IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 4, 2022, accepted April 11, 2022, date of publication April 18, 2022, date of current version April 21, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3167640

Hybrid Anomaly Detection via Multihead Dynamic Graph Attention Networks for Multivariate Time Series

LIWEN ZHOU, QINGKUI ZENG, AND BO LI
School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, Jiangsu 210044, China
Corresponding author: Liwen Zhou (20201220057@nuist.edu.cn)

This work was supported by the Postgraduate Research & Practice Innovation Program of Jiangsu Province under Grant KYCX21_1008.

ABSTRACT In the real world, a large number of multivariate time series data are generated by Internet of Things systems, which are composed of many connected sensing devices. Therefore, it is impractical to consider only a single univariate time series for decision-making. High-dimensional time series decrease the performance of traditional anomaly detection methods. Moreover, many previously developed methods capture temporal correlations instead of spatial correlations. Therefore, it is necessary to learn the temporal and spatial correlations between different time series and timestamps. In this paper, to achieve improved anomaly detection performance for multivariate time series, we propose a novel architecture based on a graph attention network (GAT) with multihead dynamic attention (MDA). This framework simultaneously learns the dependencies between sensors in both the temporal and spatial dimensions. To tackle the overfitting problem in autoencoder (AE)-based methods, we propose a hybrid approach that combines a novel generative adversarial network (GAN) architecture as a reconstruction model with a multilayer perceptron (MLP) as a prediction-based model to detect anomalies together. The detection framework proposed in this paper is called the HAD-multihead dynamic GAT (MDGAT). Extensive experiments on different public benchmarks demonstrate the superior performance of HAD-MDGAT over state-of-the-art methods.

INDEX TERMS Multivariate time series, graph attention network, anomaly detection, deep generative model, gated recurrent unit.

I. INTRODUCTION

With the rapid development of information technology, the scales of all kinds of data continue to expand. Time series anomaly detection has become a field of interest for many researchers and practitioners [1]. Anomaly detection has been applied in various domains, such as intrusion detection in cybersecurity, medical detection, economic analysis and fault diagnosis in industry [2].

Time series can be divided into univariate time series and multivariate time series. Because a univariate time series has one dimension of data, some anomaly detection algorithms can locate anomalies regarding one feature. However, in real-world scenarios, many sensors are interconnected to run and generate numerous time series data, such as in cyber-physical systems (CPSs) [3]. Data from different sensors can be related in complex, nonlinear ways (for example, pressure changes affect flow rates and water levels). Similar to these CPS data, multivariate time series data have many interconnected correlations. If a single feature (such as a univariate time series) is used to detect anomalies, it may be difficult to determine whether the system of interest runs normally. Anomalies in multivariate time series tend to be determined by multiple spatial features, and the analysis of a single feature is insufficient for correctly detecting anomalies. Therefore, it is necessary to take the correlations of multiple spatial features into consideration when addressing multiple time series.

Time series also often lack labeled samples [4]. Anomaly labeling requires high expert costs and does not guarantee coverage for all anomaly types. Thus, unsupervised deep learning methods are typically used to accomplish the task of anomaly detection. In recent years, many unsupervised anomaly detection approaches, including prediction-based methods and reconstruction-based methods, have been

The associate editor coordinating the review of this manuscript and approving it for publication was Sajid Ali.

proposed. Prediction-based methods use recurrent neural networks (RNNs), such as the deep malicious insider threat detector (DeepMIT) (Sun *et al.* [5]) and long short-term memory (LSTM) network (Hundman *et al.* [6]). DeepMIT models user behaviors as sequences and predicts the probabilities of anomalies. These approaches utilize the differences between predicted and real samples to detect anomalies. However, with the increasing dimensionality and scales of time series, it is becoming more challenging for these conventional prediction-based methods to effectively capture the temporal correlations in high-dimensional multivariate time series [7]. Reconstruction-based methods, such as the autoencoder (AE) proposed by Aggarwal [8] and the generative adversarial network for multivariate anomaly detection (MAD-GAN) proposed by Li *et al.* [9], can reconstruct samples. The reconstruction error can be obtained by the difference between the original and reconstructed samples. These methods do not simultaneously consider the temporal and spatial dimensions between sensors. Therefore, these methods do not have high accuracy when used with multivariate time series that contain many potential interrelationships. Moreover, these methods can effectively fit data according to the obtained reconstruction errors for anomaly detection. If the data include anomalies, these methods (such as AE variants) also fit anomalies well, leading to reduced anomaly detection performance. When anomalous data are very close to normal data, they are often undetectable.

However, many methods (such as the above models) do not take spatial correlations, which are important for anomaly detection, into account. In the real world, most data are generated from non-Euclidean spaces. Many deep learning methods have poor performance in terms of handling these data. In recent years, graph neural networks (GNNs) have seen increasing popularity. They can effectively model graph-structured data [10], such as molecules, and they have made great progress in terms of capturing spatial correlations. Three main types of GNNs are available, including graph convolution networks (GCNs [11]), graph attention networks (GATs [12]), and graph AEs (GAEs). GCNs can be further classified as spectral or spatial methods. A spectral GCN method uses a spectral decomposition approach, such as Laplace matrix decomposition for a graph, to aggregate node information. When the given graph is large, the whole graph must be used, resulting in decreased performance. A spatial GCN method uses the topology of the input graph to directly aggregate its neighbor node information at each layer of the GCN. Thus, this approach has greater potential to deal with large graphs than spectral GCN methods. Attention mechanisms are widely used in different domains, such as computer vision and natural language processing. Some methods must also observe time series data to mine their useful information. GNNs are no exceptions. A GAT applies an attention mechanism to assign different weights for different neighbor nodes. However, the implementations of GATs are only static: for any query, the neighbor scores are monotonic according to the per-node scores. As a result, a GAT cannot express even simple alignment problems and capture much information between different observations.

Taking the above problems into consideration, we propose a novel architecture, a HAD-multihead dynamic DAT (MDGAT), based on a GAT. The main four contributions of this paper are as follows.

- We propose a HAD-MDGAT based on a GAT. It simultaneously learns the dependencies between sensors in both the temporal and spatial dimensions. It has more robustness.
- We introduce a multihead dynamic attention (MDA) mechanism in our architecture to capture the interrelationships between different sensors. This mechanism can deal with alignment problems and model the different correlations between different keys and different queries.
- Prediction-based and reconstruction-based methods are integrated into our model. The prediction-based model can predict the next value by utilizing spatial and temporal correlations. To solve the overfitting problem, we propose re-encoding a GAN to reconstruct data. This technique uses two generators as encoders to compute differences as parts of the reconstruction errors, improving the accuracy of anomaly detection.
- Experimental results obtained on public datasets show that the HAD-MDGAT achieves the best performance in comparison with state-of-the-art baselines.

The rest of the paper is organized as follows. Section II describes the related work, and Section III presents the details of our proposed HAD-MDGAT model and how to use it for anomaly detection. In Section IV, the HAD-MDGAT is evaluated on multiple datasets, where it achieves better performance than state-of-the-art methods. Section V concludes the paper and proposes possible future work ideas.

II. RELATED WORK

As mentioned in the introduction, time series data are applied in various domains. To date, many anomaly detection methods have been proposed for industrial applications [13]–[15]. Time series can be divided into univariate and multivariate time series. Univariate time series have one dimension. Multivariate time series have many dimensions. However, many anomaly detection methods take temporal correlations into consideration while ignoring the spatial dimension. Some unsupervised anomaly detection methods have made great progress. Even though few GNN-based methods are used for time series anomaly detection, they have recently attracted increased attention.

