Complex & Intelligent Systems (2024) 10:2937–2950
https://doi.org/10.1007/s40747-023-01306-x

ORIGINAL ARTICLE

An enhanced abnormal information expression spatiotemporal model for anomaly detection in multivariate time-series

Di Ge[1] · Yuhang Cheng[2] · Shuangshuang Cao[1] · Yanmei Ma[1] · Yanwen Wu[1,3]

Received: 14 July 2023 / Accepted: 25 November 2023 / Published online: 6 January 2024

Abstract
The detection of anomalies in high-dimensional time-series has always played a crucial role in the domain of system security. Recently, with rapid advancements in transformer model and graph neural network (GNN) technologies, spatiotemporal modeling approaches for anomaly detection tasks have been greatly improved. However, most methods focus on optimizing upstream time-series prediction tasks by leveraging joint spatiotemporal features. Through experiments, we found that this modeling approach not only risks the loss of some original anomaly information during data preprocessing, but also focuses on optimizing the performance of the upstream prediction task and does not directly enhance the performance of the downstream detection task. We propose a spatiotemporal anomaly detection model that incorporates an improved attention mechanism in the process of temporal modeling. We adopt a heterogeneous graph contrastive learning approach in spatio modeling to compensate for the representation of anomalous behavioral information, thereby guiding the model through thorough training. Through validation on two widely used real-world datasets, we demonstrate that our model outperforms baseline methods. We also explore the impact of multivariate time-series prediction tasks on the detection task, and visualize the reasons behind the benefits gained by our model.

Keywords Anomaly detection · Multivariate time-series · Spatiotemporal · Abnormal information expression · Graph contrastive learning

Introduction

The detection of anomalies in multivariate time-series data based on spatiotemporal modeling is an emerging research field, aiming to capture spatiotemporal dependencies from massive multivariate time-series data and achieve more sensitive anomaly detection through richer feature representations. In the real world, spatiotemporal data are present in various domains, including industry, transportation, meteorology, and finance [1]. These not only exhibit the characteristics of time-series, but encompass diverse aspects such as physical properties and spatial–topological structures, and are multivariate, with high dimensionality and complexity. Hence, to perform automatic anomaly detection on these massive spatiotemporal data can enhance the efficiency and accuracy of data analysis, effectively reducing the risk of accidents in practical industrial production processes, and holding significant economic and safety value [2–4].

Despite the comprehensive consideration of data features from a dual perspective, it is still challenging to sensitively detect anomalies from massive amounts of multivariate time-series data through spatiotemporal modeling. Through experimental analysis, we believe that this is primarily due to a lack of sufficient abnormal behavior information in the deep modeling process to guide the model for training. There are two aspects. First, from the data characteristics, in an actual production process, the system is often reliable, so data of anomalous moments are scarce and hidden under a large amount of normal data, and the model must be trained in unsupervised conditions because of expensive labeling [5, 6]. Second, from the modeling perspective, serial decomposition or normalization of data during temporal modeling can reduce distribution differences in the time dimension [7, 8],

✉ Yanwen Wu
wyw.ccnu.edu@gmail.com

1 School of Physical Science and Technology, Central China Normal University, Wuhan 430079, China

2 Shaanxi GSXZ Technology Co., Ltd, Xi'an 710018, China

3 National Digital Learning Engineering Technology Research Centre, Central China Normal University, Wuhan 430079, China

which can effectively improve mainstream prediction performance. However, similar operations may not apply when the downstream task is anomaly detection, because some of the already sparse anomalous behavior information is lost, reducing the acuity of the model for anomalous moment capture. Spatial-dependency modeling can compensate for the learning of anomalous behavior information, providing a feasible path. However, in practice, the lack of a priori knowledge of physical features that can be translated into effective spatio constraints, such as patterns of influence between features, makes it difficult for the model to obtain adequate representations of anomalous behavioral information.

With its outstanding performance in sequence modeling and prediction tasks, numerous anomaly detection methods based on Transformer [9] have been proposed in recent years [10, 11]. Wiederer incorporated an external attention mechanism on top of self-attention to model the correlation among multivariate time-series, and proposed a regularization-based method to constrain model parameters and prevent overfitting [12]. Su focused on key issues in the prediction process, including the choice of feature embedding, impact of model depth and width, and combination of attention mechanisms and convolutional layers [13]. Anomaly Transformer leveraged the differences between abnormal time points and their local and global contexts to derive a distinguishable detection principle [14]. Some studies have deeply modeled the spatio dependencies among multivariate variables. GDN learns the correlation graph among features without prior knowledge, and utilizes graph neural networks to model the information flow between feature nodes, helping with anomaly detection by predicting the future behavior of features [15]. GTA combines Transformer and GNN in a hierarchical attention mechanism that considers the correlations between spatiotemporal features [16]. Han further advances this approach by integrating sparse self-encoders with graph neural networks, orchestrating a collaborative optimization of both the reconstruction and prediction tasks [17]. TranAD introduces a model for anomaly detection and diagnosis, leveraging a profound transformer network. This network integrates an attention-based sequence encoder, facilitating swift comprehension of overarching temporal patterns in the data [18]. The above literature models spatiotemporal features separately, obtaining future feature behavior expressions through prediction-based approaches, and conducts anomaly detection based on expression differences. However, the modeling process overlooks the impact of preprocessing methods like normalization on the loss of abnormal behavioral information, and there is still room for improvement in the modeling of spatio correlations between features and exploration of spatio constraints to strengthen model inference.

We propose an enhanced abnormal information expression spatiotemporal model for anomaly detection in multivariate time-series (EAIE-AD), which is capable of end-to-end anomaly detection under unsupervised training conditions. Our model performs simultaneous deep temporal modeling in a parallel manner. Through experiments, we have found that while Transformer and its variants have demonstrated strong modeling capabilities in prediction tasks through optimizing the mean squared error (MSE) and mean absolute error (MAE), this goal does not directly yield the final anomaly detection effect (as shown in Table 4). Hence, our goal in the prediction phase is not to optimize these metrics all the time, but to be able to learn more valid representations of anomalous behavior information. Hence, we focus on the non-stationary information in the original data but potentially lost due to normalization, and improve upon the work of non-stationary Transformer [19] by simplifying its model structure and modifying the object of action of the attention mechanism; in the spatio module, we learn the feature association graph of multivariate time-series data, and expand the homogeneous graph to a heterogeneous graph based on a GDN [15], which can more finely simulate the physical characteristics of the features, and on the graph structure, using two contrast learning strategies to find beneficial spatio constraints to strengthen our learning ability for the representation of anomalous behavior information. The contributions made by our model are summarized as follows:

(1) We propose an end-to-end spatiotemporal anomaly detection model that can simultaneously model feature temporal dependencies and spatio correlations in depth through a parallel architecture and guide the full training of the model under unsupervised conditions by enhancing abnormal information expression.

(2) In the temporal dimension, we compensate for the effective modeling of the inherent unsteadiness information in the original data, and mitigate the loss of anomalous information by improving the execution objects of the attention mechanism; in the spatio dimension, we use graph neural networks and contrast learning to model the physical properties of feature behaviors, from which we extract the hidden expression of anomalous behavior information in the spatio topology.

(3) We achieve state-of-the-art anomaly detection results on multiple datasets, with the F1-scores reaching 0.82 and 0.59 on the SWaT and WADI datasets, respectively. Then, we conduct adequate ablation experiments and data visualization. Finally, we enhance the interpretability of the model by exploring the impact of upstream multivariate timing prediction tasks on downstream anomaly detection tasks.

