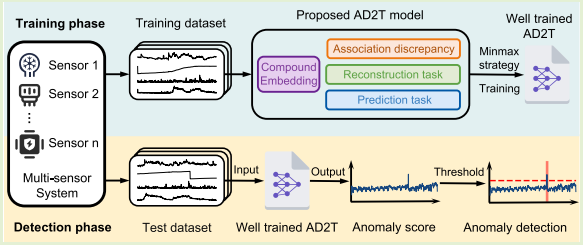
AD2T: Multivariate Time-Series Anomaly Detection With Association Discrepancy Dual-Decoder Transformer

Zezhong Li, Wei Guo, Jianpeng An, Qi Wang, Yingchun Mei, Rongshun Juan, Tianshu Wang, Yang Li, and Zhongke Gao, *Senior Member, IEEE*

***Abstract*—Multivariate time-series (MTS) anomaly detection is of great importance in both condition monitoring and malfunction identification in multisensor systems. Current MTS anomaly detection approaches are typically based on reconstruction, prediction, or association discrepancy learning algorithms. These methods detect anomalies by learning hidden representations of entire sequences, modeling dependencies at a single time-step level, or calculating an association-based metric inherently distinguishable between regular and deviant points. However, most existing methods typically fail to leverage all three types of models simultaneously to enhance overall performance as well as often disregard the correlations between different sensors. To address the issues above, this article proposes a novel deep learning-based unsupervised MTS anomaly detection algorithm called association discrepancy dual-decoder transformer (AD2T). AD2T employs a dual-decoder architecture to accommodate reconstruction, prediction, and association discrepancy learning tasks, thereby effectively utilizing information across these tasks to better characterize MTS data. We further develop a min–max training strategy to jointly optimize all the aforementioned tasks. Additionally, we propose a compound embedding module based on dilated causal convolution to simultaneously capture correlations in both temporal and sensor dimensions. Extensive empirical studies on five multisensor system datasets from the aerospace, server, and water treatment domains have demonstrated the superiority of our method, achieving an average improvement of 1.96% in the $F1$-score compared to state-of-the-art (SOTA) methods.**



***Index Terms*— Anomaly detection, association discrepancy, dual-decoder, multivariate time series (MTS), transformer.**

I. Introduction

MODERN multisensor systems (e.g., spacecraft [1], server machines [2], [3], and manufacturing process facilities [4], [5], [6]) have evolved into complex, heterogeneously integrated entities to provide rich functionalities. Ensuring the security and reliability of these systems is of great importance, as sophisticated and hidden faults can lead to unplanned downtime or even catastrophic consequences, affecting the economy, environment, and human lives. In recent years, with advancements in sensing and measurement technologies, thousands of sensors have been deployed across these systems for condition monitoring. This deployment has generated extensive multivariate time-series (MTS) data, rich with information about the intricate dynamics of systems. Therefore, it is imperative to introduce automated anomaly detection approaches for multisensor systems, which serve as a replacement for the continuous manual surveillance by inspectors, leveraging the rich MTS data to enhance detection capabilities and trigger prompt troubleshooting.

This work focuses on the challenges of MTS anomaly detection in multisensor systems. Currently, various approaches have been proposed to address these challenges. Most of them are based on an unsupervised scheme, due to the scarcity of manual labeling [7]. Benefiting from great nonlinear representation capabilities, deep learning models perform better than traditional statistical methods [8], [9] and classical machine learning (ML) approaches [10], [11], [12] in modeling complex structures in MTS, attracting widespread attention from

Received 14 January 2025; accepted 17 February 2025. Date of publication 26 February 2025; date of current version 2 April 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 62373278, and in part by the National Natural Science Foundation of China under Grant 52005365. The associate editor coordinating the review of this article and approving it for publication was Prof. Rui Yuan. *(Zezhong Li and Wei Guo are co-first authors.) (Corresponding author: Zhongke Gao.)*

Zezhong Li is with the School of Mechanical Engineering, Tianjin University, Tianjin 300350, China (e-mail: lizz1998@tju.edu.cn).

Wei Guo, Jianpeng An, Rongshun Juan, Tianshu Wang, Yang Li, and Zhongke Gao are with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: zhongkegao@tju.edu.cn).

Qi Wang and Yingchun Mei are with Huadian Heavy Industries Company Ltd., Beijing 100070, China, and also with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China.

Digital Object Identifier 10.1109/JSEN.2025.3543835

researchers in recent years. For instance, recurrent neural network (RNN)-based models [13], particularly its variants such as long short-term memory (LSTM) [14] and gated recurrent unit (GRU) networks [15], have been widely used to capture temporal patterns and reconstruct normal sequences. Subsequently, the reconstruction errors are utilized to detect anomalies.

Although RNN-based methods have achieved significant progress, they suffer from limitations in efficiency, primarily due to the sequential processing nature of recursive models and back-propagation through time (BPTT) [16]. Recent transformer models [17], by leveraging self-attention mechanisms, correlate information across all elements of sequential data, thus benefiting from the parallel computing capabilities provided by graphical processing units (GPUs). Moreover, positional encoding enables transformers to remember the order of sequences while avoiding the challenges associated with sequential data processing. Therefore, this article employs a transformer as the backbone for MTS anomaly detection, taking into account both accuracy and efficiency.

After selecting a model backbone, the design of the anomaly detection scheme also necessitates determining the calculation mechanism for the point-wise anomaly score. Depending on the calculation mechanism, unsupervised MTS anomaly detection methods can be categorized into three types.

1) Reconstruction-based methods detect anomalies by reconstructing the entire sequence, while points are reported as anomalies if the reconstruction error exceeds a specified threshold [2], [18], [19], [20], [21].
2) Prediction-based methods involve forecasting the next value of the sequence and identifying anomalies based on the prediction error [1], [22], [23].
3) Association discrepancy-based methods involve calculating an association-based metric, which is inherently distinguishable between regular and deviant points [24], [25]. Anomaly scores derived from reconstruction error or prediction error primarily provide point-wise variations in MTS for anomaly detection. In contrast, anomaly scores based on association discrepancy incorporate the relational information between individual points and the broader sequence context, with each approach having its own focus. However, existing methods have not yet achieved integration of all three types of methods. Furthermore, whether point-by-point change information or the association between points and series, both only provide temporal dynamics and lack a description of the interrelationships among different sensors. Exploring mechanisms for integrating anomaly scores of different styles and capturing intersensor relationships presents the potential to address the aforementioned challenges.

In this article, we propose a novel deep learning-based unsupervised MTS anomaly detection algorithm called association discrepancy dual-decoder transformer (AD2T). The AD2T framework comprises two main subnetworks: a shared encoder network and a dual-decoder structure. Specifically, a dilated causal convolution-based embedding module is introduced to extract both temporal and intersensor features from MTS input data in parallel and facilitates information fusion. Subsequently, an encoder equipped with anomaly attention is employed to learn the latent representations of the MTS and the association discrepancies between each time point and the series context. The learned latent representations are shared by a subsequent dual-decoder, one of which is used for reconstructing the entire sequence, while the other, a transformer-based decoder, is employed for point-wise prediction of the next value in the MTS. Furthermore, a joint optimization objective that combines reconstruction-based loss, prediction-based loss, and association discrepancy is implemented to better characterize MTS data. In summary, the main contributions of our work are as follows.

1) A novel AD2T framework is proposed for MTS anomaly detection, employing a dual-decoder architecture that jointly optimizes tasks for reconstruction, prediction, and association discrepancy to better characterize MTS data and enhance anomaly detection performance.
2) A training mechanism based on the min–max strategy is developed, aiming to simultaneously reduce reconstruction and prediction losses while amplifying the normal–abnormal distinguishability of the association discrepancy.
3) A dilated causal convolution-based embedding module is introduced to concurrently capture the temporal dynamics and the interrelationships between different sensors in MTS.
4) Extensive experiments are conducted on five real-world datasets from three fields: aerospace, server monitoring, and water treatment. Compared to state-of-the-art (SOTA) methods, AD2T achieves average improvements of 2.49%, 1.41%, and 1.94% in precision, recall, and $F1$-score, respectively, across the five datasets.

The remainder of this article is organized as follows: Section II discusses existing methods related to MTS anomaly detection. Section III formulates the anomaly detection problem and presents the preliminaries relevant to this article. Section IV details the proposed AD2T framework. Section V presents the evaluations compared with state-of-the-art methods using extensive real-world datasets and analyzes the results. Section VI concludes this article and discusses the future work.

II. Related Work

In recent years, research on MTS anomaly detection has become a hot topic. Compared to normal operating conditions in multisensor systems, anomalies occur less frequently and may not be sufficiently recorded. Furthermore, manual labeling of anomalies is a costly, labor-intensive task, which also requires specialized domain knowledge. Consequently, most MTS anomaly detection methods are unsupervised. In the early period, traditional statistical methods and classical ML approaches such as the Kalman filter (KF) [8], density estimation [9], local outlier factor (LOF) [10], and isolation forest (IF) [11] have been proposed to address the problem of MTS anomaly detection. Although these methods have been proven effective, they are sensitive to noise and struggle to model complex temporal dynamics.

With advancements in deep learning technologies, many researchers have employed deep neural networks to enhance the performance of anomaly detection methods. Previous unsupervised deep MTS anomaly detection methods generally can be divided into three categories: reconstruction-based, prediction-based, and association discrepancy-based approaches.

