2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

OSVAE-GAN: Orthogonal Self-Attention Variational Autoencoder Generative Adversarial Networks for Time Series Anomaly Detection

Zhi Li
State Key Laboratory of Synthetical Automation for Process Industries Northeastern University
Shenyang, China
2102029@stu.neu.edu.cn

Danya Xu
State Key Laboratory of Synthetical Automation for Process Industries Northeastern University
Shenyang, China
2110342@stu.neu.edu.cn

Yuzhe Li
State Key Laboratory of Synthetical Automation for Process Industries Northeastern University
Shenyang, China
yuzheli@mail.neu.edu.cn

Tianyou Chai
State Key Laboratory of Synthetical Automation for Process Industries Northeastern University
Shenyang, China
tychai@mail.neu.edu.cn

Tao Yang*
State Key Laboratory of Synthetical Automation for Process Industries Northeastern University
Shenyang, China
yangtao@mail.neu.edu.cn

***Abstract*—Time series anomaly detection is a binary classification problem with unbalanced data, which aims to identify data that fall outside of the normal behaviors. Since the proportion of the abnormal data is very small, the cost of labeling all data is prohibitively high. Therefore, unsupervised methods are more suitable than supervised methods. With the rapid development of deep learning, various multivariate time series anomaly detection methods based on deep learning have been proposed. However, existing methods do not fully capture the spatial-temporal correlations and are not robust to noise. To address these issues, we propose an unsupervised anomaly detection method called Orthogonal Self-Attention Variational Autoencoder Generative Adversarial Networks (OSVAE-GAN). To fully extract the spatial-temporal correlations, we use an orthogonal self-attention (OS) mechanism. Moreover, to increase the capability to deal with complex multivariate data, we integrate two generative adversarial networks (GANs) with the variational autoencoder (VAE). Finally, to reduce the influence of noise, we introduce the maximum mean discrepancy (MMD) loss. Experiments are conducted on five public datasets, which show that the proposed method is superior to the existing methods.**

***Index Terms*—Anomaly detection, generative adversarial networks, maximum mean discrepancy, orthogonal self-attention.**

I. INTRODUCTION

Anomaly detection (AD) is important in data mining, which aims to find outliers that are significantly different from the majority of the data. Over the last decade, research on anomaly detection has made great progress due to its wide applications in industrial production equipment monitoring, network operation and maintenance, and manufacturing quality control, see, e.g., the survey papers [1]-[5] and references therein. With the development of sensor technology, massive amounts of time series data are available [6], [7]. Therefore, multivariate time series anomaly detection is crucial.

Various multivariate time series anomaly detection methods based on supervised learning and unsupervised learning have been proposed, see, e.g., the survey papers [8]-[12] and references therein. However, the data in application fields is large, and the proportion of abnormal data is small, therefore the cost of labeling all data is prohibitively high [8], [9]. Hence, unsupervised methods are more suitable than supervised methods. Various unsupervised machine learning methods for time series anomaly detection, such as Principal Component Analysis (PCA), k-Nearest Neighbor (kNN), and Support Vector Machine (SVM) [10]-[12]. However, most existing methods ignore the inherent features of time series data, and thus are difficult to deal with time series data.

With the rapid development of deep learning, various unsupervised time series anomaly detection methods based on deep learning have been developed, to address the above issues [13]-[16]. For example, the authors of [15] and [16] propose unsupervised time series anomaly detection methods based on autoencoder (AE). AE encodes the original data to generate latent variables, which are fed to the decoder to obtain the reconstructed data, and finally realizes AD through the reconstruction error. However, AE lacks of the regularization of latent variables, and thus is difficult to deal with complex time series data. To address this problem, the authors of [17] propose a multivariate time series anomaly detection method based on variational autoencoder (VAE), originally proposed in [18] by combining Bayesian inference with AE, and by adding constraints to latent variables and a random sampling process to improve the capability to deal with complex data.

978-0-7381-4408-5/23/$31.00 ©2023 IEEE

However, it ignores the spatial-temporal correlations of the multivariate time series data.

To extract the temporal correlation, Dis-AE-LSTM [19], TAnoGAN [20] and TadGAN [21] propose time series anomaly detection methods based on GAN [22] and long short-term memory network (LSTM) [23]. There are few studies which consider the temporal correlation and the spatial correlation simultaneously. For example, the authors of [24] propose a multivariate anomaly detection method called MAD-GAN, which uses the entire variable set simultaneously to capture the spatial correlation, and the LSTM Recurrent Neural Networks (LSTM-RNN) to capture temporal correlation. However, it does not delve into the spatial correlation, and the influence of noise on detection is ignored.