Related work

Time-series anomaly detection

The classical methods employed in multivariate time-series data anomaly detection tasks are primarily reconstruction-[20, 21] or prediction-based [22, 23], which can, respectively, compress data representation or model temporal correlation [24]. In addition, dimensionality-reduction methods, such as principal component analysis (PCA), singular value decomposition (SVD), and autoencoder (AE), as commonly used in machine learning, have been shown to be effective in assisting anomaly detection. In PCA, anomalies are defined as data points that deviate from the normal data space [25]; in SVD, anomalies are defined as high-dimensional data that have not been reconstructed [26]; AE detects anomalies by learning a self-encoder from the data, where the reconstruction error of anomalous data is usually greater than that of normal data [27].

With the high dimensionality of data, by being able to automatically learn its features and better model nonlinear relationships, researchers have turned their attention to deep learning for anomaly detection [28, 29]**.** Chen proposed an anomaly detection method for multivariate temporal data, using a variational autoencoder (VAE) model to learn the latent representation of the data, and reconstruction error and the Kullback–Leibler (KL) scatter of the latent representation as anomaly metrics [30]. Kong proposed a long short-term memory (LSTM)-based method, using an attention mechanism to assign weights to temporal features [31]. Transformer with an attention mechanism as its structural core has enabled a breakthrough in deep time-series prediction. Its point-to-point attention mechanism is suitable for modeling temporal dependencies in time-series, and stackable codecs are conducive to capturing and aggregating temporal features at different time scales. Hence, Transformer-based improved anomaly detection methods have been proposed. Jeong proposed a self-supervised learning method that uses the transformer model to learn a representation of multivariate time-series data, and uses the distance of the representation to measure the degree of anomalies of data points [32]. Wiederer used Transformer to explore the variability of the association between anomalous moments and local and global moments, deriving a sensitive differentiation principle [12].

While Transformer can well model the relationships between moments, information transfer between features at any moment is also important for learning anomalous behavior. The spatio topology formed by this information transfer is in a non-Euclidean space, and this spatio dependence is difficult to model with conventional neural networks due to the sparsity of the structure [33, 34].

Anomaly detection based on spatiotemporal modeling

Graph neural networks can address the limitations of non-Euclidean space-dependent modeling, handling the relationships between features and enabling end-to-end learning with good robustness [35, 36], and hence are receiving increasing attention in anomaly detection. GGC-LSTM combines the advantages of graph convolutional neural networks and long short-term memory (LSTM) networks, which can consider both graph structure and time-series information [37]. Zhao used a deep graph convolutional neural network that can adaptively learn the relationships between time-series data with high computational efficiency [38]. GTA model spatiotemporal dependency for multivariate sensor features in IoT systems in a tandem fashion and enhances model inference efficiency through a hierarchical attention mechanism [16].

Deng proposed a general framework for spatiotemporal modeling anomaly detection network (GDN) [15], which has received attention for its excellent results on several realistic datasets. GDN learns a spatial-association graph between sensor features in the absence of a priori knowledge, uses an attention mechanism for information transfer and message updating on the graph structure, and assists the learned information in the temporal prediction task, which can more sensitively capture deviations in the future behavior of sensors and improve detection performance. However, there is room for improvement. In the temporal sequence prediction process, the model adopts a traditional normalization strategy, ignoring the expression of inherent unsteadiness in the data, which can be useful in learning behavior patterns at anomalous moments. The feature relationship graph is homogeneous, and cannot well describe physical characteristics in realistic scenarios. We propose an end-to-end spatiotemporal anomaly detection model, which can enhance abnormal information expression through modeling spatiotemporal dependency to guide the full training of the model.

Methods

We describe our proposed model. Figure 1 shows a high-level overview of EAIE-AD, with an end-to-end spatiotemporal model that incorporates temporal and spatio dependency.

Problem formulation

Assume a collection of multivariate time-series data obtained from d features at T_{train} time stamps, denoted as $S = \{S_1, \ldots, S_{T_{\text{train}}}\}$, $i \in \{1, \ldots, T_{\text{train}}\}$, $S_i \in \mathbb{R}^d$. Our model is trained in an unsupervised manner. The training and validation datasets consist of normal data, while the test dataset has

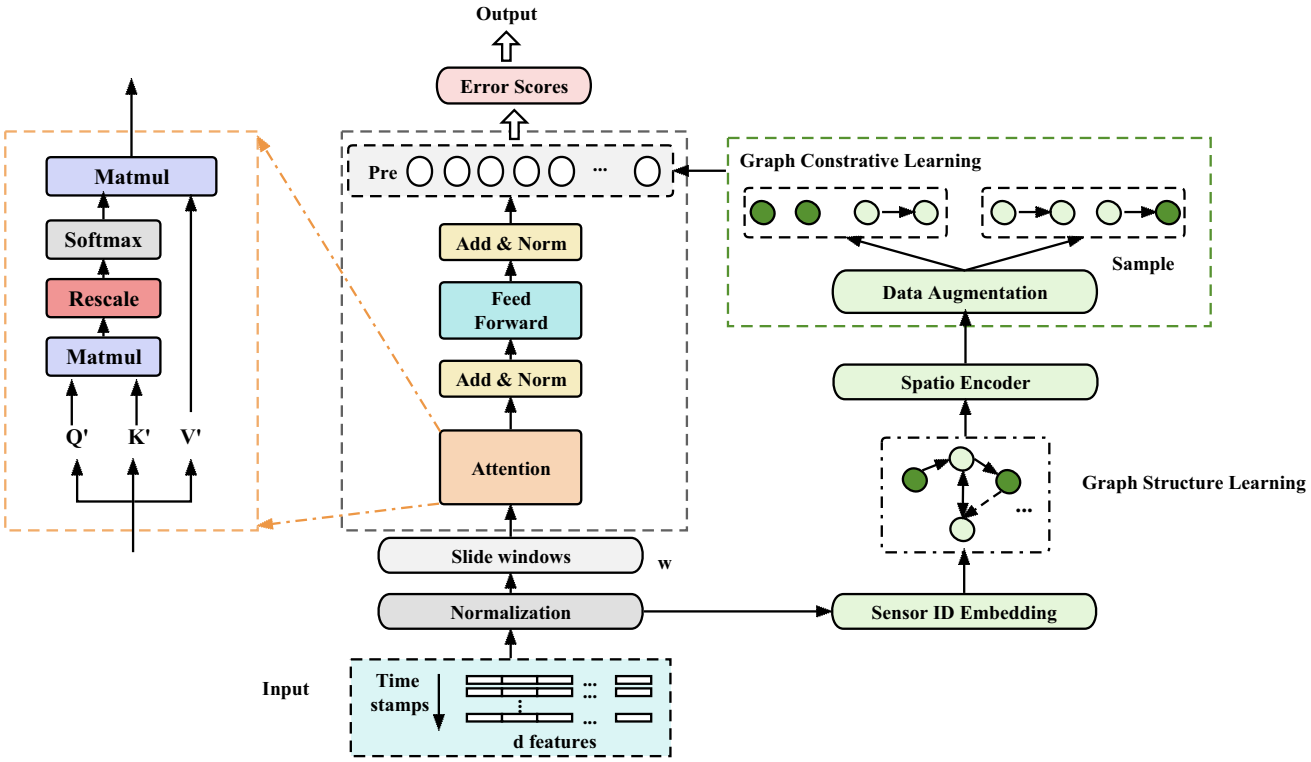

Fig. 1 Model framework

both normal and abnormal data. We seek to acquire knowledge about the behavior of the features through the training set and identify any anomalous time stamps within the test set, assigning a label to each time step in the test set, where 0 and 1 denote normal and abnormal, respectively.

Temporal dependency modeling

Our goal is to predict the behavior of features through time-series modeling. While Transformer is popular for temporal prediction due to its powerful long-time-series modeling capability, for downstream anomaly detection, to learn enough validly expressed anomalous behavior information is more important than realizing small values of MSE and MAE in prediction. Usually, non-stationary information can imply more abnormal behavior expressions, but in the traditional Transformer input, due to operations such as normalization, the input loses some non-stationary information, and may bring about data over-stationarity problems and reduce the performance of the attention mechanism.