A. Reconstruction-Based Methods

Reconstruction-based methods learn hidden representations of sequences and reconstruct them, while point-wise reconstruction losses are calculated to detect anomalies. The autoencoder (AE) is a commonly used network architecture for MTS reconstruction. Both OmniAnomaly [2] and InterFusion [21] utilize the structure of the RNN and variational AE (VAE) to model the temporal dependencies of MTS. MSCRED [18] employs a convolutional encoder–decoder and a convolutional LSTM (ConvLSTM) network to learn the intrinsic representations of MTS. It calculates the reconstruction loss of the signature matrix to detect anomalies. In addition to AEs, GAN-based architectures are frequently used to amplify the reconstruction error and enhance model performance [20]. For instance, BeatGAN [19] introduces an adversarial learning reconstruction framework and employs dynamic time warping (DTW) for data augmentation to robustly detect anomalies. Though reconstruction-based methods are effective and easy to understand, they may inadvertently reconstruct anomalies, which leads to a decline in accuracy.

B. Prediction-Based Methods

Prediction-based methods forecast the next time point's value using historical data and detect anomalies by calculating point-wise forecasting loss, which is determined by comparing forecast values with actual observations. RNNs and Transformers are commonly used network architectures for MTS prediction. For instance, LSTM-NDT was proposed by Hundman et al. [1], which constructs an LSTM for each channel of the MTS to predict future values and employs a nonparametric approach to dynamically determine thresholds for anomaly detection. Wang et al. [22] employed an attention-based LSTM to perform failure prediction on MTS data from hard disk drives, while using a multi-instance learning approach to handle the problem of data imbalance. GTA [23] combines graph learning and Transformers for MTS prediction and introduces a multibranch attention mechanism to accelerate inference speed. In general, prediction-based methods excel at modeling complex trends, cycles, and seasonal variations in normal MTS data to detect anomalies. However, these methods may falsely identify noise as anomalies, which reduces their performance.

C. Association Discrepancy-Based Methods

Association discrepancy was first proposed by Xu et al. [24] as a metric that describes the difference between prior association and series association in MTS. Prior association refers to the correlation between a data point and its adjacent points in the MTS, whereas series association refers to the correlation between a point and the overall series context. Since anomalous points typically have more difficulty establishing connections with the series context compared to normal points, the discrepancy between their prior and series associations is often insignificant. This pattern can be utilized to identify anomalies. TPAD [25] integrates association discrepancy with adversarial training to enhance the detection accuracy of subtle anomalies. In practice, association discrepancy is typically combined with other styles of loss to amplify the distinguishability between normal and anomalous points, and it is rarely used on its own.

To integrate the strengths of each method and avoid the shortcomings associated with using them independently, we propose a framework called AD2T, which can simultaneously accommodate reconstruction, prediction, and association discrepancy learning tasks. Additionally, we develop a min–max strategy for training and devise a dilated causal convolution-based embedding module to capture both temporal and intersensor correlations simultaneously. The experimental results in Section V-D demonstrate that AD2T achieves significant improvements in MTS anomaly detection compared to any single-type methods.

III. Preliminaries

A. Problem Statement

In a multisensor system, N system-dependent continuous measurement MTS are represented as $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\} \in \mathbb{R}^{N\times T}$, where $\mathbf{x}_t = \{x_t^1, x_t^2, \ldots, x_t^N\} \in \mathbb{R}^N$ represents the N different sensor (channel) values at time t, and T denotes the total number of time points. Like most MTS anomaly detection methods [26], [27], [28], we assume the training set $\mathbf{X} \in \mathbb{R}^{N\times T}$ mostly contains normal operational conditions, while another MTS $\tilde{\mathbf{X}} = \{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_{T'}\} \in \mathbb{R}^{N\times T'}$, observed from the same system, serves as the test set. The MTS anomaly detection task requires the model to learn normal data distribution from the training set and then output a scoring function $\mathrm{AS}(\cdot)$, which provides the point-wise anomaly score for the test set. Typically, the point-wise anomaly score can be converted into binary labels through a threshold, denoted as $\mathcal{Y}_{\text{test}} = \{y_1, y_2, \ldots, y_{T'}\}$, where $y_t \in \{0, 1\}$, that is, 1 (abnormal) or 0 (normal).

B. Association Discrepancy

Transformer leverages the self-attention mechanism to model the associations of each time point in the entire time series. However, in establishing associations with neighboring time points (the prior association) and with global time points (the series association), anomalous points exhibit distinct differences in capability compared to normal points. For the former, due to the scarcity of anomalies, anomalous points are only adept at forming associations with nearby points and struggle to connect with more distant points. Consequently, the series association for anomalous points is largely contributed by the prior association, resulting in minimal association discrepancy. For the latter, since normal points can establish associations with both neighboring and global points, their association discrepancy is more pronounced.

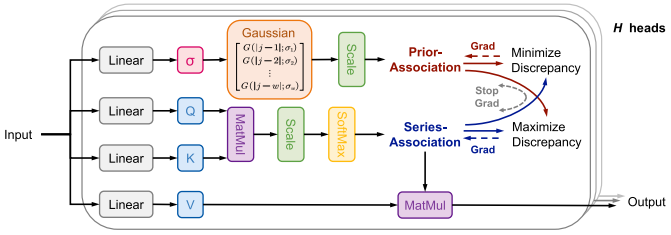


Fig. 1. Anomaly attention module in the AT, which includes the learning of prior and series associations, along with the optimization mechanism of the min–max strategy.

Recently, Xu et al. [24] introduced the anomaly Transformer (AT), whose key component, anomaly attention, leverages the aforementioned characteristic of association discrepancy to enhance anomaly detection. Fig. 1 shows the details of the anomaly attention module. The prior association is defined as a prior based on relative temporal distance, calculated using a Gaussian kernel function. Its intrinsic unimodal property pays more attention to adjacent points than to more distant points. A learnable scale parameter σ is used to control the bandwidth of the Gaussian kernel, making it adaptable to anomalous segments of varying lengths. The series-association calculation follows the classic self-attention mechanism, which adaptively learns associations from the raw sequence. For the lth layer of the anomaly attention module, given the time-series slice $\mathbf{X}^{l-1} \in \mathbb{R}^{w \times d_{\text{model}}}$ from the output of the $(l-1)$th layer as input, the overall equations for this layer are formalized as

$$\begin{aligned}
&\mathcal{Q}, \mathcal{K}, \mathcal{V}, \sigma \\
&\quad = \mathbf{X}^{l-1} W_{\mathcal{Q}}^{l}, \mathbf{X}^{l-1} W_{\mathcal{K}}^{l}, \mathbf{X}^{l-1} W_{\mathcal{V}}^{l}, \mathbf{X}^{l-1} W_{\sigma}^{l}, \\
&\mathcal{P}^{l} = \text{Rescale}\left(\left[\frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{|j-i|^2}{2\sigma_i^2}\right)\right]_{i,j \in \{1,\ldots,w\}}\right) \\
&\mathcal{S}^{l} = \text{Softmax}\left(\frac{\mathcal{Q}\mathcal{K}^{\mathrm{T}}}{\sqrt{d_{\text{model}}}}\right) \\
&\widehat{\mathcal{Z}}^{l} = \mathcal{S}^{l}\mathcal{V} \qquad (1)
\end{aligned}$$

where $\mathcal{Q}, \mathcal{K}, \mathcal{V} \in \mathbb{R}^{w \times d_{\text{model}}}$ represent the query, key, and value in the self-attention mechanism, respectively, and d_{model} denotes the dimension of anomaly attention. $\sigma \in \mathbb{R}^{w \times 1}$ represents a learnable scale parameter. These parameters are initialized by the product of the input $\mathbf{X}^{l-1}$ with the lth layer parameter matrices $W_{\mathcal{Q}}^{l}, W_{\mathcal{K}}^{l}, W_{\mathcal{V}}^{l} \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$, and $W_{\sigma}^{l} \in \mathbb{R}^{d_{\text{model}} \times 1}$, respectively. $\mathcal{P}^{l}, \mathcal{S}^{l} \in \mathbb{R}^{w \times w}$ represent the prior association and the series association, respectively, and the series association $\mathcal{S}^{l}$ multiplies with the value of self-attention $\mathcal{V}$ to get the reconstruction output of the lth layer $\widehat{\mathcal{Z}}^{l} \in \mathbb{R}^{w \times d_{\text{model}}}$. If the anomaly attention consists of L layers in total, then the association discrepancy is defined as the average symmetric KL divergence between the prior association and the series association across all L layers

$$\text{AD}(\mathcal{P}, \mathcal{S}) = \left[\frac{1}{L}\sum_{l=1}^{L}\left(\text{KL}\left(\mathcal{P}_{i,:}^{l} \parallel \mathcal{S}_{i,:}^{l}\right) + \text{KL}\left(\mathcal{S}_{i,:}^{l} \parallel \mathcal{P}_{i,:}^{l}\right)\right)\right] \quad i = 1, \ldots, w \qquad (2)$$

where $\text{AD}(\mathcal{P}, \mathcal{S})$ denotes the association discrepancy between the prior association and the series association, as well as $\text{KL}\left(\mathcal{P}_{i,:}^{l} \parallel \mathcal{S}_{i,:}^{l}\right)$ represents the KL divergence for the ith row between $\mathcal{P}^{l}$ and $\mathcal{S}^{l}$.

