STAD: Multivariate Time Series Anomaly Detection Based on Spatio-Temporal Relationship

Keyu Chen[1], Guoping Zhao[2(✉)], Zhenfeng Yao[2], and Zhihong Zhang[1]

[1] School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou, China

[2] Esunny Information Technology, Zhengzhou Commodity Exchange, Zhengzhou, China

zhaoguoping@esunny.cc

Abstract. Anomaly detection for multivariate time series is a very complex problem that requires models not only to accurately identify anomalies, but also to provide explanations for the detected anomalies. However, the majority of existing models focus solely on the temporal relationships of multivariate time series, while ignoring the spatial relationships among them, which leads to the decrease of detection accuracy and the defects of anomaly interpretation. To address these limitations, we propose a novel model, named spatio-temporal relationship anomaly detection (STAD). This model employs a novel graph structure learning strategy to discover spatial features among multivariate time series. Specifically, Graph Attention Networks (GAT) and graph structure are used to integrate each time series with its neighboring series. The temporal features of multivariate time series are jointly modeled by using Transformers. Furthermore, we incorporate an anomaly amplification strategy to enhance the detection of anomalies. Experimental results on four public datasets demonstrate the superiority of our proposed model in terms of anomaly detection and interpretation.

Keywords: Anomaly Detection · Multivariate Time Series · GAT · Transformers · Spatio-Temporal Relationship · Graph Structure Learning

1 Introduction

Anomaly detection for multivariate time series has emerged as a prominent research topic in recent times. In the areas of production and IT systems, time series data can directly reflect the working status and operating conditions of the system, which is an important basis for anomaly detection. In the past, domain experts usually utilize their expertise to establish thresholds for each indicator based on empirical observations. However, with the unprecedented explosion

X. Yang et al. (Eds.): ADMA 2023, LNAI 14176, pp. 73–87, 2023.
https://doi.org/10.1007/978-3-031-46661-8_6

in data complexity and scale due to rapid technological advancements, traditional techniques have become insufficient to effectively address the challenges posed by anomaly detection. To tackle this problem, a lot of unsupervised methods based on classical machine learning have been developed over the previous years, including density estimation-based methods [6] and distance-based methods [3,14]. Nevertheless, these approaches fail to capture the intricate and high-dimensional relationships that exist among time series.

Recently, Methods based on deep learning have contributed to the enhancement of anomaly detection for multivariate time series. For example, AutoEncoders (AE) [5], VAE [12], GAN [17], and Transformers [23] are recent popular anomaly detection methods that employ sequence reconstruction to encode time series data. In addition, Long Short-Term Memory (LSTM) networks [9] and Recurrent Neural Networks (RNN) [22] have also displayed promising results for detecting anomalies in multivariate time series. However, most of these methods fail to consider the association between various time series, moreover, they do not offer a clear explanation of which time series are correlated with each other, thus impeding the interpretation of detected anomalies. A complex set of multivariate time series are often intrinsically linked to each other.

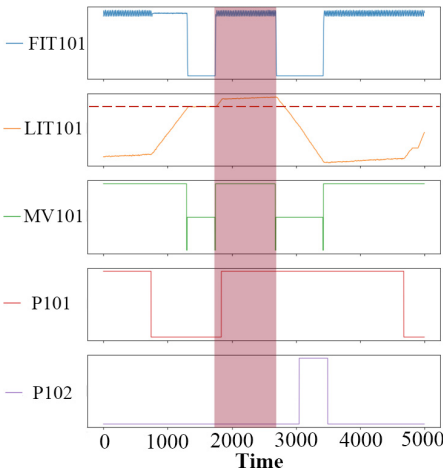

Fig. 1. Multivariate time series segments from the SWaT dataset, with anomalies shaded in red. (Color figure online)

In Fig. 1, the time series are obtained from five sensors of the same process at the SWaT water treatment testbed [16]. The red shaded region corresponds to an anomaly, indicating that the LIT101 value has exceeded the threshold. In addition, the readings of FIT101, MV101, and P101 all changed during this period, P102 changed after the anomaly has ended. Based on the fault log, we know that LIT101 serves as the level transmitter responsible for measuring the water level of the tank, and the anomalous segment corresponds to the overflow of the tank.

The fundamental reason of this anomaly is because of the premature opening of MV101 (inlet valve) in the same process. Given that the state of MV101 is limited to only two possibilities (open and closed) and irregular, identifying abnormal for it through the temporal features is a challenging task. Consequently, integrating spatial features become crucial to detect and explain the anomaly. Several methods have employed graph neural networks for anomaly detection because of its remarkable capability to leverage spatial structural information, such as MTAD-GAT [26] and GDN [8]. However, MTAD-GAT assumes a complete graph structure for the spatial characteristics of multivariate time series, which may not accurately reflect their asymmetric correlations in real-world scenarios. GDN [8] is limited to a single time point and fails to catch the detailed associations between a time point and a whole sequence. GTA [7] combines graph structures for spatial feature learning and Transformers for temporal modeling. However, it utilizes Gumbel-Softmax, which is insufficient in accurately representing the spatial relationships among multivariate time series.