The result, if not normalized, is a nonuniform feature scale and more noisy points, reducing prediction performance. Therefore, we compensate for the normalization information and optimize the object of attention. We first slice the original input data S in the form of a sliding window. For any moment sample S_t, we intercept the time window series whose historical series length is w to get $X = [X_1, \ldots, X_w]^T = [S_t, S_{t-1}, \ldots, S_{t-w+1}]^T$, $X \in \mathbb{R}^{w\times d}$, $t \geq w$. For each time window X, we normalize to obtain $X\prime = [X'_1, \ldots, X'_w]^T$, where

$$\mu_x = \frac{1}{w}\sum_{i=1}^{w} X_i,\ \sigma_x^2 = \frac{1}{w}\sum_{i=1}^{w}(X_i - \mu_x)^2,\ X'_i = \frac{1}{\sigma_x} \odot (X_i - \mu_x)\ for i \in \{1, \ldots, w\}, \tag{1}$$

where μ_x, $\sigma_X \in \mathbb{R}^d$, and $\odot$ denotes the element-wise product. At this point, for uniform characteristic scales, we use the normalization module to the input time-series. However, normalization will reduce the unsteadiness of the data distribution. On the one hand, some anomalous information will be lost. On the other hand, the input sequence of the attention mechanism may not be able to produce differentiated attention due to over-smoothing, which leads to the degradation of the model training performance. Hence, we change the execution object of the attention mechanism. The standard self-attention mechanism is

$$\mathrm{Att}(Q, K, V) = \mathrm{Softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right)V, \tag{2}$$

where Q, K, $V \in \mathbb{R}^{w\times d_k}$ are queries, keys, and values, respectively. Softmax is an exponential normalization

function. With the normalization of Eq. 1, each feature variable in the sequence has the same variance, so σ_X can be converted to a scalar. Because the embedding and feedforward layers have linear properties, $Q' = \left(Q - 1\mu_Q^{\mathrm{T}}\right)/\sigma_X$ is formed by the projection of X', where $\mu_Q \in \mathbb{R}^{d_k}$, and we can obtain

$$\begin{aligned}&\mathrm{Softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right)\\&= \mathrm{Softmax}\left(\frac{\sigma_x^2 Q'(K')^{\mathrm{T}} + 1\left(\mu_Q^{\mathrm{T}}K^{\mathrm{T}}\right) + (Q\mu_K)\,1^{\mathrm{T}} - 1\left(\mu_Q^{\mathrm{T}}\mu_K\right)1^{\mathrm{T}}}{\sqrt{d_k}}\right),\end{aligned} \tag{3}$$

where $1h\mathbb{R}^{w\times 1}$, $Q\mu_K \in \mathbb{R}^{w\times 1}$, $\mu_Q^{\mathrm{T}}\mu_K$ is a scalar, and $Softmax(\cdot)$ is invariant to the same translation on the row dimension of input, and we have

$$\mathrm{Softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right) = \mathrm{Softmax}\left(\frac{\sigma_x^2 Q'(K')^{\mathrm{T}} + 1\left(\mu_Q^{\mathrm{T}}K^{\mathrm{T}}\right)}{\sqrt{d_k}}\right). \tag{4}$$

In this way, we obtain an improved attention calculation that can benefit from the predictability of a stationary sequence while maintaining the inherent temporal correlation of the original. However, the assumption that the linear embedding and feedforward layer have linear properties holds with difficulty in practice, and there are often numerous nonlinear activation factors. We need to compensate for this nonlinear information based on σ_x^2 and $1\left(\mu_Q^{\mathrm{T}}K^{\mathrm{T}}\right)$ by multilayer perceptron (MLP) to learn two hyperparameters, to obtain non-stationary attention mechanism calculation formula [19]

$$\begin{aligned}&\mathrm{Att}\left(Q',\ K',\ V'\right)\\&= \mathrm{Softmax}\left(\frac{\mathrm{MLP}\left(\sigma_x^2\right)Q'(K')^{\mathrm{T}} + MLP\left(\mu_Q^{\mathrm{T}}K^{\mathrm{T}}\right)}{\sqrt{d_k}}\right)V'.\end{aligned} \tag{5}$$

To our knowledge, ours is the first work to simplify and introduce this attention mechanism to multivariate temporal anomaly detection. Hence, we can obtain the values of all features at any moment t by a feedforward neural network (FNN) [11] based on historical serial time window data as

$$Y_t = \mathrm{FFN}\big(\mathrm{Att}\big(Q',\ K',\ V'\big)\big), \tag{6}$$

$$\mathrm{FFN}(x) = wx + b, \tag{7}$$

where w is the weight matrix and b is the bias term, $Y_t \in R^d$. Then, we calculate the MSE loss as:

$$\zeta_{\mathrm{MSE}} = \frac{1}{T_{\mathrm{train}}}\sum_{t\in T_{\mathrm{train}}}(Y_t - S_t)^2. \tag{8}$$

Spatial-dependency modeling

To comprehend the interconnections and relationships among features enables us to acquire insights through contrastive learning techniques based on the graph structure. This provides useful supervisory information that enhances abnormal information expression learning.

Graph structure learning

Following the work of GDN, the original training data feature d types of sensors at different graph nodes. Any sensor will be randomly initialized to the d_1 dimension embedding vector based on the sequence ID, and represented as

$$O_i \in \mathbb{R}^{d_1},\ for\quad i \in \{1, \ldots, d\}. \tag{9}$$

We calculate the similarity between sensor representations O_i and O_j for each time stamp

$$e_{ij} = \frac{O_i^{\mathrm{T}}O_j}{\|O_i\| \cdot \|O_j\|}. \tag{10}$$

$$A_{ji} = 1\{j \in topK(\{e_{ki} : k \in \mathcal{C}_i\})\}. \tag{11}$$

For a given sensor node$i$, we select the top K nodes with the highest similarity to as the candidate relations $\mathcal{C}_i$, where A_{ij} represents an edge from node i to node$j$, so as to obtain the homogeneous graph with only one node type and edge type $G(O, A)$.

In actual industrial systems, the various models of sensor functions can be placed in one of two categories according to the nature of their work. One is to perform control operations, and the other to monitor the indicators of the working environment, which we, respectively, call actuators and monitors, and we classify nodes according to these attributes. According to the node classification, we can get two types of edges, which connect nodes of either the same or different type. This constitutes our heterogeneous association graph $G_0(O, A, \alpha, \beta)$, where α and β, respectively, denote node and edge types.

Graph contrastive learning

Our main goal of graph contrastive learning based on the heterogeneous graph G_0 is to find beneficial spatio constraints. Its two main steps are data augmentation and sampling. We perform an initial spatio embedding of the original data of the graph nodes into the d_1-dimensional space to obtain a

representation of any node in the graph G_0, denoted as v_i, $v_i \in G_0$, $v_i \in \mathbb{R}^{d_1}$. It is worth noting that we do not introduce sensor sequence embedding information here, and O_i is only used to learn the graph structure.

We randomly lose a certain number of edges and node features in the graph G_0 with mask ratio ε. Then, we repeat this operation twice to obtain two new graphs G_1 and G_2. We next perform message aggregation and node updating for the two graphs by GNN, and for any node v_i, we obtain its re-characterization as

$$v_i^{l+1} = \sigma\left(\sum_{r\in R}\sum_{j\in N_i^r}\frac{1}{C_{i,r}}W_r^l v_j^l + W_0^l v_i^l\right), \tag{12}$$

where v_i^{l+1} is the feature vector of node i in layer l, R is the set of relations, W_r^l is the weight matrix of relations in layer i, W_0^l is the bias vector in layer l, $C_{i,r}$ the normalization factor, and N_i^r is the set of nodes whose relation to node i is r. In simpler terms, during the encoding process for each node, we calculate its feature vector by taking a weighted sum of the feature vectors of all the nodes connected to it in the previous layer. The weights assigned to each node feature vector are determined by a weight matrix associated with their relationship, along with a bias vector. This weighted sum is passed through a nonlinear activation function to introduce nonlinearity. Throughout this process, the weights undergo normalization to mitigate the influence of node degrees.

