Multivariate Time Series Anomaly Prediction Based on Forecasting and Reconstruction Using Transformer with Temporal and Feature-wise Attention

Chihiro Maru✉[1], Masato Oguchi[2], and Ichiro Kobayashi[2]

[1] Faculty of Science and Engineering, Chuo University, Tokyo, Japan
cmaru671@g.chuo-u.ac.jp
[2] Graduate School of Humanities and Sciences, Ochanomizu University, Tokyo, Japan
{oguchi, koba}@is.ocha.ac.jp

Abstract. Anomaly detection has been actively studied, enabling the high-accuracy detection of anomalies. However, because anomaly detection assumes that an anomaly has already occurred, detecting future anomalies before they occur and preventing them from happening is impossible. Therefore, we develop a Transformer-based Anomaly Prediction (TranAP) method, which is designed to detect future anomalies. TranAP predicts future values from previous time series and uses reconstruction techniques to detect signs of anomalies using the predicted results. Detecting these precursors requires a correct understanding of the temporal characteristics of the multivariate time series (MTS). Because the timing of behavior leading to an anomaly may differ for each feature, we apply multi-head attention (ATTN) in the time dimension for each feature. Additionally, TranAP captures the dependencies between different features that the conventional ATTN could not. Because the effect of ATTN is partially diminished within the attention block, even after improvement to capture detailed information in MTS, we modify the operation of the block to preserve this effect. We demonstrate the effectiveness of TranAP by comparing it with state-of-the-art models. This improved attention mechanism of TranAP allows for a better understanding of behavior that leads to anomalies.

Keywords: anomaly prediction, multivariate time series forecasting, reconstruction, Transformer, attention block, multi-head attention

1 Introduction

Anomaly detection has been extensively studied and has demonstrated high performance. An anomaly, also known as an outlier or novelty, refers to an unusual, irregular, inconsistent, unexpected, rare, faulty, or simply a strange observation, depending on the context. Anomaly detection aims at identifying unexpected patterns or data points in real-world applications. Anomaly detection for multivariate time series (MTS) requires handling time series with several

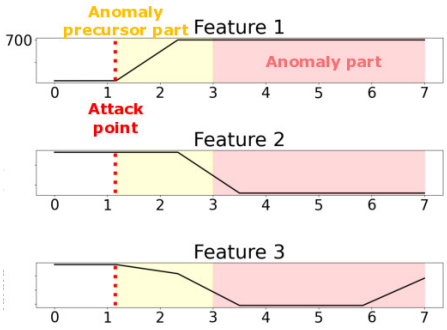


Fig. 1: Example of an anomaly precursor.

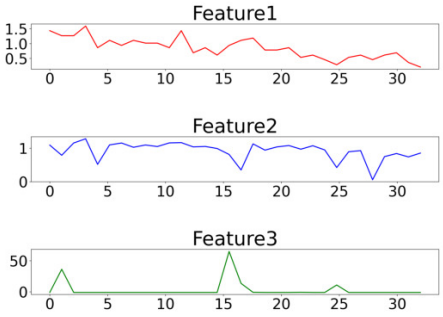


Fig. 2: Univariate time series for each feature.

features, and numerous deep learning (DL)-based models have been proposed to address this task [26, 16]. Most of these models focus on accurately detecting anomalies that have already occurred, whereas detecting future anomalies before they manifest is increasingly expected. Anomaly prediction is the process of identifying current patterns or signs that may indicate upcoming abnormal events [12]. The goal is to detect these precursors before the occurrence of anomalies, thereby enabling proactive preventive actions.

Figure 1 illustrates the transition of data points for each feature of an MTS obtained from a real-world system. An abnormal event occurs in the anomaly part owing to an external attack that sets the value of feature 1 to 700. However, this anomaly does not occur immediately after the attack. A time lag (the anomaly precursor part in Figure 1) exists between the attack and the actual occurrence of a critical abnormal event, which is recognized as an anomaly. During this period, the effects of the attack spill over to features other than feature 1, which was initially attacked, and signs triggering an abnormal event can be observed in several features.

PAD proposed an anomaly prediction model that requires anomalous data and an anomaly detection model for training [12]. We propose a Transformer-based Anomaly Prediction (TranAP) method that uses only normal data for training and does not require an anomaly detection model. TranAP uses MTS forecasting, which predicts future values from previous values and determines whether an anomaly will occur in the future by reconstructing the prediction results.

Anomaly prediction requires an accurate understanding of the temporal characteristics of the MTS. The behavior leading to an anomaly may occur at different times for each feature, as shown in Figure 2. However, general multi-head attention (ATTN) cannot capture each characteristic. Furthermore, because MTS anomaly prediction targets data with several features, more detailed MTS characteristics can be extracted by capturing the dependencies between the features. Therefore, we enhance ATTN to better capture the different temporal dependencies for each feature as well as the dependencies between features.

In addition, the attention block comprises ATTN, residual connection (RES) [9], and layer normalization (LN) [5], which contribute to the Transformer performance [32]. Previous studies on natural language processing (NLP) revealed that other components cancel the effects of ATTN [13]. To the best of our knowledge, studies on the attention block of the MTS are yet to be conducted. Further research is needed to confirm whether the same observations can be made in MTS as in natural language.

The contributions of this study are as follows:

- We proposed a novel framework specialized for anomaly prediction tasks by utilizing MTS forecasting and reconstruction. Unlike the conventional anomaly prediction model that requires anomalous data and an anomaly detection model for training, the proposed framework does not need them.
- We applied ATTN in the time direction for each feature and also in the feature direction to focus on the behavior leading to anomalies in MTS.
- We found that RES partially cancels the effect of ATTN in the attention block. Based on this finding, we modified the attention block to preserve the strong effect of the improved attention mechanism.
- We evaluated TranAP on five real-world datasets, demonstrating that anomaly prediction using MTS forecasting and reconstruction is effective. Various experiments showed that improving attention mechanisms helps capture the detailed characteristics of MTS.

2 Related Work

Anomaly Detection in MTS. Because anomaly detection requires handling time series with multiple features, numerous anomaly detection models using DL have been proposed [38, 34, 31, 35]. Most of these models focus on detecting anomalies that have already occurred, and cannot detect future anomalies.

Some anomaly detection models use techniques such as autoregression and reconstruction. These models detect anomalies by predicting or reconstructing data points within a given input and comparing them with actual values. TranAP is similar to the aforementioned models because it performs anomaly predictions based on forecasting and reconstruction. However, TranAP predicts future unseen MTS whose actual values are unknown from the given input, and thus cannot perform comparisons during anomaly prediction. Therefore, we adopt a framework to evaluate the results of MTS forecasting by reconstruction (see Section 3.2).