According to the previous analysis, the association discrepancy for anomalous points tends to be relatively small. Therefore, the AT is designed around the association discrepancy to formulate the model's objective function, and it introduces a min–max learning mechanism for the joint optimization of both the reconstruction and association discrepancy learning tasks. For details on the min–max mechanism, refer to [24].

IV. Methodology

In this article, we propose a novel approach called AD2T to solve the unsupervised MTS anomaly detection problem. Fig. 2 gives an overview of the proposed framework. In general, the proposed model first employs a preprocessing procedure to divide MTS into training sequences (inputs for the encoder) and label sequences (inputs for $\text{decoder}_{\text{pre}}$). The training sequence is a local context window that includes the current timestamp. As the segment of the training sequence closest to the current timestamp, the label sequence is intended to provide recent information for prediction [29]. Second, both training sequences and label sequences are processed through embedding and positional encoding. Specifically, a dilated causal convolution-based embedding module is proposed to extract features from both the temporal and sensor dimensions of the training sequences in parallel, thereby obtaining a richer representation of the MTS. Third, the anomaly attention in the encoder enables the adaptive learning of the prior association and series association of the input MTS data, facilitating the calculation of the corresponding association discrepancy. The encoder collaborates with $\text{decoder}_{\text{rec}}$ and $\text{decoder}_{\text{pre}}$ to form an encoder–decoder structure, respectively, for both reconstruction and prediction. The association discrepancy, reconstruction loss, and prediction loss together construct the joint loss and anomaly score of the AD2T model. Finally, we develop a min–max strategy to perform joint optimization of the model during the training phase. In the test phase, the trained model is used to calculate the anomaly score for the test set to detect anomalies.

A. Data Preprocessing

The data preprocessing consists of two procedures: data normalization and temporal window segmentation. To eliminate the impact of varying sensor signal amplitudes in the MTS on the model, the data in both the training and testing sets are normalized as follows:

$$\mathbf{x}_t = \frac{\mathbf{x}_t - \min(\mathbf{X})}{\max(\mathbf{X}) - \min(\mathbf{X})} \qquad (3)$$

where $\min(\mathbf{X})$ and $\max(\mathbf{X})$ represent the minimum and maximum values of $\mathbf{X}$, respectively.

To capture more comprehensive local temporal information, we segment the MTS into fixed-length temporal context windows, with a sliding step of one timestamp between adjacent windows. The input to the encoder is a temporal context

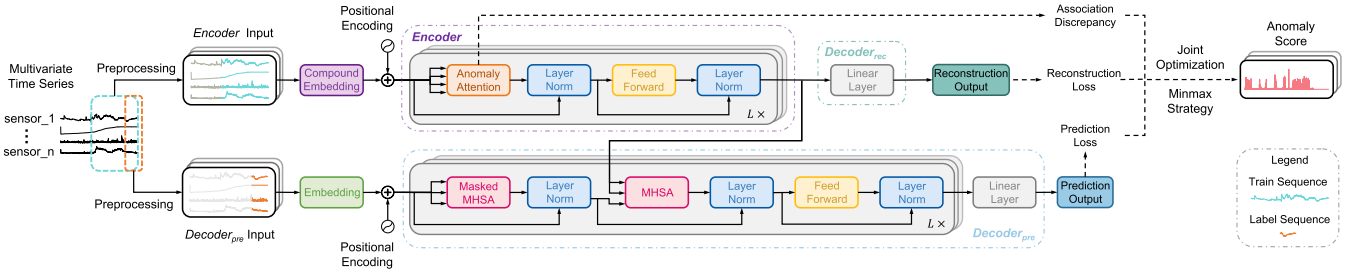


Fig. 2. Overview of the proposed AD2T framework.

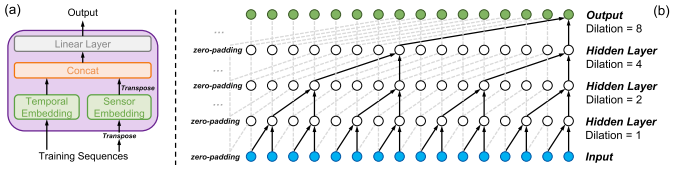


Fig. 3. (a) Overview of the proposed compound embedding module. (b) Both temporal and sensor (channel) embedding consist of a stack of dilated causal convolution layers with an initial convolution kernel size of 2 and dilation factors of {1, 2, 4, and 8}.

window of length w, denoted as $\mathbf{X}_{\text{enc}} = \{\mathbf{x}_{t-w+1}, \ldots, \mathbf{x}_t\} \in \mathbb{R}^{N\times w}$, while the input to decoder$_{\text{pre}}$ is represented as $\mathbf{X}_{\text{dec}} \in \mathbb{R}^{N\times(L_{\text{label}}+L_{\text{pre}})}$. The initialization of $\mathbf{X}_{\text{dec}}$ includes two parts: the trailing segment of the encoder's input $\mathbf{X}_{\text{enc}}$ with length L_{label} to provide recent information and placeholders of length L_{pre} filled with zero scalars, formalized as follows:

$$\begin{aligned}\mathbf{X}_{\text{dec}} &= \text{Concat}\left(\mathbf{X}_{\text{enc}(-L_{\text{label}}:)}, \mathbf{X}_0\right)\\ \mathbf{X}_{\text{enc}(-L_{\text{label}}:)} &= \left\{\mathbf{x}_{t-L_{\text{label}}+1}, \ldots, \mathbf{x}_t\right\}\\ \mathbf{X}_0 &= \underbrace{[0, \ldots, 0]}_{L_{\text{pre}}}^{\text{T}}\end{aligned} \tag{4}$$

where $\mathbf{X}_{\text{enc}(-L_{\text{label}}:)} \in \mathbb{R}^{N\times L_{\text{label}}}$ represents the trailing part of $\mathbf{X}_{\text{enc}}$ with length L_{label} that provides historical information to aid in prediction, while $\mathbf{X}_0$ denotes a placeholder of length L_{pre} filled with zeros, where L_{pre} indicates the length of the prediction sequence.

B. Compound Embedding

Existing dedicated Transformers for time series [16], [24], [30] often utilize a simple dense layer or a 1-D convolutional layer to project the input time series into a higher-dimensional space, combined with positional encoding to incorporate temporal order information. However, such embedding structures inherently fail to adequately represent temporal dependencies and intersensor correlations, which are critical for effective time-series modeling.

To address these limitations, we propose a compound embedding module based on dilated causal convolution, which is designed to capture intrasensor temporal patterns and intersensor correlation features simultaneously and efficiently. As shown in Fig. 3(a), the input training sequences are processed through two parallel paths: the left path passes the sequences through a temporal embedding layer, while the right path transposes the sequences and feeds them into a sensor (channel) embedding layer. To better capture temporal and intersensor correlations, the design of the module draws inspiration from the temporal convolutional network (TCN) [31]. Both the temporal and sensor embedding layers share the same structure: a stack of dilated causal convolution layers designed to efficiently capture temporal patterns (or intersensor correlations for transposed input), with zero padding to ensure that the output tensor maintains the same length as the input tensor. Empirically, we initialize the convolution kernel size to 2 and set the dilation factors to $\{1, 2, 4, 8\}$, as shown as Fig. 3(b). Subsequently, the extracted temporal dynamic features and the transposed intersensor correlation representations are concatenated and processed through a linear layer to project them into a d_{model}-dimensional space. Overall, the proposed compound embedding module can be formalized as follows:

$$\mathbf{E}_{\text{enc}} = \text{Linear}\left(\text{Concat}\left(\text{DCNs}(\mathbf{X}_{\text{enc}}), \left(\text{DCNs}(\mathbf{X}_{\text{enc}}^{\text{T}})^{\text{T}}\right)\right)\right) \tag{5}$$

where DCNs$(\cdot)$ denotes a stack of four dilated causal convolution networks and $\mathbf{E}_{\text{enc}} \in \mathbb{R}^{w\times d_{\text{model}}}$ represents the output of the compound embedding module. Note that the label sequences are typically shorter than the receptive field of the compound embedding module. Therefore, we use a 1-D convolution with a kernel size of 3 for the embedding module on the decoder$_{\text{pre}}$ side, which is formalized as follows:

$$\mathbf{E}_{\text{dec}} = \text{Conv1d}(\mathbf{X}_{\text{dec}}) \tag{6}$$

where Conv1d$(\cdot)$ denotes the 1-D convolution and $\mathbf{E}_{\text{dec}} \in \mathbb{R}^{(L_{\text{label}}+L_{\text{pre}})\times d_{\text{model}}}$ represents the embedding of the label sequences.

C. Encoder

As shown in Fig. 2, the encoder is built by alternating stacking of anomaly attention modules and feed-forward layers, which is designed to create an encoded representation of the training sequences while computing the prior and series associations. Suppose the training sequences processed by compound embedding and positional encoding are denoted as $\mathcal{X}_{\text{enc}}^0$, which serve as the input for the lth layer of the encoder. Then, the overall equations for the lth layer of the encoder can be formalized as follows:

$$\begin{aligned}\mathcal{Z}_{\text{enc}}^l &= \text{Layer-Norm}\left(\text{Anomaly Attention}\left(\mathcal{X}_{\text{enc}}^{l-1}\right) + \mathcal{X}_{\text{enc}}^{l-1}\right)\\ \mathcal{X}_{\text{enc}}^l &= \text{Layer-Norm}\left(\text{Feed-Forward}\left(\mathcal{Z}_{\text{enc}}^l\right) + \mathcal{Z}_{\text{enc}}^l\right)\end{aligned} \tag{7}$$

where $\mathcal{X}_{\text{enc}}^{l} \in \mathbb{R}^{w \times d_{\text{model}}}, l \in \{1, 2, \ldots, L\}$ denotes the output of the lth layer and $\mathcal{Z}_{\text{enc}}^{l} \in \mathbb{R}^{w \times d_{\text{model}}}$ represents the hidden states representation of the lth layer. Anomaly attention$(\cdot)$ is used to compute prior and series associations, as formalized by (1).