After data augmentation and graph node re-characterization, we perform positive and negative sampling. The traditional graph contrastive learning sampling strategy is to randomly fix the node of one of the views as the anchor point; only the same point in another view constitutes a positive sample pair, and the rest are negative sample pairs, with repeat traversal to obtain the set of all positive and negative sample pairs [39]. While this is intuitive and easy to implement, we found through experiments that such methods have limitations in heterogeneous graphs, and it is difficult to effectively use the representation of heterogeneous information between different node and edge types. To improve the sampling strategy, we sample positive and negative sample pairs in G_1 and G_2, and classify the relationship between sample pairs into two categories, one unrelated and the other inconsistent, where unrelated refers to sample pairs that are not directly connected, and inconsistent to those whose edges are connected but whose node types are inconsistent.

We first randomly sample a node in G_1 to obtain $v_i \in G_1$, $v_i \in \mathbb{R}^{d_1}$. Then, we find the set M of neighboring nodes of node i in G_2 to obtain v_j. A positive sample pair can be expressed as $P_i = (v_i, v_j)$, $v_i \in G_1$, $(v_j \in G_2) \cap (v_j \in \mathrm{M})$, and a negative sample pair as $N_i = (v_i, v_j)$, $v_i \in G_1$, $(v_j \in G_2) \cap (v_j \notin M)$. Then, we perform pooling on the sample pairs. For the generalization of the model, we use sum-pooling to obtain P_i, $N_i \in \mathbb{R}^{d_1}$. We repeat sampling K_1 times to obtain the set of positive and negative sample pairs denoted as $P = \{P_1, .., P_{k_1}\}$, $P \in \mathbb{R}^{k_1\times d_1}$, $N = \{N_1, .., N_{k_1}\}$, $N \in \mathbb{R}^{k_1\times d_1}$, and the objective function is

$$\zeta_1 = -\log\sum_{i=0}^{k_1}\sum_{j=0}^{\theta k_1}\frac{\exp\left(P_i(P_j)^T\right)}{\exp\left(N_i(N_j)^T/\tau\right)}, \tag{13}$$

where τ is the temperature coefficient. It is worth noting that the number of positive and negative samples is the same K_1 at this time. However, through experimental analysis, we found that increasing the number of negative samples will improve the learning ability of the model, for which we will learn a hyperparameter θ to control the sampling ratio of each group of positive and negative samples. The above sampling strategy is based on uncorrelated node structures.

The second sampling strategy is based on the inconsistency of node attributes. The difference with the first sampling strategy is that the first focuses on the characteristics of edges, in short, on whether or not they are connected as a basis for sampling positive and negative samples in two graphs. The second sampling strategy focuses on node characteristics, sampling different types of nodes on the two graphs as positive and negative sample pairs when they are already connected. Specifically, we randomly sample a node $v_i' \in G_1$, by finding the neighboring nodes of the v_i' node combined with M′ in G_2, in which a node with the same node type is randomly selected to form a positive sample pair. When learning the graph structure, we mentioned that our graph structure divides the nodes into actuator and monitor types, and we can denote the set composed of the two types of sensor nodes, respectively, as $\mathcal{A} = \{v_i'|type(v_i') = actuator\}$, $\mathcal{B} = \{v_i'|type(v_i') = monitor\}$. A positive sample pair can be expressed as $P_i' = (v_i', v_j')$, $v_i' \in G_1$, $v_j' \in \mathrm{M}'$, $\mathrm{type}(v_i') = type(v_j')$, and a negative sample pair as $P_i' = (v_i', v_j')$, $v_i' \in G_1$, $v_j' \in \mathrm{M}'$, $\mathrm{type}(v_i') \neq type(v_j')$. Then, we perform sum-pooling on the sample pairs to obtain P_i', $N_i' \in R^{d_1}$. We repeat sampling k_1 times, obtain the respective sets of positive and negative sample pairs $P' = \{P_1', .., P_{k_1}'\}$, $P' \in \mathbb{R}^{k_1\times d_1}$, $N' = \{N_1', .., N_{k_1}'\}$, $N' \in \mathbb{R}^{k_1\times d_1}$, and obtain the objective function

$$\zeta_2 = -\log\sum_{i=0}^{k_1}\sum_{j=0}^{\theta k_1}\frac{\exp\left(P_i'(P_j')^T\right)}{\exp\left(N_i'(N_j')^T/\tau\right)}. \tag{14}$$

Table 1 Details of SWaT and WADI datasets

Dataset	Features	Monitor	Actuator	Train	Test	Anomalies
Swat	51	24	27	47,520	44,991	14.01%
WADI	127	87	40	117,297	17,280	6.28%

In this way, we obtain the spatial-dependency modeling objective function

$$\zeta_{\text{spatio}} = \zeta_1 + \zeta_2, \tag{15}$$

which introduces a constraint on our temporal prediction task, thereby yielding the comprehensive objective function for our model

$$\mathcal{L} = \zeta_{\text{MSE}} + \zeta_{\text{spatio}}. \tag{16}$$

Anomaly detection

Having integrated spatio constraints into our prediction framework for sensor behavior, we compute an anomaly score to provide an explanation for anomalous behavior [15]. We calculate the discrepancy between the predicted and observed values of a sensor i at each time stamp t within the test set

$$E_{t,i} = \left|Y_{t,i} - S_{t,i}\right|, t \in T_{\text{test}}, i \in \{1, ..., d\}, \tag{17}$$

where T_{test} denotes all the time stamps in the test set, $Y_{t,i} \in Y_t$. Considering the varying sensitivities among the sensors, we apply robust normalization to the calculated deviations

$$\psi_{t,i} = \frac{E_{t,i} - \mu_i}{\sigma_i}, \tag{18}$$

where μ_i and σ_i are the median and inter-quartile range, respectively. Subsequently, we employ the maximum function to aggregate the anomaly scores of all sensors at time t, yielding the time stamp anomaly score $\psi_{t,i}$. If this surpasses the predefined threshold, we classify it as an anomaly occurring at that time. The way the thresholds are chosen can be optimized differently depending on the direction and distance [40], but to ensure fairness with baseline experiments, we use the maximum value of the system anomaly score at all moments in the validation set as the threshold [15, 18]. It is important to note that our training and validation sets solely consist of normal sample data, and only the test set contains abnormal samples. This is an important condition for us to use unsupervised training and to set thresholds.

Experiment

We performed experiments and conducted quantitative and qualitative analyses to compare our method with baseline approaches.

Dataset: The scarcity of high-dimensional series data originating from real-world industrial systems, incorporating anomalous instances, poses a challenge. However, there are two extensively employed cyber-physical systems (CPS) datasets available for research in time-series anomaly detection. These datasets, Secure Water Treatment (SWaT) and Water Distribution (WADI) [1], were generated and released by the iTrust Center for Research in Cybersecurity at the Singapore University of Technology and Design. Details are shown in Table 1.