This paper presents a method for anomaly detection by leveraging the spatio-temporal relationships among multivariate time series. The proposed approach leverages the joint optimization of Graph Attention Networks (GAT) and Transformers for unsupervised anomaly detection. In order to explore complex temporal and spatial dependencies among diverse time series, a novel graph structure learning strategy is proposed, which considers multivariate time series as separate nodes and learns attention weights of each node to obtain a bidirectional graph structure. The proposed method employs GAT and graph structure to integrate information of nodes with their neighbors, while the temporal features of time series are modeled utilizing Transformers. The utilization of Transformers in the proposed approach is motivated by their capability to capture long-term dependencies, compute global dependencies, and enable efficient parallel computation. To further enhance the detection performance, an anomaly amplification strategy based on local and global differences is also introduced. In summary, this paper makes the following major contributions:

- We propose a new method for learning the graph structure in multivariate time series.
- We propose an novel method for multivariate time series anomaly detection, which efficiently captures spatio-temporal information using GAT and Transformers.
- Extensive experiments are conducted on four popular datasets to demonstrate the effectiveness of our proposed method. And, ablation studies are conducted to understand the impact of each component in our architecture.

2 Related Work

2.1 Traditional Anomaly Detection for Multivariate Time Series

Traditional methods for time series anomaly detection typically contain distance-based methods and clustering-based methods. LOF (Local Outlier Factor) [6] is

a density-based method, which determines the degree of anomaly by comparing the local density between each data point and its surrounding neighboring data points. KNN [3] is a distance-based outlier detection method, which detects anomalies by calculating the distances between each data point and its K nearest neighbors. IsolationForest [14] uses a tree structure to decompose data and quantifies the distances between nodes to identify outliers. Traditional unsupervised methods for anomaly detection are limited in their ability to identify anomalies, as they do not take into account the spatio-temporal relationships inherent in the data.

2.2 Deep Learning Anomaly Detection for Multivariate Time Series

Prediction-Based Models: LSTMNDT [10] leverages LSTM [9] network to predict time series collected from spacecraft, but it ignores the spatial correlations. MTAD-GAT [26] employs two GAT layers to model the spatio-temporal relationships simultaneously, but MTAD-GAT assumes that the spatial characteristics of multivariate time series are a complete graph, in most cases, time series are typically associated in an asymmetric manner. GDN [8] uses node embedding for graph structure learning, encodes spatial information using GAT. However, GDN is limited to a single time point and cannot catch the detailed associations between a time point and a whole sequence.

Reconstruction-Based Models: LSTM-VAE [19] utilizes a LSTM network and a variational autoencoder (VAE) [12] for the reconstruction of time series. DAGMM [27] combines a deep autoencoder with Gaussian Mixture Model. But the Gaussian Mixture Model is not suitable for complex distributed datasets. OmniAnomaly [22] employs a new stochastic RNN based on the LSTM-VAE model for anomaly detection. GANS [13,17,20] uses generators for reconstruction. Anomaly transformer [25] leverages a prior-association and series-association and compares them to better identify anomalies. USAD [4] uses a deep autoencoder trained with adversarial training to learn and detect anomalies in new data. However, all the methods mentioned above only take into account either temporal or spatial associations, without learning both associations, and they may lack sufficient ability to accurately localize anomalies. GTA [7] combines graph structures for spatial feature learning and Transformers for temporal modeling. However, it utilizes Gumbel-Softmax, which is insufficient in accurately representing the spatial relationships between multivariate time series.

3 Method

In this section, we give the details of the proposed spatio-temporal relationship anomaly detection (STAD) for multivariate time series. At first, we present the problem statement and the overall architecture of STAD. Next, we will elaborate the particulars of the graph structure learning, the GAT-based spatial model, Transformers and anomaly amplification modules.

3.1 Problem Statement

In our study, time series is represented by $\{\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_d\}$. For the time series i, $\mathbf{X}_i = [x_{1i}, x_{2i}, \cdots, x_{Ni}]$, where $\mathbf{X}_i \in \mathbb{R}^N$ denotes the observed value of time series i. N is the length of $\mathbf{X}$ and d is the number of multivariate time series. Our goal is to model multivariate time series data in order to identify any anomalous behaviors.

3.2 Overview

The overall architecture of the model in this paper is shown in Fig. 2. It consists of three main components:

(1) Graph Structure Learning: Learn a graph structure that represents spatial relationships between multivariate time series.
(2) GAT-based spatial model: Fusing time series with spatial features using GAT and graph structure.
(3) Transformers based on anomaly amplification strategy: Transformers are used to reconstruct the spatio-temporal relationships of each time series. Anomaly amplification strategy is used to amplify anomalies.