Anomaly Prediction. The currently proposed anomaly prediction model called PAD [12] uses training data consisting of normal and pseudo-anomalous data. In the training phase, PAD requires anomaly detection and prediction models; the latter model is trained to imitate the results of the former. Specifically, the anomaly prediction model receives the MTS at timestep $t = T - 1$ to predict whether an anomaly will occur at $t = T$. It is trained to predict the same outcome as the anomaly detection result at $t = T$ obtained by the anomaly detection model. However, it is difficult to prepare all combinations of anomalies and their precursors, and the trained anomaly prediction model has difficulty detecting

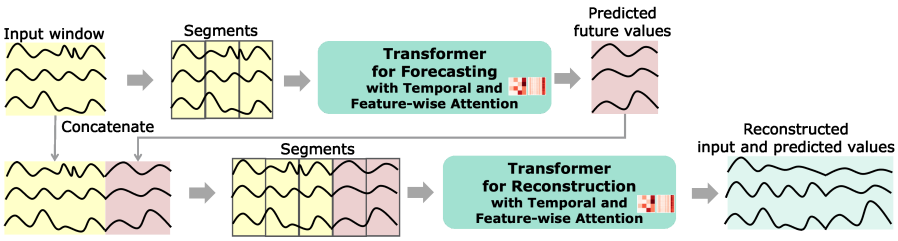


Fig. 3: TranAP architecture. The proposed model consists of Transformer-based forecasting and reconstruction.

precursors correctly for unknown anomalies that are not included in the training data [6, 36]. Furthermore, the anomaly prediction model requires an anomaly detection model for training, which makes it difficult to train them efficiently.

Transformer Attention Block Analysis. The ATTN, a key component of Transformer, has been analyzed in several studies on NLP [2, 14, 27, 19, 21, 10]. Transformer is composed of not only ATTN but also these components such as RES and LN. Previous studies have shown that other components cancel the effect of ATTN [13]. However, most Transformer-based models have only improved the ATTN and have not considered the operation of the entire attention block, e.g.,[25, 37, 34, 31, 35].

MTS forecasting using Transformer. MTS forecasting is the task of predicting future MTS values from previous values [18, 30, 22]. Transformer-based forecasting models have been proposed [39, 33, 20, 40, 7, 25]. These models improve ATTN and focus on efficiently extracting long-term time dependencies with less computational complexity. Because the MTS has several features, more detailed information about the data can be extracted by capturing the dependencies between features. Crossformer [37] is a Transformer-based model for MTS forecasting that improves ATTN to capture the dependencies between features.

3 METHODOLOGY

3.1 Problem Statement

Consider the MTS $\mathcal{X}$ consisting of M data points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M\}$, where each data point $\mathbf{x}_t \in \mathbb{R}^D$ is collected at a certain timestep t. $D (D > 1)$ is the number of features in the MTS. We adopt a window-based approach in the anomaly prediction task, similar to [12]. That is, $\mathcal{X}$ is divided into a set of windows of input length T such as $\{\mathbf{x}_{1:T}, \mathbf{x}_{1+step\_size:T+step\_size}, \ldots, \mathbf{x}_{M-T+1:M}\}$, and the input to the model is in window units.

The problem with anomaly prediction is that, given an input window of input length T, we predict whether an anomaly will occur during the next τ timesteps. For example, given an input window $\mathbf{x}_{1:T} \in \mathbb{R}^{T \times D}$ from timestep 1 to T, we

predict whether anomalies may occur between timestep $T+1$ and $T+\tau$; that is, whether an unseen $\mathbf{x}_{T+1:T+\tau} \in \mathbb{R}^{\tau \times D}$ contains anomalies. Here, τ denotes the prediction length. In this case, we output $y^{pred}_{T+1:T+\tau} \in \{0, 1\}$ (where 1 denotes an anomalous window) for each given window $\mathbf{x}_{1:T}$ as the anomaly prediction result for $\mathbf{x}_{T+1:T+\tau}$. This problem is evaluated for each testing window as in [12]. Given the MTS $\hat{\mathcal{X}}$ for testing, it is divided into a set of windows of input length T as in training. For example, given $\hat{\mathbf{x}}_{1:T}$, we can predict whether an unseen future $\hat{\mathbf{x}}_{T+1:T+\tau}$ is an anomalous window using the trained model. The correct label used for the evaluation is $\hat{y}_{T+1:T+\tau} \in \{0, 1\}$, and if $\{\hat{y}_{T+1}, \hat{y}_{T+2}, \ldots, \hat{y}_{T+\tau}\}$ contains one or more abnormal labels, we denote $\hat{y}_{T+1:T+\tau}$ as one.

In the following, we denote an input window as $\mathbf{x}_{1:T}$ and a predicted window as $\mathbf{x}_{T+1:T+\tau}$ for simplicity.

3.2 Model Structure

TranAP mainly consists of Transformer-based MTS forecasting and reconstruction, as illustrated in Figure 3. We use only normal data for training [8, 3]. TranAP enables the detection of any combination of anomalies and their precursors not included in the training dataset. Because the model has already been trained with normal data, it can successfully predict future values when a normal window is given. However, given a window that deviates from normality, the results will not be correct predictions of future trends. Such deviations can be detected as precursors of an anomaly. Similar to TranAP, some anomaly detection models use prediction techniques. However, while these models predict values within an input window, allowing for comparison with actual values, TranAP predicts a future unseen window and thus cannot be compared with actual values. Therefore, we utilize the Transformer for reconstruction to evaluate the results of MTS forecasting. The MTS, which concatenates the input and predicted values, is reconstructed. It is also trained using only normal data, enabling successful reconstruction when it receives a predicted MTS with normality. However, when it receives a predicted MTS that deviates from normality, the reconstruction fails, and the window is determined as a precursor of an anomaly.

3.3 Segmentation of Time Series

In MTS tasks, the segmentation of the input time series contributes to the accuracy of each task [37, 25]. Figure 4 shows the interactions between data points when a window is given to the trained TranAP. The (i, j) cell indicates the extent to which the jth data point on the key side contributes to the computation of the output corresponding to the ith data point on the query side. From Figure 4, we observe that the interactions tend to be divided into

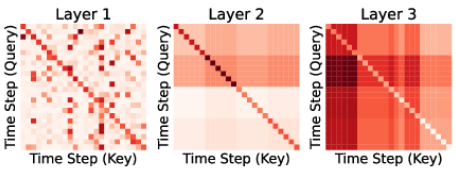


Fig. 4: Interactions between data points in each layer.

segments, particularly after the second layer. As the characteristics of temporally close data points in the time series are similar, they have similar interactions. Moreover, aggregating information from multiple data points into segments reduces the computational complexity while maintaining the accuracy of the anomaly prediction (see Section 5.1). We divide the input window into segments when performing MTS forecasting and reconstruction.