Baselines: As our model is designed for anomaly detection based on multivariate time-series forecasting, our baselines fall into two categories. The first comprise outstanding work focused on detecting anomalies in multivariate time-series data, and can provide a visual comparison of our model's performance. The second category consists of transformer models that have recently demonstrated excellent performance in multivariate time-series forecasting. The baselines are as follows:

PCA [41]: Discovers a low-dimensional projection that effectively captures the majority of variance present in the data. The anomaly score, in this context, refers to the reconstruction error associated with this projection;

KNN [42]: Employs the distance between each data point and its top k nearest neighbor as an anomaly score;

DAGMM [43]: Combines deep autoencoders and a Gaussian mixture model to generate a low-dimensional representation and reconstruction error for each observation;

AE [44]: Consisting of an encoder and a decoder, reconstructs data samples, utilizing the reconstruction error as a metric for detecting anomalies;

LSTMVAE [45]: To leverage the advantages of both LSTM and VAE, the feedforward network in a VAE is replaced by LSTM, allowing for the computation of the reconstruction error, which serves as an error score;

Mad-GAN [21]: By employing generative adversarial networks (GANs) in conjunction with a reconstruction-based approach, error scores are computed for each sample;

GDN [15]: Can capture both spatio and temporal dependencies, representing multivariate time-series data as graphs, and utilizing GNNs to learn the representations of nodes and

edges within them. Learned representations are fed into a sensor future behavior prediction module, which enables the detection of anomalies in time-series data;

TranAD [18]: Utilizing an innovative self-attentive mechanism, it incorporates self-regulation grounded in focus scores for resilient multi-modal feature extraction. The model employs adversarial training for stability and integrates reconstruction loss for anomaly detection.

Informer [46]: Employing a multilayer Transformer architecture, this model enhances the weight calculation method for attention, and incorporates techniques such as time-varying positional encoding and length masking, which enable efficient processing of long sequences and accurate predictions across multiple time steps;

Autoformer [47]: This adaptive transformer model introduces an adaptive feature selection module and adaptive transformation module to dynamically learn the crucial features and transformation methods of time-series data, so as to enhance the accuracy and generalization of sequence prediction.

Non-stationary Transformer (Nsformer) [19]: Designed for non-stationary time-series data, a progressive learning mechanism allows for adaptive learning of the dynamic nature of a sequence. Information from both historical and future data is leveraged during the prediction process, resulting in improved accuracy of sequence prediction.

Evaluation metrics: To ensure generalizability and fairness, we chose the evaluation indicators in the literature: precision, recall, and F1-score [15, 21]

$$\text{Pre} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{19}$$

$$\text{Rec} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{20}$$

$$F_1 = 2 \times \frac{\text{Pre} \times \text{Rec}}{\text{Pre} + \text{Rec}}, \tag{21}$$

where TP is the correctly detected anomaly, FP is the falsely detected anomaly, TN is the correctly assigned normal, and FN is the falsely assigned normal.

Implementation: We used the PyTorch-1.8.1 library to train all the models, and split the trained time-series into 90% training data and 10% validation data. We used the Adam optimizer with a learning rate of 0.01 and an epoch of 10. Some important hyperparameters are as follows: in the temporal module, the sliding window length was w $= 15$, the number of transformer encoder layers was $L = 3$, the number of heads was 4, and $d_k = 64$; in the spatio module, $\varepsilon = 0.2$, $\mathrm{d}_1 = 64$, $k = 20$, $k_1 = 10$, $\theta = 5$, $\tau = 0.25$.

Research question 1: anomaly detection performance

We present the anomaly detection performance of our model and the baseline approaches in Table 2, in terms of precision, recall, and F1-score on the SWaT and WADI datasets. The results indicate that our model outperforms the baselines in terms of recall and F1-score on both datasets, achieving F1-scores of 0.82 and 0.59 for SWaT and WADI, respectively. While the GDN baseline achieves higher precision scores, the trade-off between precision and recall is inevitable. In practice, maintenance technicians with domain expertise tend to prioritize high sensitivity over specificity to avoid missing any critical events worthy of future reference [36]. Therefore, the goal of our model optimization is to maximize the recall while optimizing the F1-score.

We observe that the improvement rate on the WADI dataset is higher than on the SWaT dataset. We attribute this to its larger data volume and feature dimensions, which result in a more complex spatio topology. By utilizing graph-based contrastive learning, our model can uncover more valuable spatio constraints and guide the learning process to enhance the representation of anomalous behavior. By analyzing the experimental data, it is worth noting that the TranAD model performs very well in SWaT, especially the Rec metrics, but does not perform as well as our model on the WADI dataset. This is because TranAD is able to learn feature relevance through adversarial learning and meta-learning. However, meta-learning uses limited data and lowers the learning threshold, so although it can detect more anomalous moments, it can easily misclassify some anomalous moments, which makes the accuracy much lower. Our model, on the other hand, has a good ability to learn the non-stationarity of the original data through improved attention, which is more expressive on the WADI dataset with higher data dimensions, and is able to balance the conditions of the two metrics Prec and Rec, to obtain excellent F1 values.

Research question 2: ablation

To demonstrate the necessity of our model components in achieving the optimal detection performance shown in Table 2, we conducted an ablation study, whose results are presented in Table 3.

Temporal: When we replace the temporal modeling component of our model with a regular Transformer network that only includes an encoder, there is a significant decrease in precision, recall, and F1-score. Specifically, the F1-scores decreased by 0.07 and 0.03 on the SWaT and WADI datasets, respectively. This indicates that applying the attention mechanism directly to normalized data resulted in the loss of

Table 2 Anomaly detection performance on SWaT and WADI datasets in terms of precision (Prec) (%), recall (Rec) (%), and F1-score (F1)

Method	SWaT			WADI		
	Prec	Rec	F1	Prec	Rec	F1
PCA	24.92	21.63	0.23	39.53	5.63	0.10
KNN	7.83	12.13	0.08	7.76	7.75	0.08
AE	72.63	52.63	0.61	34.35	34.35	0.34
DAGMM	27.46	69.52	0.39	54.44	26.99	0.36
LSTMVAE	96.24	59.91	0.74	54.44	26.99	0.36
Mad-GAN	98.97	63.74	0.77	41.44	33.92	0.37
GDN	**99.35**	68.12	0.81	**97.50**	40.19	0.57
TranAD	97.21	**72.42**	**0.83**	35.25	**82.96**	0.49
ours	98.29	70.84	0.82	93.42	43.23	**0.59**

Best performance for each evaluation metric is bolded; second-best is underlined. Results are partly from work of Deng [15]

Table 3 Ablation test results

Method	SWaT			WADI		
	Prec	Rec	F1	Prec	Rec	F1
Ours	**98.29**	**70.84**	**0.82**	**93.42**	**43.23**	**0.59**
Temporal	91.31	63.21	0.75	88.22	41.50	0.56
Spatio	96.82	68.13	0.79	90.32	40.34	0.55

Best performance for each evaluation metric is bolded

Table 4 Prediction task (MSE, MAE) and detection task (Prec, Rec, F1) performance test data

Method	SWaT					WADI				
	MSE	MAE	Prec	Rec	F1	MSE	MAE	Prec	Rec	F1
Trans [5]	0.19	0.27	91.31	63.21	0.75	0.22	0.28	88.22	41.50	0.56
In [34]	0.15	0.19	98.13	60.15	0.74	**0.17**	**0.23**	**94.13**	40.41	0.58
Auto [35]	**0.13**	**0.17**	**99.24**	62.81	0.77	0.19	0.24	91.35	41.53	0.57
Ns [13]	0.17	0.21	95.13	67.41	0.79	0.20	0.25	92.42	41.23	0.57
ours	0.16	0.23	98.29	**70.84**	**0.82**	0.21	0.24	93.42	**43.23**	**0.59**

Best performance for each evaluation metric is bolded

anomalous information present in the original data. Additionally, it caused excessive stationarity during deep model training, leading to attention weights that were difficult to differentiate across sequences. Our model emulates the attention mechanism on the original data, which helps capture and express the information related to anomalous behavior.

Spatio: When we directly remove the spatio dependency modeling by not calculating ζ_{spatio}, and rely solely on the time-series prediction results for detection, there are decreases in the F1-score of 0.03 and 0.04 on the SWaT and WADI datasets, respectively. This indicates that utilizing graph contrastive learning to search for supervisory signals can guide the model to learn spatio dependencies between features. It strengthens the constraints during model training, allowing for more comprehensive training of the model.