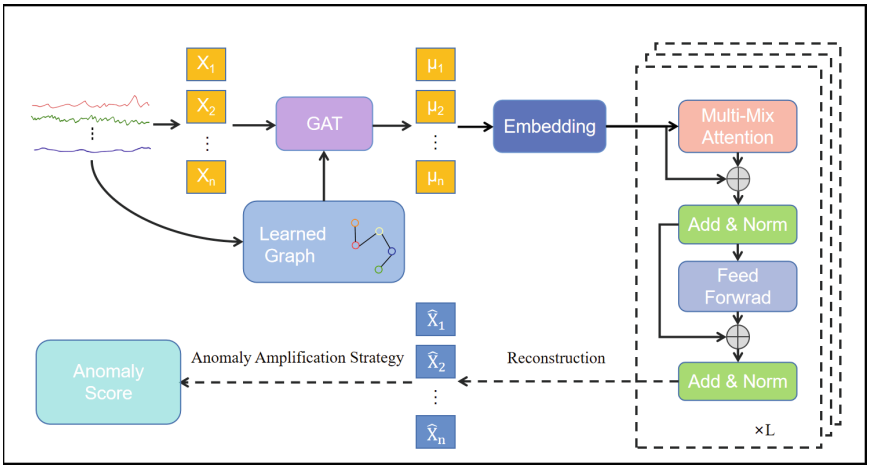

Fig. 2. An overview of the proposed STAD method.

3.3 Graph Structure Learning

For our model, the primary task is to reconstruct spatial and temporal relationships for multivariate time series. For spatial modeling, we utilize a learnable

graph to represent the relationships between multivariate time series. We consider each time series as a node, and the relationships between the time series are represented as edges in the graph. An adjacency matrix $A \in \mathbb{R}^{d \times d}$ is used to express this graph, where A_{ij} denotes that there are edges between node i to node j. Our proposed framework has flexibility and can automatically learn the relationships of the graph without prior knowledge about the graph structure. In order to obtain the hidden dependencies between nodes, we designed a framework that, unlike the GDN [8], does not use the node embedding learning graph structure. We learn a weight matrix that assigns a weight score to each node based on its own features and similarity to other nodes, and then use top k to filter the most relevant sets for the graph structure:

$$e_{ij} = \text{LeakyReLU}\left(\mathbf{w}^{\mathrm{T}} \cdot (\mathbf{X}_i \oplus \mathbf{X}_j)\right) \tag{1}$$

$$\rho_{ij} = \frac{\exp\left(e_{ij}\right)}{\sum_{k=1}^{d} \exp\left(e_{ik}\right)} \tag{2}$$

$$A_{ij} = 1\Big\{ j \in \text{TopK}\left(\left\{\rho_{\text{ik}} : \text{k} \in \text{C}_\text{i}\right\}\right) \Big\} \tag{3}$$

where $\oplus$ stands for stitching two nodes together. $\mathbf{X}_i \in \mathbb{R}^N$ is the feature vector of node i, $\mathbf{w} \in \mathbb{R}^{2N}$ is a learnable parameter vector, LeakyReLU is a nonlinear activation function, $\rho_{ij} \in \mathbb{R}^{d \times d}$ is the weight score between source node i and target node j. Next, we define a GAT-based spatial model that utilizes the learned adjacency matrix A to model the spatial features of multivariate time series.

3.4 GAT-Based Spatial Model

We use GAT and graph structure learning to fuse the information of the nodes with their neighbors. For the input multivariate time series $\mathbf{X} \in \mathbb{R}^{N \times d}$, we compute the aggregated representation μ_i of node i as follows:

$$\mu_i = \text{ReLU}\left(\alpha_{i,i}\mathbf{W}\mathbf{X}_i + \sum_{j \in N(i)} \alpha_{i,j}\mathbf{W}\mathbf{X}_j\right) \tag{4}$$

where, $\mathbf{X}_i \in \mathbb{R}^N$ is the input feature of node i, $N(i) = \{j | A_{ij} > 0\}$ represents the neighborhood set of node i and its values are obtained from matrix A, $\mathbf{W} \in \mathbb{R}^{N \times N}$ is the trainable weight matrix with a linear transformation for each node. Unlike GDN [8], we connect the node features to the weight score ρ so that not only the local spatial dependencies but also the global spatial dependencies in the graph can be captured. The attention coefficient $\alpha_{i,j}$ is computed using the following calculation method:

$$Concat_i = \rho_i \oplus \mathbf{W}\mathbf{X}_i \tag{5}$$

$$\pi_{i,j} = \text{LeakyReLU}\Big(\mathbf{a}^{\text{T}}\big(Concat_i \oplus Concat_j\big)\Big) \tag{6}$$

$$\alpha_{i,j} = \frac{\exp\left(\pi_{i,j}\right)}{\sum_{k\in N(i)\cup\{i\}} \exp\left(\pi_{i,k}\right)} \tag{7}$$

where $\oplus$ denotes concatenation, $Concat_i$ concatenates the weight scores ρ_i and $\mathbf{WX}_i$, the vector $\mathbf{a}$ represents the learnable coefficients of the attention mechanism. LeakyReLU is used to calculate the attention coefficients and we employ the Softmax function to normalize the computed coefficients. Next, we use Transformers to model temporal features.