A. TIME SERIES ANOMALY DETECTION

Univariate time series anomaly detection methods only take the dependencies of the current timestamp and the previous timestamps, such as temporal correlations, into consideration. However, methods for multivariate time series also consider the correlations between different observations.

Some methods deal with both kinds of time series anomaly detection. Among these methods, deep learning approaches have attracted the most attention from researchers. One category, unsupervised learning methods, does not need labeled samples. Classic anomaly detection methods can be divided into proximity-based methods, prediction-based methods and reconstruction-based methods [16].

Proximity-based methods, such as K-nearest neighbors (KNN) [17] and the local outlier factor [18], measure the degrees to which values deviate from anomaly objections. These methods ignore the temporal correlations between observations and need prior knowledge, such as the number of anomalies that are present.

Prediction-based methods are commonly used. Their main idea is that anomalies are identified according to the differences between the predicted values and the real values; such approaches include the autoregressive integrated moving average (ARIMA) [19], gradient boosting regression tree (GBRT) [20], and LSTM [21] methods. The ARIMA has a certain lag and is sensitive to anomalies; at the same time, much smoothness testing and parameter estimation are required. The GBRT approach is applied to detect anomalies for data with stable patterns and periodic characteristics. Due to the uncertainty of single regression tree generation, the differences among the results are large. The ARIMA and GBRT techniques do not consider temporal correlations. However, deep learning methods can tackle these problems. RNNs [22] can detect anomalies by predicting time series data. They capture the temporal correlations between different observations. However, RNNs have lower performance, while the input time series are becoming longer. This means that RNNs cannot capture long-term series [23].

To date, reconstruction-based methods, such as AEs [24], variational AEs (VAEs) [25], the LSTM-VAE [26], unsupervised anomaly detection (USAD) [27], OmniAnomaly [28], the MAD-GAN [12], and a GAN with an attention network and bidirectional LSTM (AMBi-GAN) [29], have also been widely investigated. Such an approach learns a model to reconstruct data that are as similar as possible to the original data. Anomalies are identified by their high anomaly scores. An AE is a basic model. To improve the performance of the original AE, Chen *et al.* proposed a VAE. A VAE additionally considers Kullback-Leibler divergence to measure the difference between the estimated and prior distributions. It combines reconstruction error and distribution error to detect anomalies, but it ignores the temporal correlations in the data. The LSTM-VAE was proposed to capture temporal correlations. USAD is an unsupervised method based on reconstruction and consists of three parts, an encoder and 2 decoders that share the same encoder network. It also uses LSTM to capture temporal correlations. OmniAnomaly uses a VAE with gated recurrent units (GRUs) to detect anomalies. However, OmniAnomaly does not amplify the reconstruction error. When processing time series data, LSTM serves as the basic architecture of the generator and the discriminator to capture temporal correlations. However, LSTM exhibits gradient instability and model collapse problems. The AMBi-GAN consists of bidirectional LSTM and an attention mechanism, and it can capture temporal correlations. Recently, Nguyen *et al.* [30] proposed an LSTM-based method to detect anomalies. It uses LSTM to predict time series and employs an AE-LSTM with a one-class support mechanism to reconstruct time series. Prediction-based and reconstruction-based methods have also been combined to detect anomalies. However, reconstruction-based methods can effectively fit the input data. Thus, these methods fit anomalies when they are close to normal data [31]. Therefore, the resulting overfitting problem decreases the accuracy of anomaly detection.

Even though the above methods are effective, they do not take spatial correlations into consideration.

B. ANOMALY DETECTION WITH GNNS

Deep learning can achieve great success in terms of data representation. The patterns of anomalies can be learned by deep learning methods. However, many deep learning methods have poor performance when handling non-Euclidean data. GNNs have been proposed to tackle graph-structured data. A GNN is based on deep learning. It enhances the capability of the resulting model to process graph pattern information. The anomalies can be easily identified according to the extracted representation [32]. GCNs have been proposed as the convolutional networks of computer vision. Wu *et al.* [33] proposed the multitask GNN (MTGNN) for multivariate time series forecasting problems. The MTGNN consists of a graph convolution module and a temporal convolution module to capture the spatiotemporal dependencies between time series. Weber *et al.* [34] proposed EvolveGCN to detect anomalies in financial transaction networks. Their approach uses a GCN as the feature extractor. The k most influential nodes represent all the information contained in the network at a certain moment. However, the overall characteristics of the network are ignored. The graph deviation network (GDN) [35] also uses the top-k method employed by the MTGNN to construct a graph. It treats each time series as a node on the graph, but the connections between the nodes are learned. GATs are also used for feature extraction. A GAT evaluates a graph deviation score as the difference between the expected value and the observed value. Wang *et al.* [36] proved that a GNN can effectively model multirelation data. Attention mechanisms have been widely applied to sequence-based tasks. GNNs also benefit from this concept by using an attention mechanism during aggregation, integrating the outputs of multiple models, and generating random walks that are oriented to important targets. A GAT is a spatial-based GCN. It uses an attention mechanism to determine the weights of node neighborhoods when aggregating feature information, and it considers the correlations between different time series. Huang *et al.* [37] proposed a hybrid-order GAT (HO-GAT) to detect anomalies in attributed networks. This network uses an HO self-attention mechanism to learn node and motif instance representations. Two encoders are

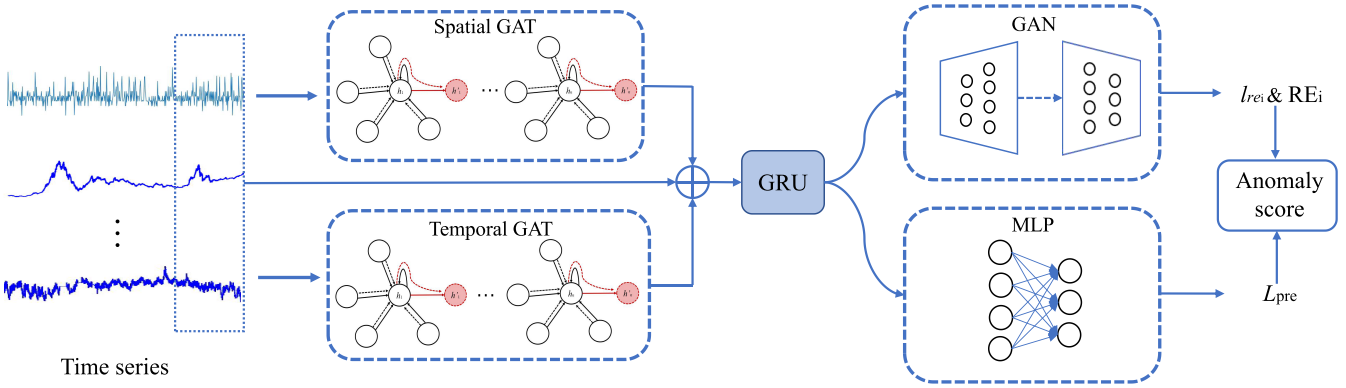


FIGURE 1. The architecture of the HAD-MDGAT.

used to reconstruct the attribute information. The reconstruction errors can serve as anomaly scores for detecting anomalies. Wu *et al.* [38] proposed Event2Graph, which uses a dynamic bipartite graph structure to capture the interdependencies between observations. Event2Graph converts the predicted event edges into anomaly scores. If an anomaly score is higher than the threshold, the corresponding event is classified as an anomaly. Fan *et al.* [39] proposed AnomalyDAE, which uses a GAT in its structural encoder to learn the importance levels among nodes and their neighbors. Thus, AnomalyDAE can efficiently capture structural information.