To address these issues, we propose a novel unsupervised multivariate time series anomaly detection method called Orthogonal Self-Attention Variational Autoencoder Generative Adversarial Networks (OSVAE-GAN). The proposed method uses two GANs, where one generator is used as the encoder of the VAE to obtain latent variables, and the other generator is used as the decoder of the VAE to reconstruct the original time series. In addition, two discriminators are used to measure the difference between the original and the reconstructed time series as well as the difference between the distribution of latent variables and the prior distribution, respectively. Finally, in order to detect the abnormal multivariate time series data, the anomaly score is calculated based on the output of the discriminator and the reconstruction error.

The main contributions of this work are as follow:

- A novel unsupervised AD method for multivariate time series data is proposed. Compared with Dis-AE-LSTM [19] and LSTM-VAE [25], we use two generators as the encoder and the decoder of the VAE, which has better capability to deal with complex time series data.
- Compared with TAnoGAN [20], TadGAN [21], and MAD-GAN [24], two sets of orthogonal multi-head self-attention networks are used to form an orthogonal self-attention (OS) mechanism, which significantly improves the capability to capture spatial-temporal correlations.
- Compared with TadGAN and MAD-GAN, the maximum mean discrepancy (MMD) [26]-[28] loss is introduced to further constrain latent variables, which improves the robustness to noise.
- Experiments are conducted on five public datasets, i.e., SWaT [29], WADI [30], SMD [31], SMAP [32], MSL [32]. The comparison with LSTM-VAE, TAnoGAN and TadGAN shows that the proposed method outperforms these existing methods.

The remainder of this paper is organized as follows. Section II presents the proposed anomaly detection method OSVAE-GAN. Section III provides the detection performance of the proposed method by various experiments. Section IV concludes the paper.

II. Anomaly Detection with OSVAE-GAN

Given a multivariate time series $\boldsymbol{X} = [x_1, x_2, ..., x_T] \in \mathbb{R}^{d \times T}$, where T is the total length of the time sequence, d is the number of measurements, and $x_i = [x_i^1; x_i^2; ...; x_i^d] \in \mathbb{R}^d$ denotes d measurements at time i. The multivariate time series $\boldsymbol{X}$ is processed through a sliding window with the window size w_t and the step size w_s, which results in a set of samples $X = \{X_{seq}^i\}_{i=1}^N$, where $N = \left\lfloor \frac{T - w_t}{w_s} \right\rfloor$ and $X_{seq}^i \in \mathbb{R}^{d \times w_t}$.

In this section, we first provide the overview of the proposed OSVAE-GAN method. In the training phase, as illustrated in Fig. 1, the pre-processed sample X_{seq} is fed into the OS to extract the hidden spatial-temporal correlations in the original sample, which will be detailed in subsection II-A. Then the output of the OS $\bar{X}_{seq}$ is input to the VAE to obtain the reconstructed sample $\widehat{X}_{seq}$. The discriminator $\mathcal{D}_x$ of the generator $\mathcal{G}$ and the discriminator $\mathcal{D}_z$ of the generator $\mathcal{E}$ are introduced for adversarial training, which will be detailed in subsection II-B. Once the OSVAE-GAN model is trained with normal data, in the testing phase, the anomaly score for a given sample is computed. By comparing with the detection threshold, whether the data is abnormal can be determined, which will be detailed in subsection II-C.

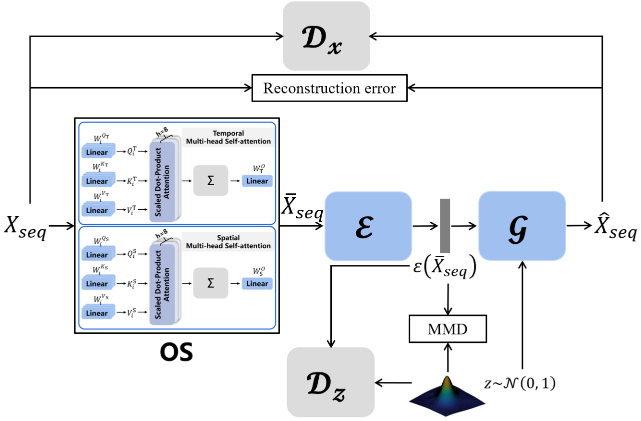

Fig. 1. The overall framework of the proposed OSVAE-GAN.

A. Orthogonal self-attention mechanism

In this subsection, we introduce the OS containing Temporal Multi-head Self-attention and Spatial Multi-head Self-attention to capture temporal and spatial correlations, respectively.

In the seminal work [33], multi-head attention is proposed, which maps its input to multiple different subspaces through the nonlinear transformation, and then uses these subspaces to find the final point in the new space. The essence of multi-head attention is multiple independent attention calculations as an integrated function, which can capture richer feature information, improve expression ability, and prevent overfitting. Therefore, we use multi-head self-attention networks as models for Temporal Multi-head Self-attention and Spatial Multi-head Self-attention.

