Effectively Detecting Operational Anomalies in Large-scale IoT Data Infrastructures by using a GAN-based Predictive Model

Peng Chen[1], Hongyun Liu[2], Ruyue Xin[2], Thierry Carval[3], Jiale Zhao[4], Yunni Xia *[4] and Zhiming Zhao *[2]

[1]*School of Computer and Software Engineering, Xihua University, Chengdu, 610039, China and Multiscale Networked Systems research group, University of Amsterdam, 1098XH, Amsterdam.*
[2]*Multiscale Networked Systems research group, University of Amsterdam, 1098XH, Amsterdam, the Netherlands.*
[3]*Euro-Argo Eric, Plouzané, 29280, France.*
[4]*Schoole of Computer Science, Chongqing University, Chongqing, 400044, China*
Email: xiayunni@hotmail.com; z.zhao@uva.nl

Quality of data services is crucial for operational large-scale internet-of-things (IoT) research data infrastructure, in particular when serving large amounts of distributed users. Effectively detecting runtime anomalies and diagnosing their root cause helps to defend against adversarial attacks, thereby essentially boosting system security and robustness of the IoT infrastructure services. However, conventional anomaly detection methods are inadequate when facing the dynamic complexities of these systems. In contrast, supervised machine learning methods are unable to exploit large amounts of data due to the unavailability of labeled data. This paper leverages popular GAN-based generative models and end-to-end one-class classification to improve unsupervised anomaly detection. A novel heterogeneous BiGAN-based anomaly detection model Heterogeneous Temporal Anomaly-reconstruction GAN (HTA-GAN) is proposed to make better use of a one-class classifier and a novel anomaly scoring function. The Generator-Encoder-Discriminator BiGAN structure can lead to practical anomaly score computation and temporal feature capturing. We empirically compare the proposed approach with several state-of-the-art anomaly detection methods on real-world datasets, anomaly benchmarks, and synthetic datasets. The results show that HTA-GAN outperforms its competitors and demonstrates better robustness.

Keywords: Data Infrastructure; Anomaly Detection; Generative Adversarial Networks (GAN); Unsupervised Learning

Received 00 January 2009; revised 00 Month 2009

1. INTRODUCTION

Scientific research data infrastructures bring together facilitates, resources, research data and services used by scientific communities to conduct datacentric research and establish best practice for science, and foster innovation. In the environmental and earth sciences domain, there are various existing research data infrastructures in operation, e.g., EPOS[1] of solid earth and ACTRIS[2] of atmosphere. Among them, Euro-Argo[3] is a typical example, in which thousands of ocean observation sensors, namely Argo devices, are deployed to monitor the ocean variables. These sensors provide large amounts of timely updated status data, and the accumulated research data are well-organized in large-scale IoT data infrastructures to serve user communities for different purposes, e.g., for realtime decisions, oceanographic research, and education, as shown in Figure 1. In such kind of data infrastructures, the quality of the data services, e.g., for retrieving and accessing data products, is crucial for

[1] https://epos-eu.org/
[2] http://actris.nilu.no/
[3] http://www.euro-argo.eu/

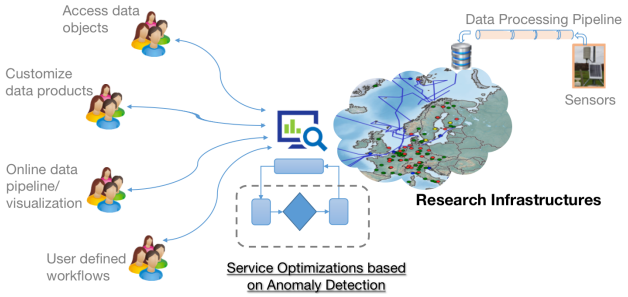

FIGURE 1. Typical Service Optimization for Large-scale IoT Data Infrastructures.

enabling applications with critical business values such as realtime early warning.

Monitoring the service performance and detecting abnormal system behavior is a meaningful way to diagnose the system status and assure service quality. Anomaly detection, a.k.a. outlier detection or novelty detection, is referred to as the process of detecting data instances that have significantly different characteristics from the majority of data instances. Generally, the system monitor data, especially service logs, are organized in time series. In addition, we are focusing on point anomaly detection in this paper instead of complicated anomalies like contextual or collective ones. Thus, we limit anomaly detection as "to detect a point in time series where the system's behavior is significantly different from the previous normal status". Anomaly detection has been an active research area for several decades [1, 2, 3].

Although anomaly detection has been active since the 1960s and it is highlighted in various technologies, including data mining, machine learning, computer vision, and statistics, it remains challenging because of: i) Unknownness: anomalies are associated with many unknowns until their actual occurrence, such as novel frauds and network intrusions. ii) Heterogeneity: one class of anomalies may be completely different from another class. iii) Rarity: anomalies are typically rare data instances, compared with normal instances that often dominate and occupy a most proportion of the data. Therefore, this rarity leads to the unavailability of large-scale labeled data in almost all situations.

In recent years, deep learning has demonstrated tremendous capabilities in learning feature representations of complex data such as high-dimensional non-linear data, temporal data, spatial data, and graph data, gaining great success in various applications. As for anomaly detection, deep learning for anomaly detection, deep anomaly detection for short, aims to learn feature representations or anomaly scoring via deep neural networks to detect anomalies. Unlike traditional distance-based, one-class classification, probabilistic or cluster-based methods, there are mainly three categories in deep anomaly detection: feature extraction, learning feature representations of normality, and end-to-end anomaly scoring. Recently, deep anomaly detection approaches have continuously pushed the boundaries from different aspects for detecting challenging real-world anomalies.

Supervised deep anomaly detection still suffers from limited available high-quality labeled datasets. Generative Adversarial Networks (GAN) [4] framework has shown to be wildly successful in generating realistic-looking images, real-valued time series sequences, or even melodious polyphonic music. Thus GAN-based anomaly detection has attracted much attention, and several GAN-based methods have been proposed recently. However, due to the high complexity of massive high-dimensional, non-linear, and non-independent data and the difficulties of GAN's adversarial training model, we still face some challenges in applying it in anomaly detection. First of all, low anomaly detection precision and recall rate in high-dimensional and/or non-independent data. Secondly, convergence issues and mode collapse occur frequently during adversarial model training. Last, it is time-consuming for neural network training, which may not be acceptable for an enormous amount of real-world applications.

As for real-time operational anomaly detection for large-scale scientific research data infrastructure, limited exploration for GAN-based anomaly detection approaches has been conducted. Inspired by Generative Adversarial Active Learning [5] to generate anomalies instead of normal instances via the minimax game, we are motivated to propose the Heterogeneous Temporal Anomaly-reconstruction GAN (HTA-GAN) to detect point anomalies from time series. HTA-GAN can satisfy better detecting precision, recall, and real-time performance for operating data infrastructure services. Our method adopts a novel heterogeneous generator and discriminator architecture, which can empower temporal feature capturing via recurrent neural networks, make the best use of convolutional neural networks' strong pattern classification capability, and reduce overfitting issues. We define a novel anomaly scoring function using a weighted combination of the discriminator's binary cross-entropy and the generator's anomaly reconstruction error. BiGAN [6] architecture is leveraged to improve computational efficiency of the reverse mapping of the generator while generating anomalies. The main contributions of the proposed HTA-GAN are as follows:

- Capture normal multivariate time series data regularities while discriminating generated anomalies in an unsupervised one-class classification manner by providing an effective heterogeneous GAN-based architecture.
- Exploit end-to-end one-class discrimination and the reconstruction error of anomalies by developing a novel anomaly score function.

- Improve the computational efficiency of the anomaly score by introducing a BiGAN-based structure.

The rest of this paper is organized as follows. Section 2 presents a brief review of the related works. Section 3 introduces the proposed HTA-GAN architecture and corresponding anomaly score function. In Section 4, we introduce seven test datasets, including real operational logs from the Euro-Argo infrastructure, anomaly bench-marks, and synthetic datasets, and then show the experimental results of the proposed HTA-GAN with other seven state-of-the-art methods on these test datasets. Section 5 presents further discussions on the results and service optimization for Euro-Argo Infrastructures. Finally, Section 6 summarizes the whole paper and suggests possible future work.

2. RELATED WORK

We briefly review existing work on point anomaly detection, especially recent progress for time series. We start from the classic anomaly detection methods, followed by deep-learning-based ones. More comprehensive literature reviews can be found in recent surveys [7, 8, 9].

2.1. Classic Methods

Due to the inherent unavailability of labeled anomaly data for supervised learning, anomaly detection is mostly based on unsupervised methods. Most classic anomaly detection methods are as straightforward as model-based methods to establish a model for all samples and then predict anomalies as those having large deviations from the established profiles of time series. These model-based methods are typically based on linear models, distance models, and probability models.

Principal Component Analysis (PCA) [10] is a typical example of linear models. Most linear-model-based anomaly detection methods basically preserve the important variability information extracted from dimension reduction for vast amounts of correlated data. K-Nearest Neighbor (KNN) [11] is a popular approach of distance-based methods; it computes the average distance to its k nearest neighbors and obtains anomaly scores based on this distance. Local Outlier Factor (LOF) [12] is another example of distance-based methods. Most linear or distance-based models are only effective for highly correlated data and require the data to comply with some assumptions, such as following multivariate Gaussian distribution [13]. The probabilistic-based methods were proposed as improvements of distance-based methods by paying more attention to the data distributions. For example, the Angle-Based Outlier Detection (ABOD) [14] and Isolation Forest (IF) [15] deal with data by considering variable correlations. However, these methods are unable to take into consideration the temporal correlation along with time steps and thus do not work well for complicated datasets. One-Class Support Vector Machine (OC-SVM) [16] does not make any assumptions about the data distribution. It aims to find a hyperplane that can separate the vast majority of data from the origin in the projected high-dimensional space.