III. METHODOLOGY

Through the above description, we know that current deep anomaly detection methods only concentrate on temporal correlations while ignoring spatial correlations. In addition, some methods overfit anomalies. In this section, we first state the current problems, propose the HAD-MDGAT framework for capturing the temporal and spatial correlations between different observations, then discuss the proposed novel GAN framework, and finally compute anomaly scores for anomaly detection.

A. PROBLEM STATEMENT

In our work, we focus on anomaly detection in multivariate time series. We execute the HAD-MDGAT on real-world datasets to find anomalous samples that are apparently different from other observations. In our work, the datasets are derived from sensors at timestamp T; the sensor data are denoted as $X = \{x_1, x_2, \ldots, x_T, \ldots, x_n\} \in \mathbb{R}^{N*n}$. At timestamp i, $x_i \in \mathbf{R}^N$, $i = 1, 2, \ldots, n$, is an N-dimensional vector determined from N sensors, where N is the number of features and n is the length of the input data. The inputs are generated by a sliding window. The final output is a vector $y \in R^n$, where $y_t \in \{0, 1\}$ and $y_t = 1$ indicates that the observation at time t is anomalous.

Our work simultaneously learns the dependencies between sensors in both the temporal and spatial dimensions with two MDGATs. Then, a GRU is applied to capture the pattern features of the given time series. Next, we propose a hybrid approach that combines a prediction-based method and a reconstruction-based method to detect anomalies. With respect to the overfitting problem faced by reconstruction-based methods, we propose a novel GAN as the reconstruction-based approach.

To further understand GNNs, we provide the following foundational concepts.

1) GRAPH

A directed graph is defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, v_2, \cdots, v_n\}$ denotes the nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the edges. Here, v_i (with the same dimensionality x_i) denotes the feature vector for each node and $\mathbf{e}_{i,j}$ represents an edge from a node v_j to a node vi. An undirected graph has bidirectional edges.

2) NODE NEIGHBORS

The neighbors of node v_i are defined as $\mathcal{N}(i) = \{j \in \mathcal{V} \mid \mathbf{e}_{i,j} \in \mathcal{E}\}$.

3) PEAK OVER THRESHOLD (POT)

We apply the POT approach [40] as the threshold selection method. It can automatically select the appropriate threshold for a time series. Moreover, it does not make any assumptions about the data distribution and fits the tails of a probability distribution via a generalized Pareto distribution (GPD) with parameters.

B. HAD-MDGAT FRAMEWORK

The HAD-MDGAT framework (shown in Fig. 1) simultaneously learns the dependencies between sensors in both the temporal and spatial dimensions through MDGATs. Then, the two GAT layers and the output of the 1-D convolution layer are concatenated to extract the features of the input time series. The concatenated vector is fed into a prediction-based module and a reconstruction-based module (shown in Fig. 3). The predicted results, the reconstructed samples and the real values are used to compute anomaly scores. A threshold is set by the POT method. If an anomaly score exceeds this

threshold, we can identify that the corresponding sample is anomalous.

1) DATA PROCESSING

In multivariate time series, different variables have various dimensions. This affects the selected threshold and the robustness of hybrid modules. Thus, we process the input time series by executing the maximum-minimum normalization method on the training and testing data:

$$\widehat{x}_i = \frac{x_i - \min X_{\text{Training}}}{\max X_{\text{Training}} - \min X_{\text{Training}}}. \tag{1}$$

2) MULTIHEAD DYNAMIC ATTENTION (MDA)

Due to increases in data volumes and the number of connected sensory devices, it is difficult to achieve high accuracy in the multivariate anomaly detection task. Many deep learning methods concentrate on temporal correlations instead of spatial correlations. Therefore, we introduce a GAT with MDA to capture the temporal and spatial correlations between different observations. The output of each node computed by the GAT layer is shown as follows:

$$h_i = \sigma\left(\sum_{j=1}^{L} \alpha_{ij} v_j\right) \tag{2}$$

$$e_{ij} = a^{\top} \text{LeakyReLU}\left(w \cdot \left(v_i \| v_j\right)\right) \tag{3}$$

$$\alpha_{ij} = \text{softmax}\left(e_{ij}\right) = \frac{\exp\left(e_{ij}\right)}{\sum_{j' \in \mathcal{N}(i)} \exp(e_{ij'})} \tag{4}$$

where h_i denotes the output representation of a node; α_{ij} measures the correlation degree between v_i and v_j; $\|$ denotes the concatenation of node representations; $a \in \mathbb{R}^{2N'}$, $w \in \mathbb{R}^{2N}$ are trainable parameters; a leaky rectified linear unit (LeakyReLU) is used as the activation function to consider the attention weights between node pairs (i, j) for the representation of node i; and j' denotes node i's adjacent neighbors.

A multihead attention mechanism is also applied in the HAD-MDGAT. After the feature vectors calculated by the K-head attention mechanism are concatenated, the corresponding output feature vectors are denoted as follows:

$$h_i' = \prod_{k=1}^{K} \sigma\left(\sum_{j' \in \mathcal{N}(i)} \alpha_{ij'}^{k} w^{k} h_{j'}\right) \tag{5}$$

where $\prod$ denotes vector concatenation; σ is the sigmoid activation function; K indicates that K attention heads are used to calculate the attention scores; $\alpha_{ij'}^{k}$ is the attention score obtained after the calculation of the k-th attention mechanism head; and w_k is the parameter matrix of the linear transformation of the input vector. From [12], if we apply MDA on the last layer of the network, the concatenation method is no longer sensible. Therefore, concatenation is used in the intermediary layers. Regarding the output of the last layer, the concatenation approach does not achieve good results. Therefore, the averaging approach is applied. The output of the last layer calculated by the MDA mechanism is denoted as follows:

$$h_i' = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j' \in \mathcal{N}(i)} \alpha_{ij'}^{k} w^{k} h_{j'}\right) \tag{6}$$

h_i' is the feature vector that is input into the GRU after feature extraction is performed by the HAD-MDGAT.

For multivariate time series anomaly detection, we use two kinds of graph attention layers with MDA (the MDGAT) to learn the dependencies in both the temporal and spatial correlations.

a: SPATIAL LAYER

To capture spatial correlations, we view a multivariate time series as a complete graph. Every node is a value of one feature across n timestamps, and an edge represents the dependency between two nodes. N denotes the number of features (nodes). x_i is denoted as $x_i = \left\{x_{t,i} \mid t \in [0, N)\right\}$. An MDA mechanism is applied to calculate $\vec{h}^{\text{S}}$ for a certain node. The spatial layer is shown in Fig. 2.

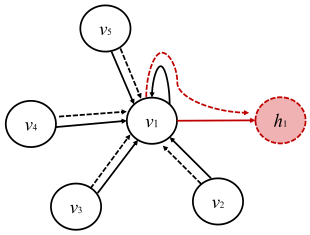


FIGURE 2. The spatial graph attention layer. The final output is shown in the red circle.

b: TEMPORAL LAYER

We also use MDA to capture the temporal correlations between different observations. Each node $x_t = \left\{x_{i,t} \mid i \in [0, n)\right\}$ denotes one timestamp with N features (or sensors). The output of the spatial layer is an $N \times n$ matrix ($\vec{h}^{\text{Temp}}$). The output of the temporal layer is an $n \times N$ matrix.