3.4 Temporal and Feature-wise ATTN

We perform ATTN in the time dimension (temporal ATTN) and feature dimension (feature-wise ATTN) in Transformer-based MTS forecasting and reconstruction.
Temporal ATTN. General Transformer-based models capture the temporal dependencies between input representations by performing ATTN in the time dimension. This computation fails to capture the temporal characteristics of each feature because all features of each input representation share the same attention map and the information of all features at a timestep is aggregated into a single embedding.

Because the timing of behaviors leading to an anomaly differs for each feature, capturing each temporal characteristic is essential for anomaly prediction. Therefore, we performe ATTN on each feature separately in the univariate time series.

Temporal ATTN receives $\mathbf{H} \in \mathbb{R}^{N \times D \times d_{\text{model}}}$ as input, where N is the number of segments. Note that $\mathbf{H}$ is a vector after trainable linear projection added with a positional embedding. We define $\mathbf{H}_{:,d}$ as a vector of all segments with feature $d (1 \leq d \leq D)$. After temporal ATTN ($\text{ATTN}^{\text{time}}$), we obtain the output $\mathbf{H}^{\text{time}}$:

$$\begin{aligned} \hat{\mathbf{H}}^{\text{time}}_{:,d} &= \text{LN}\left(\text{ATTN}^{\text{time}}\left(\mathbf{H}_{:,d}, \mathbf{H}_{:,d}, \mathbf{H}_{:,d}\right) + \mathbf{H}_{:,d}\right), \\ \mathbf{H}^{\text{time}} &= \text{LN}\left(\text{FF}\left(\hat{\mathbf{H}}^{\text{time}}\right) + \hat{\mathbf{H}}^{\text{time}}\right), \end{aligned} \tag{1}$$

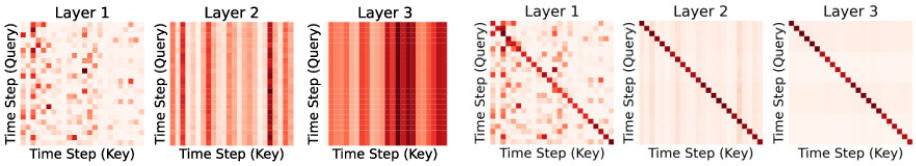
where FF denotes the feedforward network.
Feature-wise ATTN. Temporal ATTN alone does not capture feature-wise dependencies. We apply ATTN in the feature dimension ($\text{ATTN}^{\text{feature}}$) after performing temporal ATTN:

$$\begin{aligned} \hat{\mathbf{H}}^{\text{feature}}_{i,:} &= \text{LN}\left(\text{ATTN}^{\text{feature}}\left(\mathbf{H}^{\text{time}}_{i,:}, \mathbf{H}^{\text{time}}_{i,:}, \mathbf{H}^{\text{time}}_{i,:}\right) + \mathbf{H}^{\text{time}}_{i,:}\right), \\ \mathbf{H}^{\text{feature}} &= \text{LN}\left(\hat{\mathbf{H}}^{\text{feature}} + \text{FF}\left(\hat{\mathbf{H}}^{\text{feature}}\right)\right), \end{aligned} \tag{2}$$

where $\mathbf{H}_{i,:}$ is a vector of all features of the ith $(1 \leq i \leq N)$ segment.

3.5 Effect of ATTN

Although many Transformer-based models have improved the attention mechanism, [13] reported that RES cancels the effect of ATTN in NLP. Therefore, we investigate the operation in the attention block when dealing with the MTS.



(a) ATTN mixes input representations other than its own.

(b) RES preserves the original information.

Fig. 5: Interactions between representations in each layer.

Transformer consists of layers with an attention block. The attention block comprises three components: ATTN, RES, and LN.

$$\widetilde{\mathbf{H}} = \mathrm{LN}\underbrace{(\mathrm{ATTN}\,(\mathbf{H},\mathbf{H},\mathbf{H}) + \mathbf{H})}_{\mathrm{RES}}, \tag{3}$$

where $\mathbf{H} := [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_L] \in \mathbb{R}^{L \times d_{\mathrm{model}}}$ is the sequence of input representations, and $\mathbf{h}_i \in \mathbb{R}^{d_{\mathrm{model}}}$ is the ith input representation. $\widetilde{\mathbf{H}} := [\widetilde{\mathbf{h}}_1, \widetilde{\mathbf{h}}_2, \ldots, \widetilde{\mathbf{h}}_L] \in \mathbb{R}^{L \times d_{\mathrm{model}}}$ is the sequence of output representations, and $\widetilde{\mathbf{h}}_i \in \mathbb{R}^{d_{\mathrm{model}}}$ is the output corresponding to $\mathbf{h}_i$.

Among these components, ATTN and RES have contrasting effects on the computation of output representations. While ATTN mixes the input representations, RES preserves the original input representations.

We can visualize the interactions between the representations after ATTN and RES in each layer when a window is provided to the trained TranAP in Figure 5. The (i, j) cell indicates how strongly the key input $\mathbf{h}_j \in \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_L\}$ contributes to computing the query output $\widetilde{\mathbf{h}}_i$. The diagonal elements correspond to the effect of preserving the original input information, that is, preserving $\mathbf{h}_{j=i}$ when computing $\widetilde{\mathbf{h}}_i$. ATTN in Figure 5a mixes information from input representations other than its own. On the other hand, RES in Figure 5b loses the mixing effect of ATTN and strongly preserves the information from the original input representation. These results indicate that RES cancels the effect of ATTN on the MTS.

We modify Eq. (3) to adjust for the mixing effect of ATTN and the preservation effect of RES:

$$\widetilde{\mathbf{H}} = \mathrm{LN}\left(\mathrm{ATTN}\left(\mathbf{H},\mathbf{H},\mathbf{H}\right) + \lambda\mathbf{H}\right), \tag{4}$$

where $\lambda (0 \leq \lambda \leq 1)$ is a parameter that adjusts each effect. The lower λ, the stronger the influence of ATTN.

We calculate the mixing ratio r_i for ATTN, RES, and LN, which represents the ratio of the mixing effect to the sum of the mixing and preservation effects. A higher mixing ratio indicates that the mixing effect is stronger than the preservation effect. Table 1 shows the mixing ratio after performing ATTN, RES,

Table 1: Mean value of the mixing ratio for each component.

Components	$\lambda = 0$	$\lambda = 0.25$	$\lambda = 1$
Multi-head attention	95.8	96.0	95.9
Residual connection	91.7	84.5	74.2
Layer normalization	91.8	84.5	74.2

and LN with $\lambda = 0, 0.25, 1$. The values in Table 1 represent the mean ratios of the heads and layers.