3.5 Transformers Based on Amplifying Anomalies Strategy

We supply $\mu \in \mathbb{R}^{N\times d}$ to the Transformers for reconstruction by alternately stacking Multi-Mix Attention and feedforward layers. This structure better captures the details and patterns present in time series data. Among them, the overall equation of layer l is as follows:

$$\mathbf{Z}^{l} = \text{Add\&Norm}\Big(\text{Multi-Mix Attention}\big(\mu^{l-1}\big) + \mu^{l-1}\Big) \tag{8}$$

$$\mu^{l} = \text{Add\&Norm}\Big(\text{Feed-Forward}\big(\mathbf{Z}^{l}\big) + \mathbf{Z}^{l}\Big) \tag{9}$$

where $\mu^{l} \in \mathbb{R}^{N\times d_{model}}$, $l \in \{1, 2, \cdots L\}$ represents the output of layer l, featuring d_{model} channels. Initial input $\mu^0 = \text{Embeding}(\mu)$. $\mathbf{Z}^{l} \in \mathbb{R}^{N\times d_{model}}$ is the hidden representation of layer l.

Multi-mix Attention: Inspired by Anomaly Transformer [25], we propose the Multi-Mix Attention with local associations and global associations to amplify anomalies. Local associations are derived from a learnable Gauss function. The Gauss function can focus on adjacent layers and amplify local associations. To prevent the weights from decaying too rapidly or overfitting, we design the scale parameter σ as a learnable parameter, which allows the function to better adapt to different patterns of time series. In addition, we use Transformers' self-attentive scores as the global associations, which can adaptively find the most effective global distributions. The Multi-Mix Attention of layer l is as follows:

$$\mathbf{Q}, \mathbf{K}, \mathbf{V}, \sigma = \mu^{l-1}\mathbf{M}_{\mathbf{Q}}^{l}, \mu^{l-1}\mathbf{M}_{\mathbf{K}}^{l}, \mu^{l-1}\mathbf{M}_{\mathbf{V}}^{l}, \mu^{l-1}\mathbf{M}_{\sigma}^{l} \tag{10}$$

$$\text{Local-Association}: \mathbf{G}^{l} = \text{Rescale}\Big(\Big[\frac{1}{\sqrt{2\pi}\sigma_i}\exp\big(-\frac{|j-i|^2}{2\sigma_i^2}\big)\Big]_{i,j\in\{1,\cdots,N\}}\Big) \tag{11}$$

$$\text{Global-Association}: \mathbf{S}^{l} = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{T}}{\sqrt{d_{model}}}\right) \tag{12}$$

$$\text{Reconstruction}: \widehat{\mathbf{Z}}^l = \mathbf{S}^l \mathbf{V} \tag{13}$$

where $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{N \times d_{model}}$, $\sigma \in \mathbb{R}^{N \times 1}$ denote query, key, self-attentive value and learning scale respectively. $\mathbf{M}_{\mathbf{Q}}^l, \mathbf{M}_{\mathbf{K}}^l, \mathbf{M}_{\mathbf{V}}^l \in \mathbb{R}^{d_{model} \times d_{model}}$, $\mathbf{M}_{\sigma}^l \in \mathbb{R}^{d_{model} \times 1}$ denote the parameter matrices of the $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ and σ in the l-th layer respectively. We use Gaussian kernels to calculate the association weights between each two points, and then convert these weights into a discrete distribution through row-wise normalization with Rescale to obtain $\mathbf{G}^l \in \mathbb{R}^{N \times N}$. $\mathbf{S}^l \in \mathbb{R}^{N \times N}$ is the attention map of Transformers. We found that it contains abundant information and can be utilized as a global learning association. $\widehat{\mathbf{Z}}^l \in \mathbb{R}^{N \times d_{model}}$ is the hidden representation after the Multi-Mix Attention in the l-th layer.