Overall, choosing a suitable model and parameters are the key to the classic methods, which significantly depends on the prior expertise. However, due to the complexities of high-dimension, non-linear and non-stationary multivariate time series data, traditional methods face significant challenges, including high computational costs and the "curse of dimensionality."

2.2. Deep Learning Methods

In recent years, deep learning has proven to be amazingly promising in various applications. Thus, deep-learning-based unsupervised anomaly detection methods have gained visible popularity, especially for multivariate time series. Generally, these well-studied methods could be categorized into three groups.

Firstly, deep feature extraction approaches directly leverage popular pre-trained deep learning models such as AlexNet [17] to extract low-dimensional feature representations from high-dimensional, non-linear, and non-stationary data for downstream anomaly detection. The anomaly detection performance would be pretty impressive if there are proper mature pre-trained models for corresponding specific applications such as images or videos.

The second one is to learn feature representations of normality, to which most existing unsupervised deep methods belong. Auto-Encoder (AE) and its variants like Variational Auto-Encoders are typical models by inspecting its reconstruction errors. Deep Autoencoding Gaussian Mixture Model (DAGMM) [18] and LSTM En-coder-Decoder [19] have reported good performance for multivariate anomaly detection. The most straightforward way is to follow the same procedure as the conventional use of AEs by adapting the network architecture to the type of input data including multivariate time series, such as CNN-AE [20, 21], Conv-LSTM-AE [22], and GCN-AE [23]. There are also deep distance-based anomaly detection methods aiming to learn feature representations that are specifically optimized for a specific type of distance-based anomaly measures. There have been a number of effective distance-based anomaly measures introduced, such as DB outliers [24, 25], k-nearest neighbor distance [26], average k-nearest neighbor distance [11], relative distance [27], and random nearest neighbor distance [28, 29]. Next, the predictability-based model, which learns feature representations by predicting the current data instances using the representations of the previous instances within a temporal window as the context, is widely used for sequence representation

learning and anomaly detection [30, 31, 32, 33]. The predictability-based methods should capture the temporal or sequential dependence within a given time window or sequence length to achieve accurate predictions. Specifically, GAN-based [4] anomaly detection methods emerge quickly after their early use in [34] due to GAN's capability to capture the deep representation of real data through a mini-max game and achieved state-of-the-art performance in a variety of applications. The majority of GAN-based methods fall into this group as the GAN-based generative methods. AnoGAN [34] generally aims to learn a latent feature space of a generative network G so that the latent space well captures the normality underlying the given data. Some forms of residual between the real instance and the generated instance are then defined as anomaly score. Then, EGAN [35] and fast AnoGAN [36] have been proposed to improve AnoGAN's computational efficiency. GANomaly [37] further improves the generator by adopting an encoder-decoder-encoder network for the generator. There have been some other GAN-based improvements from various aspects [38, 39, 40, 41] in this group.

The last group is the end-to-end anomaly scoring approach, which is not dependent on existing anomaly measures. Similar to typical end-to-end applications, it has a neural network that directly learns the anomaly scores. Novel loss functions are often required to drive the anomaly scoring network, e.g., the ranking-model-based approach devises ordinal regression-based loss functions to drive the anomaly scoring neural network [42, 43]. It is worth mentioning that GAN can also be adopted as an end-to-end one-class classification model in this group. The key idea is to train the discriminator to be a one-class classifier for normality to discriminate those normal instances from generated anomalies. Adversarially Learned One-class Classification (ALOCC) is first studied in [44]. This idea is explored more in [45]. One-class adversarial network (OCAN) is introduced in [46] to leverage the idea of badGAN [47] to generate fringe instances based on the distribution of the normal training data. Generative Adversarial Active Learning (GAAL) [5] adopts generative adversarial learning to generate informative potential outliers directly. Meanwhile, one widely used assumption for this group would be that the anomalies are not as concentrated as the normal instances. Furthermore, GAN can be also adopted to deal with data imbalance for intrusion detection [48].

In summary, there are impressive achievements by deep-learning-based unsupervised anomaly detection approaches. However, there are still three difficulties, including i) the detecting performance still could not satisfy real-world applications, especially for classification measurements such as precision or recall. ii) the real-time performance suffer from multiple problems during neural network training, such as failure or slowness to converge and mode collapse [49]. iii) feature representation learning could be biased by infrequent regularities and the presence of anomalies in the training data.

3. PROPOSED METHOD: HTA-GAN

To resolve these difficulties and detect operational anomalies for data infrastructure, we propose a novel GAN-based approach HTA-GAN. The organization of this section is as follows: In section 3.1, we present the problem formulation of anomaly detection. Section 3.2 introduces the architecture of HTA-GAN. Section 3.3 presents anomaly scoring computation to satisfy real-time requirements. The heterogeneous structure for better feature learning is introduced in Section 3.4.

3.1. Problem Formulation

Given a dataset $X = \{x_1, x_2, \ldots, x_N\}, x_i \in \mathbb{R}^D$ with unobservable labels $Y = \{\mathrm{y_1}, \mathrm{y_2}, \ldots, \mathrm{y_N}\}, \quad \mathrm{y_i} \in \{0, 1\}$, let $Z \in \mathbb{R}^{\mathrm{K}}(\mathrm{K} \ll \mathrm{N})$ be a feature space, goal of deep anomaly detection is to learn a feature mapping function $\phi(\cdot) : X \mapsto \mathcal{Z}$ or an anomaly scoring function $\tau(\cdot) : X \mapsto \mathbb{R}$ in order that anomalies can be easily distinguished from normality in the space yielded by the ϕ or τ function, where both ϕ and τ are neural networks with $H \in \mathrm{N}$ hidden layers with weight matrix $\Theta = \{\mathrm{M}^1, \mathrm{M}^2, \ldots, \mathrm{M}^{\mathrm{H}}\}$. Specifically for the feature mapping $\phi(\cdot)$, an additional step is required to calculate the anomaly score of each data instance in the new representation space. As an anomaly scoring function, $\tau(\cdot)$ can directly infer the anomaly scores with raw data inputs. More significant τ outputs indicate a greater degree of being anomalous. Therefore, deep anomaly detection can be formulated as:

For $\phi(\cdot) : X \mapsto Z$, there is:

$$\{\Theta^*, \ \mathrm{W}^*\} = \underset{\Theta,\ \mathrm{W}}{argmin} \sum_{\mathrm{x} \in \chi} \ell(\psi(\phi(\mathrm{x}; \Theta); \mathrm{W})) \tag{1}$$

$$S_{\mathrm{x}} = f(\mathrm{x}, \phi\Theta^*, \psi\mathrm{W}^*) \tag{2}$$

While $\tau(\cdot) : X \mapsto \mathbb{R}$, there is:

$$\Theta^* = \underset{\Theta}{argmin} \sum_{\mathrm{x} \in \chi} \ell(\tau(\mathrm{x}; \Theta)) \tag{3}$$

$$S_{\mathrm{x}} = \tau(\mathrm{x}; \Theta^*) \tag{4}$$

Where ϕ maps the original data to the feature space Z, ψ is parameterized by W is a learning task that operates on the feature space Z and is dedicated to learning normal data's regularities and distributions, ℓ is a model relevant loss function, and f is a scoring function that utilizes ϕ and ψ to calculate the anomaly score s.

As in most existing GAN-based generative methods, the parameters of generator G and discriminator D are updated based on the outputs of D until Nash Equilibrium. On the one hand, train the discriminator

D to be as sensitive as possible to assign correct labels to either real or fake data instances. On the other hand, simultaneously turn the generator G to be as ingenious as possible to fool the discriminator after sufficient iterations. As can be imagined, the generator G will have a better understanding of datasets' hidden distributions of the training dataset and be able to generate realistic samples. Theoretically, in a typical GAN model, the generator, which takes randomly generated noises as input, can directly generate informative potential anomalous instances that occur far from the real normal data through the guide of the discriminator. An anomaly score could be adopted inversely for such fake anomaly generation. As a result, the discriminator can identify anomalies based on the anomaly score by describing a division boundary separating potential anomalies from the normal instances. There is an assumption that the class distributions of the datasets should be unbalanced between anomalies and normal data. This assumption is widely used for anomaly detection. In addition, as we mainly focus on point anomaly detection for time series, we only consider generating point anomaly, i.e., single anomalous point in time series, via the GAN model.