These results demonstrate that, while general Transformer-based models (i.e., $\lambda = 1$) lose the mixing effect of ATTN after RES, increasing the influence of ATTN in the attention block can preserve the mixing effect even after RES. We set λ to preserve a strong mixing effect.

3.6 Anomaly Prediction Flow

Here, we define the Transformer for forecasting operations as $\mathrm{TF}^{\mathrm{pred}}$ and the Transformer for reconstruction operations as $\mathrm{TF}^{\mathrm{reconst}}$.

Training. TranAP is trained using only normal data. Given an input window $\mathbf{x}_{1:T}$, the Transformer for forecasting predicts the following future τ timesteps $\mathbf{x}_{T+1:T+\tau}$:

$$\mathbf{x}^{\mathrm{pred}}_{T+1:T+\tau} = \mathrm{TF}^{\mathrm{pred}}(\mathbf{x}_{1:T}), \tag{5}$$

where $\mathbf{x}^{\mathrm{pred}}_{T+1:T+\tau}$ denotes the predicted values. We utilize the mean squared error (MSE) to compute the difference between the predicted values and ground truth during the training phase of the Transformer for forecasting. The Transformer is trained to minimize the following objective function:

$$\mathcal{L}^{\mathrm{pred}} = \left\| \mathbf{x}^{\mathrm{pred}}_{T+1:T+\tau} - \mathbf{x}_{T+1:T+\tau} \right\|_2^2. \tag{6}$$

The Transformer for reconstruction reconstructs a vector $\mathbf{x}_{1:T} \oplus \mathbf{x}^{\mathrm{pred}}_{T+1:T+\tau} \in \mathbb{R}^{(T+\tau)\times D}$ that combines the original input $\mathbf{x}_{1:T}$ and the predicted $\mathbf{x}^{\mathrm{pred}}_{T+1:T+\tau}$:

$$\mathbf{x}^{\mathrm{reconst}}_{1:T+\tau} = \mathrm{TF}^{\mathrm{reconst}}(\mathbf{x}_{1:T} \oplus \mathbf{x}^{\mathrm{pred}}_{T+1:T+\tau}), \tag{7}$$

where $\oplus$ is the operation of the concatenation of two vectors, and $\mathbf{x}^{\mathrm{reconst}}_{1:T+\tau}$ represents the reconstructed values. The Transformer for reconstruction is trained to minimize the following objective function to reconstruct values similar to the input:

$$\mathcal{L}^{\mathrm{reconst}} = \left\| \mathbf{x}^{\mathrm{reconst}}_{1:T+\tau} - \mathbf{x}_{1:T+\tau} \right\|_2^2. \tag{8}$$

Anomaly Prediction. The trained Transformer for forecasting receives an input window $\hat{\mathbf{x}}_{1:T}$ and predicts the future $\hat{\mathbf{x}}^{\mathrm{pred}}_{T+1:T+\tau}$. Subsequently, the trained Transformer for reconstruction reconstructs $\hat{\mathbf{x}}_{1:T} \oplus \hat{\mathbf{x}}^{\mathrm{pred}}_{T+1:T+\tau}$:

$$\begin{aligned} \hat{\mathbf{x}}^{\mathrm{pred}}_{T+1:T+\tau} &= \mathrm{TF}^{\mathrm{pred}}(\hat{\mathbf{x}}_{1:T}), \\ \hat{\mathbf{x}}^{\mathrm{reconst}}_{1:T+\tau} &= \mathrm{TF}^{\mathrm{reconst}}(\hat{\mathbf{x}}_{1:T} \oplus \hat{\mathbf{x}}^{\mathrm{pred}}_{T+1:T+\tau}), \end{aligned} \tag{9}$$

where $\hat{\mathbf{x}}_{1:T+\tau}^{\text{reconst}}$ denotes the reconstructed values. The anomaly prediction score $\mathcal{A}_{\text{AP}}(\hat{\mathbf{x}}_{T+1:T+\tau}|\hat{\mathbf{x}}_{1:T})$ is defined as

$$\mathcal{A}_{\text{AP}}(\hat{\mathbf{x}}_{T+1:T+\tau}|\hat{\mathbf{x}}_{1:T}) = \left\| \hat{\mathbf{x}}_{1:T+\tau}^{\text{reconst}} - \hat{\mathbf{x}}_{1:T} \oplus \hat{\mathbf{x}}_{T+1:T+\tau}^{\text{pred}} \right\|_2^2. \tag{10}$$

$\mathcal{A}_{\text{AP}}(\hat{\mathbf{x}}_{T+1:T+\tau}|\hat{\mathbf{x}}_{1:T})$ means the degree to which an anomaly can occur in future $\hat{\mathbf{x}}_{T+1:T+\tau}$ (i.e., $\hat{\mathbf{x}}_{1:T}$ can be a precursor of an anomaly) given an input window $\hat{\mathbf{x}}_{1:T}$. A window $\hat{\mathbf{x}}_{1:T}$ with an anomaly prediction score $\mathcal{A}_{\text{AP}}(\hat{\mathbf{x}}_{T+1:T+\tau}|\hat{\mathbf{x}}_{1:T})$ that exceeds a predefined threshold is determined to be a precursor of an anomaly (i.e., $y_{T+1:T+\tau}^{\text{pred}} = 1$). Then, the result of the anomaly prediction $y_{T+1:T+\tau}^{\text{pred}} \in \{0, 1\}$ is compared with the correct label $y_{T+1:T+\tau} \in \{0, 1\}$ to evaluate the success or failure of the anomaly prediction.

4 Experiments

4.1 Anomaly Prediction in MTS

Datasets. We assess the performance of the proposed TranAP on five real-world datasets: SWaT [23], PSM [1], SMD [29], SMAP [11], and NIPS-TS-GECCO [15, 24].

Baselines. We compare TranAP with nine anomaly detection models and one anomaly prediction model **PAD**.

Anomaly detection models consist of reconstruction-based models: **LSTM-AE** [28], **MAD-GAN** [17], **USAD** [4], **CAE-M** [38], **TranAD** [31], **Anomaly Transformer** [34], and **DCdetector** [35]; the autoregression-based model **LSTM** [11]; and the density-estimation model **DAGMM** [41]. Anomaly detection models can be applied to anomaly prediction tasks by considering the precursors of anomalies as an anomaly (see Appendix A.3).