Research question 3: interpretability

We investigate an interesting question that we discovered during our experiments. In our prediction-based multivariate time-series anomaly detection model, the upstream task is one of general multivariate time-series prediction, and downstream is a binary classification anomaly detection task. In multivariate time-series prediction tasks, MSE and MAE are commonly used performance evaluation metrics [46, 47]. We wish to explore whether the downstream anomaly detection

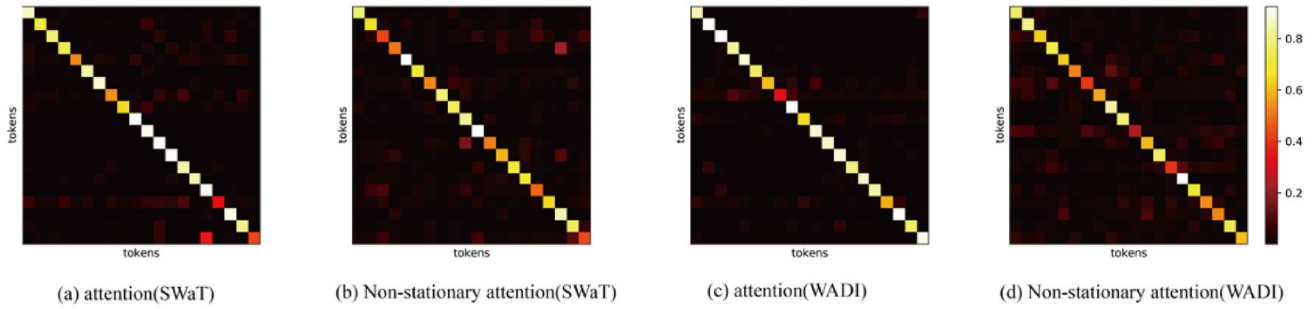

Fig. 2 Attention visualization on WADI dataset

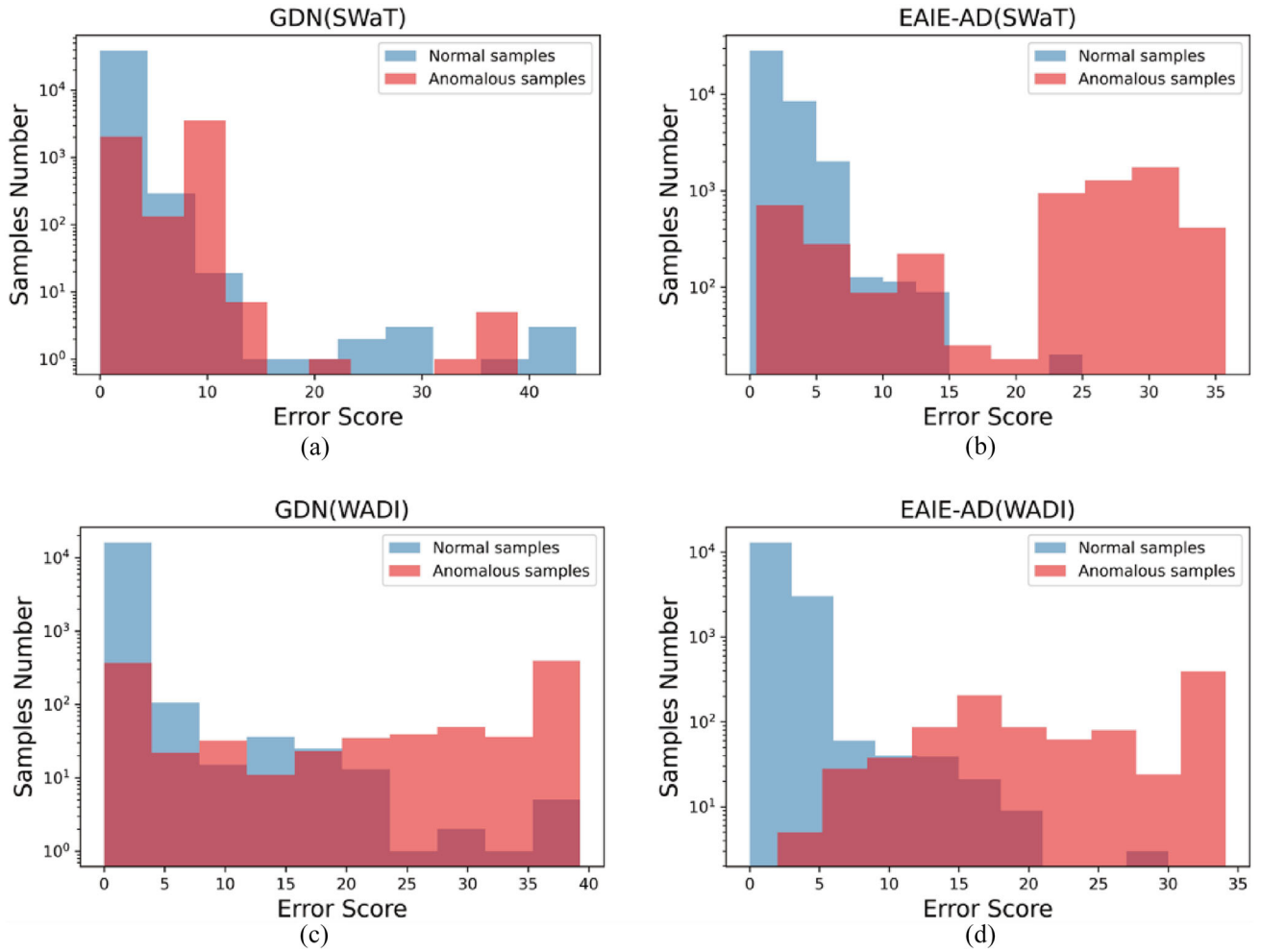

Fig. 3 Sample error score distribution

task can directly benefit from the optimized MSE and MAE metrics in the prediction task. To study this question, we employed various state-of-the-art transformer models that have shown excellent performance in multivariate time-series prediction tasks. We observed the impact of MSE and MAE during the prediction phase on the precision, recall, and F1-score in the detection task. The results are presented in Table 4.

As mentioned in Research Question 1, we place more emphasis on recall and F1-score as our primary objectives in the practical process. On the SWaT dataset, Autoformer demonstrates the best performance in the prediction task, with MSE and MAE values of 0.13 and 0.17, respectively. However, its recall and F1-scores are not as high as those of Ns [19] and our model. On the WADI dataset, Informer performs best in the prediction task, with MSE and MAE

