomalous samples) to learn a more discriminative normal profile should be considered. Simultaneously, practical applications in satellite operations necessitate the construction of anomaly detection models as simply and rapidly as possible, so future work should investigate how to automate the search for hyperparameters and the network structure of the encoder.

Author Contributions: Conceptualization, G.G. and T.H.; methodology, G.G.; investigation, G.G.; formal analysis, G.G., T.H. and T.Z.; writing—original draft preparation, G.G.; writing—review and editing, G.G., T.H., T.Z., H.L. and Y.L.; funding acquisition, T.H. and Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Chinese Academy of Sciences Project, grant number XDA15040100.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors thank the Space Science Mission Operations Center (SMOC) for providing satellite telemetry data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hassanien, A.E.; Darwish, A.; Abdelghafar, S. Machine learning in telemetry data mining of space mission: Basics, challenging and future directions. *Artif. Intell. Rev.* **2020**, *53*, 3201–3230. [CrossRef]
2. Baireddy, S.; Desai, S.R.; Mathieson, J.L.; Foster, R.H.; Chan, M.W.; Comer, M.L.; Delp, E.J. Spacecraft time-series anomaly detection using transfer learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 1951–1960.
3. Ji, X.Y.; Li, Y.Z.; Liu, G.Q.; Wang, J.; Xiang, S.H.; Yang, X.N.; Bi, Y.Q. A brief review of ground and flight failures of Chinese spacecraft. *Prog. Aerosp. Sci.* **2019**, *107*, 19–29. [CrossRef]
4. Ibrahim, S.K.; Ahmed, A.; Zeidan, M.A.E.; Ziedan, I.E. Machine learning methods for spacecraft telemetry mining. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *55*, 1816–1827. [CrossRef]
5. Aggarwal, C.C. An introduction to outlier analysis. In *Outlier Analysis*; Springer: Cham, Switzerland, 2017; pp. 1–34.
6. Pang, G.; Shen, C.; Cao, L.; Hengel, A.V.D. Deep learning for anomaly detection: A review. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–38. [CrossRef]
7. Zhang, X.; Mu, J.; Zhang, X.; Liu, H.; Zong, L.; Li, Y. Deep anomaly detection with self-supervised learning and adversarial training. *Pattern Recognit.* **2022**, *121*, 108234. [CrossRef]
8. Yairi, T.; Takeishi, N.; Oda, T.; Nakajima, Y.; Nishimura, N.; Takata, N. A data-driven health monitoring method for satellite housekeeping data based on probabilistic clustering and dimensionality reduction. *IEEE Trans. Aerosp. Electron. Syst.* **2017**, *53*, 1384–1401. [CrossRef]
9. Takeishi, N.; Yairi, T. Anomaly detection from multivariate time-series with sparse representation. In Proceedings of the 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), San Diego, CA, USA, 5–8 October 2014; pp. 2651–2656.
10. Chalapathy, R.; Menon, A.K.; Chawla, S. Anomaly detection using one-class neural networks. *arXiv* **2018**, arXiv:1802.06360.
11. Zong, B.; Song, Q.; Min, M.R.; Cheng, W.; Lumezanu, C.; Cho, D.; Chen, H. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
12. Lv, P.; Yu, Y.; Fan, Y.; Tang, X.; Tong, X. Layer-constrained variational autoencoding kernel density estimation model for anomaly detection. *Knowl.-Based Syst.* **2020**, *196*, 105753. [CrossRef]
13. Jin, W.; Sun, B.; Li, Z.; Zhang, S.; Chen, Z. Detecting anomalies of satellite power subsystem via stage-training denoising autoencoders. *Sensors* **2019**, *19*, 3216. [CrossRef]
14. Schlegl, T.; Seeböck, P.; Waldstein, S.M.; Schmidt-Erfurth, U.; Langs, G. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In Proceedings of the Information Processing in Medical Imaging: 25th International Conference, IPMI 2017, Boone, NC, USA, 25–30 June 2017; pp. 146–157.
15. Yu, J.; Song, Y.; Tang, D.; Han, D.; Dai, J. Telemetry data-based spacecraft anomaly detection with spatial–temporal generative adversarial networks. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–9. [CrossRef]
16. Geiger, A.; Liu, D.; Alnegheimish, S.; Cuesta-Infante, A.; Veeramachaneni, K. Tadgan: Time series anomaly detection using generative adversarial networks. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA. 10–13 December 2020; pp. 33–43.