D. Dual-Decoder Architecture

To make efficient use of the MTS and obtain richer representations, we propose a dual-decoder architecture that can simultaneously accommodate both reconstruction and prediction tasks. After being processed by the L-layer encoder, the encoded representation $\mathcal{X}_{\text{enc}}^{L} \in \mathbb{R}^{w \times d_{\text{model}}}$ is shared with both $\text{decoder}_{\text{rec}}$ and $\text{decoder}_{\text{pre}}$, as shown in Fig. 2. For the branch $\text{decoder}_{\text{rec}}$, the encoded representation is used as the input for the reconstruction task; for the other branch, it is used as the cross-information to help the $\text{decoder}_{\text{pre}}$ refine prediction results. In this article, the $\text{decoder}_{\text{rec}}$ consists of a single linear layer, and the overall equation for $\text{decoder}_{\text{rec}}$ is shown as follows:

$$\hat{\mathbf{X}}_{\text{rec}} = \text{Linear}(\mathcal{X}_{\text{enc}}^{L}) \tag{8}$$

where $\hat{\mathbf{X}}_{\text{rec}} \in \mathbb{R}^{N \times w}$ denotes the reconstruction output. As one of the important components of the final joint loss for model training, the reconstruction loss $\mathcal{L}_{\text{rec}}$ is formalized as

$$\mathcal{L}_{\text{rec}}(\widehat{\mathbf{X}}_{\text{rec}}, \mathbf{X}_{\text{enc}}) = \|\mathbf{X}_{\text{enc}} - \widehat{\mathbf{X}}_{\text{rec}}\|_F^2 \tag{9}$$

where $\|\cdot\|_F^2$ indicates the Frobenius norm.

The $\text{decoder}_{\text{pre}}$ mainly contains two parts: a Transformer-style decoder, which includes self-attention layers with a lower triangular mask [17] to predict the next timestamp while preventing the leakage of future information, and a linear layer to restore the dimension of the predicted values to the input size. Suppose there are L decoder layers, and the label sequences processed by embedding and positional encoding are denoted as $\mathcal{X}_{\text{dec}}^{0}$, then the overall equations of the Transformer-style decoder are formalized as follows:

$$\begin{aligned}
\mathcal{Z}_{\text{dec}}^{l,1} &= \text{Mask}\left(\text{MultiHeadAtt}\left(\mathcal{X}_{\text{dec}}^{l-1}, \mathcal{X}_{\text{dec}}^{l-1}, \mathcal{X}_{\text{dec}}^{l-1}\right)\right) \\
\mathcal{Z}_{\text{dec}}^{l,2} &= \text{Layer-Norm}(\mathcal{X}_{\text{dec}}^{l-1} + \mathcal{Z}_{\text{dec}}^{l,1}) \\
\mathcal{Z}_{\text{dec}}^{l,3} &= \text{Layer-Norm}(\mathcal{Z}_{\text{dec}}^{l,2} + \text{MultiHeadAtt}(\mathcal{X}_{\text{enc}}^{L}, \mathcal{X}_{\text{enc}}^{L}, \mathcal{Z}_{\text{dec}}^{l,2})) \\
\mathcal{X}_{\text{dec}}^{l} &= \text{Layer-Norm}(\mathcal{Z}_{\text{dec}}^{l,3} + \text{Feed-Forward}(\mathcal{Z}_{\text{dec}}^{l,3}))
\end{aligned} \tag{10}$$

where $\mathcal{X}_{\text{dec}}^{l} \in \mathbb{R}^{(L_{\text{label}}+L_{\text{pre}}) \times d_{\text{model}}}$ denotes the output of the lth layer, and $\mathcal{Z}_{\text{dec}}^{l,1}, \mathcal{Z}_{\text{dec}}^{l,2}, \mathcal{Z}_{\text{dec}}^{l,3} \in \mathbb{R}^{(L_{\text{label}}+L_{\text{pre}}) \times d_{\text{model}}}$ represent the hidden states representations in the lth layer, respectively. At last, the linear layer in $\text{decoder}_{\text{pre}}$ maps $\mathcal{X}_{\text{dec}}^{L}$ to $\hat{\mathbf{X}}_{\text{dec}} \in \mathbb{R}^{N \times (L_{\text{label}}+L_{\text{pre}})}$. In this article, $\text{decoder}_{\text{pre}}$ is used to predict the next point in time, then L_{pre} is set to 1. As another important component of the joint loss for model training, the prediction loss $\mathcal{L}_{\text{pre}}$ is formalized as follows:

$$\mathcal{L}_{\text{pre}}(\hat{\mathbf{X}}_{\text{dec}}, \mathbf{X}_{\text{dec}}) = \sqrt{(\hat{\mathbf{X}}_{\text{dec}[:,-L_{\text{pre}}:]} - \mathbf{X}_{\text{dec}[:,-L_{\text{pre}}:]})^2} \tag{11}$$

where $\hat{\mathbf{X}}_{\text{dec}[:,-L_{\text{pre}}:]}$ represents the last part of the predicted output $\hat{\mathbf{X}}_{\text{dec}}$ with a length of L_{pre}.

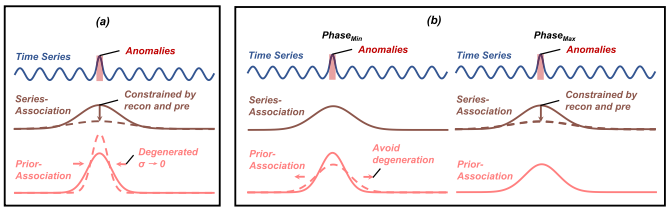


Fig. 4. Two training strategies for AD2T. Note that the series-association distribution is also depicted as a Gaussian distribution to facilitate the visualization of the distance between the series association and prior association distributions (i.e., the association discrepancy). (a) Direct minimization strategy. (b) Developed min–max training strategy.

E. Joint Optimization

The three types of loss functions incorporated in AD2T push the model to learn multiview relationships in MTS: the reconstruction loss L_{rec} is primarily used to capture the temporal dependencies across entire sequences, the prediction loss L_{pre} is mostly used to forecast the trend of the next point, and the association discrepancy $\text{AD}(\mathcal{P}, \mathcal{S})$ is used to learn the relationships between individual points and the broader sequence context. To better characterize the latent representations of the original MTS input, the total loss L_{total} is defined as the weighted sum of the three losses as follows:

$$L_{\text{total}} = (1-\beta)\, L_{\text{rec}} + \beta L_{\text{pre}} - k\, \|\text{AD}(\mathcal{P}, \mathcal{S})\|_1 \tag{12}$$

where $\beta \in [0, 1]$ balances the task weights between reconstruction and prediction and $k \geqslant 0$ balances the weight of association discrepancy learning with the first two tasks. Since the training phase aims to amplify the distinction between anomalous and normal points, the association discrepancy should be enlarged, meaning its optimization direction is opposite to L_{rec} and L_{pre}. In practice, $\beta = 0.6$ and $k = 0.7$. The hyperparameter selection will be discussed in Section V-C.

To optimize (12), a straightforward approach is to adopt a minimization strategy. In this approach, the learning algorithm encourages the series association to focus more on points with a broader context, as anomalies are generally harder to reconstruct and predict. Meanwhile, the prior association would focus more on points adjacent to the anomalies, resulting in the degeneration of the Gaussian distribution ($\sigma \to 0$) [32], thereby enlarging the distance between the series association and the prior association (i.e., the association discrepancy), as shown in Fig. 4(a). However, such a prior association would be meaningless. To address the aforementioned issue, we extended the min–max strategy from [24] to meet two key requirements: 1) effectively control the excessive reduction of σ while enlarging the association discrepancy and 2) maintain constraints on the reconstruction loss and prediction loss. The developed min–max joint optimization strategy is defined as follows:

$$\text{Phase}_{\text{Min}} : (1-\beta)\, L_{\text{rec}} + \beta L_{\text{pre}} + k\, \|\text{AD}(\mathcal{P}, \mathcal{S}_{\text{detach}})\|_1 \tag{13}$$

$$\text{Phase}_{\text{Max}} : (1-\beta)\, L_{\text{rec}} + \beta L_{\text{pre}} - k\, \|\text{AD}(\mathcal{P}_{\text{detach}}, \mathcal{S})\|_1 \tag{14}$$

where $*_{\text{detach}}$ denotes stopping the gradient backpropagation. As shown in Fig. 4(b), during $\text{Phase}_{\text{Min}}$, the gradient updates of the series association are locked. The learning algorithm guides the prior association's distribution to align more closely

with the series association (temporarily reducing the association discrepancy) to minimize (13), thereby preventing the excessive reduction of σ. In $\text{Phase}_{\text{Max}}$, the gradient updates of the prior association are locked. Under the constraints of reconstruction and prediction, the series association distribution diverges from the prior association to minimize (14), thereby maximizing the association discrepancy. The developed min–max training strategy reduces reconstruction and prediction losses while increasing the association discrepancy, simultaneously controlling the excessive degeneration of the Gaussian distribution corresponding to the prior association, which lays a solid foundation for the effective training of AD2T.