Experimental Settings. All models follow the experimental setup with an input length $T = 48$ and prediction lengths $\tau \in \{24, 36, 48, 72, 96\}$. The attention ratio λ is set to 0.5 for the SMD and SMAP datasets and 0.25 for the other datasets. The segment length L_{seg} is set to 24 for the SMD and NIPS-TS-GECCO datasets and 12 for the other datasets. We choose the F1-score as evaluation metrics to compare the performance of TranAP with those of the other models. All experiments are repeated five times and the mean of the metrics is reported.

4.2 Main Results

We first evaluate the anomaly prediction performance, as shown in Table 2. Overall, we achieve state-of-the-art results for almost all datasets. The mean F1-score over all the datasets is 8.1 points higher than that of the baselines. The F1-score is 2.5–6.8 points higher than that of the baselines for datasets with relatively clear precursors of anomalies, such as the SWaT and PSM datasets. On the other hand, for datasets with small anomaly precursors, such as the SMD and SMAP datasets, the F1-score is comparable to that of the baselines. The

NIPS-TS-GECCO dataset is a challenging dataset with several types of anomalies. However, TranAP is successful, whereas the baselines fail to predict anomalies. The standard deviations of the F1-score of TranAP in the five repetitions of the experiments are within 0.14%–4.09% for all datasets.

Despite a fixed input length and different prediction lengths, the F1-score does not change significantly. In actual operations, it is desirable to detect the precursors of anomalies further into the future. Therefore, the prediction length should be increased to the extent that the evaluation metrics, such as the F1-score, do not change significantly.

5 Analysis

5.1 Ablation Study

Input Length.

We investigate the effect of input length by changing the input lengths $T \in \{12, 24, 48, 96, 168\}$ with a fixed prediction length $\tau = 96$ in Figure 6. The F1-score is the highest when $T = 12$ for the SWaT dataset and $T = 24$ for the PSM dataset. In these datasets, the behavior leading to anomalies is likely to have occurred at slightly earlier time steps when the prediction length is 96. The longer the input length, the lower the value of the F1-score. If the input length is too long, the input contains information that is irrelevant to anomaly prediction.

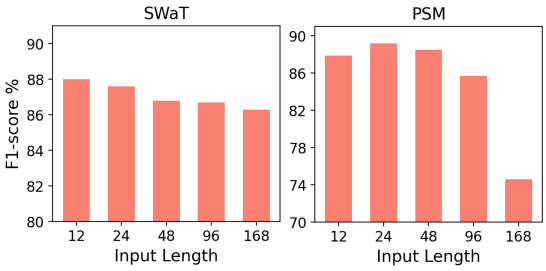


Fig. 6: F1-score when changing input lengths.

Effect of Segment.

We investigate the average running time per iteration and F1-score for different segment lengths in the SWaT and PSM datasets in Figure 7. We set an input length $T = 48$, prediction length $\tau = 48$, and segment lengths $L_{\text{seg}} \in \{1, 3, 6, 12, 24, 48\}$. The average running time per iteration is reduced by 55.4%–81.5% without decreasing the F1-score. Note that the F1-score is the highest with $L_{\text{seg}} = 24$ for the SWaT dataset and $L_{\text{seg}} = 6$ for the PSM dataset. In the PSM dataset with $L_{\text{seg}} = 48$, the F1-score decreases because the time steps with different temporal characteristics are grouped into a single segment.

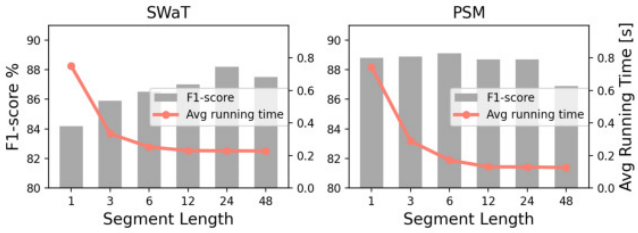


Fig. 7: Running time and F1-score when changing segment lengths.

Table 2: MTS anomaly prediction results. We use an input length $T = 48$ and prediction lengths $\tau \in \{24, 36, 48, 72, 96\}$. The evaluation metrics are the precision (P), recall (R), and F1-score (F1). The best results are in **bold**, and the second best are underlined. All values are in percentages.