The processes of Temporal Multi-head Self-attention and Spatial Multi-head Self-attention are illustrated in Fig. 2. For

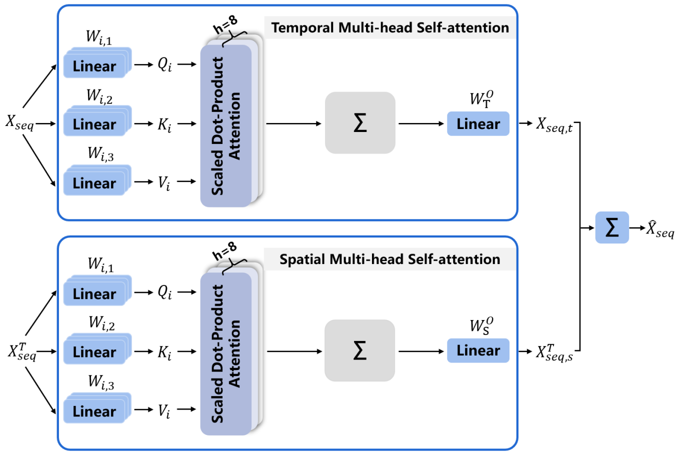

Fig. 2. The OS composed of Temporal Multi-head Self-attention and Spatial Multi-head Self-attention.

example, through Temporal Multi-head Self-attention, we get the sample $X_{seq,t}$ containing the temporal correlation by

$$X_{seq,t} = \mathrm{MultiHead_T}(X_{seq}) = \left(\sum_{i=1}^{h} head_i\right) W_{\mathrm{T}}^O, \quad (1)$$

where $W_{\mathrm{T}}^O \in \mathbb{R}^{d_v \times d_{in}}$ is a learnable parameter matrix, h is the number of heads for the multi-head self-attention, $d_v = d_{in}$, d_{in} is the number of columns of the input, which is equal to w_t for the temporal case, and $head_i$ is calculated by

$$head_i = \mathrm{Attention}(Q_i W_{i,1}, K_i W_{i,2}, V_i W_{i,3}), \quad (2)$$

where Attention indicates that the input sample is processed using a scaled dot-product attention

$$\mathrm{Attention}(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) V, \quad (3)$$

where $softmax$ is the activate function, $W_{i,1}$, $W_{i,2} \in \mathbb{R}^{d_{in} \times d_{in}}$, $W_{i,3} \in \mathbb{R}^{d_{in} \times d_v}$ are learnable parameter matrices, and

$$Q_i = X_{seq} W_{i,1}, \;\; K_i = X_{seq} W_{i,2}, \;\; V_i = X_{seq} W_{i,3}. \quad (4)$$

For Spatial Multi-head Self-attention, the input sample X_{seq} is first transposed, and then follows the process similar to Temporal Multi-head Self-attention to obtain $X_{seq,s}$ containing the spatial correlation. Due to the space limitation, we have omitted the details.

Finally, $X_{seq,t}$ and $X_{seq,s}$ respectively contains the temporal correlation and the spatial correlation are added as

$$\bar{X}_{seq} = X_{seq,t} + X_{seq,s}^T. \quad (5)$$

B. Adversarial learning

In this subsection, we introduce the proposed adversarial learning strategy based on VAE, GANs and MMD. The process of the adversarial learning is as follows. As illustrated in Fig. 1, the output of the OS $\bar{X}_{seq}$ is input to the encoder $\mathcal{E}$ of the VAE to obtain latent variables $\mathcal{E}(\bar{X}_{seq})$. Then, the decoder $\mathcal{G}$ is used to obtain the reconstructed sample $\widehat{X}_{seq}$. In order to make the reconstruction of the VAE more robust, the discriminator $\mathcal{D}_x$ of the generator $\mathcal{G}$ is introduced to play an adversarial game, forcing the reconstructed sample $\widehat{X}_{seq}$ as close to the original sample X_{seq} as possible. Moreover, to deal with complex multivariate data, the discriminator $\mathcal{D}_z$ of the generator $\mathcal{E}$ is introduced to play an adversarial game, which makes latent variables and the data follow from the prior standard normal distribution indistinguishable. Finally, in order to mitigate the influence of noise, the MMD is introduced to measure the difference between the latent variable distribution and the prior distribution.

In order to train the OSVAE-GAN, we need to solve an optimization problem to be specified in Eq. (11), whose objective function contains the loss of the VAE, objective functions of two GANs, and the loss of the MMD.