Finally, we concatenate the outputs of the two layers and the processed vector $\widehat{x}_i$. This forms an $n \times 3N$ matrix containing spatial and temporal correlations.

C. RE-ENCODING GANS

As mentioned in Section II, the prediction-based methods and reconstruction-based methods all have their own advantages. Therefore, we use the two types of methods to detect anomalies together. The output of the GRU is input into the prediction-based method (a multilayer perceptron, MLP) and reconstruction-based method (re-encoding GANs) simultaneously. However, a better reconstruction performance often results in the overfitting of anomalies. This reduces the accuracy of anomaly detection. Moreover, model collapse is a common situation during GAN training. Therefore, we propose a novel GAN (shown in Fig. 3) as the

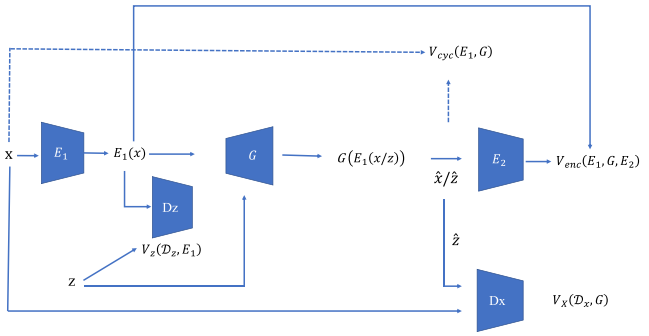


FIGURE 3. The architecture of the proposed GAN.

reconstruction-based method to generate samples. To deal with model collapse in GANs, we use the Wasserstein loss [41] to train the GAN. GRU cells have the advantage of generating time series. Therefore, we apply fully connected neural networks with GRU cells to achieve improved anomaly detection performance. The two generators serve as encoders (E_1 and E_2). Each generated sample is encoded again into the latent space. The re-encoding loss can be obtained by the difference between the two values in the latent space. The output of D_x is a probability score ranging from 0 to 1, which can be used as a part of the anomaly score for detecting anomalies. D_z identifies whether the input is obtained from random noise or the encoded latent space. This enables the z distribution be as close as possible to the X (original time series) distribution and can deal with the overfitting problem.

The objective functions of D_z and D_x are as follows:

$$\min_{G} \max_{D_x \in \mathbf{D_x}} V_x(D_x, G) = \mathbb{E}_{x\sim\mathbb{P}_X}[D_x(x)] - \mathbb{E}_{z\sim\mathbb{P}_z}[D_x(G(z))] \tag{7}$$

$$\min_{G} \max_{D_z \in \mathbf{D}_z} V_z(D_z, E_1) = \mathbb{E}_{x\sim\mathbb{P}_X}[D_z(z)] - \mathbb{E}_{z\sim\mathbb{P}_z}[D_z(E_1(x))] \tag{8}$$

However, during training, it is not guaranteed that the learned mapping can map each individual input x_i to the desired $\hat{x}_i$ by relying on the adversarial and Wasserstein loss functions alone. When a network has a sufficiently large capacity, any random arrangement of the input data can be mapped to the output distribution that matches the target. The cycle consistency loss [42] was proposed by Zhu *et al.* It ensures that images in the corresponding domains have a one-to-one correspondence and prevents conflicts between the samples generated by the two generators. Therefore, the two generators can transform the generated samples back to their original states. Our GAN maps the input x_i to the target z_i in the latent space via E_1 and then generates $\hat{x}$ with a generator. To reduce the size of the space derived from function mapping, the learned function should be cyclically consistent to keep the mappings G and E_1 from contradicting each other. E_1, E_2, and G are trained with the adaptive cycle consistency loss:

$$\min_{\{G,E_1\}} V_{cyc}(E_1, G) = E_{x\sim\mathbb{P}_X}\left[\|x - G(E_1(x))\|_2\right] \tag{9}$$

To further improve the accuracy of anomaly detection, we use two encoders in the GAN to amplify anomalies. The re-encoding loss is used to detect anomalies according to the observed differences in the latent space. When the comparison is conducted in the latent space after encoding, anomalies can be more effectively detected. By minimizing the input features and the encoded features of the generator, the differences between them allow the generator to learn how to encode real samples into the corresponding latent space. Therefore, the generator can address the overfitting problem encountered by the reconstruction-based method. The object re-encoding process is as follows:

$$\min_{\{G,E_1,E_2\}} V_{enc}(E_1, G, E_2) = \mathbb{E}_{x\sim\mathbb{P}_X}\left[\|E_1(x) - E_2(G(E_1(x)))\|_2\right] \tag{10}$$

D. PREDICTION-BASED METHOD

We combine prediction-based and reconstruction-based methods to detect anomalies. Fully connected layers form the basic architecture of the prediction-based method. The loss is as follows:

$$l_{pre} = \sqrt{\sum_{i=1}^{N}\left(x_{n,i} - \hat{x}_{n,i}\right)^2} \tag{11}$$

where x_n is the next timestamp. $x_{n,i}$ denotes the value of the i-th feature of x_n.

E. ANOMALY SCORES

1) RECONSTRUCTION ERROR

Dynamic time warping (DTW) can identify areas with small differences over a long period of time and can address time drift issues. Each series is linearly deflated to perform some "twisting" operation to achieve better alignment. The best match of a given time series is calculated to measure the similarity between local regions. Thus, we use DTW to measure the differences between real and reconstructed samples. There are two time series X and $\hat{X}$, and a $2*l*2*l$ matrix is used to compare the two time series. The warping path traverses this matrix, and the k-th element of the warping path is denoted as $w_k = (i, j)_k$, which is the minimum distance between x_i and $\hat{x}_j$.

$$RE_i = W^* = \mathrm{DTW}(X, \hat{X}) = \min_{W}\left[\frac{1}{K}\sqrt{\sum_{k=1}^{K} w_k}\right] \tag{12}$$

where $X = (x_{i-l}, x_{i-l+1}, \ldots, x_{i+l})$ and $\hat{X} = \left(\hat{x}_{i-l}, \hat{x}_{i-l+1}, \ldots, \hat{x}_{i+l}\right)$ are the real and reconstructed samples for the i-th feature, respectively.

2) RE-ENCODING LOSS

The re-encoding loss is computed as follows:

$$l_{re_i} = \frac{\sum_{i=1}^{T}\left\|E_1\left(\hat{x}_i\right) - E_2\left(G\left(E_1\left(\hat{x}_i\right)\right)\right)\right\|_1}{T} \tag{13}$$

3) PREDICTION LOSS

We calculate the as_is for the N features. The final anomaly score produced by the prediction-based method is the sum of the scores of all features.

$$L_{pre} = \sum_{i=1}^{N} \left(\hat{x}_i - x_i\right)^2 \tag{14}$$

The final anomaly score is computed as follows:

$$\text{AS} = \sum_{i=1}^{N} as_i = (1-\lambda) \cdot L_{pre} + \lambda \cdot \sum_{i=1}^{N} (RE_i + l_{re_i}) \tag{15}$$

where λ is a hyperparameter used to combine the prediction-based and reconstruction-based errors. The default value is 0.5. According to the POT technique, if AS exceeds the threshold, the corresponding samples can be identified as anomalies.

IV. PERFORMANCE ANALYSIS

First, we describe the utilized experimental datasets, baseline models and evaluation metrics. Then, we conduct experiments to demonstrate the performance of our method. Finally, to illustrate the effectiveness of the proposed modules, we conduct an ablation study on five datasets to validate the GAT, the prediction-based module and the reconstruction-based module, which contribute to the performance improvement achieved by the proposed approach.