To be more concrete, the fundamental intuition of this GAN-based generative approach is that, given any data instance x, it aims at an instance z from latent feature space of the generative network G so that the corresponding generated instance G(z) and x are as similar as possible. Since the latent space is enforced to learn the training dataset's underlying key feature, anomalies are expected to have less similarity to generated counterparts than normal instances. Specifically, a GAN is first trained with the following conventional objective:

$$\min_{\Theta_G} \max_{\Theta_D} V(D,G) = \mathrm{E}_{x\sim px}[\log[D(x)]+E_{x\sim pz}[\log[1-D(G(z))] \tag{5}$$

Where G and D are respectively parameterized by Θ_G and Θ_D, and V is the value function of the two-player minimax game. After that, for each x, to find its best z, two loss functions - residual loss and discrimination loss - are used to guide the search. The residual loss is defined as:

$$\ell_R\left(X, Z_y\right) = \left\|X - G\left(Z_y\right)\right\|_1 \tag{6}$$

While the discrimination loss is defined based on the feature matching:

$$\ell_{fm}\left(X, Z_y\right) = \left\|h(x) - h\left(G\left(Z_y\right)\right)\right\|_1 \tag{7}$$

Where γ is the index of the search iteration step, and h is a feature mapping from an intermediate layer of D. The search starts with a randomly sampled z, followed by updating z based on the gradients derived from the loss function, i.e., anomaly scoring function:

$$s_x = (1-\alpha)\ell_R\left(X, Z_\gamma\right) + \alpha\ell_{fm}\left(X, Z_\gamma\right) \tag{8}$$

Where α is a hyper parameter. The anomaly score is accordingly defined upon the similarity between x and z obtained at the last step $\gamma*$:

$$s_x = (1-\alpha)\ell_R\left(X, Z_{\gamma^*}\right) + \alpha\ell_{fm}\left(X, Z_{\gamma^*}\right) \tag{9}$$

The end-to-end one-class classification approaches emerge mainly due to the combination between GAN and the concept of one-class classification, i.e., adversarial learning of one-class classifier. The key idea is to learn a one-class discriminator of the normal instances in an end-to-end way. As a result, the discriminator would be able to differentiate those instances from generated fake anomalies well. The intuition is that G can well reconstruct or even augment normal instances, but it can be fooled by inputted anomalies and consequently generates distorted anomalies. Through the minimax optimization, the discriminator D learns to be a one-class classifier, which can better discriminate normal instances from the anomalies than using the original data instances. Thus, D(G(z)) can be directly used as τ to detect anomalies.

Based on the two GAN-relevant methods, the three difficulties as mentioned earlier are transferred into three key problems: i) How to exploit a single GAN's generator and discriminator architecture to combine the two models' advantages. ii) How to define a computational efficient anomaly scoring function. iii) How to conduct simultaneous temporal representation capture and one-class classifier for complex logs of multivariate time series. To perform effective anomaly detection in the real-time operation of data infrastructure, we should handle them reasonably. Moreover, the engineering problem discussed in this paper is originated from the service optimization of Euro-Argo. To enable Euro-Argo data infrastructures to adapt to real-time operation and dynamic provisioning, the maximum duration for analyzing the integrated data should be less than 10 minutes, including data preprocessing, training the model in a sliding window, and calculating anomaly scores. The 10 minutes maximum time window to log analysis is defined by Euro-Argo based on both business value and real-world operation. If it is too long, anomaly detection is not effective enough for real-time operation. However, if a short interval, like less than 1 minute, infrastructure provisioning like VM initialization may not even finish. Therefore, we must take time cost into account.

3.2. HTA-GAN Architecture

To address the first key problem, i.e., how to exploit a single GAN's generator and discriminator architecture to combine the two models' advantages, we need first to look into the differences between the GAN-based generative methods and the end-to-end one-class classification approaches. In brief, there are

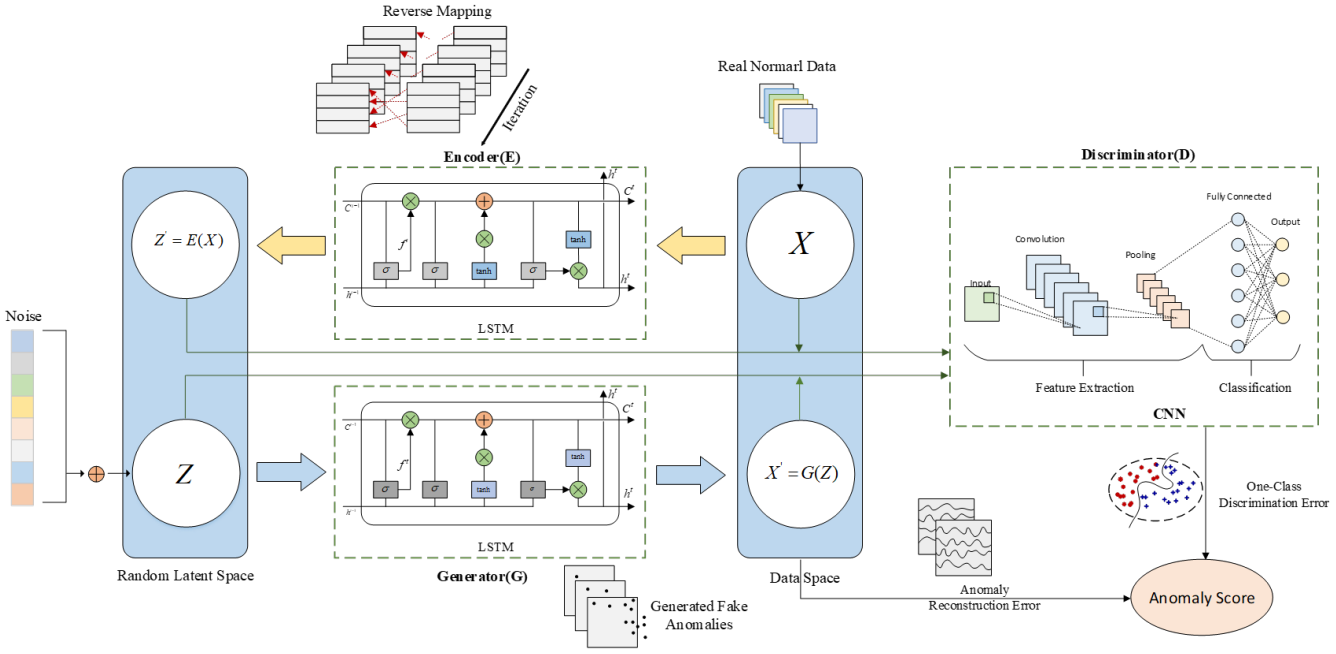

FIGURE 2. HTA-GAN architecture overview. Z is the latent space while X is the original data space.

two main differences: i) the GAN-based methods aim at learning generative characteristics to be maximally similar to the real data distribution, obtaining a generative model that nicely captures the normality of the training data instances; while the end-to-end one-class classification based approaches aim to optimize a discriminative model to separate normal instances from generated anomalous instances. ii) the GAN-based generative methods define the anomaly scores based on the reconstruction residual between the real instances and the corresponding generated instances, whereas the end-to-end one-class classification based methods here directly use the discriminator to recognize anomalies in a binary classification manner, i.e., the discriminator D acts as τ in (3). Next, there are two necessary assumptions, which are consistent with most existing researches [50]. One is that inherent properties of anomalies have significantly different characteristics from normal data, and the class distributions are extremely unbalanced between anomalies and normal data. The other assumption is that anomalies are not as clustered as the normal data.

As shown in Figure 2, on the basis of comparison between GAN-based generative framework and end-to-end one-class classification models, we propose HTA-GAN (Heterogeneous Temporal Anomaly-reconstruction GAN) to exploit and unify both models into a single GAN for operational anomaly detection in large-scale scientific research data infrastructures. Specifically, the discriminator D plays a one-class classifier role and the generator G as an anomalous instance reconstructor. The training procedure falls into a similar manner as end-to-end one-class classification models: The generator G generates fake anomalies from noises from latent space while the discriminator D is to differentiate generated anomalies from real normal data. After sufficient adversarial training epochs, the discriminator D is trained as a robust one-class classifier to detect anomalies directly. Here we must notice that it is crucial for the generator G to learn anomalies' representation when the discriminator is sensitive enough to assign correct labels to an anomaly or normal instance. Therefore, the reconstruction error of the generator G would also be exploited for anomaly scoring as well as direct binary cross-entropy from the discriminator D, i.e., D(G(z)). The reconstruction error indicates the residuals between real-time testing samples and reconstructed samples, i.e., fake anomalies by G based on the mapping from the GAN latent space to real-time space. Here the main difference of the proposed reconstruction error from the typical GAN-based generative framework is that the proposed reconstruction error is more significant for normal data instances due to the fact that the generator generates anomalous instances G. As can be figured out, the proposed reconstruction error is smaller for anomalies while more significant for normal data. Moreover, regarding unknown anomalies, since our model learns the features of normal data and adopts the deviation between an instance and normal instances to decide if it is anomalous, it can detect unknown anomalies.

Therefore, the anomaly scoring function would be appropriately defined by a combination of binary cross-entropy from the discriminator D and the anomaly reconstruction error from generator G, which, according to (8), can be formulated as follows:

$$s_x = \alpha \ell_{OC}(X) - (1-\alpha)\ell_{AR}(X) \tag{10}$$

Where ℓ_{OC} is the one-class classification error, which can be obtained from binary cross-entropy of the discriminator D. However, the anomaly reconstruction error ℓ_{AR} could not be calculated directly, which is the second key problem to be handled.