17. Bergman, L.; Hoshen, Y. Classification-based anomaly detection for general data. *arXiv* **2020**, arXiv:2005.02359.
18. Qiu, C.; Pfrommer, T.; Kloft, M.; Mandt, S.; Rudolph, M. Neural Transformation Learning for Deep Anomaly Detection Beyond Images. In Proceedings of the 38th International Conference on Machine Learning, Online, 18–24 July 2021; Volume 139, pp. 8703–8714.
19. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In Proceedings of the International Conference on Machine Learning, Online, 13–18 July 2020; pp. 1597–1607.

20. Jaiswal, A.; Babu, A.R.; Zadeh, M.Z.; Banerjee, D.; Makedon, F. A survey on contrastive self-supervised learning. *Technologies* **2020**, *9*, 2. [CrossRef]
21. Wang, F.; Liu, H. Understanding the behaviour of contrastive loss. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2021; pp. 2495–2504.
22. Ahmed, T. Online anomaly detection using KDE. In Proceedings of the GLOBECOM 2009-2009 IEEE Global Telecommunications Conference, Honolulu, HI, USA, 30 November–4 December 2009; pp. 1–8.
23. Li, K.L.; Huang, H.K.; Tian, S.F.; Xu, W. Improving one-class SVM for anomaly detection. In Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 03EX693), Xi'an, China, 5 November 2003; Volume 5; pp. 3077–3081.
24. Li, L.; Hansman, R.J.; Palacios, R.; Welsch, R. Anomaly detection via a Gaussian Mixture Model for flight operation and safety monitoring. *Transp. Res. Part C Emerg. Technol.* **2016**, *64*, 45–57. [CrossRef]
25. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: Identifying Density-Based Local Outliers. *SIGMOD Rec.* **2000**, *29*, 93–104. [CrossRef]
26. Ruff, L.; Vandermeulen, R.; Goernitz, N.; Deecke, L.; Siddiqui, S.A.; Binder, A.; Müller, E.; Kloft, M. Deep one-class classification. In Proceedings of the International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 4393–4402.
27. Sohn, K.; Li, C.L.; Yoon, J.; Jin, M.; Pfister, T. Learning and evaluating representations for deep one-class classification. *arXiv* **2020**, arXiv:2011.02578.
28. Tack, J.; Mo, S.; Jeong, J.; Shin, J. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 11839–11852.
29. van den Oord, A.; Li, Y.; Vinyals, O. Representation learning with contrastive predictive coding. *arXiv* **2018**, arXiv:1807.03748.
30. Wu, Z.; Xiong, Y.; Yu, S.X.; Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 3733–3742.
31. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 9729–9738.
32. Caron, M.; Misra, I.; Mairal, J.; Goyal, P.; Bojanowski, P.; Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9912–9924.
33. Kalantidis, Y.; Sariyildiz, M.B.; Pion, N.; Weinzaepfel, P.; Larlus, D. Hard negative mixing for contrastive learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 21798–21809.
34. Robinson, J.; Chuang, C.Y.; Sra, S.; Jegelka, S. Contrastive learning with hard negative samples. *arXiv* **2020**, arXiv:2010.04592.
35. Chuang, C.Y.; Robinson, J.; Lin, Y.C.; Torralba, A.; Jegelka, S. Debiased contrastive learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 8765–8775.
36. Huynh, T.; Kornblith, S.; Walter, M.R.; Maire, M.; Khademi, M. Boosting contrastive self-supervised learning with false negative cancellation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; pp. 2785–2795.
37. Thota, M.; Leontidis, G. Contrastive domain adaptation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 2209–2218.
38. Lin, S.; Liu, C.; Zhou, P.; Hu, Z.Y.; Wang, S.; Zhao, R.; Zheng, Y.; Lin, L.; Xing, E.; Liang, X. Prototypical graph contrastive learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**. [CrossRef] [PubMed]
39. Bahri, D.; Jiang, H.; Tay, Y.; Metzler, D. Scarf: Self-supervised contrastive learning using random feature corruption. *arXiv* **2021**, arXiv:2106.15147.
40. Asano, Y.M.; Rupprecht, C.; Vedaldi, A. Self-labelling via simultaneous clustering and representation learning. *arXiv* **2019**, arXiv:1911.05371.
41. Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 1–9.
42. Gidaris, S.; Singh, P.; Komodakis, N. Unsupervised representation learning by predicting image rotations. *arXiv* **2018**, arXiv:1803.07728.
43. Lee, K.; Lee, K.; Lee, H.; Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In Proceedings of the Advances in Neural Information Processing Systems 31 (NeurIPS 2018), Montreal, QC, Canada, 3–8 December 2018.
44. Kamoi, R.; Kobayashi, K. Why is the mahalanobis distance effective for anomaly detection? *arXiv* **2020**, arXiv:2003.00402.