A. DATASETS

We use five real-world datasets to validate the performance of the HAD-MDGAT, namely, Secure Water Treatment (SWaT), Water Distribution (WADI), Mars Science Laboratory Rover (MSL), Soil Moisture Active Passive (SMAP), and the Server Machine dataset (SMD). SWaT[1] and WADI[2] come from a water treatment test bed coordinated by Singapore's Public Utility Board and a network, respectively. MSL and SMAP contain spacecraft telemetry signals provided by NASA.[3] The SMD[4] is a five-week dataset obtained from a large Internet company. The dataset is derived from 28 machines, and the anomalies in the training dataset are labeled by experts.

The five datasets contain different numbers of anomalies, and the location of each anomaly is known. Table 1 provides the details of each dataset (including their anomaly ratios, etc.).

B. BASELINE MODELS

We implement 7 state-of-the-art baseline models for a performance comparison with the HAD-MDGAT.

1) DAGMM

The deep autoencoding Gaussian mixture model (DAGMM) contains a compression network and an estimation network.

[1]https://itrust.sutd.edu.sg/testbeds/secure-water-treatment-swat/
[2]https://itrust.sutd.edu.sg/testbeds/water-distribution-wadi/
[3]https://s3-us-west-2.amazonaws.com/telemanom/data.zip
[4]https://github.com/NetManAIOps/OmniAnomaly/tree/master

TABLE 1. Dataset statistics.

Dataset	Features	Data Points	Training	Testing	Ratio
SMD	38	1416825	708405	708420	4.16%
SMAP	25	562800	337680	225120	13.13%
MSL	55	132046	79227	52819	10.72%
SWaT	51	946719	568031	378688	11.97%
WADI	127	577658	346594	231064	5.99%

The compression network is an AE. The estimation network can obtain the features and reconstruction errors of the middle hidden layer for anomaly detection purposes.

2) LSTM-VAE

LSTM serves as the basic architecture of the encoder. However, it does not consider the temporal correlations between observations.

3) LSTM-NDT [43]

LSTM is applied to detect anomalies in multivariate time series. This approach utilizes an unsupervised, nonparametric algorithm for threshold determination.

4) OMNIANOMALY

This method learns latent representations through a GRU and a VAE. It takes dependence and stochastic factors into consideration and applies reconstruction probabilities for anomaly detection.

5) USAD

An AE is the basic architecture of USAD. USAD conducts two-phase training in an adversarial manner to reconstruct samples. The input is identified as an anomaly if its corresponding anomaly score is higher than the threshold.

6) MAD-GAN

The MAD-GAN applies LSTM to capture temporal correlations and embeds the captured dependencies into a GAN. It uses reconstruction errors to detect anomalies in multivariate time series.

7) GDN

The GDN also learns the interrelationships between variables in multivariate time series. It directly applies GATs to capture features and uses graph deviation scores to detect anomalies.

C. EVALUATION METRICS

We apply the accuracy (Prec), recall (Rec) and F_1 score (F_1) metrics to evaluate the anomaly detection performance of the HAD-MDGAT.

$$Prec = \frac{TP}{TP + FP} \tag{16}$$

$$Rec = \frac{TP}{TP + FN} \tag{17}$$

$$F_1 = 2 \times \frac{Prec \times Rec}{Prec + Rec} \tag{18}$$

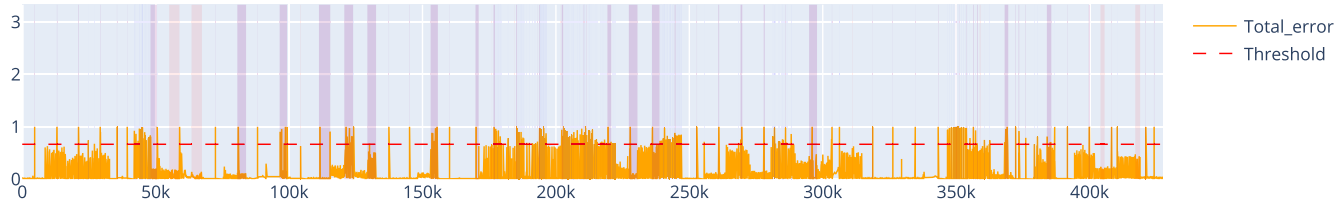


FIGURE 4. Testing set results with the total error in orange, number of predicted anomalies in blue, number of true anomalies in red, and number of correct detections in purple.

The true positive (TP) indicator is the number of samples that a detection model correctly identifies as anomalies. The false positive (FP) indicator represents the number of normal samples that are identified as anomalies. The false negative (FN) metric is the number of anomalous samples that are identified as normal. Moreover, the true negative (TN) measure indicates the number of normal samples that are correctly recognized as the normal type. The model is more robust if the values of the three above metrics (precision, recall, and F1 score) are higher. To evaluate the robustness of the HAD-MDGAT, dynamic Gaussian mixture noise [44] with different signal-to-noise ratios (SNRs) is added to the original data.

D. EXPERIMENTAL SETTINGS

The experiments are implemented in Python 3.6 with PyTorch 1.9 and are performed on a PC with a Ubuntu 18.04.5 LTS, an Intel ® Xeon(R) E5-2678 v3 CPU, 2 RTX 3060 GPUs, and 32 GB of RAM. We use the same sliding window $w = 100$ for all datasets. We set the kernel size $k_0 = 7$ for 1D convolutions. The dimensionalities of the GRU layer (k_1) and the fully connected layers (k_2) are all set to 150. Our model is trained with the Adam optimizer for 300 epochs. The initial learning rate is 0.001. In the GAN, the generator uses the tanh activation function, and the discriminator uses the sigmoid activation function. The numbers of GRUs in the generators and discriminators are all 4.

E. RESULTS

We evaluate the performance of the HAD-MDGAT and compare it with that of with 7 other baselines on five datasets. An ablation study on different components is conducted to determine the impacts of these components on the performance of the HAD-MDGAT. The appropriate thresholds for anomaly detection are used for all models, and the optimal F1 scores are obtained. The optimal results obtained by all models on the public datasets are shown in Table 2 and marked in bold. The HAD-MDGAT detects anomalies on the SMAP training set (shown in Fig. 4). It can detect the most anomalies.

Table 2 shows that the HAD-MDGAT significantly outperforms the other state-of-the-art baselines by achieving the highest mean F1 value (0.929) across all public datasets. The second- and third-best methods in terms of overall performance are the GDN (0.881) and OmniAnomaly (0.785), respectively. The HAD-MDGAT outperforms them by 5.45% and 18.34%, respectively. We make the following observations. (1) The LSTM-VAE is used for classification. It exhibits robustness to imbalanced data and has fast convergence. However, it does not take the temporal correlations between different observations into account. Its performance on the five datasets is not good, especially its value of 0.38 on WADI. The DAGMM is suitable for balanced datasets. However, it exhibits slow convergence for unbalanced data, and its generalization ability is not decent. Additionally, it has poor performance on high-dimensional datasets, such as WADI (0.201). USAD applies a VAE as its basic architecture; it has a fast training speed, and it has the second-best performance on MSL (0.911). However, it achieves a lower value on WADI (0.43). The MAD-GAN has many hyperparameters, making it unsuitable for training. It also has poor generalizability. However, it has better performance on SWaT (0.81) and the SMD (0.872). USAD, the LSTM-VAE, the DAGMM and the MAD-GAN are not suitable for high-dimensional datasets. (2) LSTM-NDT has better performance on SWaT (0.804). However, it does not perform well on MSL (0.564) and the SMD (0.604). This means that LSTM-NDT is sensitive to different scenarios because it cannot conduct effective modeling for all of the cases. OmniAnomaly has better performance on SWaT (0.833), MSL (0.901) and the SMD (0.931). However, it does not take the spatial correlations between observations into consideration. (3) The GDN has good performance on the SMD (0.92) and the other datasets. However, strong connections cannot be merely determined by the tightness of their spatial distances. This approach has a poorer performance on WADI (0.815) than on the other datasets.