We use KL divergence to represent the difference between local and global associations [18]. By averaging multiple layers of association differences, more information can be fused, and the combined association differences is:

$$\text{Dis}\left(\mathbf{G}, \mathbf{S}\right) = \Big[\frac{1}{L}\sum_{l=1}^{L}\Big(\text{KL}(\mathbf{G}_{i,:}^l \,||\, \mathbf{S}_{i,:}^l) + \text{KL}(\mathbf{S}_{i,:}^l \,||\, \mathbf{G}_{i,:}^l)\Big)\Big]_{i=1,\cdots,N} \tag{14}$$

where, $\text{KL}(\cdot \parallel \cdot)$ corresponds to the Kullback-Leibler divergence between the associations of $\mathbf{G}^l$ and $\mathbf{S}^l$ for each row. $\text{Dis}\left(\mathbf{G}, \mathbf{S}\right) \in \mathbb{R}^{N \times 1}$ is the degree of deviation of input time series with local-association $\mathbf{G}$ and global-association $\mathbf{S}$. Since the Gaussian function has local single-peakedness, so that the Gaussian distribution will show fluctuations on both normal and anomalous data, while normal data tends to exhibit smoother performance with the global association, which indicates that the Dis value of the abnormal points will be smaller than the Dis value of the normal points, so Dis has good anomaly differentiation.

3.6 Joint Optimization

Finally, we optimize the spatio-temporal model. We employ additional losses to amplify the Dis, which can further amplify the difference. The loss functions are:

$$\mathbf{L}_1 = \|\mu - \mathbf{X}\|_F^2 \tag{15}$$

$$\mathbf{L}_2 = \left\|\widehat{\mathbf{X}} - \mu\right\|_F^2 \tag{16}$$

$$\mathbf{L}_{total} = \beta \times \mathbf{L}_1 + (1-\beta) \times \mathbf{L}_2 - \lambda \times \|\text{Dis}\left(\mathbf{G}, \mathbf{S}\right)\|_1 \tag{17}$$

where $\widehat{\mathbf{X}}$ represents the reconstruction of μ through the use of Transforms. $\|\cdot\|_F$, $\|\cdot\|_K$ represents the Frobenius and k-norms, β denotes a balance parameter that lies within the interval [0, 1], λ represents the weighting of the loss terms. When $\lambda > 0$, the optimization is to amplify Dis.

Note that excessively amplifying differences can compromise the accuracy of Gaussian kernel [18], rendering the local-association devoid of meaningful interpretation. To avoid this, Anomaly Transforms proposes a minimax strategy [25].

In the minimization phase, the local association $\mathbf{G}$ is optimized to approximate the sequence association $\mathbf{S}$ learned from the original sequence. For the maximization stage, we optimize the global association to increase the difference. The loss functions of the two stages are as follows:

$$\text{MinimizePhase}: \mathbf{L}_{total} = \beta \times \mathbf{L}_1 + (1-\beta) \times \mathbf{L}_2 + \lambda \times \left\| \text{Dis}\left(\mathbf{G}, \mathbf{S}_{detach}\right) \right\|_1 \quad (18)$$

$$\text{MaximizePhase}: \mathbf{L}_{total} = \beta \times \mathbf{L}_1 + (1-\beta) \times \mathbf{L}_2 - \lambda \times \left\| \text{Dis}\left(\mathbf{G}_{detach}, \mathbf{S}\right) \right\|_1 \quad (19)$$

where detach refers to the discontinuation of backpropagating the gradient and $\lambda > 0$. During the minimization phase, the backpropagation of the gradient of $\mathbf{S}$ is halted, enabling $\mathbf{G}$ to approximate $\mathbf{S}$. Conversely, during the maximization phase, the gradient backpropagation of $\mathbf{G}$ is stopped while $\mathbf{S}$ is optimized to amplify anomalies.

Anomaly Score: By combining association differences with joint optimization, we obtain the anomaly score:

$$Score = \text{Softmax}\Big(- \text{Dis}\left(\mathbf{G}, \mathbf{S}\right) \Big) \odot \Big(\beta \times \mathbf{L}_1 + (1-\beta) \times \mathbf{L}_2 \Big) \quad (20)$$

where $\odot$ is the element multiplication method. This design allows the reconstruction error and anomaly amplification strategies to synergistically improve the detection performance.

4 Experiments

4.1 Datasets

To evaluate our method, we carry out detailed experiments on four datasets. The characteristics of these datasets are summarized in Table 1.

- Secure Water Treatment Testbed (SWaT): The SWaT dataset is derived from genuine industrial control system data obtained from a water treatment plant [16]. It contains 51 sensors.
- Water Distribution Testbed (WADI): This is an extension of the SWaT system, but has a larger and more complex data scale compared to the SWaT dataset [2].
- Server Machine Dataset (SMD) [22]: SMD consisting of 38-dimensional data collected over a 5-week period from a major Internet corporation. Only a subset of the dataset is used for evaluation due to Service Changes, which affected some machines in the dataset. The subset consists of 7 entities (machines) that did not undergo any service changes.
- Pooled Server Metrics (PSM) [1]: The PSM dataset is provided by eBay, reflects the status of servers, 25 dimensions in total.