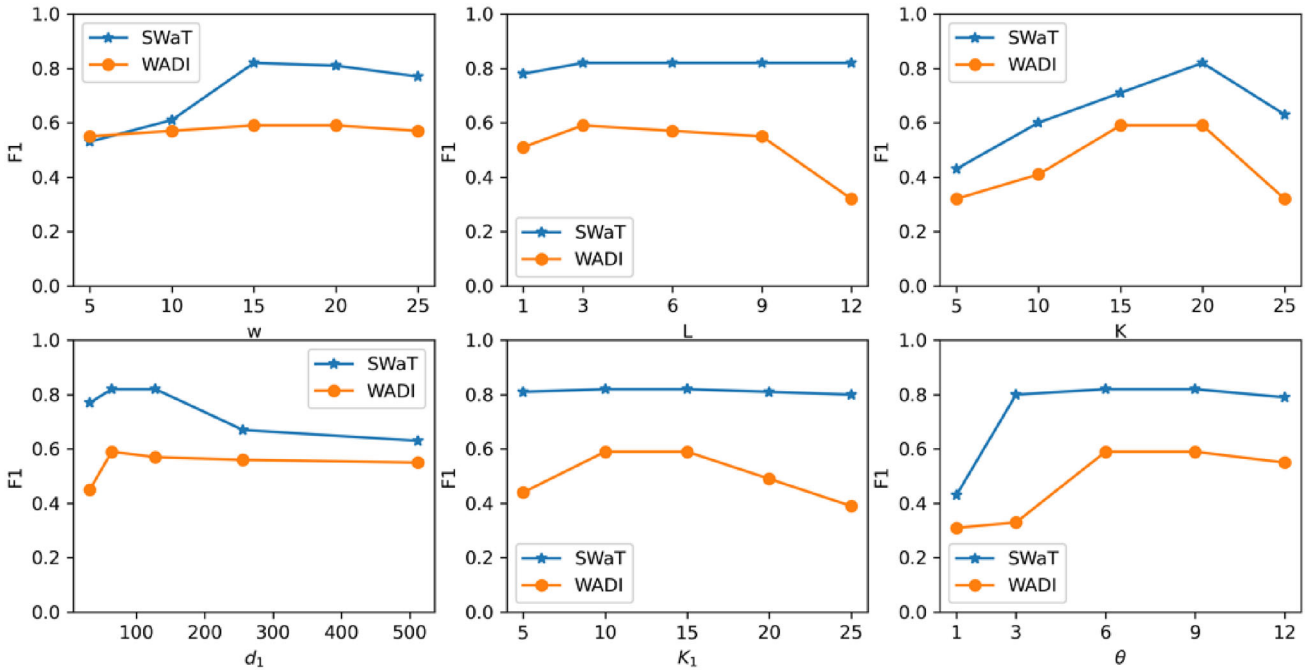

Fig. 4 Hyperparametric sensitivity experiments

values of 0.17 and 0.23, respectively. Nevertheless, the final detection performance is still not optimal. Through experimental data analysis, we argue that optimizing the MSE and MAE metrics for the upstream prediction task is a nonlinear relationship in terms of gain for the downstream detection task and that there is a critical point that prevents sustained gain. Hence, we introduced spatial-dependence modeling, and although the MSE and MAE of the prediction task are numerically inferior to those of other models, the recall and F1-score of the final detection task can be surpassed. In further analysis, we believe that we should focus more on whether our model can learn enough information representation of anomalous behavior and get enough constraints in the prediction phase. Hence, our model tries to strengthen the anomalous behavior information expression at two levels. First, different from the mainstream Transformer, we execute the attention mechanism on the original un-normalized non-stationary data to avoid ignoring anomalous behavior information lost due to normalization. Second, we can get more spatio constraints on the anomalous behavior by mining the spatio dependencies between the features through graph contrastive learning.

To further enhance interpretability, we visualize self-attention plots on our dataset WADI using the standard attention mechanism and the non-stationary attention mechanism employed in our model.

As shown in Fig. 2, the horizontal and vertical axes are the normalized input data. Figure 2(a) and (c) visualizes the execution of standard attention, and Fig. 2(b) and (d) visualizes the non-stationary attention mechanism. Comparing the figures in the same dataset, we can clearly find that in Fig. 2(a) and (c), due to the effect of normalization, the attention weights focus on the diagonal line, and all input tokens tend to focus on themselves, thus producing an over-stationarity problem, while in Fig. 2(b) and (d), our model can focus on more information of other tokens in the input sequence and produce effective variability in attention, helping our model learn to express more information about the anomalous behavior.

To further illustrate the ability of the error scores constructed by our model to discriminate between abnormalities, we visualize a comparative plot of the distribution of abnormal scores for positive and negative samples. Given the balance of precision and recall, we visualize the distribution of GDN abnormality scores for comparison and contrast to facilitate comparative observations, as shown in Fig. 3. Compared to GDN, our model obtains a better distribution of normal/abnormal data, especially in the SWaT dataset, where error scores of normal data remain low and concentrated, indicating that our model can more effectively separate normal from abnormal embedding. In the WADI dataset, there are still many normal sample points with excessive error scores, and the presence of these noisy points is one of the main reasons why detection performance on the WADI dataset is inferior to that on SWaT.

To explore the sensitivity of the hyperparameters in our model, we conducted tests on six important hyperparameters,

Table 5 The detail in module time complexity and parameter

Block	Time complexity	Parameter
Temporal	$O(3d_1^2w^2)$	121,344
Spatial	$O(20L^2d_1^2) + k_1^2d_1$	17,120

as shown in Fig. 4, sliding window length (w), the number of transformer encoder layers (L), the number of node neighbors (K), the graph embedding dimension (D_1), the contrastive learning sample times(K_1), and sampling ratio (θ). We can find many hyperparameters which have multiple optimal values. Considering the model inference speed, we chose relatively small values for all hyperparameters. It is also worth noting that the two most influential parameters are K and θ. As for K, we need to ensure that each node has enough neighbors, so that we are able to capture more useful information when we perform information transfer in the graph, but when our number of neighbors exceeds 20, an over-smoothing phenomenon occurs, and the flow of information in the whole graph tends to be like a fully connected graph. As for θ, when theta is too small, the difference between positive and negative samples cannot be fully explained and the model is underfitted, and when theta is too large a large amount of noise is learned and the inference efficiency is significantly reduced. Finally, the details in module time complexity and parameter are shown in Table 5.

Conclusion

In this work, we proposed a novel end-to-end multivariate time-series spatial–temporal anomaly detection model (EAIE-AD), which is capable of deep modeling in both temporal and spatio dimensions, compensating for anomalous behavioral information. Our model is versatile, catering to both temporal and graph domains. It employs self-supervised learning specifically tailored for sparse data, making it well suited for scenarios characterized by data sparsity and complexity. This adaptability is particularly valuable in applications that demand the simultaneous capture of spatial–temporal characteristics, such as traffic flow detection and anomaly detection in intelligent systems. Through experiments, we verified that the performance of our model outperforms the baseline on two generic real datasets, while we explored the relationship between performance in the prediction task and performance gain in the final detection task in mainstream models through visualization and analysis of experimental results, enhancing the interpretability of our model. In our future work, we will further dig deeper into the spatio topology properties, including spatio-localization and path compensation, as a way to provide more realistic application scenarios for our model.

Acknowledgements Thanks to the original dataset provider: iTrust, Centre for Research in Cyber Security, Singapore University of Technology and Design.

Author contributions DG and YW contributed to conception and design of the study. YC administered the experiments. YM performed code development. SC performed data preprocessing. DG wrote the first draft of the manuscript. All authors contributed to manuscript revision, and read and approved the submitted version.

Funding This work was developed with internal research funding from the National Natural Science Foundation of China (61937001).

Data availability The dataset in this article can be found at https://itrust.sutd.edu.sg/itrust-labs-home/itrust-labs_iot/.

Declarations

Conflict of interest The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Chen Z, Peng Z, Zou X, Sun H (2022) Deep learning based anomaly detection for muti-dimensional time series: a survey. Commun Comput Inf Sci. https://doi.org/10.1007/978-981-16-9229-1_5
2. Zhang J, Pan L, Han Q-L, Chen C, Wen S, Xiang Y (2022) Deep learning based attack detection for cyber-physical system cybersecurity: a survey. IEEE/CAA J Autom Sin 9:377–391. https://doi.org/10.1109/jas.2021.1004261
3. Song X, Wu N, Song S, et al. Switching-Like Event-Triggered State Estimation for Reaction–Diffusion Neural Networks Against DoS Attacks[J]. Neural Processing Letters 2023:1–22
4. Dong X, He S, Stojanovic V (2020) Robust fault detection filter design for a class of discrete-time conic-type non-linear markov jump systems with jump fault signals. IET Control Theory Appl. https://doi.org/10.1049/iet-cta.2019.1316
5. Xie X, Ning W, Huang Y, Li Z, Yu S, Yang H (2022) Graph-based Bayesian network conditional normalizing flows for multiple time series anomaly detection. Int J Intell Syst 37:10924–10939. https://doi.org/10.1002/int.23027
6. Song X, Wu N, Song S, Zhang Y, Stojanovic V (2023) Bipartite Synchronization for Cooperative-Competitive Neural Networks with Reaction-Diffusion Terms via Dual Event-Triggered Mechanism. Neurocomputing. https://doi.org/10.1016/j.neucom.2023.126498