TABLE 2. Performance comparison among the best F1 scores of different baseline methods on five datasets.

Baselines	SWaT	WADI	NASA		SMD	Mean
			SMAP	MSL		
DAGMM	0.797	0.201	0.764	0.852	0.697	0.662
LSTM-VAE	0.804	0.38	0.684	0.579	0.768	0.643
LSTM-NDT	0.813	0.507	0.891	0.564	0.604	0.676
OmniAnomaly	0.833	0.406	0.853	0.901	0.931	0.785
USAD	0.812	0.43	0.817	0.911	0.901	0.774
MAD-GAN	0.81	0.624	0.381	0.124	0.872	0.562
GDN	0.901	0.815	0.874	0.892	0.921	0.881
HAD-MDGAT*	**0.917**	**0.906**	**0.915**	**0.969**	**0.937**	**0.929**

1) PERFORMANCE COMPARISON

The HAD-MDGAT achieves the best performance on all datasets. As shown in Fig. 5, the HAD-MDGAT outperforms the other baselines and scores 65.3% higher than the MAD-GAN (0.562). The HAD-MDGAT outperforms USAD (20.03%), OmniAnomaly (18.34%) and the LSTM-VAE (44.48%). The reconstruction-based methods are prone to overfitting anomalies, which leads to low performance. However, these methods have good performance on low-dimensional datasets. We use an extra encoder to address the overfitting problem. The HAD-MDGAT has a higher F1 score than the DAGMM (40.33%). Similar to the LSTM-VAE, the DAGMM does not consider temporal correlations. This indicates that temporal correlations are critical for anomaly detection. As introduced in Section III, we use a temporal layer to capture temporal correlations. The GDN scores 12.23% higher than OmniAnomaly. The GDN applies a GAT to extract the temporal and spatial correlations between different observations. OmniAnomaly does not consider spatial correlations. Therefore, it is essential to utilize spatial correlations when reconstructing samples for anomaly detection. However, the GDN cannot achieve high performance on high-dimensional datasets such as WADI. We use a spatial layer to learn the spatial dependencies between different observations. In our GAN, we propose the use of a re-encoder to amplify the reconstruction error and improve the efficiency of anomaly detection; however, OmniAnomaly does not have a similar effect. The HAD-MDGAT has better performance than LSTM-NDT (37.43%). LSTM-NDT has better performance on SWaT than on MSL, WADI and the SMD. The reconstruction-based methods mostly achieve better performance than the prediction-based methods (the DAGMM and LSTM-NDT) on WADI, the SMD and MSL. This means that the prediction-based and reconstruction-based methods all have separate advantages in terms of anomaly detection. The HAD-MDGAT uses a hybrid method that combines both kinds of approaches to detect anomalies. This technique improves the performance of the HAD-MDGAT on all five datasets.

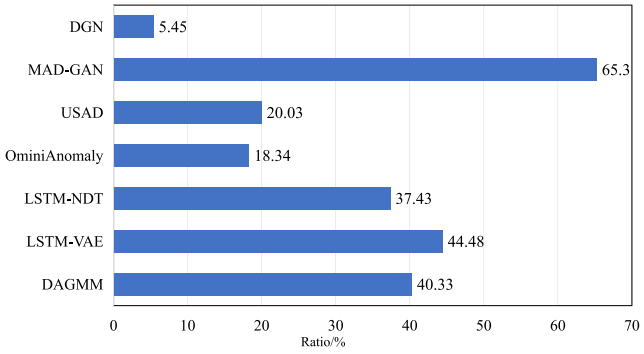


FIGURE 5. The 7 baselines are compared with the HAD-MDGAT based on their average F1 scores on the overall datasets. The HAD-MDGAT outperforms all baselines.

2) ROBUSTNESS

Gaussian noise based on SNRs is typically used to evaluate the robustness of models [7]. We set different SNRs to evaluate the HAD-MDGAT. The results are shown in Table 3. Even though the F1 value decreases with increasing SNRs, our HAD-MDGAT is still more competitive than the other baselines, especially at an SNR of 10. Overall, the HAD-MDGAT is less impacted by noise.

TABLE 3. Test performance on five datasets with dynamic Gaussian mixture noise based on different SNRs.

SNRs	SWaT	WADI	NASA		SMD
			SMAP	MSL	
10	0.896	0.885	0.891	0.913	0.901
20	0.884	0.862	0.873	0.887	0.868
30	0.898	0.843	0.885	0.876	0.890
40	0.864	0.821	0.861	0.859	0.879
Clean	**0.917**	**0.906**	**0.915**	**0.969**	**0.937**

F. ABLATION STUDY

To illustrate the effectiveness of each component of our method, we conduct an ablation study on the same five datasets. The results validate the improvements provided by the MDA mechanism, the hybrid architecture and the spatial layer. The different components are denoted as follows: w/o MDA: disabling the MDA mechanism; w/o prediction-based: disabling the prediction-based method; w/o reconstruction-based: disabling the reconstruction-based method; and w/o spatial layer: preventing the GAT from learning spatial correlations (only the temporal layer remains). The results are shown in Table 4.

TABLE 4. F1 scores obtained in the ablation study.

HAD-MDGAT	SWaT	WADI	NASA		SMD
			SMAP	MSL	
w/o MDA	0.885	0.869	0.878	0.918	0.897
w/o prediction-based	0.886	0.862	0.891	0.947	0.919
w/o reconstruction-based	0.852	0.846	0.862	0.915	0.904
w/o spatial layer	0.845	0.829	0.842	0.881	0.861

Fig. 6 shows that the spatial layer and the reconstruction-based method achieve good performance based on their mean F1 scores. The reconstruction-based method outperforms the prediction-based method. Moreover, the MDA mechanism also contributes to the anomaly detection accuracy.

The MDA mechanism can model the different correlations between different keys and different queries to assign node neighbor scores. The GAT with MDA can fit unbalanced data, and it has superior robustness. We find that the GAT with MDA can also capture the interrelationships between non-adjacent timestamps. The HAD-MDGAT scores are 10.86% higher than those of the version without the spatial layer. When a sample is anomalous, its spatial correlation is greatly different from that of a normal sample in Fig. 7 (which includes 13 features). A darker block indicates a higher spatial correlation and vice versa. This means that it is critical to

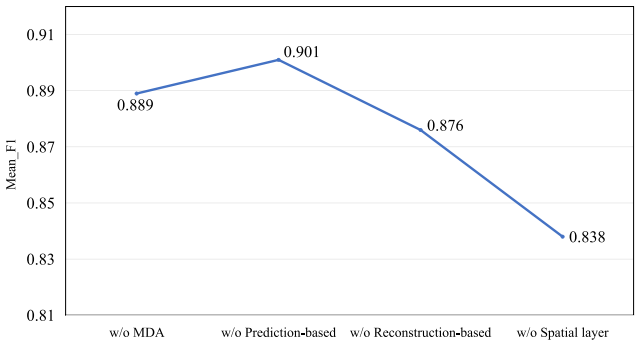


FIGURE 6. Components' average F1 scores on the overall datasets.