Table 1. Details of the datasets.

Dataset	SWaT	WADI	SMD	PSM
Training size	396000	838857	144546	105984
Validation size	99000	209714	36135	26497
Testing size	449919	172801	180682	87841
Number of Sensors	51	123	38	25
Anomaly rate(%)	12.13	5.99	8.80	27.76

4.2 Baseline and Evaluation Metrics

We compared our STAD with several baseline approaches, including traditional methods: Isolation Forest [14], and deep-learning-based models: USAD [4], GDN [8], OmniAnomaly [22], LSTM-VAE [19], DAGMM [27], and Anomaly transformer [25]. We use Precision, Recall, and F1 scores to evaluate the performance of our method, which are widely used in anomaly detection.

4.3 Implementation Details

Adhering to the established protocol in Anomaly Transformer [25], we use a non-overlapping sliding window approach to obtain subsets. The fixed size of the sliding window is uniformly set to 100. We utilized grid search to obtain the anomaly threshold and hyperparameters that result in the highest F1 score. The top-K values for SWaT, PSM, WADI, and SMD are 10, 5, 30, and 15 respectively. The Transformer model consists of 3 layers, we set the number of heads to 8 and the d_{model} dimension to 512. The value of λ is set to 3, β to 0.5 and we employ the Adam optimizer [11] with the learning rate of 10^{-4}. Training process employs an early stopping strategy and batch size is set to 32. All experiments were conducted using a single NVIDIA Titan RTX 12GB GPU in PyTorch. To ensure that any timestamps during an anomaly event can be detected, we utilized a widely adopted point adjustment strategy [21,22,24]. In order to maintain fairness, the same point adjustment strategy was implemented across all baseline experiments.

4.4 Result Analysis

In many real-world anomaly detections, failure to detect anomalies can result in severe consequences. Therefore, detecting all genuine attacks or anomalies is more crucial than achieving high accuracy. As shown in Table 2, our proposed STAD outperforms other methods in terms of F1 performance. It is noteworthy that while most methods perform well on datasets such as SWaT, PSM, and SMD, as their anomalies are more easily detectable, our model still outperforms them in F1 score. When dealing with more complex MTS datasets like WADI, most existing methods yield poor results, while our model shows a significant

improvement compared to others. We also observe that: (1) Compared to traditional unsupervised methods, deep learning-based techniques generally demonstrate superior detection performance; (2) Compared to models that solely learn a single relationship, the concurrent acquisition of temporal and spatial relationships significantly amplifies the anomaly detection efficacy.

Table 2. Experimental results on four public datasets.(%)

Method	SWaT			WADI			SMD			PSM		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
iForest [14]	49.29	44.95	47.02	62.41	61.55	61.98	59.45	85.64	68.31	76.09	92.45	83.48
LSTM-VAE [19]	76.00	89.50	82.20	46.32	32.20	37.99	87.36	79.63	83.84	73.62	89.92	80.96
DAGMM [27]	89.92	57.84	70.40	22.28	19.76	20.94	69.13	87.25	76.67	93.49	70.03	80.08
OmniAnomaly [22]	81.42	84.30	82.83	26.52	97.99	41.74	96.79	94.37	96.20	88.39	74.46	80.83
GDN [8]	99.35	68.12	81.17	97.35	40.11	57.17	67.83	95.78	77.01	54.92	99.92	70.88
USAD [4]	98.70	74.02	84.60	64.51	32.20	42.96	93.46	95.65	90.24	56.44	92.69	70.15
Anomaly-Transformer [25]	91.55	96.73	94.07	79.70	93.83	85.91	95.86	94.71	95.15	96.91	98.90	97.89
Ours	93.97	99.84	**96.46**	85.57	97.98	**91.34**	95.94	96.64	**96.43**	98.45	98.42	**98.32**

4.5 Ablation Experiments

To investigate the efficacy of each constituent of our methodology, we conducted ablation experiments to observe how the model performance varies on the four datasets. Firstly, we investigated the significance of using GAT to model spatial dependency relationships. We directly applied the raw data as input to the Transformers. Secondly, we used a static graph to replace the learned graph to prove the effectiveness of our proposed graph structure learning. Finally, to validate the necessity of Multi-Mix Attention, we removed it and only use spatial relations to reconstruct.