The loss function of the VAE consists of the reconstruction error and the Kullback-Leibler (KL) divergence, given by

$$\begin{aligned} L_{\mathrm{VAE}}(\mathcal{E}, \mathcal{G}) = \mathbf{E}_{X_{seq} \sim \mathrm{P}_x}\left[\left\|X_{seq} - \mathcal{G}(\mathcal{E}(\bar{X}_{seq}))\right\|_2\right] \\ + \beta D_{KL}(\mathrm{P}_{\mathcal{E}} || \mathrm{P}_z), \end{aligned} \quad (6)$$

where P_x is the distribution of the input sample X_{seq}, $\mathrm{P}_{\mathcal{E}}$ is the distribution of latent variables $\mathcal{E}(\bar{X}_{seq})$, P_z is the standard normal distribution, β is a trade-off parameter, and

$$D_{KL}(P||Q) = \sum_{a \in \Omega} P(a) \log \frac{P(a)}{Q(a)},$$

where Ω is the sample space of the distributions of P and Q.

For the decoder $\mathcal{G}$ and its discriminator $\mathcal{D}_x$, the objective is

$$\min_{\mathcal{G}} \max_{\mathcal{D}_x \in \mathbf{D}_x} V_X(\mathcal{D}_x, \mathcal{G}),$$

with

$$\begin{aligned} V_X(\mathcal{D}_x, \mathcal{G}) \;&= \mathbf{E}_{X_{seq} \sim \mathrm{P}_x}[\mathcal{D}_x(X_{seq})] - \mathbf{E}_{z \sim \mathrm{P}_z}[\mathcal{D}_x(\mathcal{G}(z))] \\ &\quad + \lambda_1 \mathbf{E}_{z \sim \mathrm{P}_z}\left[(\|\nabla_z \mathcal{D}_x(\mathcal{G}(z))\|_2 - 1)^2\right], \end{aligned} \quad (7)$$

where $\mathbf{D}_x$ is a set of Lipschitz continuous functions with Lipschitz constant being 1, i.e., $\|f(a) - f(b)\| \leq \|a - b\|, \forall a, b \in dom\ f$, and $dom\ f$ refers to the domain of the function f. In addition, the gradient penalty term penalizes the gradient that is not equal to 1 [34], and λ_1 is a trade-off parameter.

Similarly, for the encoder $\mathcal{E}$ and its discriminator $\mathcal{D}_z$, the objective is

$$\min_{\mathcal{E}} \max_{\mathcal{D}_z \in \mathbf{D}_z} V_Z(\mathcal{D}_z, \mathcal{E}),$$

with

$$\begin{aligned} V_Z(\mathcal{D}_z, \mathcal{E}) = \mathbf{E}_{z \sim \mathrm{P}_z}[\mathcal{D}_z(z)] - \mathbf{E}_{X_{seq} \sim \mathrm{P}_x}[\mathcal{D}_z(\mathcal{E}(\bar{X}_{seq}))] \\ + \lambda_2 \mathbf{E}_{X_{seq} \sim \mathrm{P}_x}\left[(\|\nabla_{\bar{X}_{seq}} \mathcal{D}_z(\mathcal{E}(\bar{X}_{seq}))\|_2 - 1)^2\right]. \end{aligned} \quad (8)$$

The loss of the MMD is given by

$$\begin{aligned} L_{\mathrm{MMD}}(\mathrm{P}_z, \mathrm{P}_{\mathcal{E}}) = \Big\| \mathbf{E}_{z \sim \mathrm{P}_z}[\Phi(z)] \\ - \mathbf{E}_{\mathcal{E}(\bar{X}_{seq}) \sim \mathrm{P}_{\mathcal{E}}}[\Phi(\mathcal{E}(\bar{X}_{seq}))] \Big\|_{\mathcal{H}}^2, \end{aligned} \quad (9)$$

where $\mathcal{H}$ denotes the reproducing kernel Hilbert space (RKHS), and $\Phi(\cdot)$ is a function that maps the data to a RKHS.

Finally, by combining all objective functions given in Eqs. (6)-(9), the overall objective function is

$$\begin{aligned} V_F(\mathcal{D}_x, \mathcal{G}, \mathcal{D}_z, \mathcal{E}) \; &= L_{\mathrm{VAE}}(\mathcal{E}, \mathcal{G}) + V_X(\mathcal{D}_x, \mathcal{G}) \\ &\quad + V_Z(\mathcal{D}_z, \mathcal{E}) + L_{\mathrm{MMD}}(\mathrm{P}_z, \mathrm{P}_{\mathcal{E}}), \end{aligned} \tag{10}$$

and the overall objective is

$$\min_{\{\mathcal{E}, \mathcal{G}\}} \max_{\{\mathcal{D}_x \in \mathbf{D}_x, \mathcal{D}_z \in \mathbf{D}_z\}} V_F(\mathcal{D}_x, \mathcal{G}, \mathcal{D}_z, \mathcal{E}). \tag{11}$$

A pseudo code of the above procedure is summarized in Algorithm 1, where we have used X_i to denote X_{seq}^i for the notational simplification.