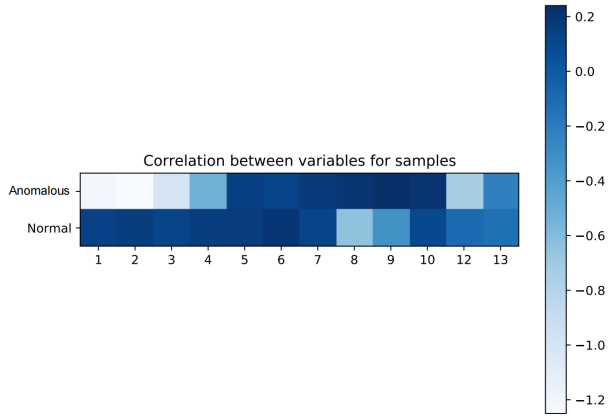


FIGURE 7. Heat map of the spatial correlations between the normal samples and anomalous samples.

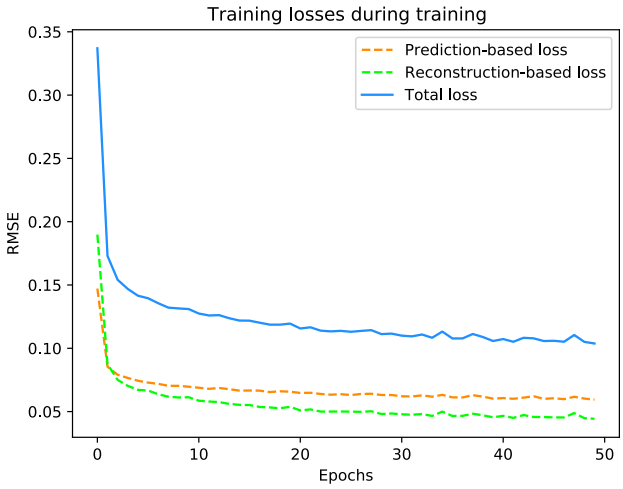


FIGURE 8. Losses incurred during training.

capture spatial correlations when conducting anomaly detection. The prediction-based method is sensitive to random time series.

However, the reconstruction-based method trains a model to learn the distribution of the input data; this model is less affected by noise and other perturbations. The overfitting

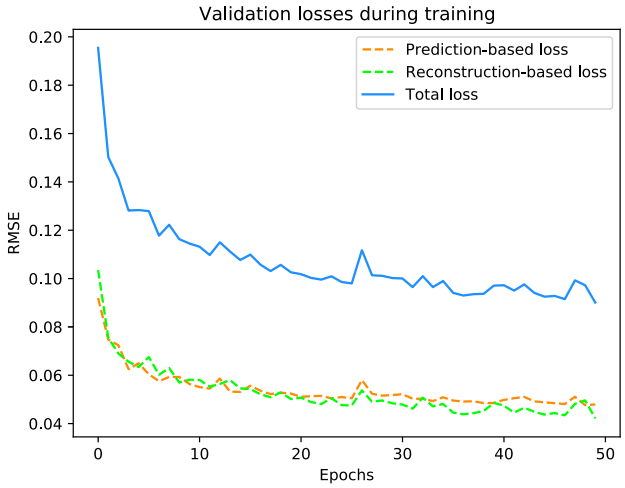


FIGURE 9. Validation losses incurred during training.

problem is a limitation of the reconstruction-based method. If an anomaly is very close to the normal data, it may be undetectable by the reconstruction-based method. We propose a novel GAN to reconstruct data and compare the differences between the representations of two encoders in the latent space, thereby amplifying the errors between the normal and anomalous samples. The prediction-based method can detect anomalies that are sudden time series perturbations. Hence, the hybrid method, which combines both types of methods, can achieve higher anomaly detection accuracy. In order to further optimize proposed model, the Adam gradient descent method is implemented in HAD-MDGAT. Mini-batch algorithm is applied to improve the efficiency of HAD-MDGAT. Data can divided into batches by mini-batch algorithm. In gradient descent training, only a portion of the data set instead of all training set is used and updates the parameters by batch. Therefore, a set of data in a batch jointly determines the direction of this gradient, reducing randomness. As shown in Fig. 8 and Fig. 9, the hybrid method exhibits fast convergence on the training and validating sets.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a hybrid method based on a GAT called the HAD-MDGAT. A GAT with MDA is proposed to learn the temporal and spatial correlations between different observations. Ablation study shows that MDA makes greate contributions to improving anomaly detection accuracy. HAD-MDGAT scores are 10.86% higher than those of the version without the spatial layer. The prediction-based method can detect anomalies that are sudden time series perturbations and the reconstruction-based method trains a model to learn the distribution of time series. The combination of two methods makes that HAD-MDGAT is less affected by noise and other perturbations.In order to evaluate the robustness of HAD-MDGAT, we set different SNRs to evaluate the HAD-MDGAT. HAD-MDGAT are still more

competitive than other baselines. We use the re-encoding loss as a portion of the final anomaly score. Two encoders control the fitting of the learned features in the reconstruction-based method so that the GAN can deal with the overfitting problem. What's more, HAD-MDGAT scores are 6.05% higher than those of the version without the reconstruction-based method from ablation study. Experiments show that the HAD-MDGAT achieves improved anomaly detection performance and outperforms the other seven tested baselines.

For GAN-based anomaly detection models, choosing an appropriate sliding window length is difficult. Additionally, a GAN is unstable during training. In the future, we will investigate these issues and combine other prediction-based methods with GATs.

REFERENCES

[1] T. Barbariol, F. D. Chiara, D. Marcato, and G. A. Susto, "A review of tree-based approaches for anomaly detection," *Control Charts Mach. Learn. Anomaly Detection Manuf.*, pp. 149–185, Aug. 2021, doi: 10.1007/978-3-030-83819-5_7.

[2] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," *ACM Comput. Surveys*, vol. 54, no. 3, pp. 1–33, Jun. 2021, doi: 10.1145/3444690.

[3] J. Goh, S. Adepu, M. Tan, and Z. S. Lee, "Anomaly detection in cyber physical systems using recurrent neural networks," in *Proc. IEEE 18th Int. Symp. High Assurance Syst. Eng. (HASE)*, Jan. 2017, pp. 140–145.

[4] A. Abdulaal, Z. Liu, and T. Lancewicki, "Practical approach to asynchronous multivariate time series anomaly detection and localization," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 2485–2494.

[5] D. Sun, M. Liu, M. Li, Z. Shi, P. Liu, and X. Wang, "DeepMIT: A novel malicious insider threat detection framework based on recurrent neural network," in *Proc. IEEE 24th Int. Conf. Comput. Supported Cooperat. Work Design (CSCWD)*, May 2021, pp. 335–341.

[6] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 387–395.

[7] W. Zheng and J. Hu, "Multivariate time series prediction based on temporal change information learning method," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 4, 2022, doi: 10.1109/TNNLS.2021.3137178.

[8] C. C. Aggarwal, "An introduction to outlier analysis," in *Outlier Analysis*. Cham, Switzerland: Springer, 2017, pp. 1–34.

[9] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2019, pp. 703–716.

[10] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3844–3852.

[11] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[12] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *Stat*, vol. 1050, p. 20, Oct. 2017.

[13] H. Liang, L. Song, J. Wang, L. Guo, X. Li, and J. Liang, "Robust unsupervised anomaly detection via multi-time scale DCGANs with forgetting mechanism for industrial multivariate time series," *Neurocomputing*, vol. 423, pp. 444–462, Jan. 2021.