3.3. BiGAN-based Anomaly Score Calculation

The second key problem, i.e., how to define a computational efficient anomaly scoring function, although binary cross-entropy from the discriminator D is easy for calculation, the calculation of anomaly scoring should be paid much attention. Theoretically, the testing samples should be mapped back into the latent space to calculate the corresponding reconstruction error based on the difference between the reconstructed samples and the actual testing samples. However, reverse mapping is not available directly since the generator G only implements only the mapping from latent space to real data space. Enlightened by the concept of BiGAN [6], the overall architecture would be updated accordingly by adopting BiGAN's encoder-generator-discriminator structure, which can result in simultaneous adversarial training for encoder E, the generator G and the discriminator D. The encoder E would be the reverse mapping from real data space back to the latent space, leading to notably reduced computational complexity. Thus G(E(x)) can be regarded as the reconstructed instance to compare with the real input instance x, which could be pretty straightforward as:

$$\ell_{AR}(X) = \|x - G(E(X))\|_1 \tag{11}$$

With the BiGAN improvement, we can also update our overall minimax game as follows: From

$$\min_{\Theta_G} \max_{\Theta_D} V(D,G) = \mathrm{E}_{x\sim px}[\log[D(x)] + E_{x\sim pz}[\log[1-D(G(z))] \tag{12}$$

To

$$\begin{aligned}\Theta_E \min_{\Theta_G} \max_{\Theta_D} \mathrm{E}_{\mathrm{x}\sim px}\left[\mathrm{E}_{x\sim pE(\cdot|x)}\log[D(x,z)]\right] \\ +\mathrm{E}_{x\sim pz}\left[\mathrm{E}_{x\sim pG(\cdot|z)}[\log(1-D(x,z))]\right]\end{aligned} \tag{13}$$

3.4. Heterogeneous Generator and Discriminator

To deal with properly the third key problem, i.e., how to conduct simultaneous temporal representation capture and one-class classifier for complex logs of multivariate time series, we construct the GAN's generator and discriminator heterogeneously. As to the generator, we adopt Long Short-Term Memory (LSTM) due to its proven temporal feature learning and representation capability. In contrast, Convolutional Neural Networks (CNN) are leveraged for discriminator because of their out-standing performance in pattern classification, as shown in the corresponding part of Figure 2. Following a typical End-to-end one-class GAN-based classifier framework, the generator (G) generates fake anomalies with sequences from a random latent space as its inputs. It passes the generated anomalies to the discriminator (D), which will try to distinguish the generated, i.e., anomalous data instances from the real, i.e., normal training data sequences as a generic one-class classifier. This improved heterogeneous GAN structure is capable of making the best use of LSTM and CNN for their respective merits, which fits for the generator to bring forward anomalous instances actively and for the discriminator to detect anomalies. Furthermore, this heterogeneous structure can also effectively mitigate the overfitting issue. Considering real-time performance and practicality, we adopt shallow layers, i.e., no more than three layers for either LSTM or CNN. We choose Adam and SGD as optimizers during the training process.

4. EVALUATION

4.1. Experimental Settings

This section provides details of the datasets, evaluation metrics, baseline methods, and parameter settings for subsequent experiments.

4.1.1. Datasets

To verify the proposed anomaly detection method, we conduct experiments on three different kinds of datasets, covering both real-world and synthetic datasets.

Euro-Argo Integrated Data Service log: The Euro-Argo scientific research data infrastructure is the European contribution to the global Argo program, which currently has more than 3500 autonomous float instruments globally deployed over the world ocean to measure and report temperature, salinity, and other properties of the oceans. The collected raw data from deployed floats, massive but low-dimensional, is processed into scientific research data, which are key assets for conducting environmental and interdisciplinary scientific research. They are then made available via the Euro-Argo data portal, and research communities can access them from various methods. To guarantee this, the Euro-Argo infrastructure needs to allocate sufficient resources for the storage of data, execution of service requests, and bandwidth for down and uploads. Analyzing access patterns of different data services will help Euro-Argo understand more about how its data products and services are used, particularly in the context of open access to scientific data. Like an ordinary data center, among access pattern analysis and optimization for quality of service, one critical and unavoidable factor is anomaly detection, which is, in data infrastructure context, to detect service failure, SLA unsatisfaction, traffic outliers, and network intrusion. The Euro-Argo

TABLE 1. GENERAL INFORMATION OF SEVEN TEST DATASETS

Dataset	Abbr.	Variables	Samples	Type	Anomaly Percentage
Euro-Argo	Eu	49	28311	Real-world	2.84%
vertebral	Ve	6	240	Benchmark	12.50%
optdigits	Op	64	5216	Benchmark	2.88%
Synthetic 1	S1	200	33000	Synthetic	15.00%
Synthetic 2	S2	100	12000	Synthetic	5.00%
Synthetic 3	S3	50	4000	Synthetic	5.00%
Synthetic 4	S4	50	18000	Synthetic	15.00%

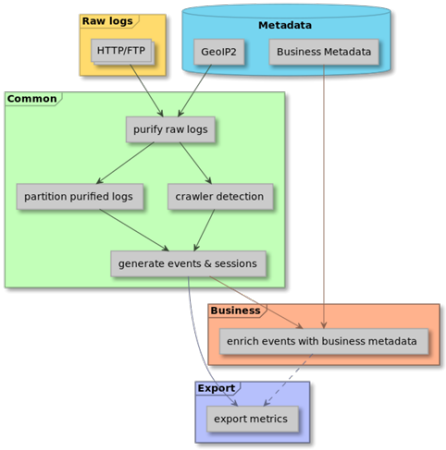

FIGURE 3. Euro-Argo Integrated Data Service logs.

Integrated Data Service Log data, as can be seen from Figure 3, is collected for one month's continuous data services running 24 hours per day. Various anomalies occurred with different intents and divergent lasting durations from a few minutes to some hours. There are 49 variables from different data services measured for one month. There are 28311 samples collected by sampling every 10 minutes from the 4094157 raw log data. In the anomaly detection process, we subdivide the original long multiple sequences into smaller time series by taking a sliding window across raw streams. To enable effective adaptation, Euro-Argo requires that the point anomaly detection accuracy is no less than 90 percent. Meanwhile, another crucial requirement of Euro-Argo is real-time performance since the logs are generated continuously per each request. The time duration for analyzing the integrated service logs should be less than 10 minutes to empower Euro-Argo data infrastructures to adapt to real-time operation and dynamic provisioning. This criterion is a trade-off between dynamic adaptation and resource utilization, e.g., shorter duration may lead to better and faster adaptation but a lower resource utilization or even resource unpreparedness.

Open Anomaly Detection Datasets: as for the open anomaly detection datasets, we choose two benchmark datasets: "vertebral" and "optdigits" to evaluate the performance for small sample sizes. These two benchmarks often appear in the anomaly or outlier detection literature with lower performance due to smaller sample sizes and ranged anomaly rates, e.g., there are only 240 and 5216 samples for "vertebral" and "optdigits", respectively. Generally, the F1 Scores for most tested anomaly detection methods in existing experiments are lower than 0.5.

Synthetic Datasets: We adopt four synthetic datasets for further performance comparison. Specifically, we use the PyOD[4] toolset to generate four random synthetic datasets with different volume, dimension, and anomaly rate combinations: 1) Data volume: training sample size considered from 3000 to 30000 with test sample size from 1000 to 3000 are used to evaluate the computational complexity of anomaly detectors on a group of datasets with different scale; 2) Data dimension: is designed to explore their influence on the performance of different algorithms. Here, from 50 to 200, which are most typical for high-dimensional data, is synthesized for evaluation; 3) Anomaly Distribution and Percentage: the distribution and percentage of anomalies are used to assess the sensitivity for different anomaly class and ratios. Note that, for all synthetic datsets, normal data comply with multivariate Gaussian distribution, and anomalous data is generated by a Uniform distribution with an Anomaly Percentage, i.e., Contamination Rate (CR) in PyOD, from 5% to 15%.

Table 1 summarizes general information about these datasets.

4.1.2. Evaluation Measures

Considering anomaly detection as a one-class classification, we follow generic measurements based on the confusion matrix. Generally, given anomalies are the rare part of the entire datasets, Precision, Recall, and $F1$ score are adopted to measure the overall detection accuracy, which is more suitable for these highly skewed datasets. The definition of *Precision*, *Recall*, and $F1$ scores are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

[4] https://pyod.readthedocs.io/en/latest/pyod.html

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{16}$$

Where TP is True Positives, FP is False Positives, TN is True Negatives and FN is False Negatives.

Another important metric is the time duration, measured in seconds, especially for real-world applications like Euro-Argo. According to Euro-Argo's requirement, it is of much importance that anomaly detection should be carried out within 10 minutes. Meanwhile, to comprehensively evaluate the real-time performance, we also compare the time duration of all approaches on the other six datasets.

4.1.3. State-of-the-art Anomaly Detection Methods

We evaluate the anomaly detection performance of HTA-GAN on the datasets mentioned earlier. We compare the anomaly detection performance with classic and popular K-Nearest Neighbour (KNN) [11], Angle-Based Outlier Detection (ABOD) [14], Isolation Forest (IF) [15], and Auto-Encoder (AE) [18] that are popular unsupervised anomaly detection methods on the datasets. As to the GAN-based method, we compare HTA-GAN with the Efficient GAN (EGAN) [35] method and Generative Adversarial Active Learning (GAAL) method [5]. EGAN is a typical GAN-based generative approach, while GAAL is also based on anomaly generation and one-class classification. Note that GAAL is actually including two variants: Single-Object Generative Adversarial Active Learning (SO GAAL) and Multi-Object Generative Adversarial Active Learning (MO GAAL).