F. Anomaly Score

In the testing phase, we combine the association discrepancy, reconstruction loss, and prediction loss to define the anomaly score. The reconstruction loss L_{rec} and prediction loss L_{pre} are balanced by the weight β, consistent with (12). To amplify the distinguishability between normal points and anomalies, elementwise multiplication is used to combine the normalized association discrepancy with the reconstruction and prediction losses, as shown as follows:

$$\text{AS} = \text{Softmax}\left(-\text{AD}(\mathcal{P}, \mathcal{S})\right) \odot \left[(1-\beta)\, L_{\text{rec}} + \beta L_{\text{pre}}\right] \quad (15)$$

where $\odot$ denotes the Hadamard product and $\text{AS} \in \mathbb{R}^{w\times 1}$ represents the point-wise anomaly score of the time window in the test dataset $\tilde{\mathbf{X}}$. Since anomalous points are more difficult to reconstruct and predict, their corresponding reconstruction and prediction errors are larger. Additionally, anomalous points exhibit smaller association discrepancies, leading to higher overall anomaly scores and making the distinction between anomalous and normal points more pronounced.

Based on the point-wise anomaly score, a threshold δ needs to be set to convert the anomaly score into binary anomaly labels $\mathcal{Y}_{\text{test}} = \{y_1, y_2, \ldots, y_{T'}\}$. For each point in the test set, if its corresponding anomaly score exceeds δ, its binary label is set to 1 (abnormal); otherwise, it is set to 0 (normal). In this article, threshold selection methods are not the primary focus. Existing techniques such as peaks-over-threshold (POT) [33] can be employed to guide threshold selection.

V. Experiments

In this section, we conduct comprehensive experiments on real-world datasets to compare the performance of our proposed AD2T with seven competitors in terms of anomaly detection accuracy. We first describe the datasets, metrics, competing methods, and implementation details. Then, we evaluate the overall performance of the proposed AD2T against seven SOTA competitors. To explore the impact of different hyperparameters on the effectiveness of AD2T, we conduct a sensitivity analysis of the key parameters. Additionally, we perform an ablation study to investigate the contribution of each component in AD2T. Lastly, through visualization analysis, we demonstrate how AD2T works in MTS anomaly detection tasks.

Algorithm 1 Training Procedure of AD2T

Input: Training dataset $\mathbf{X}$, number of hidden channels d_{model}, window length w, batch size B, train epochs I, the number of model layers L, and hyperparameters $L_{\text{label}}, L_{\text{pre}}, k, \beta$.
Output: Well trained model.
1: Normalize the dataset $\mathbf{X}$ and apply time window slicing according to Eq. (4).
2: **for** epoch $i = 1, 2, \ldots, I$ **do**
3: **while** not end of data **do**
4: Sample a batch to construct $\mathbf{X}_{\text{enc}} \in \mathbb{R}^{B\times N\times w}$ and $\mathbf{X}_{\text{dec}} \in \mathbb{R}^{B\times N\times (L_{\text{label}}+L_{\text{pre}})}$.
5: Calculate the embedding representations $\mathbf{E}_{\text{enc}}$ and $\mathbf{E}_{\text{dec}}$ for $\mathbf{X}_{\text{enc}}$ and $\mathbf{X}_{\text{dec}}$ using Eqs. (5)∼(6), respectively.
6: Add positional encoding to $\mathbf{E}_{\text{enc}}$ and $\mathbf{E}_{\text{dec}}$ to obtain $\mathcal{X}^0_{\text{enc}}$ and $\mathcal{X}^0_{\text{dec}}$, respectively.
7: Compute $\mathcal{X}^L_{\text{enc}}, \mathcal{P}^l, \mathcal{S}^l, l = 1, \ldots, L$ by $\mathcal{X}^0_{\text{enc}}$ according to Eq. (1) and Eq. (7).
8: *// Encoder Layer*
9: Caculate $\text{AD}(\mathcal{P}, \mathcal{S})$ by $\mathcal{P}^l, \mathcal{S}^l, l = 1, \ldots, L$ according to Eq. (2).
10: *// Association Discrepancy*
11: Compute $\hat{\mathbf{X}}_{\text{rec}}$ by $\mathcal{X}^L_{\text{enc}}$ according to Eq. (8).
12: Compute $\mathcal{L}_{\text{rec}}$ by $\hat{\mathbf{X}}_{\text{rec}}$ and $\mathbf{X}_{\text{rec}}$ according to Eq. (9).
13: *// Reconstruction Loss*
14: Calculate $\mathcal{X}^L_{\text{dec}}$ by $\mathcal{X}^0_{\text{dec}}$ and $\mathcal{X}^L_{\text{enc}}$ based on Eq. (10).
15: $\hat{\mathbf{X}}_{\text{dec}} = Linear(\mathcal{X}^L_{\text{dec}})$.
16: Compute $\mathcal{L}_{\text{pre}}$ by $\hat{\mathbf{X}}_{\text{dec}}$ and $\mathbf{X}_{\text{dec}}$ based on Eq. (11).
17: *// Prediction Loss*
18: Minimize Eq. (13) and compute the gradients, then backpropagate while retaining the computational graph for further use.
19: *// $Phase_{Min}$*
20: Minimize Eq. (14) and compute the gradients, then backpropagate.
21: *// $Phase_{Max}$*
22: Update the model parameters using the optimizer.
23: **end for**
24: **Return** the well trained model.

A. Experimental Setup

1) Datasets: We demonstrate the effectiveness of our method on five multisensor system public datasets. All datasets are available on the website.[1] Details about these datasets are presented in Table I. Notably, except for server machine dataset (SWaT), the other datasets have omitted localization meta-attributes (i.e., sensor or channel information) in their original publications due to anonymization requirements.

1) *MSL and SMAP [1]:* Mars Science Laboratory (MSL) and Soil Moisture Active Passive (SMAP) public datasets are both collected by NASA. These datasets include telemetry anomaly data reported by the spacecraft monitoring system and used by inspectors to

[1]https://drive.google.com/drive/folders/1RaIJQ8esoWuhyphhmMaH-VCDh-WIluRR

TABLE I
DATASET STATISTICS

Dataset	Field	Dimension	Train	Test	Anomaly Rate (%)
MSL	Aerospace	55	58317	73729	10.48
SMAP	Aerospace	25	135183	427617	12.83
SWaT	Water	51	495000	449919	12.14
PSM	Server	25	132481	87841	27.76
SMD	Server	38	708405	708420	4.16

address unexpected events that pose potential risks during postlaunch spacecraft operations. MSL contains sensor and actuator data from the Mars rover itself, while SMAP focuses on soil moisture measurements collected by the rover. Both the training and testing sets of MSL and SMAP contain anomalies, but only the latter provides corresponding labels.

2) *SMD [2]:* The secure water treatment (SMD) dataset consists of data collected over five weeks from 28 physical servers in a large Internet company, describing server machine metrics such as TCP retransmissions, memory usage, and CPU load, with a time interval of one minute between consecutive observations. The first five days of the dataset contain only normal data, serving as the training set, while all anomalies (e.g., overload) occur intermittently during the last five days, serving as the test set.

3) *PSM [3]:* Similar to SMD, the Pooled Server Metrics (PSM) dataset is collected from multiple application server nodes at eBay. It includes 25 channels describing server machine metrics such as CPU usage and memory status. The anomalies in this dataset include both injected anomalies and unplanned anomalies. While anomalies exist in both the training and testing sets, only the testing set contains corresponding labels.

4) *SWaT [4]:* The SWaT dataset is collected from a water treatment testbed over 11 days, comprising data from 51 sensors (e.g., level sensors and differential pressure sensors). This dataset is used to study the impact of cyberattacks on critical infrastructure and the corresponding system responses. During the last 4 days, anomalies are injected using various types of cyberattacks (e.g., manipulating sensors to cause the raw water tank to overflow), serving as the test set, while the first 7 days contain only normal operation data, which serves as the training set.

2) Metrics: For all methods, we evaluate the anomaly detection accuracy by calculating the precision, recall, and $F1$-score, which are defined as follows:

$$\begin{aligned} \text{Precision} &= \frac{\text{TP}}{\text{TP}+\text{FP}} \\ \text{Recall} &= \frac{\text{TP}}{\text{TP}+\text{FN}} \\ F_1 &= 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision}+\text{Recall}} \end{aligned} \tag{16}$$

where true positive (TP) denotes the number of correctly predicted anomaly samples, false positive (FP) indicates the count of incorrectly predicted anomaly samples, and false negative (FN) represents the number of incorrectly predicted normal samples. Precision measures the ratio of correctly predicted anomaly samples to the total predicted anomaly samples, while recall represents the ratio of correctly predicted anomaly samples to the total ground-truth anomaly samples. The $F1$-score harmonically averages precision and recall to evaluate the overall performance of the model. Since anomalies typically occur consecutively in practice, like many previous works [2], [30], [34], we adopt the point-adjust strategy to finetune the metrics in this study.