Algorithm 1 OSVAE-GAN.

```
Require: $n$, batch size.
         $epoch$, number of iterations over the data.
         $n_{disc}$, number of iterations of the discriminator per epoch.
         $\eta$, learning rate.
for each $epoch$ do
    for each $k = 0, ..., n_{disc}$ do
        Sample $\{X_i\}_{i=1}^n$ from the input sample $X$.
        Sample $\{z_i\}_{i=1}^n$ from standard normal distribution.
        $\bar{X}_i = \mathrm{MultiHead}(X_i)$.
        $g_{\mathcal{D}_x} = \nabla_{w_{\mathcal{D}_x}} (\frac{1}{n}\sum_{i=1}^n \mathcal{D}_x(X_i) - \frac{1}{n}\sum_{i=1}^n \mathcal{D}_x(\mathcal{G}(z_i)))$
                $+ \lambda_1(\|\nabla_z \mathcal{D}_x(\mathcal{G}(z_i))\|_2 - 1)^2$
        $w_{\mathcal{D}_x} = w_{\mathcal{D}_x} + \eta \cdot \mathrm{adam}(w_{\mathcal{D}_x}, g_{\mathcal{D}_x})$
        $g_{\mathcal{D}_z} = \nabla_{w_{\mathcal{D}_z}} (\frac{1}{n}\sum_{i=1}^n \mathcal{D}_z(z_i) - \frac{1}{n}\sum_{i=1}^n \mathcal{D}_z(\mathcal{E}(\bar{X}_i)))$
                $+ \lambda_2(\|\nabla_{\bar{x}_i} \mathcal{D}_z(\mathcal{E}(\bar{X}_i))\|_2 - 1)^2$
        $w_{\mathcal{D}_z} = w_{\mathcal{D}_z} + \eta \cdot \mathrm{adam}(w_{\mathcal{D}_z}, g_{\mathcal{D}_z})$
    end for
    Sample $\{X_i\}_{i=1}^n$ from the input sample $X$.
    Sample $\{z_i\}_{i=1}^n$ from standard normal distribution.
    $g_{w_{\mathcal{G},\mathcal{E}}} = \nabla_{w_{\mathcal{G}}, w_{\mathcal{E}}} \big(\frac{1}{n}\sum_{i=1}^n \mathcal{D}_x(X_i) - \frac{1}{n}\sum_{i=1}^n \mathcal{D}_x(\mathcal{G}(z_i))$
            $+ \frac{1}{n}\sum_{i=1}^n \mathcal{D}_z(z_i) - \frac{1}{n}\sum_{i=1}^n \mathcal{D}_z(\mathcal{E}(\bar{X}_i))$
            $+ \big\|\frac{1}{n}\sum_{i=1}^n \Phi(z_i) - \frac{1}{n}\sum_{i=1}^n \Phi(\mathcal{E}(\bar{X}_i))\big\|_{\mathcal{H}}^2$
            $+ \big\|\frac{1}{n}\sum_{i=1}^n X_i - \frac{1}{n}\sum_{i=1}^n \mathcal{G}(\mathcal{E}(\bar{X}_i))\big\|_2 \big)$
    $w_{\mathcal{G},\mathcal{E}} = w_{\mathcal{G},\mathcal{E}} + \eta \cdot \mathrm{adam}(w_{\mathcal{G},\mathcal{E}}, g_{w_{\mathcal{G},\mathcal{E}}})$
end for
```

C. Anomaly score

Our proposed anomaly score is motivated by [21], which is calculate by the reconstruction error and the output of the discriminator $\mathcal{D}_x$.

To calculate the reconstruction error between the original sample X_{seq} and the reconstructed sample $\widehat{X}_{seq}$, we use the root-mean square error (RMSE) given by

$$L_R(X_{seq}) = \sqrt{\frac{1}{d \cdot w_t} \sum_{i=1}^d \sum_{j=1}^{w_t} (x_{ij} - \widehat{x}_{ij})^2}, \tag{12}$$

where x_{ij} and $\widehat{x}_{ij}$ are the elements at the corresponding positions of X_{seq} and $\widehat{X}_{seq}$, respectively.

The output of the discriminator $\mathcal{D}_x$ is given by

$$L_{\mathcal{D}_x}(X_{seq}) = \mathcal{D}_x(X_{seq}). \tag{13}$$

A larger reconstruction error indicates a higher probability that the X_{seq} is abnormal. However, a smaller output of the discriminator $\mathcal{D}_x$ indicates a higher probability that the X_{seq} is abnormal. Therefore, the reconstruction error $L_R(X_{seq})$ and the discriminator output $L_{\mathcal{D}_x}(X_{seq})$ cannot be calculated directly to obtain the anomaly score without processing. We apply a z-score transformation to normalize $L_R(X_{seq})$ and $L_{\mathcal{D}_x}(X_{seq})$ into distributions $Z_{L_R}(X_{seq})$ and $Z_{D_x}(X_{seq})$ with the standard normal distribution. Their multiplication leads to the anomaly score $A(X_{seq})$, given by

$$A(X_{seq}) = Z_{L_R}(X_{seq}) Z_{D_x}(X_{seq}). \tag{14}$$

During the testing phase, a given sample is abnormal if the anomaly score is greater than the threshold. There are two common methods to choose the appropriate threshold. When the input samples are relatively consistent with the normal distribution, the detection threshold can be determined by using the 3$\boldsymbol{\sigma}$ rule [35]. The detection threshold can also be determined according to the abnormal rate of the verification dataset (a subset of the testing dataset) [36]. We adapt these two methods and choose the better one in the experiment.