Models		Anomaly Detection Models for Anomaly Prediction Tasks																									Anomaly Prediction Models					
		LSTM			LSTM-AE			DAGMM			MAD-GAN			USAD			CAE-M			TranAD			Ano Trans			DCdetector			PAD			**TranAP**		
Metric		P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
SWaT	24	15.2	92.8	26.1	96.4	70.9	81.7	67.1	82.7	74.0	92.3	75.2	82.9	78.2	81.7	79.6	62.2	80.2	70.1	90.7	76.2	82.8	18.1	91.6	30.2	98.6	68.6	80.9	69.2	58.3	56.1	92.8	81.7	**86.9**
	36	15.2	91.0	26.1	96.7	74.0	83.8	76.8	79.3	77.1	84.2	73.9	77.1	78.3	81.3	79.4	62.4	79.9	70.1	90.8	75.9	82.6	18.3	91.1	30.5	99.4	68.9	81.3	31.1	84.5	33.6	93.3	81.8	**87.1**
	48	15.4	91.1	26.4	97.0	74.8	84.5	76.9	79.8	77.4	92.8	74.0	82.3	76.4	82.1	78.8	62.7	80.2	70.4	89.7	77.2	82.8	18.4	91.2	30.7	99.3	68.3	80.9	49.8	75.6	43.3	93.4	81.4	**87.0**
	72	15.5	90.7	26.5	97.0	75.2	84.8	83.2	78.6	80.1	81.6	77.6	78.5	77.4	81.6	79.1	63.1	79.7	70.4	90.1	78.1	83.6	18.6	92.0	31.0	99.2	67.2	80.1	65.6	68.4	48.8	92.9	82.3	**87.3**
	96	15.8	90.6	26.9	97.1	74.4	84.3	81.8	78.2	79.0	91.5	76.9	83.5	82.0	79.9	80.7	63.4	79.2	70.4	91.4	77.6	83.9	18.9	91.0	31.2	97.2	52.6	64.6	32.1	89.0	33.8	93.4	81.0	**86.8**
PSM	24	60.5	89.1	72.0	67.9	83.2	74.6	78.2	87.4	82.6	77.3	87.2	81.9	81.7	86.8	84.1	70.9	88.2	78.6	85.9	85.2	85.3	40.1	56.4	46.8	86.9	83.0	78.9	39.0	87.0	45.9	92.7	86.4	**89.2**
	36	64.0	84.2	72.4	67.3	85.2	75.1	69.8	87.6	77.7	68.6	87.9	77.0	73.5	87.5	79.9	71.4	87.3	78.6	75.3	86.8	80.5	51.4	78.0	61.9	83.6	81.5	82.5	35.1	83.9	48.4	96.4	82.4	**88.8**
	48	60.6	87.5	71.6	67.3	77.4	72.0	78.8	85.8	82.1	80.0	84.6	82.0	74.4	86.5	79.9	71.9	86.3	78.5	86.7	80.9	83.3	55.6	73.8	63.3	46.9	44.5	45.6	37.0	60.5	29.4	97.5	81.4	**88.7**
	72	62.8	85.8	72.4	68.0	76.2	71.8	71.9	85.7	78.2	70.7	86.0	77.6	75.1	85.4	79.9	72.8	84.8	78.4	78.5	84.6	81.4	39.8	56.5	46.7	56.6	54.1	55.3	32.2	99.9	48.7	93.9	83.4	**88.2**
	96	64.5	84.0	72.9	69.1	75.8	72.2	80.3	82.9	81.6	80.4	82.5	81.3	76.4	83.4	79.7	73.7	83.3	78.2	92.5	77.3	84.0	55.0	75.0	63.4	76.5	60.8	64.9	42.2	80.6	41.3	97.0	81.4	**88.5**
SMD	24	46.5	55.8	50.4	41.2	56.1	47.2	41.2	76.1	53.4	34.7	77.8	48.0	37.7	76.9	50.6	43.5	74.0	54.8	38.3	76.4	50.9	45.4	29.1	35.1	33.4	85.3	46.3	6.5	47.6	11.3	45.0	73.9	**55.9**
	36	47.7	52.8	50.1	47.2	51.5	49.0	41.8	75.5	53.8	43.1	73.1	54.2	42.3	72.3	53.4	45.3	72.6	55.5	43.8	69.5	53.7	29.8	43.0	32.4	44.9	77.1	52.7	5.8	66.5	10.0	46.9	68.8	**55.8**
	48	46.3	51.5	48.5	48.1	50.2	48.9	44.2	72.9	55.0	41.4	73.6	52.9	42.3	73.3	53.6	47.1	69.9	56.3	43.1	71.3	53.7	79.9	42.8	55.1	41.6	79.0	48.0	5.5	67.6	10.0	54.2	62.1	**57.7**
	72	49.1	46.5	47.5	52.0	47.1	49.4	47.8	71.2	57.2	48.5	69.5	57.1	48.6	69.0	57.0	50.8	68.0	**58.1**	47.5	69.2	56.2	69.0	24.6	36.2	36.2	90.2	48.0	7.3	99.1	13.6	56.0	55.4	55.7
	96	49.6	47.9	48.3	53.6	46.5	49.6	52.1	70.9	60.1	46.9	73.8	57.3	48.9	72.2	58.3	54.5	68.0	**60.5**	46.2	73.7	56.8	60.6	33.1	42.7	43.5	92.2	57.1	7.8	74.4	13.4	48.9	69.5	57.0
SMAP	24	74.8	53.1	62.1	78.5	52.4	62.9	85.6	49.1	62.3	87.2	49.1	62.8	88.5	48.6	62.8	85.5	49.6	62.8	89.7	48.4	62.9	57.0	42.9	48.9	77.1	49.8	45.3	10.8	56.4	17.5	77.7	62.1	**68.9**
	36	74.9	53.0	62.1	78.5	52.1	62.7	85.5	48.9	62.2	86.8	49.2	62.8	88.6	48.6	62.8	85.7	49.6	62.8	88.6	48.6	62.8	57.6	42.3	48.7	64.6	33.6	38.7	11.0	76.7	19.3	88.2	52.3	**65.7**
	48	74.3	52.7	61.7	78.9	52.2	**62.9**	85.6	48.7	62.1	87.0	48.9	62.6	88.9	48.3	62.6	85.8	49.4	62.7	88.7	48.4	62.6	71.4	53.9	61.4	69.8	47.7	48.7	13.6	60.1	14.4	79.4	51.9	62.8
	72	74.5	52.6	61.7	79.3	51.9	62.8	87.0	49.0	62.7	88.9	49.5	63.6	89.6	49.6	63.9	86.1	49.1	62.5	89.0	48.7	62.9	73.0	54.2	62.2	68.9	59.6	61.5	5.6	40.0	9.8	83.6	51.8	**64.0**
	96	76.0	52.5	62.1	79.8	51.7	62.8	86.6	51.6	64.7	88.8	51.8	65.5	90.4	51.7	**65.8**	87.1	51.2	64.5	88.5	50.8	64.5	43.7	31.5	36.6	77.0	49.3	56.0	11.4	80.0	20.0	80.6	52.1	63.3
GECCO	24	83.0	22.3	35.2	83.6	22.3	35.2	40.2	85.7	26.2	85.7	26.2	40.2	85.7	26.2	40.2	87.1	26.2	40.3	85.7	26.2	40.2	44.1	14.9	21.7	14.3	31.5	19.3	47.2	31.5	15.9	38.3	77.5	**51.3**
	36	82.1	20.0	32.1	84.7	20.0	32.4	37.6	87.0	24.0	87.0	24.0	37.6	87.0	24.0	37.6	88.2	24.0	37.7	87.0	24.0	37.6	49.4	12.7	20.2	17.5	46.2	23.3	15.1	45.3	10.3	42.4	74.4	**54.0**
	48	84.9	18.4	30.2	87.1	18.4	30.3	88.0	22.5	35.8	88.2	22.5	35.8	88.0	22.5	35.8	89.2	22.5	35.9	88.0	22.5	35.8	29.7	8.8	13.6	19.7	52.3	27.8	13.1	41.0	3.9	45.8	74.1	**56.6**
	72	93.4	16.2	27.7	93.9	16.2	27.7	89.7	20.4	33.3	33.3	89.7	20.4	89.7	20.4	33.3	90.7	20.4	33.3	89.7	20.4	33.3	37.0	11.4	14.5	28.0	65.0	35.2	32.4	51.0	12.9	53.7	73.8	**62.1**
	96	94.0	14.9	25.7	98.6	14.9	25.9	90.9	19.2	31.6	90.9	19.2	31.6	90.9	19.2	31.6	91.8	19.2	31.7	90.9	19.2	31.6	50.2	13.5	20.5	42.2	46.9	43.5	4.7	60.6	6.2	55.6	73.6	**63.2**
Average		57.6	59.9	47.9	76.4	54.8	54.7	70.7	67.1	61.6	74.3	64.1	61.1	75.2	62.4	62.8	71.7	61.7	61.4	79.1	60.7	59.9	45.3	50.2	38.8	60.9	62.2	59.7	25.2	67.4	24.7	73.6	71.9	**70.9**

Table 4: F1-score when changing attention mechanisms.