3) Baselines: To comprehensively evaluate our superiority, we selected SOTA anomaly detection methods for comparison, including *Clustering-based method:* DeepSVDD [35]; *Reconstruction-based methods:* USAD [20], TranAD [36], and AMFormer [37]; *Prediction-based methods:* GDN [38] and GTA [23]; *Discrepancy-based methods:* AT [24] and DCDetector [34]. Since not every model provides an appropriate threshold selection paradigm, to ensure a fair comparison, we decided to present the best precision, best recall, and best $F1$-score by enumerating all possible anomaly thresholds, thereby comparing the upper bound of each model's accuracy.

4) Implementation Details: We set the number of hidden channels d_{model} and the number of layers L to 512 and 3, respectively, for both the encoder and decoder$_{\text{pre}}$. The window length w for encoder input is set to 100. The label length L_{label} and prediction length L_{pre} are set to 10 and 1, respectively, to provide recent information and enable single-point prediction. The hyperparameters k and β are set to 0.7 and 0.6, respectively, to balance the weights of the association discrepancy, reconstruction, and prediction tasks. To prevent overfitting, dropout is employed in the AD2T framework, with a dropout rate set to 0.1. During the training phase, 30% of the training data is used to construct a validation set. We use the Adam optimizer with an initial learning rate of $1e^{-4}$ and run for 5 epochs with early stopping, where the training phase terminates if the validation loss does not improve. For all baselines, we adopt parameter settings consistent with those recommended in their respective articles. All experiments are conducted using Python 3.8 on an Ubuntu 18.04 server equipped with a 10-core Intel Xeon Silver 4210R CPU, 128GB RAM, and one NVIDIA A100 GPU.

B. Overall Performance

Table II records the precision (P), recall (R), and $F1$-score of all models across the five datasets. As a supplement, Fig. 5 shows the average scores of all models on these datasets. Among all algorithms, DeepSVDD and USAD exhibit the most inferior performance, which may be due to their simple network structures and inability to capture temporal correlations. In contrast, TranAD and AMFormer utilize the Transformer architecture for capturing temporal dependencies and demonstrate better performance. However, their neglect of learning cross-sensor relationships results in weaker generalization. AMFormer achieves competitive detection results on the MSL, PSM, and SMAP datasets, but its poor performance on the SWaT dataset further proves this limitation. GDN and GTA employ a graph structure to capture complex intersensor relationships, thereby enhancing their performance. However,

TABLE II
DATASETS QUANTITATIVE RESULTS FOR AD2T AND BASELINES IN FIVE REAL-WORLD DATASETS. FOR THESE THREE METRICS (%), BOLD BLACK TEXT INDICATES THE BEST PERFORMANCE IN EACH EXPERIMENT, WHILE UNDERLINED TEXT DENOTES THE SECOND-BEST RESULTS

Method	MSL			PSM			SMAP			SMD			SWaT		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
DeepSVDD	83.40	98.50	90.32	98.17	95.43	96.78	77.92	75.70	76.79	76.95	76.44	76.70	**97.59**	71.40	82.46
USAD	86.70	84.66	85.67	81.25	86.98	84.02	73.99	88.05	80.41	84.52	79.44	81.90	87.26	79.32	83.10
TranAD	92.67	82.88	87.50	97.84	86.17	91.63	77.09	90.12	83.10	78.61	81.10	79.84	95.47	74.13	83.46
AMFormer	**94.75**	89.44	92.02	95.71	97.71	96.70	**97.44**	96.55	96.99	81.73	90.36	85.83	42.18	78.30	54.83
GDN	88.71	90.63	89.66	87.77	91.40	89.55	81.92	92.54	86.91	80.15	91.27	85.35	92.22	86.80	89.43
GTA	91.04	91.17	91.11	90.57	92.88	91.72	89.11	91.76	90.41	85.86	88.72	87.27	94.83	88.10	91.34
AT	91.55	94.73	93.11	97.97	97.52	97.74	94.60	**98.70**	96.61	91.56	88.75	90.14	93.41	92.50	92.95
DCDetector	92.46	**98.81**	**95.54**	97.14	**98.74**	97.94	94.33	98.44	96.34	85.75	83.07	84.39	93.11	99.96	96.42
Ours	93.03	97.34	95.14	**98.97**	97.86	**98.41**	95.90	98.67	**97.26**	**93.60**	**92.18**	**92.89**	93.72	**100.00**	**96.76**

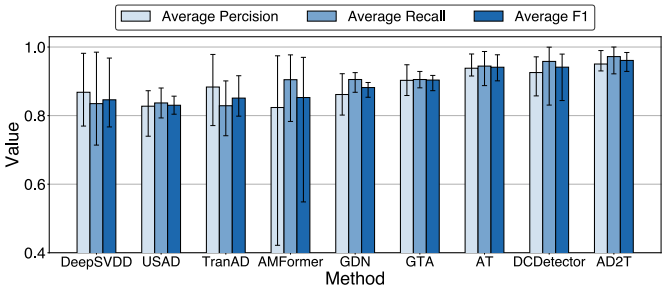


Fig. 5. Average precision, recall, and $F1$ scores for AD2T and baselines across five real-world datasets. The top and bottom of the error bars represent the highest and lowest values of the metrics across the five datasets, respectively.

the failure to consider the intrinsic discrepancy between points and series results in their underwhelming performance across the five datasets, preventing further improvement.

Both AT and DCDetector follow the same principle: anomalous points have difficulty establishing connections with other points, whereas normal points do so easily. AT formulates this principle as the difference between prior association and series association to distinguish between anomalous and normal points, while DCDetector adopts a dual-attention asymmetry design and uses a purely contrastive loss to guide the learning process. Discrepancy-based methods exhibit highly competitive and stable performance across almost all datasets. However, the discrepancy is just one perspective for making anomalies distinguishable, and there remains room for improvement in their overall performance. Additionally, both AT and DCDetector also neglect the learning of relationships between different sensor features.

AD2T exhibits optimal $F1$-scores across the majority of datasets while achieving suboptimal results on the MSL dataset. Specifically, AD2T introduces the compound embedding module to capture relationships between sensors. In addition, the dual-decoder architecture integrates multiple tasks, including reconstruction, prediction, and association discrepancy learning. The adopted min–max training strategy amplifies the score difference between anomalous and normal points, improving detection accuracy. Quantitatively, AD2T demonstrates significant improvements over all SOTA methods. Compared to DeepSVDD (clustering-based), AD2T shows a relative average improvement of **11.48%** in the $F1$-score. Against the best-performing reconstruction-based method, TranAD, AD2T achieves an average improvement of **10.98%** in the $F1$-score. In comparison to GTA (prediction-based), AD2T achieves a notable average $F1$-score enhancement of **5.72%**. Finally, compared to the best-performing discrepancy-based method, DCDetector, AD2T demonstrates an average $F1$-score improvement of **1.96%**.

C. Parameter Analysis

We conduct a study on the impact of the key parameters in AD2T. Note that when we change the value of one parameter, all other parameters are kept at their default settings.

1) Effect of Weighting Parameter k and β: The parameter k controls the weight of the association discrepancy term in L_{total}, while β controls the weight of the reconstruction and prediction terms in both L_{total} and the anomaly score AS. A large k will cause the model to focus primarily on learning the association discrepancy during training while neglecting the reconstruction and prediction tasks. Conversely, a small k will result in the model overlooking the association discrepancy. Similarly, a large β will cause the model and anomaly score to prioritize L_{pre} over L_{rec}, whereas a small β will make the model focus more on L_{rec} while ignoring L_{pre}. We test both k and β with values ranging from 0 to 1, in increments of 0.1, across five datasets using the hyperparameter tuning framework Optuna.[2] Due to space constraints, we only present the search results for the PSM, SMAP, and SWaT datasets, as shown in Fig. 6. It can be seen that the values corresponding to the yellow stars in Fig. 6(a)–(c) yield relatively optimal $F1$-scores across the three datasets, with $k = 0.7$ and $\beta = 0.6$. This suggests that assigning balanced weight parameters to the various terms in L_{total} and AS contributes to better detection results for AD2T.

2) Effect of Window Size w: The parameter w controls the window size of the encoder input. We test w in the range of 40–140, with a step size of 20, and record AD2T's average recall, precision, $F1$-score, and time-consuming across the five datasets. As depicted in Fig. 7(a), with the increase in w, the average $F1$-score of AD2T initially rises and then declines, while the average time-consuming exhibits a near-linear growth. This phenomenon indicates that a smaller w

[2]https://github.com/optuna/optuna

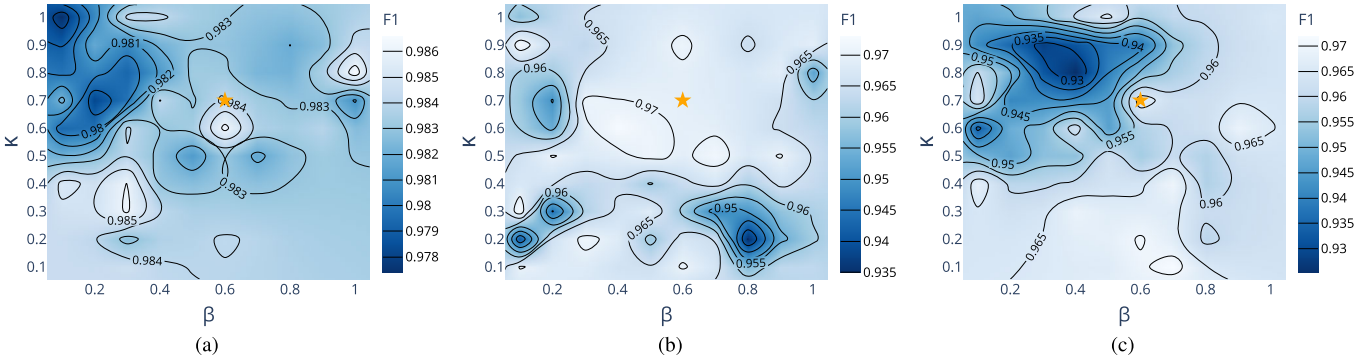


Fig. 6. Impact of varying β and k on different datasets. The white regions correspond to relatively high $F1$-scores, while the blue regions correspond to relatively low $F1$-scores: (a) PSM. (b) SMAP. (c) SWaT.