4.1.4. Experimental Environments

We perform all the experiments on a single server with Intel Xeon Processor 4 Core Skylake CPU and an Nvidia T4 GPU. The software environments include Anaconda 3, Python 3.8, Cuda 11.1, and necessary running libraries such as TensorFlow-GPU 2.2.0, Cudnn 7.6.5, PyOD 0.8.6, and Sktime 0.5.1. Implementation of our methods is based on Keras[5], and all compared anomaly detection methods are implemented on a common anomaly detection framework PyOD.

To better compare with those competing methods, we search their optimal parameters in a range of values. For KNN and ABOD, since their performance will be dramatically affected by the size of the neighborhood set and the number of base estimators, we tune it in the range of 2, 4, 8, 10, 20, 40, 80, 100, 200, 400. For IF, the number of base estimators counts a lot, and we choose the best from the range of 20, 40, 80, 100, 200, 400, 800, 1000. For AE, considering the comparison between AE and GAN-based methods, three layers of dense are adopted with hidden neuron numbers (128, 64, 64, 128) and 30 training epochs, which could lead to better performance as well as reasonable time, which are similarly considered in GAN-based methods. Finally, regarding GAN-based methods including SO GAAL, MO GAAL, EGAN, and the proposed HTA-GAN, we use a relatively stringent parameter setting: (i) Three layers of neural networks are adopted for all generators, discriminators, and potential encoders. (ii) The number of hidden neurons is no more than 128. (iii) Adopt the Sigmoid activation function for the output layer of discriminator and ReLU or Leaky ReLU for the remaining layers. (iv) The training epochs are all limited to 30. (v) Use the SGD optimizer with the learning rate of 0.0001 for the generator G and the encoder E and 0.01 for the discriminator D. (vi) a mini-batch size m = min (500, n) for training, and (vii) stop training generator when the downward trend of its loss tends to be slow.

To mitigate experiments' randomness, we run each method for each dataset ten times with different randomized seeds, and we record the average results of the ten runnings in the result tables.

4.2. Experimental Results

In this section, we present results intending to demonstrate the global performance of HTA-GAN on both real-world and synthetic datasets and give some insights into the performance comparison among different anomaly detection methods.

Figure 4 demonstrates the overall F1 Scores for the compared eight approaches on the seven datasets. Table 2 , Table 3, and Table 4 show the detailed results for Euro-Argo, Anomaly benchmarks, and synthetic datasets, respectively. To make the result clear, we show the best performance among the 4 popular unsupervised methods, including K-Nearest Neighbour (KNN), Angle-Based Outlier Detection (ABOD), Isolation Forest (IF), and Auto-Encoder (AE) with underlines. The best result among the 4 GAN-based methods, including EGAN, SO GAAL, MO GAAL, and the proposed HTA-GAN with double underlines, and the overall best performance in bold for all the compared 8 methods. The last column of the tables indicates the ranking of each method's performance within each dataset. Note that the ranking is mainly based on the F1 score.

4.2.1. Results of Euro-Argo Datasets

From Table 2, we observe the following details:

For the Euro-Argo dataset, we focused on the results chosen by best F1 since F1 could demonstrate a balance between precision and recall. Generally, with large-scale samples for feature learning, deep-learning-based models like AE or GAN perform better. AE

[5] https://tensorflow.google.cn/api_docs/python/tf/keras/

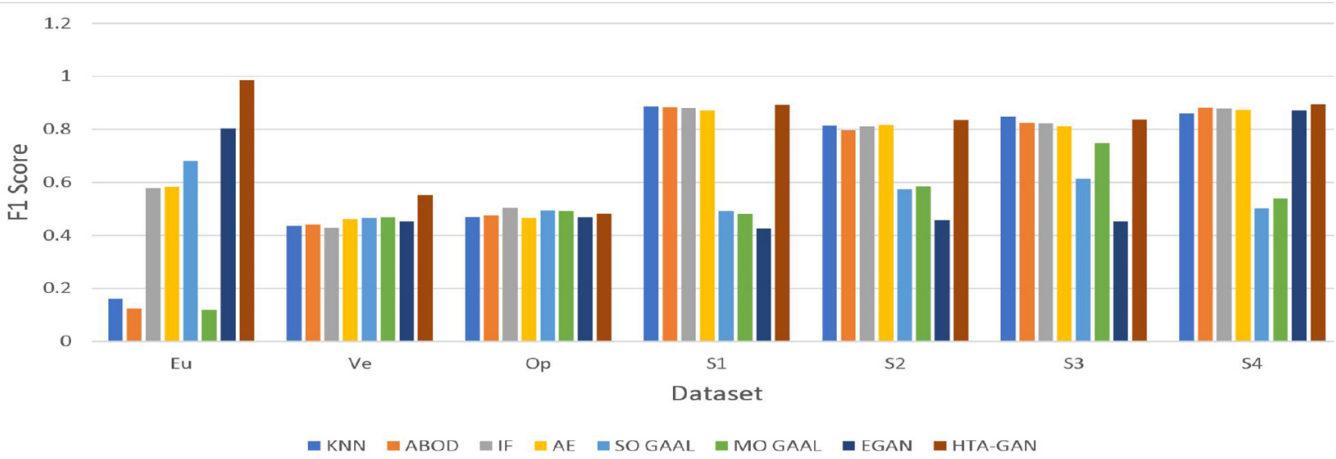

FIGURE 4. The overall F1 Scores for all the compared anomaly detection methods on the given 7 test datasets. EA for Euro-Argo real-world dataset. Ve and Op for the two benchmarks while S1-S4 for the 4 synthetic datasets.

TABLE 2. EXPERIMENT RESULTS FOR EURO-ARGO DATASETS

Dataset	Method	Precision	Recall	F1	Duratiom	Rank
Euro-Argo	KNN	0.5678	0.5216	0.1608	**1.1542**	6
	ABOD	0.5657	0.5039	0.1239	4.6947	7
	IF	0.5901	0.5707	0.5778	1.3863	5
	AE	0.6308	0.7875	0.5835	31.425	4
	SO GAAL	0.6755	0.8599	0.6802	89.5091	3
	MO GAAL	0.5391	0.5012	0.1193	948.5885	8
	EGAN	0.7586	0.9278	0.8032	87.356	2
	HTA-GAN	**0.9961**	**0.974**	**0.9847**	183.2797	1

* The best result among KNN, ABOD, IF, and AE is underlined.
* Double underline is for the best among 4 GAN-based methods.
* The overall best for each metric is bold.

demonstrates the best performance among the four classic methods (KNN, ABOD, IF, and AE). We can see, the proposed HTA-GAN outperformed AE by 30.12% and achieves almost 100% precision and recall. HTA-GAN detects anomalies with 36.53% and 18.65% more for precision and recall than AE, respectively, at the cost of longer time duration for training.

As to the GAN-based method sets (SO GAAL, MO GAAL, and EGAN), according to previously mentioned reasonably similar hyper parameters' settings, although EGAN performs well both in recall and F1, HTA-GAN still obviously leads to an all-around better performance from all metrics except for time cost during training. In a word, the proposed HTA-GAN achieved nearly 100% precision and recall, resulting in detecting almost all the anonymous points correctly for Euro-Argo without false alarms.

As required by Euro-Argo, anomaly detection must satisfy no less than 90 percent accuracy and no more than 10 minutes time cost. The proposed HTA-GAN with three layers of neural networks for generator G, encoder E, and discriminator D would fully satisfy both requirements by achieving 0.9847 F1 and finishing computation in about 3 minutes, as shown in Table 2. Regarding other models, only EGAN can satisfy only Recall by 0.9278 for accuracy. Meanwhile, all models are able to finish the detection within 10 minutes except for MO-GAAL.

4.2.2. Results of Anomaly Benchmarks

We compare performance between different methods for two popular anomaly benchmarks, and we put the result for both datasets together, as shown in Table 3. We can see, here following observations would be found:

Compared methods, either the four popular or the four GAN-based ones, behave differently for the two benchmarks. Generally speaking, less samples will lead to lower performance because a larger sample size could be crucial for adequate feature learning. Specifically, for "vertebral", it is evident that HTA-GAN outperforms the others for all the metrics except for the time duration, which is always an inherent disadvantage for deep learning methods. The 0.5519 F1 score is the only one more than 0.5. However, there are not many differences among all these methods for "optdigits" because the variation is not quite noticeable, e.g., the F1 score ranges from 0.4682 to 0.5032 while Precision from 0.486 to 0.5105.

TABLE 3. EXPERIMENT RESULTS FOR BENCHMARKS

Dataset	Method	Precision	Recall	F1	Duratiom	Rank
vertebral	KNN	0.4302	0.4405	0.4353	**0.0186**	7
	ABOD	0.4318	0.4524	0.4419	0.0559	6
	IF	0.4286	0.4286	0.4286	0.3529	8
	AE	0.4634	0.4583	0.4613	2.8225	4
	SO GAAL	0.4686	0.4642	0.4662	3.5628	3
	MO GAAL	0.486	0.4881	0.4866	21.8672	2
	EGAN	0.4598	0.4464	0.4517	3.5947	5
	HTA-GAN	**0.5767**	**0.5436**	**0.5519**	9.0861	1
optdigits	KNN	0.4865	0.4579	0.47	2.7611	6
	ABOD	0.4928	0.4738	0.4749	3.3089	5
	IF	0.5105	0.5361	0.5032	**0.9948**	1
	AE	0.486	0.4509	0.4658	7.2231	8
	SO GAAL	0.503	0.5093	0.4948	13.6825	3
	MO GAAL	**0.5044**	**0.516**	**0.4927**	131.3917	2
	EGAN	0.4863	0.4556	0.4686	15.3933	7
	HTA-GAN	0.4947	0.4834	0.4822	41.5841	4

* The best result among KNN, ABOD, IF, and AE is underlined.
* Double underline is for the best among 4 GAN-based methods.
* The overall best for each metric is bold.