Table 3. Experimental results of STAD and its variants.(%)

Method	SWaT			WADI			SMD			PSM		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Without-GAT	90.31	93.73	93.76	70.13	91.67	85.37	95.31	94.27	93.47	94.51	96.75	96.29
Without-Graph Learning	90.15	93.20	91.09	80.61	84.34	78.32	93.82	91.97	92.43	92.49	93.87	96.42
Without-Multi-Mix Attention	95.94	66.45	78.52	81.95	47.45	60.01	68.49	99.65	80.13	58.65	99.46	73.79
Ours	93.97	99.84	**96.46**	85.57	97.98	**91.34**	95.94	96.64	**96.43**	98.45	98.42	**98.32**

The summarized results are presented in the Table 3. Furthermore, the following observations are provided based on the results: (1) The difference between the models that do not learn graph structure and our proposed model highlights the significance of spatial features in addressing anomaly detection for multivariate time series data. (2) Our structure learning is more effective than using

a static graph as the graph structure. (3) The transformer architecture with the Multi-Mix Attention performs remarkable performance in handling time series data. Overall, It is evident that each component of our model is effective and indispensable, thereby endowing the framework with powerful capabilities for detecting anomalies in multivariate time series.

4.6 Interpretability

We visualize the anomaly amplification strategy section, as seen in Fig. 3, for real-world datasets, our model can correctly detect anomalies. For the SWaT dataset, our approach has shown the ability to detect anomalies at an early stage, indicating its potential for practical applications such as providing early warning for faults.

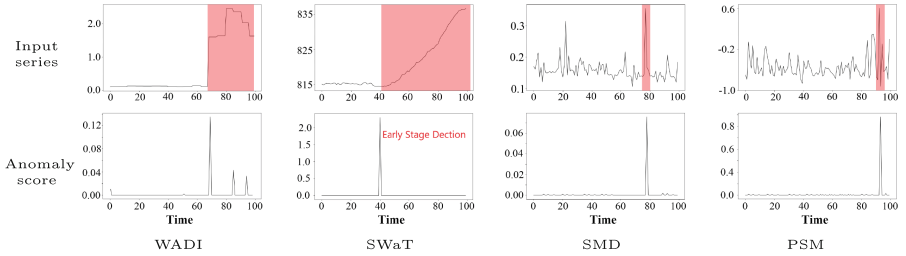

Fig. 3. Visualization of model learning in a real-world dataset. Anomalies are marked by red shading. (Color figure online)

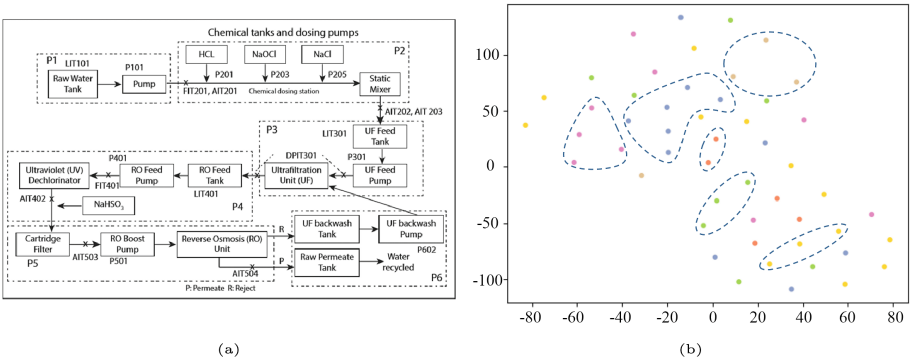

Fig. 4. Visualization of graph structure learning.

Additionally, the visualization of the learned graph structure further demonstrates the effectiveness of our proposed model. Figure 4(a) is the process diagram

of the secure water treatment testbed [16]. It can be observed that the SWaT system is mainly divided into 6 processes, and sensors in the same process stage are more likely to be interdependent. Figure 4(b) displays the t-SNE [15] plot of the sensor embeddings learned by our model on the SWaT dataset, where most nodes belonging to the same process cluster together. This demonstrates the effectiveness of our graph structure learning.

4.7 Case Analysis

We use the example in Fig. 1 to illustrate why our model helps with anomaly interpretation. From the previous anomaly analysis, we know that the anomaly is manifested as water tank overflow, but the root anomaly is caused by the early opening of MV101. It is hard to find anomalies of MV101 with its irregular switch status. However, through Fig. 5(a), we can see that our model successfully detected the anomaly in MV101.

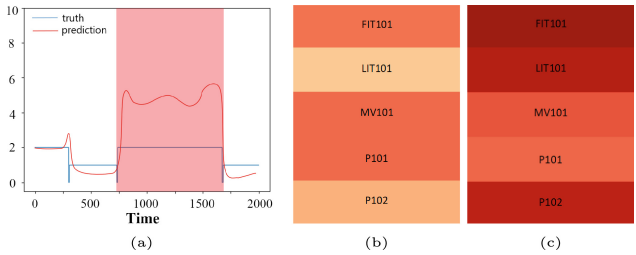

Fig. 5. Case study showing the attack in SWaT.