Attention mechanism		Original	F-ATTN	T-ATTN	TF-ATTN
SWaT	24	83.2	86.3	85.4	**86.9**
	96	85.3	86.4	86.6	**86.8**
PSM	24	88.4	88.7	**90.2**	89.2
	96	87.3	87.3	**88.7**	88.5
SMD	24	38.6	53.6	53.3	**55.9**
	96	44.1	51.4	53.1	**54.7**
SMAP	24	59.3	62.5	61.9	**63.1**
	96	61.0	62.6	**63.3**	63.0
GECCO	24	50.0	50.5	50.7	**51.3**
	96	57.3	58.9	63.0	**63.2**

Effect of ATTN.

We examine the impact of the mixing effect of ATTN and the preservation effect of RES on the anomaly prediction performance by adjusting the attention ratio λ in Eq. (4). Table 3 denotes the F1-score with an input length $T = 48$, prediction length $\tau = \{24, 96\}$, and attention ratio $\lambda = \{0, 0.25, 1\}$ in the SWaT and PSM datasets. Although increasing the mixing effect improves the F1-score, completely eliminating the preservation effect (i.e., $\lambda = 0$) tends to degrade the performance of anomaly prediction. Therefore, the increased mixing effect of ATTN leads to improved performance, and the preservation effect of RES also contributes to anomaly prediction in the attention block.

Table 3: F1-score when changing attention ratios.

Attention ratio		0	0.25	1
SWaT	24	86.8	**86.9**	86.6
	96	86.5	**86.8**	86.6
PSM	24	88.0	**89.2**	87.3
	96	87.2	**88.5**	88.0

Effect of Temporal and Feature-wise ATTN. We perform anomaly prediction on all datasets using four different attention mechanisms: general (**Original**), feature-wise (**F-ATTN**), temporal (**T-ATTN**), and temporal and feature-wise ATTN (**TF-ATTN**).

Table 4 denotes the F1-score with an input length $T = 48$ and prediction lengths $\tau \in \{24, 96\}$. In all cases, the F1-score is higher than that of the other attention mechanisms for **Original**. **F-ATTN** achieves a higher F1-score than **Original** for all prediction lengths, indicating that it is important to reflect the dependencies between features in the final computed representations. The longer the prediction length, the more **T-ATTN**, which treats each feature independently, functions. Therefore, each feature has a different temporal dependency, which is important for anomaly prediction. This indicates that performing **TF-ATTN** enables detailed extraction of the MTS characteristics and contributes to improving anomaly prediction.

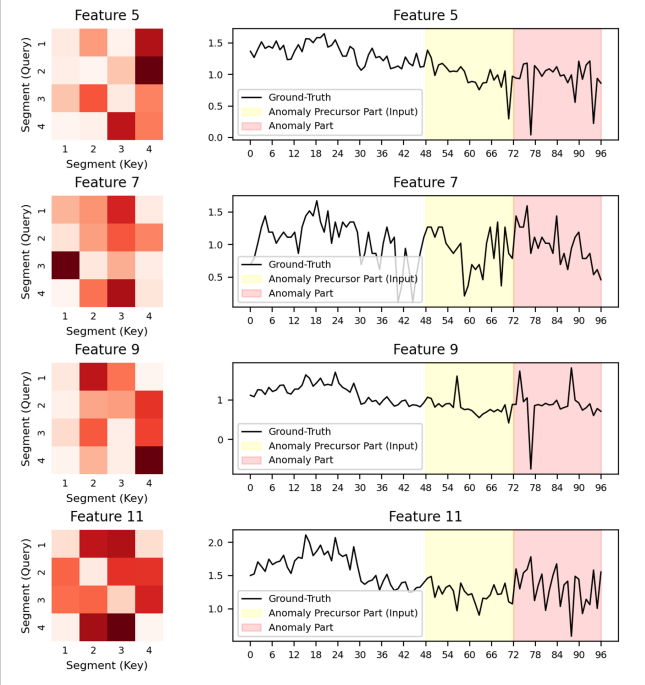


Fig. 8: Attention maps after temporal ATTN.

5.2 Visualization of Attention Maps

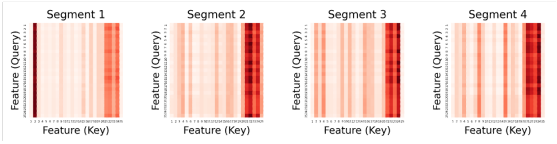


Fig. 9: Attention maps after feature-wise ATTN.

We visualize the interactions between representations after temporal and feature-wise ATTN to confirm the effectiveness of the improved attention mechanism. A window $\hat{\mathbf{x}}_{48:71}$ from the PSM dataset with an input length $T = 24$ and segment length $L_{\text{seg}} = 6$ which contains the precursors of the anomalies (yellow highlighted area of Figure 8 to the right) is fed to the trained encoder. An anomaly occurs in $\hat{\mathbf{x}}_{72:95}$ (red highlighted area of Figure 8 to the right).

Figure 8 shows the attention maps of $\hat{\mathbf{x}}_{48:71}$ for each feature after performing temporal ATTN. The attention maps of each feature differ in time direction, indicating that each feature has different temporal characteristics. In the attention maps, the attention weights of the segments (keys) that contain behaviors leading to anomalies tends to be higher for each segment (query). This indicates that temporal ATTN can focus on the behaviors leading to the anomaly.

Figure 9 shows the attention maps between features when performing feature-wise ATTN after temporal ATTN. The dependencies between features are cap-

tured, and this information is essential for capturing the MTS characteristics. When we compare each attention map of the segments, all segments exhibit similar dependencies between features. These segments are close in time and thus capture similar feature-wise dependencies.

6 Conclusions

In this paper, we proposed an anomaly prediction framework based on MTS forecasting and reconstruction using Transformer. Our model is trained to predict future trends using only normal data. Therefore, when given a time series exhibiting deviations from normal features, the results will not be accurate predictions of future trends, which can then be detected as precursors to anomaly occurrences through reconstruction. Detecting precursors of anomalies requires an accurate understanding of the temporal characteristics of the MTS. We modified the attention mechanism of each Transformer to perform ATTN in the time and feature directions. However, an NLP study reported that the effect of the ATTN was diminished after applying RES, despite using improved attention mechanisms. Therefore, we confirmed the same phenomenon in the context of MTS. We successfully preserved the strong effects of the improved attention mechanism by modifying the operation of the attention block. We enhanced the anomaly prediction performance by introducing these improvements.

Acknowledgements

This work was supported by the 4th Research Grant from the Hagiwara Foundation of Japan and by JSPS KAKENHI Grant Number JP24999361 (Grant-in-Aid for Early-Career Scientists).