III. Experimental Evaluation

In this section, we evaluate the proposed method and compare with the existing methods, such as LSTM-VAE [25], TAnoGAN [20], and TadGAN [21], on five public datasets, i.e., SWaT [29], WADI [30], SMD [31], SMAP [32], MSL [32]. First, the description of datasets and experiment settings are introduced in subsection III-A. Next, experimental results and ablation study are given in subsection III-B.

A. Datasets description and experiment settings

1) Datasets description: The five publicly datasets are as listed in Table I. The Mars Science Laboratory Rover (MSL) dataset and the Soil Moisture Active Passive (SMAP) satellite dataset are telemetry data provided by two different spacecrafts. The Secure Water Treatment (SWaT) dataset is collected at real water treatment plants. The Water Distribution (WADI) dataset consists of data collected under normal conditions and under attack scenarios in a water distribution testbed. The Server Machine Dataset (SMD) collects the resource utilization in a computer cluster.

2) Experiment settings: PyTorch framework is utilized and all experiments are performed on a GeForce RTX 3090 graphics card. In addition, the network architecture is detailed in Table II. Both the encoder $\mathcal{E}$ and the decoder $\mathcal{G}$ consist of a 3-layer bidirectional LSTM network and a fully connected layer. The discriminator $\mathcal{D}_x$ of the decoder $\mathcal{G}$ consists of three fully connected layers. The discriminator $\mathcal{D}_z$ of the encoder $\mathcal{E}$ consists of two fully connected layers. The number of heads in

TABLE I
DATASETS DESCRIPTION.

Dataset name	Training set size	Testing set size	Number of dimensions
MSL	58317	73729	55
SMAP	135183	427617	25
SWaT	3000	5000	1
WADI	40000	17281	127
SMD	708405	708420	38

two multi-head self-attention networks is 8. For the proposed OSVAE-GAN, the batch size is 32, the number of epoch is 100, and the learning rate is 10^{-4}. We set the window sizes of SMAP, WADI, SMD, MSL, and SWaT are 50, 50, 80, 100, and 200, respectively. We set the time steps of the sliding window are $\lfloor \frac{w_t}{2} \rfloor$. We use F1-Score (F1) to evaluate the performance of our proposed method and the existing methods, which is given by $\mathrm{F1} = \frac{2 \times \mathrm{Preision} \times \mathrm{Recall}}{\mathrm{Preision} + \mathrm{Recall}}$, where $\mathrm{Preision} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$. A higher F1 score indicates a better performance.

TABLE II
ARCHITECTURE OF NETWORKS.

Name of Networks	Layer	Parameters
OS	FC	-
	softmax	-
The encoder $\mathcal{E}$	LSTM	num_layers=3
The decoder $\mathcal{G}$		bidirectional=True
		batch_first=True
	FC	-
	LeakyReLU	-
	Dropout	-
The discriminator $\mathcal{D}_x$	FC	-
	LeakyReLU	-
	Dropout	-
The discriminator $\mathcal{D}_z$	FC	-
	LeakyReLU	-
	Dropout	-

B. Experimental results

The experimental results of the proposed OSVAE-GAN method and the existing methods are listed in Table III, which shows that the proposed method outperforms these existing methods by having the highest averaged F1 score (97.71%) across all the datasets. The averaged F1 score of LSTM-VAE is 95.43%. LSTM-VAE is an anomaly detection method based on the VAE, which uses the LSTM as the encoder and the decoder. However, LSTM-VAE lacks any auxiliary network to strengthen the reconstruction process, as well as additional restrictions on latent variables, which may lead to its poor performance when dealing with sparse or imbalanced time series data. The average F1 score of TAnoGAN is 85.83%, which is much lower than other methods in our experiments. TAnoGAN is also an anomaly detection method based on the GAN, which uses a discriminator and two generators. TAnoGAN does not utilize VAE for feature extraction, but instead a fully connected layer (FC), which may lead to the worse performance when dealing with complex multivariate time series data. The average F1 score of TadGAN method is 95.25%. TadGAN is also an anomaly detection method based on two GANs, which only uses bidirectional LSTM to extract the temporal correlation. In addition, TadGAN has two generators as the AE and constrains latent variables only through the discriminator, which could result in poor robustness to noise.