7. Charco JL, Roque-Colt T, Egas-Arizala K, Pérez-Espinoza CM (2021) Cruz-Chóez A (2021) Using multivariate time series data via long-short term memory network for temperature forecasting. In: Botto-Tobar M, Zamora W, Larrea Plúa J, Bazurto Roldan J, Santamaría Philco A (eds) Systems and Information Sciences. Springer International Publishing, Cham, pp 38–47
8. Maini S, Aggarwal AK (2018) Camera position estimation using 2D image dataset. Int J Innov Eng Technol 10:199–203
9. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN et al (2017) Attention is all you need. Curran Associates Inc
10. Tuli S, Casale G, Jennings NR (2022) Tranad: deep transformer networks for anomaly detection in multivariate time series data. Proc VLDB Endow 15:1201–1214. https://doi.org/10.14778/3514061.3514067
11. Maru C, Brandherm B, Kobayashi I (2022) Combining transformer with a discriminator for anomaly detection in multivariate time series. In: 2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS). p. 1–7
12. Wiederer J, Bouazizi A, Troina M, Kressel U, Belagiannis V (2021) Anomaly detection in multi-agent trajectories for automated driving. arXiv preprint arXiv:211007922
13. Su Y, Zhao Y, Niu C, Liu R, Sun W, Pei D (2019) 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. p. 2828–2837
14. Xu J, Wu H, Wang J, Long M (2021) Anomaly transformer: Time series anomaly detection with association discrepancy. arXiv preprint arXiv:211002642
15. Deng A, Hooi B (2021) Graph neural network-based anomaly detection in multivariate time series. Proc Innov Appl Artif Intell 35:4027–4035. https://doi.org/10.1609/aaai.v35i5.16523
16. Chen Z, Chen D, Zhang X, Yuan Z, Cheng X (2022) Learning graph structures with transformer for multivariate time-series anomaly detection in IoT. IEEE Internet Things J 9:9179–9189. https://doi.org/10.1109/jiot.2021.3100509
17. Han S, Woo SS (2022) "Learning Sparse Latent Graph Representations for Anomaly Detection in Multivariate Time Series." In: Zhang A, Rangwala H (eds), KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 2977–2986). Washington, DC, USA: ACM. DOI: https://doi.org/10.1145/3534678.3539117
18. Tuli S, Casale G, Jennings NR (2022) Tranad: deep transformer networks for anomaly detection in multivariate time series data. Proc VLDB Endowment 15(6):1201–1214
19. Liu Y, Wu H, Wang J, Long M (2022) Non-stationary transformers: exploring the stationarity in time series forecasting. arXiv preprint arXiv:220514415
20. Yokkampon U, Chumkamon S, Mowshowitz A, Hayashi E (2020) Anomaly detection using variational autoencoder with spectrum analysis for time series data. In: 2020 Joint 9th International Conference on Informatics, Electronics & Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision & Pattern Recognition (icIVPR). IEEE
21. Fu Y, Xue F (2022) MAD: Self-supervised masked anomaly detection task for multivariate time series. In: 2022 International Joint Conference on Neural Networks (IJCNN). IEEE
22. Malhotra P, Vig L, Shroff GM, Agarwal P (2015) Long short term memory networks for anomaly detection in time series. In: The European Symposium on Artificial Neural Networks
23. Sagheer A, Kotb M (2019) Unsupervised pre-training of a deep LSTM-based stacked autoencoder for multivariate time series forecasting problems. Sci Rep 9:19038. https://doi.org/10.1038/s41598-019-55320-6
24. Han S, Woo SS (2022) Learning sparse latent graph representations for anomaly detection in multivariate time series. Association for Computing Machinery, New York, NY, USA, p. 2977–2986
25. Ahmad S, Purdy S (2016) Real-time anomaly detection for streaming analytics. arXiv preprint arXiv:160702480
26. Ye-Kui Q, Ming C (2010) A multivariate online anomaly detection algorithm based on SVD updating. J Electron Inform Technol 32:2404–2409. https://doi.org/10.3724/SP.J.1146.2009.01342
27. Malhotra P, Ramakrishnan A, Anand G, Vig L, Agarwal P, Shroff G (2016) LSTM-based encoder-decoder for multi-sensor anomaly detection. arXiv preprint arXiv:160700148
28. Aggarwal AK (2020) "Atmospheric Delay Correction of RINEX GPS Data"
29. Aggarwal AK (2023) "A hybrid approach to GPS improvement in urban canyons"
30. Chen N, Tu H, Duan X, Hu L, Guo C (2023) Semisupervised anomaly detection of multivariate time series based on a variational autoencoder. Appl Intell 53:6074–6098. https://doi.org/10.1007/s10489-022-03829-1
31. Kong F, Li J, Jiang B, Wang H, Song H (2023) Integrated generative model for industrial anomaly detection via bidirectional LSTM and attention mechanism. IEEE Trans Ind Inform 19:541–550. https://doi.org/10.1109/TII.2021.3078192
32. Jeong Y, Yang E, Ryu J, Park I, Kang M (2023) AnomalyBERT: self-supervised transformer for time series anomaly detection using data degradation scheme
33. Aggarwal AK, Sato T, Oishi T, Ono S, Ikeuchi K (2014) "Improving GPS Position Accuracy by Identification of Reflected GPS Signals Using Range Data for Modeling of Urban Structures." SEISAN KENKYU
34. Xiao J, Aggarwal AK, Kiran U, Katiyar V, Avtar R (Year) "Deep Learning-based spatiotemporal fusion of unmanned aerial vehicle and satellite reflectance images for crop monitoring"
35. Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:160902907
36. Veličković P, Cucurull G, Casanova A, Romero A, Lio P, Bengio Y (2017) Graph attention networks. arXiv preprint arXiv:171010903
37. He Q, Wang G, Wang H, Chen L (2023) Multivariate time-series anomaly detection via temporal convolutional and graph attention networks. J Intell Fuzzy Syst 44:5953–5962. https://doi.org/10.3233/jifs-222554
38. Zhao H, Wang Y, Duan J, Huang C, Cao D, Tong Y, et al. (2020) Multivariate time-series anomaly detection via graph attention network. In: 2020 IEEE International Conference on Data Mining (ICDM). IEEE
39. Zhu Y, Xu Y, Yu F, Liu Q, Wu S, Wang L (2020) Deep graph contrastive representation learning. arXiv preprint arXiv:200604131
40. Aggarwal AK, Banno A, Ono S, Oishi T, Ikeuchi K (2013) "Global Coordinate Adjustment of the 3D Survey Models under Unstable GPS Condition." SEISAN KENKYU
41. Shyu M-L, Chen S-C, Sarinnapakorn K, Chang L (2003) A novel anomaly detection scheme based on principal component classifier. In: IEEE Foundations and New Directions of Data Mining Workshop, in conjunction with the Third IEEE International Conference on Data Mining (ICDM'03)
42. Angiulli F, Pizzuti C (2002) Fast outlier detection in high dimensional spaces. Principles of data mining and knowledge discovery. Springer, pp 15–27
43. Zong B, Song Q, Min MR, Cheng W, Lumezanu C, Cho D et al (2018) Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. Springer
44. Aggarwal C (2013) Outlier analysis. Springer

45. Park D, Hoshi Y, Kemp CC (2018) A Multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. IEEE Robot Autom Lett 3:1544–1551. https://doi.org/10.1109/lra.2018.2801475
46. Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H et al (2021) Informer: beyond efficient transformer for long sequence time-series forecasting. Proc Innov Appl Artif Intell 35:11106–11115. https://doi.org/10.1609/aaai.v35i12.17325
47. Wu H, Xu J, Wang J, Long M (2021) Autoformer: decomposition transformers with auto-correlation for long-term series forecasting. Curran Associates Inc, pp 22419–22430