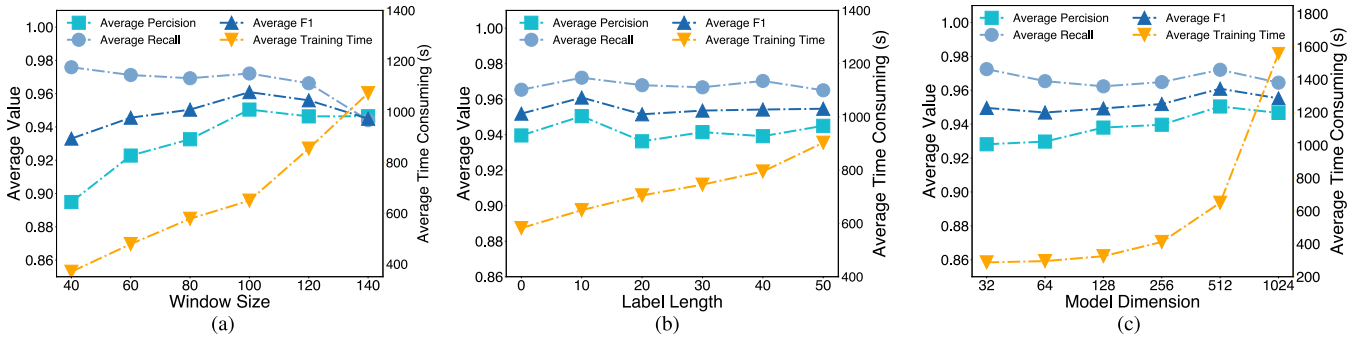


Fig. 7. Impact of different window sizes, label length, and model dimension on AD2T performance across different datasets: (a) Window size. (b) Label length. (c) Model dimension.

may lead to insufficient context information in the input, whereas a larger w could introduce the risk of dilution of anomaly signals. Overall, considering both detection accuracy and efficiency, we select $w = 100$.

3) Effect of Label Length L_{label}: The parameter L_{label} controls the label window size. We test L_{label} in the range of 0–50, with a step size of 10. As shown in Fig. 7(b), it is found that $L_{\text{label}} = 10$ corresponds to the highest average $F1$-score and relatively low average time consuming, indicating that a too small L_{label} leads to insufficient latest context information for the auxiliary prediction task, while a too large L_{label} may cause the model to overfit past patterns and ignore short-term abrupt anomalies.

4) Effect of Model Dimension d_{model}: The parameter d_{model} controls the number of hidden channels in AD2T. We test d_{model} in the range of $\{32, 64, 128, 256, 512, 1024\}$. Fig. 7(c) shows that $d_{\text{model}} = 512$ corresponds to the best average $F1$-score, as well as a relatively acceptable average time-consuming (just before a rapid increase), indicating that too small d_{model} leads to an overly simple model that struggles to fit complex nonlinear situations, while too large d_{model} makes the model prone to memorizing details in the training set, reducing its generalization performance.

D. Ablation Studies

In this section, we test the contribution of each component in AD2T, including the compound embedding module, association discrepancy learning module, reconstruction branch, and prediction branch. The results are presented in Table III. Here, "Com" denotes the utilization of the compound embedding module, "Ass" represents the association discrepancy learning module (i.e., anomaly attention and the corresponding min–max learning strategy), "Rec" refers to the reconstruction branch, and "Pre" indicates the prediction branch. "Time" and "Sens" represent using only the temporal embedding branch or the sensor (channel) embedding branch in the compound embedding module, respectively. Note that the original AT can be represented as "Ass + Rec." Through the analysis of Table III, we obtain the following observations.

1) Using original AT (Ass + Rec) as a baseline, the inclusion of the association discrepancy component (Ass) in both the training process and the anomaly score calculation significantly affects the model's performance. When Ass is included, the average $F1$-score reaches the **90%** level, while without it, the score remains at the **80%** level. This demonstrates that the association discrepancy metric, when combined with the reconstruction loss and prediction loss, can effectively amplify the distinguishability between normal and anomalous points.
2) Compared to the original AT (Ass + Rec), adding the compound embedding module (Com + Ass + Rec) improves the model's performance across all five datasets, increasing the average $F1$-score from **94.10%** to **95.03%**. Furthermore, compared to AD2T with all components (Com + Ass + Rec + Pre), removing the compound embedding module (Ass + Rec + Pre) results in a decrease in the average $F1$-score from **96.09%** to **94.86%**. This indicates that the parallel

TABLE III
ABLATION EXPERIMENT RESULTS FOR AD2T IN FIVE REAL-WORLD DATASETS

Modules				MSL			PSM			SMAP			SMD			SWaT			Average F1
Com	Ass	Rec	Pre	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	
✓		✓		92.56	84.53	88.36	99.42	93.40	96.32	76.79	77.04	76.91	83.89	78.84	81.28	93.81	76.41	84.22	85.42
✓			✓	91.09	86.93	88.96	97.97	96.15	97.05	89.12	58.92	70.94	92.05	79.35	85.23	92.72	79.96	85.87	85.61
✓		✓	✓	91.14	86.93	88.98	98.75	94.50	96.58	71.50	77.69	74.47	93.25	78.28	85.11	93.29	76.41	84.02	85.83
	✓	✓		91.55	94.73	93.11	97.97	97.52	97.74	94.60	98.70	96.61	91.56	88.75	90.14	93.41	92.50	92.95	94.10
	✓	✓	✓	92.00	96.75	94.31	99.01	97.25	98.12	94.93	98.42	96.65	93.73	88.62	91.10	89.45	99.28	94.11	94.86
✓	✓	✓		92.05	97.92	94.90	98.65	97.73	98.19	94.81	98.89	96.81	93.53	87.18	90.25	92.22	97.96	95.01	95.03
Time	✓	✓	✓	93.68	95.19	94.43	98.71	97.90	98.30	95.03	98.61	96.79	91.55	90.26	90.90	92.82	99.57	96.08	95.30
Sens	✓	✓	✓	94.97	92.06	93.50	98.86	97.31	98.08	94.55	98.76	96.61	90.77	91.04	90.90	92.93	100.00	96.33	95.08
✓	✓	✓	✓	93.03	97.34	**95.14**	98.97	97.86	**98.41**	95.90	98.67	**97.26**	93.60	92.18	**92.89**	93.72	100.00	**96.76**	**96.09**

capture of temporal and intersensor relationships by the compound embedding module has a significant impact on the detection results. Additionally, whether using only the temporal embedding branch (Time + Ass + Rec + Pre) or only the sensor (channel) embedding branch (Sens + Ass + Rec + Pre), the obtained average $F1$-score is lower than that implemented with the complete compound embedding module (Com + Ass + Rec + Pre). This indicates that both the temporal and sensor embedding branches in the compound embedding module are indispensable.

3) When association discrepancy learning is not applied, integrating the dual-decoder structure (Com + Rec + Pre) yields better performance compared to using only a single branch (Com + Rec or Com + Pre). Similarly, when association discrepancy learning is incorporated, the dual-decoder AD2T (Com + Ass + Rec + Pre) achieves a higher average $F1$-score compared to the model without the prediction branch (Com + Ass + Rec) (**96.09%** versus **95.03%**). This indicates that, regardless of whether association discrepancy learning is introduced, the dual-decoder structure that integrates both reconstruction and prediction branches consistently outperforms models with any single branch.

E. Visualization Analysis

We demonstrate how AD2T works by visualizing anomaly detection results on a piece of the SMD dataset. As shown in Fig. 8, the ground-truth anomalies include two abrupt anomalies (caused by sensor_10 and sensor_15, respectively) and a collective anomaly segment (caused by sensor_1 and sensor_15). In real-world scenarios, applying point adjustment is a reasonable strategy for collective anomaly detection. This means that if at least one predicted anomaly point falls in the ground-truth collective anomaly segment, the entire segment is considered successfully detected.

Rival methods such as DeepSVDD, USAD, and TranAD can detect nearly all ground-truth anomalies, but they are easily misled by the fluctuating trends in sensor_1 signal, resulting in many false alarms. In contrast, our proposed AD2T successfully detects all anomalies without any false positives, which may be attributed to the association discrepancy's amplification effect on the difference between normal and abnormal points. AT missed all abrupt anomalies caused by different sensors, which may suggest that the addition of the prediction branch and the compound embedding module enhances the model's robustness in detecting abrupt anomalies.