TABLE 4. EXPERIMENT RESULTS FOR SYNTHETIC DATASETS

Dataset	Method	Precision	Recall	F1	Duratiom	Rank
Sample=33000, Dim=200, CR=0.15	KNN	0.9724	0.8344	0.8866	451.7546	2
	ABOD	0.9717	0.83	0.883	455.0937	3
	IF	0.9712	0.8267	0.8803	**13.6794**	4
	AE	0.9694	0.8156	0.8712	75.126	5
	SO GAAL	0.4958	0.4972	0.4929	204.8572	6
	MO GAAL	0.481	0.4868	0.4814	2304.483	7
	EGAN	0.4157	0.4351	0.4252	248.5142	8
	HTA-GAN	**0.9736**	**0.8422**	**0.8928**	427.9677	1
Sample=12000, Dim=100,CR=0.05	KNN	0.7451	0.9726	0.8149	26.9207	3
	ABOD	0.7273	0.9684	0.7962	27.3808	5
	IF	0.7427	0.9721	0.8124	**1.7035**	4
	AE	0.7463	0.9729	0.8161	20.4199	2
	SO GAAL	0.5612	0.6124	0.5746	56.0444	7
	MO GAAL	0.5763	0.6503	0.5949	504.5886	6
	EGAN	0.472	0.4439	0.4576	57.8765	8
	HTA-GAN	**0.766**	**0.9768**	**0.8354**	123.7224	1
Sample=4000, Dim=50, CR=0.05	KNN	**0.7809**	**0.9795**	**0.8492**	1.5843	1
	ABOD	0.7551	0.9747	0.8249	2.0324	3
	IF	0.7525	0.9742	0.8223	**0.6736**	4
	AE	0.7427	0.9721	0.8123	7.4666	5
	SO GAAL	0.5897	0.6921	0.6136	16.9357	7
	MO GAAL	0.694	0.8774	0.7488	132.9787	6
	EGAN	0.4714	0.4342	0.4521	15.8845	8
	HTA-GAN	0.7688	0.9774	0.8381	37.7779	2
Sample=18000,Dim=50,CR=0.15	KNN	0.9674	0.8022	0.8599	23.963	6
	ABOD	0.9713	0.8278	0.8812	25.8787	2
	IF	0.9708	0.8244	0.8785	**1.7982**	3
	AE	0.97	0.8189	0.8739	25.2651	4
	SO GAAL	0.5061	0.5042	0.5017	74.4308	8
	MO GAAL	0.5584	0.5359	0.539	793.8951	7
	EGAN	0.9694	0.8159	0.8712	77.9439	5
	HTA-GAN	**0.974**	**0.8444**	**0.8945**	166.3911	1

* The best result among KNN, ABOD, IF, and AE is underlined.
* Double underline is for the best among 4 GAN-based methods.
* The overall best for each metric is bold.

The "vertebral" benchmark is interesting because its sample number and its dimension are relatively small, which requires anomaly detection to be data-efficient. From the results, it can be learned that the five deep-learned-based unsupervised methods work pretty well as they beat classic models for this kind of data-efficient task. Among these methods, GAN-based ones behave even better than AE. What is more, the proposed HTA-GAN demonstrates the best detecting results: 9.07% Precision, 5.55% Recall, and 6.53% F1 score better than the second-winner MO GAAL and shorter time by 12.7811 seconds.

As to the "optdigits" benchmark, every method struggles, not only for classic ones but also deep learning ones such as AE and GAN-based models. These struggles are common because, as mentioned in the previous section, we choose "vertebral" and "optdigits" due to their challenging. Among all these methods, IF would be the winner, from F1 to time duration, although the performance is still not good enough and just a little step ahead of others. AE performs poor while GAN-based methods are all just average or even below average.

4.2.3. Results of Synthetic Datasets

From Table 4, it is easy to know:

For the synthetic datasets by PyOD toolsets, a total of four datasets are generated for testing, with different datavolumes, dimensions, and contamination rates. As the PyOD toolset generates the synthetic datasets with multivariate Gaussian distribution for normal data and Uniform distribution for anomalous data, most anomaly detection methods work well since the data distributions are more straightforward to be captured than real-world complicated anomalous scenarios. Classic models such as KNN and IF could effectively capture such data regularities. However, this is challenging for GAN-based methods due to the generator's capability of learning data regularities via a mini-max game. As shown in Table 4, existing GAN-based models still struggle in most cases. The proposed HTA-GAN could perform well due to its heterogeneous structure, exploiting both temporal feature representation and one-class anomaly recognition. As can be seen from Table 4, among those GAN-based methods, only HTA-GAN works fine to get as competitive results as other non-GAN-based methods. HTA-GAN provides the best metrics for three out of four datasets, proving that the proposed HTA-GAN can achieve better robustness. However, compared with other non-GAN-based models, the training would still be a tradeoff because it takes much longer than IF and AE, although it looks acceptable compared to KNN and ABOD.

More precisely, on the one hand, classic unsupervised methods including IF, ABOD, KNN, and AE perform well, with good metrics such as more than 0.85 for F1 score and relatively shorter time, e.g., just a few seconds. On the other hand, GAN-based methods, except for the proposed HTA-GAN, struggle a bit and behave inconsistently, e.g., EGAN could catch up with main-stream metrics as ABOD and HTA-GAN for the fourth datasets but ranks the last for the other datasets. Only the proposed HTA-GAN, thanks to its heterogeneous BiGAN structure to better capture temporal dependencies and identify anomalies via an end-to-end one-class classifier, outperformed other methods consistently, especially for those datasets with large sample sizes. As for high-dimensional datasets, HTA-GAN can still achieve decent performance.

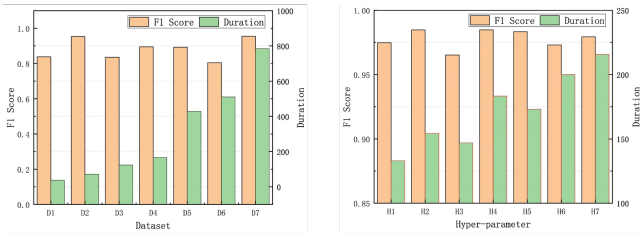

FIGURE 5. Robustness Test Results. (a) shows the F1 score of synthetic datasets D1-D7 with different volume, dimension and contamination rate. (b) demonstrates the F1 scores for different hyper parameters H1-H7.

Overall, as shown in Figure 4 HTA-GAN outperformed the popular unsupervised detection methods for most situations. To be more specific, among all these 7 test datasets, the proposed HTA-GAN ranks no.1 for five times, and for remain two datasets, it ranks no.2 and no.4. The main drawback is that it takes a much longer time to deal with longer subsequences. For the Euro-Argo project, it is acceptable. However, it will be an issue for time-critical real-time applications. It will be worthwhile to explore using other Neural Networks to incorporate the temporal correlation and consider the choice of subsequence length for future work.

4.2.4. Experiments for Robustness Analysis

As alluded to above, the robustness of HTA-GAN is an interesting issue to explore further. First of all, followed by previous synthetic datasets, we conduct more experiments for different data volumes, dimensions, and contamination rates, which could help demonstrate the proposed HTA-GAN's robustness. In addition to the existing four generated datasets, we have three more datasets with different volumes, dimensions, and contamination rates. There are seven datasets from D1 to D7.

We plot the seven datasets' F1 score on Figure 5(a), and we can see that the proposed HTA-GAN performs consistently well. To be more specific, all the F1 scores are generally decent, more than 0.83. For lower contamination rates such as D5 and D6,

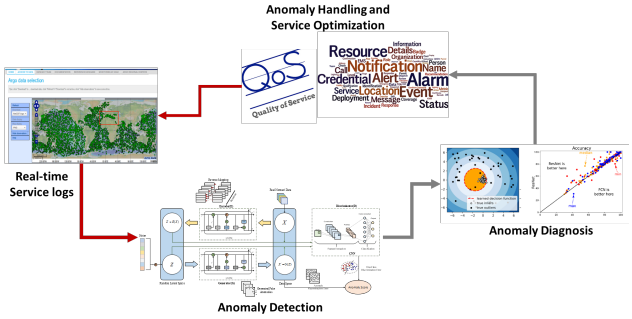

FIGURE 6. The Complete Service Optimization Pipeline for large-scale IoT Data Infrastructure.

the performance is even better, although volume and dimension vary significantly. Another observation is that the runtime increases reasonably according to the scale, decided by both volume and dimension. In a word, HTA-GAN is robust to various datasets with different data volume, dimension, and contamination rate combinations.