References

1. Abdulaal, A., Liu, Z., Lancewicki, T.: Practical approach to asynchronous multivariate time series anomaly detection and localization. Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining pp. 2485–2494 (2021)
2. Abnar, S., Zuidema, W.: Quantifying attention flow in transformers. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics pp. 4190–4197 (Jul 2020)
3. Ahmad, S., Lavin, A., Purdy, S., Agha, Z.: Unsupervised real-time anomaly detection for streaming data. Neurocomputing **262**, 134–147 (2017)
4. Audibert, J., Michiardi, P., Guyard, F., Marti, S., Zuluaga, M.A.: Usad: Unsupervised anomaly detection on multivariate time series. Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining pp. 3395–3404 (2020)
5. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)

6. Chen, Z., Chen, D., Zhang, X., Yuan, Z., Cheng, X.: Learning graph structures with transformer for multivariate time-series anomaly detection in iot. IEEE Internet of Things Journal **9**(12), 9179–9189 (2021)
7. Du, D., Su, B., Wei, Z.: Preformer: predictive transformer with multi-scale segment-wise correlations for long-term time series forecasting. ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp. 1–5 (2023)
8. Goldstein, M., Uchida, S.: A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. PloS one **11**(4), e0152173 (2016)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition pp. 770–778 (2016)
10. Htut, P.M., Phang, J., Bordia, S., Bowman, S.R.: Do attention heads in bert track syntactic dependencies? arXiv preprint arXiv:1911.12246 (2019)
11. Hundman, K., Constantinou, V., Laporte, C., Colwell, I., Soderstrom, T.: Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining pp. 387–395 (2018)
12. Jhin, S.Y., Lee, J., Park, N.: Precursor-of-anomaly detection for irregular time series. Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining p. 917–929 (2023)
13. Kobayashi, G., Kuribayashi, T., Yokoi, S., Inui, K.: Incorporating Residual and Normalization Layers into Analysis of Masked Language Models. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing pp. 4547–4568 (Nov 2021)
14. Kovaleva, O., Romanov, A., Rogers, A., Rumshisky, A.: Revealing the dark secrets of BERT. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) pp. 4365–4374 (Nov 2019)
15. Lai, K.H., Zha, D., Xu, J., Zhao, Y.: Revisiting time series outlier detection: Definitions and benchmarks. NeurIPS Datasets and Benchmarks (2021)
16. Landauer, M., Onder, S., Skopik, F., Wurzenberger, M.: Deep learning for anomaly detection in log data: A survey. Machine Learning with Applications **12**, 100470 (2023)
17. Li, D., Chen, D., Jin, B., Shi, L., Goh, J., Ng, S.K.: Mad-gan: Multivariate anomaly detection for time series data with generative adversarial networks. International conference on artificial neural networks pp. 703–716 (2019)
18. Lim, B., Zohren, S.: Time-series forecasting with deep learning: a survey. Philosophical Transactions of the Royal Society A **379**(2194), 20200209 (2021)
19. Lin, Y., Tan, Y.C., Frank, R.: Open sesame: Getting inside BERT's linguistic knowledge. Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP pp. 241–253 (Aug 2019)
20. Liu, S., Yu, H., Liao, C., Li, J., Lin, W., Liu, A.X., Dustdar, S.: Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. International conference on learning representations (2021)
21. Mareček, D., Rosa, R.: From balustrades to pierre vinken: Looking for syntax in transformer self-attentions. Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP (Aug 2019)
22. Masini, R.P., Medeiros, M.C., Mendes, E.F.: Machine learning advances for time series forecasting. Journal of economic surveys **37**(1), 76–111 (2023)

23. Mathur, A.P., Tippenhauer, N.O.: Swat: A water treatment testbed for research and training on ics security. 2016 international workshop on cyber-physical systems for smart water networks (CySWater) pp. 31–36 (2016)
24. Moritz, S., Rehbach, F., Chandrasekaran, S., Rebolledo, M., Bartz-Beielstein, T.: Gecco industrial challenge 2018 dataset: A water quality dataset for the'internet of things: Online anomaly detection for drinking water quality'competition at the genetic and evolutionary computation conference 2018, kyoto, japan. Kyoto, Japan (2018)
25. Nie, Y., H. Nguyen, N., Sinthong, P., Kalagnanam, J.: A time series is worth 64 words: Long-term forecasting with transformers. International Conference on Learning Representations (2023)
26. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep learning for anomaly detection: A review. ACM computing surveys (CSUR) **54**(2), 1–38 (2021)
27. Reif, E., Yuan, A., Wattenberg, M., Viegas, F.B., Coenen, A., Pearce, A., Kim, B.: Visualizing and measuring the geometry of bert. Advances in Neural Information Processing Systems **32** (2019)
28. Said Elsayed, M., Le-Khac, N.A., Dev, S., Jurcut, A.D.: Network anomaly detection using lstm based autoencoder. Proceedings of the 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks pp. 37–45 (2020)
29. Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., Pei, D.: Robust anomaly detection for multivariate time series through stochastic recurrent neural network. Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining pp. 2828–2837 (2019)
30. Torres, J.F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F., Troncoso, A.: Deep learning for time series forecasting: a survey. Big Data **9**(1), 3–21 (2021)
31. Tuli, S., Casale, G., Jennings, N.R.: TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data. Proceedings of VLDB **15**(6), 1201–1214 (2022)
32. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
33. Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Advances in Neural Information Processing Systems **34**, 22419–22430 (2021)
34. Xu, J., Wu, H., Wang, J., Long, M.: Anomaly transformer: Time series anomaly detection with association discrepancy. arXiv preprint arXiv:2110.02642 (2021)
35. Yang, Y., Zhang, C., Zhou, T., Wen, Q., Sun, L.: Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining p. 3033–3045 (2023)
36. Zhang, C., Zhou, T., Wen, Q., Sun, L.: Tfad: A decomposition time series anomaly detection architecture with time-frequency analysis. Proceedings of the 31st ACM International Conference on Information & Knowledge Management pp. 2497–2507 (2022)
37. Zhang, Y., Yan, J.: Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. International Conference on Learning Representations (2023)
38. Zhang, Y., Chen, Y., Wang, J., Pan, Z.: Unsupervised deep anomaly detection for multi-sensor time-series signals. IEEE Transactions on Knowledge and Data Engineering (2021)

39. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting. Proceedings of the AAAI conference on artificial intelligence **35**(12), 11106–11115 (2021)
40. Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., Jin, R.: Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. International Conference on Machine Learning pp. 27268–27286 (2022)
41. Zong, B., Song, Q., Min, M.R., Cheng, W., Lumezanu, C., Cho, D., Chen, H.: Deep autoencoding gaussian mixture model for unsupervised anomaly detection. International conference on learning representations (2018)