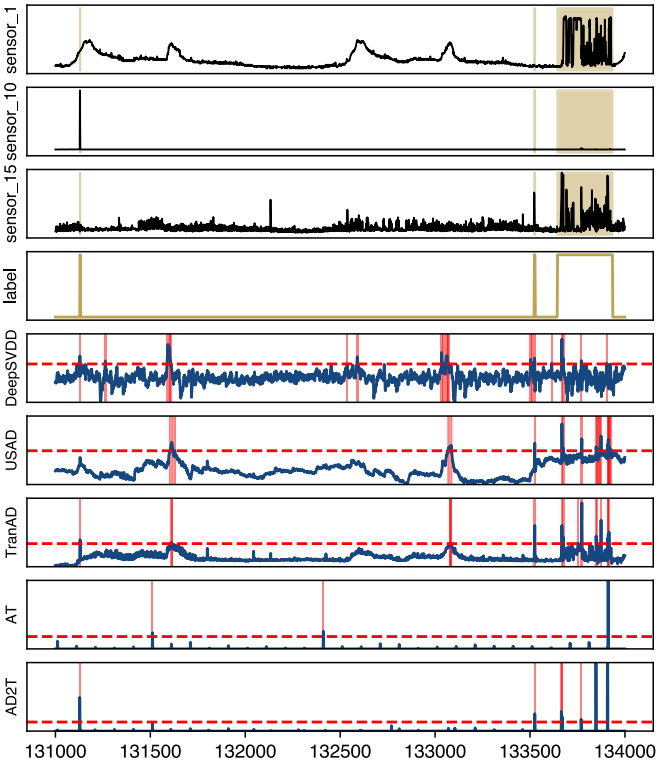


Fig. 8. Visualization of anomaly detection on a piece of the SMD dataset. The black line represents the SMD test data, the yellow area and yellow solid line represent the ground truth and anomaly labels, the blue line indicates the anomaly score, and the red area and red dashed line denote the predicted anomalies and the threshold, respectively.

VI. CONCLUSION

In this work, we propose a novel framework for MTS anomaly detection called AD2T, which integrates intersensor relationship capture, reconstruction, prediction, and association discrepancy learning to enhance anomaly detection performance. In our design, the dilated causal convolution-based embedding module is employed to capture temporal and intersensor relationships in parallel. Next, a dual-decoder architecture is applied to simultaneously accommodate reconstruction, prediction, and association

discrepancy learning tasks, enabling the model to learn both point-wise and point-series correlations. Finally, the developed min–max training strategy is introduced for the joint optimization of all tasks. Extensive experiments were conducted to quantitatively demonstrate the framework's effectiveness and robustness. The experimental results show that our AD2T outperforms the latest SOTA methods, demonstrating the appropriateness of the parameter settings and the contribution of each component to the overall performance.

The proposed AD2T framework has been validated as suitable for MTS anomaly detection tasks in industrial domains such as space telemetry, servers, and water treatment. In future studies, AD2T could be further optimized in terms of lightweight design and inference speed, broadening its applicability to anomaly detection tasks in IoT setups with limited storage capacity.

REFERENCES

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

[2] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min.*, 2019, pp. 2828–2837.

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

[4] A. P. Mathur and N. O. Tippenhauer, "SWaT: A water treatment testbed for research and training on ICS security," in *Proc. Int. Workshop Cyber–Phys. Syst. Smart Water Netw. (CySWater)*, 2016, pp. 31–36.

[5] S. Zhang et al., "Hot rolled prognostic approach based on hybrid Bayesian progressive layered extraction multi-task learning," *Expert Syst. Appl.*, vol. 249, Sep. 2024, Art. no. 123763.

[6] S. Lou et al., "TKS-BLS: Temporal kernel stationary broad learning system for enhanced modeling, anomaly detection, and incremental learning with application to ironmaking processes," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 55, no. 1, pp. 645–660, Jan. 2025.

[7] S. Chen, R. Hu, T. Jiang, and S. Chen, "An unsupervised framework based on dual-domain contrastive learning and tri-indicator joint alarm strategy for early fault detection of bearing," *IEEE Sensors J.*, vol. 24, no. 16, pp. 26889–26901, Aug. 2024.

[8] S. Wang, C. Li, and A. Lim, "A model for non-stationary time series and its applications in filtering and anomaly detection," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[9] B. Nachman and D. Shih, "Anomaly detection with density estimation," *Phys. Rev. D, Part. Fields*, vol. 101, no. 7, Apr. 2020, Art. no. 075042.

[10] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 93–104.

[11] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Min.*, Aug. 2008, pp. 413–422.

[12] M. A. Belay, A. Rasheed, and P. S. Rossi, "Multivariate time series anomaly detection via low-rank and sparse decomposition," *IEEE Sensors J.*, vol. 24, no. 21, pp. 34942–34952, Nov. 2024.

[13] Y. Zhang, Y. Chen, J. Wang, and Z. Pan, "Unsupervised deep anomaly detection for multi-sensor time-series signals," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 2118–2132, Feb. 2023.

[14] Y. Wei, J. Jang-Jaccard, W. Xu, F. Sabrina, S. Camtepe, and M. Boulic, "LSTM-autoencoder-based anomaly detection for indoor air quality time-series data," *IEEE Sensors J.*, vol. 23, no. 4, pp. 3787–3800, Feb. 2023.

[15] C. Tang, L. Xu, B. Yang, Y. Tang, and D. Zhao, "GRU-based interpretable multivariate time series anomaly detection in industrial control system," *Comput. Secur.*, vol. 127, Apr. 2023, Art. no. 103094.

[16] S. Ahmed, I. E. Nielsen, A. Tripathi, S. Siddiqui, R. P. Ramachandran, and G. Rasool, "Transformers in time-series analysis: A tutorial," *Circuits, Syst., Signal Process.*, vol. 42, no. 12, pp. 7433–7466, Dec. 2023.

[17] A. Vaswani et al., "Attention is all you need," in *Proc. NIPS*, vol. 30, 2017, pp. 6000–6010.

[18] C. Zhang et al., "A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 1409–1416.

[19] S. Liu et al., "Time series anomaly detection with adversarial reconstruction networks," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 4, pp. 4293–4306, Apr. 2023.

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

[21] Z. Li et al., "Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 3220–3230.

[22] G. Wang, Y. Wang, and X. Sun, "Multi-instance deep learning based on attention mechanism for failure prediction of unlabeled hard disk drives," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–9, 2021.

[23] Z. Chen, D. Chen, X. Zhang, Z. Yuan, and X. Cheng, "Learning graph structures with transformer for multivariate time-series anomaly detection in IoT," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9179–9189, Jun. 2022.

[24] J. Xu, H. Wu, J. Wang, and M. Long, "Anomaly transformer: Time series anomaly detection with association discrepancy," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Jan. 2021, pp. 1–12.

[25] S. Ma, S. Guan, Z. He, J. Nie, and M. Gao, "TPAD: Temporal-pattern-based neural network model for anomaly detection in multivariate time series," *IEEE Sensors J.*, vol. 23, no. 24, pp. 30668–30682, Dec. 2023.

[26] J. Kumari, J. Mathew, and A. Mondal, "MAD-MEL: Combining entity and metric learning for anomaly detection in multivariate time series," *IEEE Sensors J.*, vol. 24, no. 3, pp. 3144–3156, Feb. 2024.

[27] Q. Miao, D. Wang, C. Xu, J. Zhan, and C. Wu, "An unsupervised long- and short-term sparse graph neural network for multisensor anomaly detection," *IEEE Sensors J.*, vol. 24, no. 14, pp. 23088–23097, Jul. 2024.

[28] C. Ding, S. Sun, and J. Zhao, "MST-GAT: A multimodal spatial–temporal graph attention network for time series anomaly detection," *Inf. Fusion*, vol. 89, pp. 527–536, Jan. 2023.

[29] H. Zhao, Y. Wu, L. Ma, and S. Pan, "Spatial and temporal attention-enabled transformer network for multivariate short-term residential load forecasting," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–11, 2023.

[30] L. Kong, J. Yu, D. Tang, Y. Song, and D. Han, "Multivariate time series anomaly detection with generative adversarial networks based on active distortion transformer," *IEEE Sensors J.*, vol. 23, no. 9, pp. 9658–9668, May 2023.

[31] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.

[32] R. M. Neal, "Pattern recognition and machine learning," *Technometrics*, vol. 49, no. 3, p. 366, 2007, doi: 10.1198/tech.2007.s518.

[33] A. Siffer, P.-A. Fouque, A. Termier, and C. Largouet, "Anomaly detection in streams with extreme value theory," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 1067–1075.

[34] Y. Yang, C. Zhang, T. Zhou, Q. Wen, and L. Sun, "DCdetector: Dual attention contrastive representation learning for time series anomaly detection," in *Proc. 29th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2023, pp. 3033–3045.

[35] L. Ruff et al., "Deep one-class classification," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 4393–4402.

[36] S. Tuli, G. Casale, and N. R. Jennings, "TranAD: Deep transformer networks for anomaly detection in multivariate time series data," *Proc. VLDB Endow.*, vol. 15, no. 6, pp. 1201–1214, Feb. 2022.

[37] G. Zhong, F. Liu, J. Jiang, B. Wang, and C. L. P. Chen, "Refining one-class representation: A unified transformer for unsupervised time-series anomaly detection," *Inf. Sci.*, vol. 656, Jan. 2024, Art. no. 119914.

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