TABLE III
COMPARISON RESULTS OF THE PROPOSED METHOD AND EXISTING METHODS.

Datasets Methods	MSL	SMAP	SWaT	WADI	SMD	Avg.
LSTM-VAE	94.00	92.74	95.09	97.80	97.53	95.43
TAnoGAN	87.50	85.25	70.59	92.14	93.65	85.83
TadGAN	94.44	93.17	93.08	97.77	97.81	95.25
OSVAE-GAN	**97.91**	**96.55**	**97.16**	**98.61**	**98.33**	**97.71**

TABLE IV
RESULTS OF THE ABLATION STUDY FOR OSVAE-GAN.

Datasets Methods	MSL	SMAP	SWaT	WADI	SMD	Avg.
OSVAE-GAN without OS and MMD	94.10	91.96	97.30	97.43	97.35	95.63
OSVAE-GAN without OS	94.14	93.08	97.10	97.18	96.87	95.67
OSVAE-GAN	**97.91**	**96.55**	97.16	**98.61**	**98.33**	**97.71**

To verify the roles and contributions of the two key modules, i.e., the OS and the MMD in extracting spatial-temporal correlations and improving the robustness to noise in the OSVAE-GAN method, we conduct the ablation study. The results of ablation experiments are listed in Table IV. The average F1 score of OSVAE-GAN without the OS is 95.67%, which is lower than the average F1 score of OSVAE-GAN. This shows that the OS indeed contributes to the extract spatial-temporal correlations and improves detection performance. The average F1 score of OSVAE-GAN without the OS and the MMD is 95.63%, which is lower than the average F1 score of OSVAE-GAN without the OS and the average F1 score of OSVAE-GAN. This shows that the MMD improves the robustness to noise and improves detection performance. As shown in Table IV, for the SWaT dataset, the F1 score of the OSVAE-GAN is slightly lower than the OSVAE-GAN without these two modules. A possible reason for this is that the SWaT dataset is univariate time series data, which is relatively simple.

IV. Conclusions

In this paper, we propose an unsupervised anomaly detection method called OSVAE-GAN. First, we introduce the OS to fully extract the spatial-temporal correlations. Moreover, we integrate two GANs with the VAE to deal with complex multivariate time series data. Finally, we use the MMD loss to reduce the influence of noise. Experiments are conducted on five public datasets and the results shows that the proposed method outperforms the existing methods.

References

[1] V. Chandola, A. Banerjee and V. Kumar, “Anomaly detection: A survey”, *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.

[2] S. Agrawal and J. Agrawal, “Survey on anomaly detection using data mining techniques”, *Procedia Computer Science*, vol. 60, pp. 708–713, 2015.

[3] R. Chalapathy and S. Chawla, “Deep learning for anomaly detection: A survey”, *arXiv preprint arXiv:1901.03407*, 2019.

[4] G. Pang, C. Shen, L. Cao and A. V. D. Hengel, “Deep learning for anomaly detection: A review”, *ACM Computing Surveys*, vol. 54, no. 2, pp. 1–38, 2021.

[5] L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich and K. R. Müller, “A unifying review of deep and shallow anomaly detection”, *Proceedings of the IEEE*, vol. 109, no. 5, pp. 756–795, 2021.

[6] M. Ahmed, A. N. Mahmood and J. Hu, “A survey of network anomaly detection techniques”, *Journal of Network and Computer Applications*, vol. 60, pp. 19–31, 2016.

[7] J. Li, W. Pedrycz and I. Jamal, “Multivariate time series anomaly detection: A framework of Hidden Markov Models”, *Applied Soft Computing*, vol. 60, pp. 229–240, 2017.

[8] D. Guthrie, L. Guthrie, B. Allison and Y. Wilks, “Unsupervised anomaly detection”, in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 1624–1628, 2007.

[9] J. Audibert, P. Michiardi, P. Guyard, S. Marti and M. A. Zuluaga, “USAD: Unsupervised anomaly detection on multivariate time series”, in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3395–3404, 2020.

[10] S. Eltanbouly, M. Bashendy, N. AlNaimi, Z. Chkirbene and A. Erbad, “Machine learning techniques for network anomaly detection: A survey”, in *Proceedings of the 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies*, pp. 156–162, 2020.

[11] Y. Luo, Y. Xiao, L. Cheng, G. Peng and D. Yao, “Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities”, *ACM Computing Surveys*, vol. 54, no. 5, pp. 1–36, 2021.

[12] G. Li and J. J. Jung, “Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges”, *Information Fusion*, vol. 9, pp. 78658–78700, 2021.

[13] G. Pang, L. Cao and C. Aggarwal, “Deep learning for anomaly detection: Challenges, methods, and opportunities”, in *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pp. 1127–1130, 2021.