In addition, other sensors are expected to be correlated with MV101 when the system is functioning normally. Figure 5 presents the weight scores between the other sensors of the same process and MV101. As depicted in Fig. 5(b), our model effectively learns the features associated with MV101 under normal conditions. When anomalies occur (corresponding to the red section in Fig. 1), the sensors weight scores are visualized in Fig. 5(c). It is evident that the sensor under attack (MV101) is more closely associated (darker in color) with other sensors in the same subprocess. This is reasonable, as when an anomaly occurs, the sensors associated with the anomaly are more strongly affected.

5 Conclusion

This paper proposes a novel approach for multivariate time series anomaly detection by leveraging spatio-temporal relationships. The proposed approach utilizes a graph attention network (GAT) and a graph structure learning strategy to capture spatial associations among multivariate time series. Additionally, Transformers are used to model temporal relationships within the time series. An

anomaly amplification strategy is also employed to enhance the anomaly scores. Experimental results demonstrate that the proposed method outperforms existing approaches in identifying anomalies and is effective in explaining anomalies. Future work may involve incorporating online training techniques to better handle complex real-world scenarios.

Acknowledgements. This paper was supported by the National Natural Science Foundation of China(Grant No.U22B2051).

References

1. Abdulaal, A., Liu, Z., Lancewicki, T.: Practical approach to asynchronous multivariate time series anomaly detection and localization. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, pp. 2485–2494 (2021)
2. Ahmed, C.M., Palleti, V.R., Mathur, A.P.: 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)
3. Angiulli, F., Pizzuti, C.: Fast outlier detection in high dimensional spaces. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) PKDD 2002. LNCS, vol. 2431, pp. 15–27. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45681-3_2
4. Audibert, J., Michiardi, P., Guyard, F., Marti, S., Zuluaga, M.A.: 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)
5. Baldi, P.: Autoencoders, unsupervised learning, and deep architectures. In: Proceedings of ICML Workshop on Unsupervised and Transfer Learning, pp. 37–49. JMLR Workshop and Conference Proceedings (2012)
6. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, pp. 93–104 (2000)
7. Chen, Z., Chen, D., Zhang, X., Yuan, Z., Cheng, X.: Learning graph structures with transformer for multivariate time-series anomaly detection in iot. IEEE Internet Things J. **9**(12), 9179–9189 (2021)
8. Deng, A., Hooi, B.: Graph neural network-based anomaly detection in multivariate time series. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 4027–4035 (2021)
9. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
10. Hundman, K., Constantinou, V., Laporte, C., Colwell, I., Soderstrom, T.: 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)
11. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. iclr. 2015. arXiv preprint arXiv:1412.6980 9 (2015)
12. Kingma, D.P., Welling, M., et al.: An introduction to variational autoencoders. Foundat. Trends® Mach. Learn. **12**(4), 307–392 (2019)

13. Li, D., Chen, D., Jin, B., Shi, L., Goh, J., Ng, S.-K.: MAD-GAN: multivariate anomaly detection for time series data with generative adversarial networks. In: Tetko, I.V., Kůrková, V., Karpov, P., Theis, F. (eds.) ICANN 2019. LNCS, vol. 11730, pp. 703–716. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30490-4_56
14. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: 2008 Eighth IEEE International Conference on Data Mining, pp. 413–422. IEEE (2008)
15. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. J. Mach. Learn. Res. **9**(11) (2008)
16. Mathur, A.P., Tippenhauer, N.O.: Swat: a water treatment testbed for research and training on ics security. In: 2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater), pp. 31–36. IEEE (2016)
17. Mirza, M., Osindero, S.: Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784 (2014)
18. Neal, R.M.: Pattern recognition and machine learning. Technometrics **49**(3), 366 (2007)
19. Park, D., Hoshi, Y., Kemp, C.C.: A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. IEEE Robot. Autom. Lett. **3**(3), 1544–1551 (2018)
20. Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., Schmidt-Erfurth, U.: f-anogan: fast unsupervised anomaly detection with generative adversarial networks. Med. Image Anal. **54**, 30–44 (2019)
21. Shen, L., Li, Z., Kwok, J.: Timeseries anomaly detection using temporal hierarchical one-class network. Adv. Neural. Inf. Process. Syst. **33**, 13016–13026 (2020)
22. Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., Pei, D.: 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)
23. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems 30 (2017)
24. Xu, H., et al.: Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In: Proceedings of the 2018 World Wide Web Conference, pp. 187–196 (2018)
25. Xu, J., Wu, H., Wang, J., Long, M.: Anomaly transformer: time series anomaly detection with association discrepancy. arXiv preprint arXiv:2110.02642 (2021)
26. Zhao, H., et al.: Multivariate time-series anomaly detection via graph attention network. In: 2020 IEEE International Conference on Data Mining (ICDM), pp. 841–850. IEEE (2020)
27. Zong, B., et al.: Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In: International Conference On Learning Representations (2018)