Next, HTA-GAN is based on deep neural networks. As mentioned by many pieces of literature, several hyperparameters of neural networks are involved. As for the representation learning ability of neural networks, we adjust the number of layers from 2 to 5 and the number of hidden layer neurons from 64 to 128, respectively, to get seven models of HTA-GAN of H1-H7.

We test Euro-Argo real-world datasets on the above seven models with varied hyperparameters and plot their F1 scores on Figure 5(b). We can see that fewer neurons in hidden layers lead to a slightly worse F1 score because insufficient neurons can result in loss of representation for informative potential anomalies for generator. Moreover, excessive hidden layers may also have little impact on model performance. Certainly, with increased neurons and layers, the runtime arises correspondingly. Therefore, we can conclude that the proposed HTA-GAN is generally robust to different hyperparameter settings

5. DISCUSSION

As for high-quality data services running on data infrastructures such as Euro-Argo, the proposed HTA-GAN is able to detect real-world operational anomalies accurately and timely, which would be practical to improve the quality of scientific data services. The main features of HTA-GAN are: i) a novel anomaly reconstruction error embedded into an end-to-end one-class classification model. ii) adopting BiGAN structure for computational efficiency, and iii) heterogeneous generator and discriminator for different goals of feature learning. Moreover, here some interesting discussions would be involved further.

As shown in Figure 6, here is the complete service optimization pipeline for large-scale IoT data infrastructures. Anomaly detection is a fundamental module of the whole service optimization pipeline for data infrastructures. However, the only detection is not adequate; anomaly-based system diagnosis, or anomaly diagnosis for short, is the next topic. With HTA-GAN's decent detection and a wide variety of known anomalies, upgrading the one-class discriminator into a multiclassifier would be potentially more feasible. Further handling of anomalies would be realistic based on anomaly diagnosis and explanation. The most challenging issues would be: i) Complicated anomalies are involved in most cases. ii) Real-time performance, both for processing the continuously generated log steamings with strict time-constraint and various operational time-critical aspects.

A large-scale data infrastructure is often broadly distributed; it consists of many monitoring sensors deployed in remote areas and data management services running in different data centers. A research data infrastructure is often constructed as an aggregation of many small-scale or regional data infrastructures, as we see in EPOS and many other examples in ENVRI. Detecting real-time anomalies at different parts of the infrastructure will thus be an important operation challenge. How to tackle anomaly detection in edge-cloud continuum would be a fundamental problem here. Are we able to seamlessly migrate and embed HTA-GAN into edge systems as an anomaly detection service? Either communication cost or unavailability of complete system logs would cause massive troubles. The results achieved in this paper provide a good starting point to be applied in decentralized learning, e.g., Decentralized Learning of Generative Adversarial Networks [51] to learn multiple data collections. In addition, anomaly diagnosis is not the final step. As shown in Figure 6, anomaly handling would be more practical, e.g., we could consider anomaly-aware adaptation based on deep-reinforcement-learning-based dynamic scheduling approaches [52, 53, 54] with effective anomaly detection and diagnosis, which would also contribute to QoS improvement [55].

According to the experimental results, although demonstrating well, there are still two main limitations of HTA-GAN, including i) only detecting point anomalies due to lack of contextual anomalous instances, as most methods did. ii) real-time performance still needs improvement for latency-sensitive scenarios. To mitigate, we continue closely collaborating with Euro-Argo technical team to get more contextual or collective anomalous instances and optimizing the convergence of neural networks.

6. CONCLUSIONS

This paper proposes a novel anomaly detection approach HTA-GAN to detecting real-world operational anomalies for high-quality data services running on

large-scale scientific research data infrastructures such as Euro-Argo, which can make better use of GAN's capability as an informative fake anomaly generator and end-to-end one-class discriminator. Moreover, we exploit the heterogeneous structure of GAN to improve multivariate time series representation learning and an efficient BiGAN-based anomaly scoring function leveraging not only binary crossentropy of discriminator but anomaly's reconstruction error. HTA-GAN achieves the best metrics and ranking on the real-world Euro-Argo datasets, anomaly benchmarks, and synthetic datasets compared to several state-of-the-art point anomaly detection methods for multivariate time series. HTA-GAN demonstrates strong robustness to varying hyperparameters of neural networks and datasets with ranged data volume, dimension, and contamination rate based on further experiments. Although neural network training is involved, HTA-GAN can satisfy most real-world applications like Euro-Argo. Generally, HTA-GAN would be suitable for anomaly detection scenarios with complex anomaly distribution, large sample size, limited time cost, and high accuracy requirements. Typical applications would include operation & maintenance (O&M) in cloud platforms, data infrastructures, edge-cloud continuum, especially helpful when integrated into AIOps toolsets. For future work, as shown in Figure 6 for the Euro-Argo project, we plan to conduct further research on i) Introduce multi-classification into discriminator so that it could be capable of anomaly diagnosis and explanation, which would be of key importance for operating such data infrastructure; ii) Complex anomalies, such as contextual or collective anomalies, detection for Euro-Argo Data Service operations. How to generate such complex anomalies based on a GAN model and few-shot learning? How to generate synthetic datasets with a complex anomaly distribution? iii) A detailed study on the pipeline of the whole data-centric operational anomaly detection and handling lifecycle and iv) In terms of applications, we plan to extend HTA-GAN for predictive scheduling and fault diagnosis for edge-cloud continuum.

DATA AVAILABILITY STATEMENT

The data underlying this article were provided by Euro-Argo by permission. Data will be shared on request to the corresponding author with permission of Euro-Argo.

ACKNOWLEDGMENT

The authors wish to thank Erwan Bodere, Misha Mesarcik, Yu Wang, Tingyan Long, Lei Liu and Zhihao Gan for their informative suggestions and help. This work has been partially supported by the European Union's Horizon 2020 research and innovation programme by the ARTICONF project grant agreement No.825134, by the ENVRI-FAIR project grant agreement No.824068, by the BLUECLOUD project grant agreement No.862409, by the LifeWatch ERIC, by China Scholarship Council, Science and Technology Program of Sichuan Province under Grant No.2020JDRC0067 and No.2020YFG0326, and Talent Program of Xihua University under Grant No.Z202047.

REFERENCES

[1] Mao, J., Wang, T., Jin, C., and Zhou, A. (2017) Feature grouping-based outlier detection upon streaming trajectories. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 29, 2696-2709.

[2] Fiore, U., DeSantis, A., Perla, F., Zanetti, P., and Palmieri, F. (2019) Using generative adversarial networks for improving classification effectiveness in credit card fraud detection. *Information Sciences*, 479, 448-455.

[3] Zhang, L., Li, X., Liu, H., Mei, J., Hu, G., Zhao, J., Zou, Y., Xie, B., and Xie, G. (2016) Probabilistic-mismatch anomaly detection: do one's medications match with the diagnoses. *2016 IEEE 16th International Conference on Data Mining (ICDM)*, Barcelona, Spain, Jun 12-15, pp. 659-668. IEEE, New York, USA.

[4] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014) Generative adversarial nets. *Advances in neural information processing systems*, Montréal CANADA, Dec 8-13, Curran Associates, Inc., San Francisco, USA.

[5] Liu, Y., Li, Z., Zhou, C., Jiang, Y., Sun, J., Wang, M., and He, X. (2019) Generative adversarial active learning for unsupervised outlier detection. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 32, 1517-1528.

[6] Donahue, J., Krähenbühl, P., and Darrell, T. (2016) Adversarial feature learning. *arXiv preprint arXiv:1605.09782*

[7] Aggarwal, C. (2017) An introduction to outlier analysis. *Outlier analysis*, Springer, Berlin, Germany.

[8] Gupta, M., Gao, J., Aggarwal, C., and Han, J. (2013) Outlier detection for temporal data: A survey. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 26, 2250-2267.

[9] Pang, G., Shen, C., Cao, L., and Hengel, A. V.D. (2021) Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)*, 54, 1-38.

[10] Wold, S., Esbensen, K., and Geladi, P. (1987) Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2, 37-52.

[11] Angiulli, F. and Pizzuti, C. (2002) Fast outlier detection in high dimensional spaces. *European conference on principles of data mining and knowledge discovery*, Helsinki, Finland, August 19-23, 2002, pp. 15-27. Springer, Berlin, Germany.

[12] Breunig, M., Kriegel, H., Ng, R., and Sander, J. (2000) Lof: identifying density-based local outliers. *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, Dallas, Texas, United States, May 15-18, pp. 93-104. ACM, New York, USA.

[13] Dai, X. and Gao, Z. (2013) From model, signal to knowledge: A data-driven perspective of fault detection and diagnosis. *IEEE Transactions on Industrial Informatics (TII)*, 9, 2226-2238.

[14] Kriegel, H., Schubert, M., and Zimek, A. (2008) Angle-based outlier detection in high-dimensional data. *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, Las Vegas, Nevada, USA, Aug 24-27, pp. 444-452. ACM, New York, USA.

[15] Liu, F., Ting, K., and Zhou, Z. (2008) Isolation forest. *2008 eighth ieee international conference on data mining*, Pisa, Italy, Dec 15-19, pp. 413-422. IEEE, New York, USA.

[16] Erfani, S.M., Rajasegarar, S., Karunasekera, S., and Leckie, C. (2016) High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning. *Pattern Recognition*, 58, 121-134.

[17] Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012) Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, Lake Tahoe, Nevada, USA. Dec 3-6. pp.1097-1105. Curran Associates, Inc., San Francisco, USA.