[14] U. A. Usmani, A. Happonen and J. Watada, “A review of unsupervised machine learning frameworks for anomaly detection in industrial applications”, *Science and Information Conference*, pp. 158–189, 2022.

[15] T. Kieu, B. Yang, C. Guo and C. S. Jensen, “Outlier detection for time series with recurrent autoencoder ensembles”, in *Proceedings of the 28th International Joint Conferenceon Artificial Intelligence*, pp. 2725–2732, 2019.

[16] C. Zhang, D. Song, Y. Chen, X. Feng, C. Lumezanu, W. Cheng, J. Ni, B. Zong, H. Chen and N. V. Chawla, “A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data”, in *Proceedings of the AAAI conference on Artificial Intelligence*, vol. 33, no. 1, pp. 1409–1416, 2019.

[17] L. Li, J. Yan, J. Wang and Y. Jin, “Anomaly detection of time series with smoothness-inducing sequential variational auto-encoder”, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 3, pp. 1177–1191, 2020.

[18] D. P. Kingma and M. Welling, “Auto-encoding variational bayes”, in *Proceedings of the 2nd International Conference on Learning Representations*, 2014.

[19] S. Mao, J. Guo, T. Gu and Z. Ma, “Dis-AE-LSTM: Generative adversarial networks for anomaly detection of time series data”, in *Proceedings of the 2020 International Conference on Artificial Intelligence and Computer Engineering*, pp. 330–336, 2020.

[20] M. A. Bashar and R. Nayak, “TAnoGAN: Time series anomaly detection with generative adversarial networks”, in *Proceedings of the 2020 IEEE Symposium Series on Computational Intelligence*, pp. 1778–1785, 2020.

[21] A. Geiger, D. Liu, S. Alnegheimish, A. Cuesta-Infante and K. Veeramachaneni, “TadGAN: Time series anomaly detection using generative adversarial networks”, in *Proceedings of the 2020 IEEE International Conference on Big Data*, pp. 33–34, 2020.

[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, “Generative adversarial nets”, in *Proceedings of the 2014 Advances in Neural Information Processing Systems*, pp. 139–144, vol. 27, 2014.

[23] S. Hochreiter and J. Schmidhuber, “Long short-term memory”, *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[24] 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 *Proceedings of the 2019 International Conference on Artificial Neural Networks*, pp. 703–716, 2019.

[25] D. Park, Y. Hoshi and C. C. Kemp, “A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder”, *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1544–1551, 2018.

[26] T. Yang, X. Yu, N. Ma, Y. Zhao and H. Li, “A novel domain adaptive deep recurrent network for multivariate time series prediction”, *Engineering Applications of Artificial Intelligence*, vol. 106, p. 104498, 2021.

[27] K. M. Borgwardt, A. Gretton, M. J. Rasch, H. P. Kriegel, B. Schölkopf and A. J. Smola, “Integrating structured biological data by kernel maximum mean discrepancy”, *Bioinformatics*, vol. 22, no. 14, pp. e49–e57, 2006.

[28] M. Long, Y. Cao, J. Wang and M. Jordan, ‘Learning transferable features with deep adaptation networks”, in *Proceedings of the 32nd International Conference on Machine Learning*, pp. 97–105, 2015.

[29] A. P. Mathur and N. O. Tippenhauer, “SWaT: A water treatment testbed for research and training on ICS security”, in *Proceedings of the 2016 International Workshop on Cyber-Physical Systems for Smart Water Networks*, pp. 31–36, 2016.

[30] C. M. Ahmed, V. R. Palleti and A. P Mathur, “WADI: A water distribution testbed for research in the design of secure cyber physical systems”, in *Proceedings of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks*, pp. 25–28, 2017.

[31] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun and D. Pei, “Robust anomaly detection for multivariate time series through stochastic recurrent neural network”, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2828–2837, 2019.

[32] K. Hundman, V. Constantinou, C. Laporte, I. Colwell and T. Soderstrom, “Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding”, in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 387–395, 2018.

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N Gomez, Ł. Kaiser and I. Polosukhin, “Attention is all you need”, in *Proceedings of the 31st Advances in Neural Information Processing Systems*, pp. 6000–6010, 2017.

[34] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. C Courville, “Improved training of wasserstein GANs”, *Proceedings of the 31st Advances in Neural Information Processing Systems*, pp. 5769–5779, 2017.

[35] A. Blázquez-García, A. Conde, U. Mori and J. A. Lozano, “A review on outlier/anomaly detection in time series data”, *ACM Computing Surveys*, vol. 54, no. 3, pp. 1–13, 2021.

[36] J. An and S. Cho, “Variational autoencoder based anomaly detection using reconstruction probability”, *Special Lecture on IE*, vol. 2, no. 1, pp. 1–18, 2015.