[14] M. F. Abdelaty, R. Doriguzzi Corin, and D. Siracusa, "DAICS: A deep learning solution for anomaly detection in industrial control systems," *IEEE Trans. Emerg. Topics Comput.*, early access, Apr. 13, 2021, doi: 10.1109/TETC.2021.3073017.

[15] B. Bayram, T. B. Duman, and G. Ince, "Real time detection of acoustic anomalies in industrial processes using sequential autoencoders," *Expert Syst.*, vol. 38, no. 1, Jan. 2021, Art. no. e12564.

[16] K. K. Santhosh, D. P. Dogra, and P. P. Roy, "Anomaly detection in road traffic using visual surveillance: A survey," *ACM Comput. Surveys*, vol. 53, no. 6, pp. 1–26, Nov. 2021.

[17] F. Angiulli and C. Pizzuti, "Fast outlier detection in high dimensional spaces," in *Proc. Eur. Conf. Princ. Data Mining Knowl. Discovery*. Berlin, Germany: Springer, 2002, pp. 15–27.

[18] B. Schölkopf, J. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Comput.*, vol. 13, no. 7, pp. 1443–1471, Jul. 2001.

[19] H. Moayedi and M. Masnadi-Shirazi, "Arima model for network traffic prediction and anomaly detection," in *Proc. Int. Symp. Inf. Technol.*, vol. 4, Sep. 2008, pp. 1–6.

[20] N. Georgoulopoulos, A. Hatzopoulos, K. Karamitsios, I. M. Tabakis, K. Kotrotsios, and A. I. Metsai, "A survey on hardware failure prediction of servers using machine learning and deep learning," in *Proc. 10th Int. Conf. Modern Circuits Syst. Technol. (MOCAST)*, Jul. 2021, pp. 1–5.

[21] P. Malhotra, L. Vig, G. Shroff, and P. Agarwal, "Long short term memory networks for anomaly detection in time series," in *Proc. ESANN*, vol. 89, 2015, pp. 89–94.

[22] L. Bontemps, V. L. Cao, J. McDermott, and N.-A. Le-Khac, "Collective anomaly detection based on long short-term memory recurrent neural networks," in *Proc. Int. Conf. Future Data Secur. Eng.* Cham, Switzerland: Springer, 2016, pp. 141–152.

[23] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," *ACM Comput. Surveys*, vol. 54, no. 3, pp. 1–33, Jun. 2021.

[24] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 665–674.

[25] W. Chen, H. Xu, Z. Li, D. Pei, J. Chen, H. Qiao, Y. Feng, and Z. Wang, "Unsupervised anomaly detection for intricate KPIs via adversarial training of VAE," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Apr. 2019, pp. 1891–1899.

[26] D. Park, Y. Hoshi, and C. C. Kemp, "A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder," *IEEE Robot. Automat. Lett.*, vol. 3, no. 3, pp. 1544–1551, Jul. 2018.

[27] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "USAD: UnSupervised anomaly detection on multivariate time series," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 3395–3404.

[28] H. Zhao, Y. Wang, J. Duan, C. Huang, D. Cao, Y. Tong, B. Xu, J. Bai, J. Tong, and Q. Zhang, "Multivariate time-series anomaly detection via graph attention network," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2020, pp. 841–850.

[29] F. Kong, J. Li, B. Jiang, H. Wang, and H. Song, "Integrated generative model for industrial anomaly detection via bi-directional LSTM and attention mechanism," *IEEE Trans. Ind. Informat.*, early access, May 7, 2021, doi: 10.1109/TII.2021.3078192.

[30] H. D. Nguyen, K. P. Tran, S. Thomassey, and M. Hamad, "Forecasting and anomaly detection approaches using LSTM and LSTM autoencoder techniques with the applications in supply chain management," *Int. J. Inf. Manage.*, vol. 57, Apr. 2021, Art. no. 102282.

[31] D. X. Song and A. Perrig, "Advanced and authenticated marking schemes for IP traceback," in *Proc. Conf. Comput. Commun., 20th Annu. Joint Conf., IEEE Comput. Commun. Soc. (IEEE INFOCOM)*, Apr. 2001.

[32] X. Ma, J. Wu, S. Xue, J. Yang, C. Zhou, Q. Z. Sheng, H. Xiong, and L. Akoglu, "A comprehensive survey on graph anomaly detection with deep learning," *IEEE Trans. Knowl. Data Eng.*, early access, Oct. 8, 2021, doi: 10.1109/TKDE.2021.3118815.

[33] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th Int. Conf. Knowl. Discovery Data Mining (ACM SIGKDD)*, Aug. 2020, pp. 753–763.

[34] M. Weber, G. Domeniconi, J. Chen, D. K. I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics," 2019, *arXiv:1908.02591*.

[35] A. Deng and B. Hooi, "Graph neural network-based anomaly detection in multivariate time series," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 5, pp. 4027–4035.

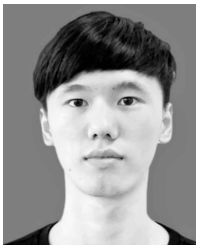
[36] Y. Wang, A. Liu, J. Fang, J. Qu, and L. Zhao, "ADQ-GNN: Next POI recommendation by fusing GNN and area division with quadtree," in *Proc. Int. Conf. Web Inf. Syst. Eng.* Cham, Switzerland: Springer, 2021, pp. 177–192.

[37] L. Huang, Y. Zhu, Y. Gao, T. Liu, C. Chang, C. Liu, Y. Tang, and C.-D. Wang, "Hybrid-order anomaly detection on attributed networks," *IEEE Trans. Knowl. Data Eng.*, early access, Oct. 6, 2021, doi: 10.1109/TKDE.2021.3117842.

[38] Y. Wu, M. Gu, L. Wang, Y. Lin, F. Wang, and H. Yang, "Event2Graph: Event-driven bipartite graph for multivariate time-series anomaly detection," 2021, *arXiv:2108.06783*.
[39] H. Fan, F. Zhang, and Z. Li, "Anomalydae: Dual autoencoder for anomaly detection on attributed networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 5685–5689.
[40] A. Siffer, P.-A. Fouque, A. Termier, and C. Largouet, "Anomaly detection in streams with extreme value theory," in *Proc. 23rd Int. Conf. Knowl. Discovery Data Mining (ACM SIGKDD)*, Aug. 2017, pp. 1067–1075.
[41] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.
[42] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2223–2232.
[43] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 387–395.
[44] Y. Wu, J. Ni, W. Cheng, B. Zong, D. Song, Z. Chen, Y. Liu, X. Zhang, H. Chen, and S. B. Davidson, "Dynamic Gaussian mixture based deep generative model for robust forecasting on sparse multivariate time series," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 1, 2021, pp. 651–659.

LIWEN ZHOU received the bachelor's degree from the Binjiang College, Nanjing University of Information Science and Technology, China. He is currently pursuing the master's degree with the Nanjing University of Information Science and Technology. His research interests include anomaly detection, deep learning, and machine learning.

QINGKUI ZENG received the M.S. degree in computer technology from the College of Computer Science and Engineering, Anhui University of Science and Technology, China, in 2021. He is currently pursuing the Ph.D. degree with the Nanjing University of Information Science and Technology. His research interests include privacy protection and federated learning.



BO LI received the bachelor's degree from Jining Medical University, China. He is currently pursuing the master's degree with the Nanjing University of Information Science and Technology, China. His research interest includes anomaly detection for encrypted malicious traffic in cyberspace security.