[18] Zong, B., Song, Q., Min, M.R., Cheng, W., Lumezanu, C., Cho, D., and Chen, H. (2018) Deep autoencoding gaussian mixture model for unsupervised anomaly detection. *International conference on learning representations*. Vancouver, Canada. April 30 - May 3. University of Massachusetts, Massachusetts, USA.

[19] Habler, E., and Shabtai, A. (2018) Using lstm encoder-decoder algorithm for detecting anomalous ads-b messages. *Computers & Security*, 78, 155-173.

[20] Gao, H.H., Qiu, B.Y., Ramon, J., Duran, B., Walayat, H., Xu, Y.S., and Wang, X.H. (2022) TSMAE: A Novel Anomaly Detection Approach for Internet of Things Time Series Data Using Memory-Augmented Autoencoder. *IEEE Transactions on Network Science and Engineering (TNSE)*, 99, 1-14.

[21] Zhang, C., Song, D., Chen, Y., Feng, X., Lumezanu, C., Cheng, W., Ni, J., Zong, B., Chen, H., and Chawla, N.V. (2019) A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. *Proceedings of the AAAI Conference on Artificial Intelligence*, Honolulu, Hawaii, USA. January 27 -February 1. pp. 1409-1416. AAAI, New York, USA.

[22] Luo, W., Liu, W., and Gao, S. (2017) Remembering history with convolutional lstm for anomaly detection. *2017 IEEE International Conference on Multimedia and Expo (ICME)*, Hong Kong, China, Jul 10-14, pp. 439-444. IEEE, New York, USA.

[23] Ding, K., Li, J., Bhanushali, R., and Liu, H. (2019) Deep anomaly detection on attributed networks. *Proceedings of the 2019 SIAM International Conference on Data Mining*, Iowa State University Ames, Iowa. October 19-20. pp. 594-602. SIAM, Philadelphia, USA.

[24] Knorr, E.M. and Ng, R.T. (1999) Finding intensional knowledge of distance-based outliers. *25th VLDB Conference*, September 7-10. pp. 211-222. Morgan Kaufmann Publishers Inc. Francisco, USA.

[25] Knorr, E.M., Ng, R.T., and Tucakov, V. (2000) Distance-based outliers: algorithms and applications. *The VLDB Journal*, 8, 237-253.

[26] Ramaswamy, S., Rastogi, R., and Shim, K. (2000) Efficient algorithms for mining outliers from large data sets. *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, Dallas, Texas, United States, May 15-18, pp. 427-438. ACM, New York, USA.

[27] Zhang, K., Hutter, M., and Jin, H. (2009) A new local distance-based outlier detection approach for scattered real-world data. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Bangkok, Thailand. April 27-30. pp. 813-822. Springer, Berlin, Germany.

[28] Pang, G., Ting, K.M., and Albrecht, D. (2015) Lesinn: Detecting anomalies by identifying least similar nearest neighbours. *2015 IEEE international conference on data mining workshop (ICDMW)*, Atlantic City, NJ, USA, Nov 14-17, pp. 623-630. IEEE, New York, USA.

[29] Sugiyama, M. and Borgwardt, K. (2013) Rapid distance-based outlier detection via sampling. *Advances in Neural Information Processing Systems*, pp. 467-475. Curran Associates, Inc., San Francisco, USA.

[30] Hsieh, J.T., Liu, B., Huang, D.A., Fei-Fei, L., and Niebles, J.C. (2018) Learning to decompose and disentangle representations for video prediction. *arXiv preprint arXiv:1806.04166*

[31] Liao, B., Zhang, J., Wu, C., McIlwraith, D., Chen, T., Yang, S., Guo, Y., and Wu, F. (2018) Deep sequence learning with auxiliary information for traffic prediction. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, London, United Kingdom, Aug 19-23, pp. 537-546. ACM, New York, USA.

[32] Gao, H.H., Xiao, J.S., Yin, Y.Y., Liu, T., and Shi, J.G. (2022) A Mutually Supervised Graph Attention Network for Few-shot Segmentation: The Perspective of Fully Utilizing Limited Samples. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 1-13.

[33] Xu, R., Cheng, Y., Liu, Z., Xie, Y., and Yang, Y. (2020) Improved long short-term memory based anomaly detection with concept drift adaptive method for supporting iot services. *Future Generation Computer Systems*, 112, 228-242.

[34] Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., and Langs, G. (2017) Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. *International conference on information processing in medical imaging*, Boone, NC, USA, June 25-30. pp. 146-157. Springer, Berlin, Germany.

[35] Zenati, H., Foo, C.S., Lecouat, B., Manek, G., and Chandrasekhar, V.R. (2018) Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*

[36] Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., and Schmidt-Erfurth, U. (2019) F-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical image analysis*, 54, 30-44.

[37] Akcay, S., Atapour-Abarghouei, A., and Breckon, T.P. (2018) Ganomaly: Semi-supervised anomaly detection via adversarial training. *Asian conference on computer vision*, Perth, Australia. December 2-6. pp. 622-637. Springer, Berlin, Germany.

[38] Arjovsky, M., Chintala, S., and Bottou, L. (2017) Wasserstein generative adversarial networks. *International conference on machine learning*, June 26-30. pp. 214-223. PMLR.

[39] Gao, H.H., Xu, K.L., Cao, M., Xiao, J.S., Xu, Q., and Yin, Y.Y. (2022) The Deep Features and Attention Mechanism-Based Method to Dish Healthcare Under Social IoT Systems: An Empirical Study With a Hand-Deep Local-Global Net. *IEEE Transactions on Computational Social Systems (TCSS)*, 9, 336-347.

[40] Geiger, A., Liu, D., Alnegheimish, S., Cuesta-Infante, A., and Veeramachaneni, K. (2020) Tadgan: Time series anomaly detection using generative adversarial networks. *2020 IEEE International Conference on Big Data (Big Data)*, Atlanta, Georgia, US, Dec 10-13, pp.33-43. IEEE, New York, USA.

[41] Niu, Z., Yu, K., and Wu, X. (2020) Lstm-based vae-gan for time-series anomaly detection. *Sensors*, 20, 3738-3750.

[42] Yin, Y.Y, Huang, Q., Gao, H.H., and Xu, Y.S. (2021) Personalized APIs Recommendation with Cognitive Knowledge Mining for Industrial Systems. *IEEE Transactions on Industrial Informatics (TII)*, 17, 6153-6161.

[43] Pang, G., Yan, C., Shen, C., Hengel, A.v.d., and Bai, X. (2020) Self-trained deep ordinal regression for end-to-end video anomaly detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Washington, D.C., June 16-20. pp. 12173-12182. IEEE, New York, USA.

[44] Sabokrou, M., Khalooei, M., Fathy, M., and Adeli, E. (2018) Adversarially learned one-class classifier for novelty detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, Utah, USA. June 18-22. pp. 3379-3388. IEEE, New York, USA.

[45] Ngo, P.C., Winarto, A.A., Kou, C.K.L., Park, S., Akram, F., and Lee, H.K. (2019) Fence gan: Towards better anomaly detection. *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, Tanwan, China, Mar 18-20, pp. 141-148. IEEE, New York, USA.

[46] Zheng, P., Yuan, S., Wu, X., Li, J., and Lu, A. (2019) One-class adversarial nets for fraud detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, Honolulu, Hawaii, USA. January 27 - February 1. pp. 1286-1293. AAAI, New York, USA.

[47] Dai, Z., Yang, Z., Yang, F., Cohen, W.W., and Salakhutdinov, R. (2017) Good semi-supervised learning that requires a bad gan. *arXiv preprint arXiv:1705.09783*

[48] Andresini, G., Appice, A., De Rose, L., and Malerba, D. (2021) Gan augmentation to deal with imbalance in imaging-based intrusion detection. *Future Generation Computer Systems*, 123, 108-127.

[49] Metz, L., Poole, B., Pfau, D., and Sohl-Dickstein, J. (2016) Unrolled generative adversarial networks. *arXiv preprint arXiv:1611.02163*

[50] Steinwart, I., Hush, D., and Scovel, C. (2005) A classification framework for anomaly detection. *Journal of Machine Learning Research*, 6, 211-232.

[51] Yonetani, R., Takahashi, T., Hashimoto, A., and Ushiku, Y. (2019) Decentralized learning of generative adversarial networks from multi-client non-iid data. *arXiv preprint arXiv:1905.09684*

[52] Chen, P., Xia, Y.N., and Yu, C. (2021) A Novel Reinforcement-Learning-Based Approach to Workflow Scheduling upon Infrastructure-As-a-Service clouds. *International Journal of Web Service Research*, 18, 21-33.

[53] Wang, B., Liu, F.G., and Lin, W.W. (2021) Energy-efficient VM Scheduling Based on Deep Reinforcement Learning. *Future Generation Computer Systems*, 125, 616-628.

[54] He, Q., Cui, G.M., Zhang, X.Y., Chen, F.F., Deng, S.G., Jin, H., and Yang, Y. (2020) A Game-Theoretical Approach for User Allocation in Edge Computing Environment. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 31, 515-529.

[55] Xia, Y.N., Zhou, M.C., Luo, X., Zhu, Q.S., Li, J., and Huang, Y. (2015) Stochastic Modeling and Quality Evaluation of Infrastructure-as-a-Service Clouds. *IEEE Transactions on Automation Science and Engineering (TASE)*, 12, 162-170.