Temporal-Frequency Masked Autoencoders for Time Series Anomaly Detection

Yuchen Fang[1], Jiandong Xie[2], Yan Zhao[3,*], Lu Chen[4], Yunjun Gao[4], Kai Zheng[1,*]
[1]University of Electronic Science and Technology of China, China
[2]Huawei Cloud Database Innovation Lab, China
[3]Aalborg University, Denmark
[4]Zhejiang University, China
fyclmiss@gmail.com, xiejiandong@huawei.com, yanz@cs.aau.dk, {luchen,gaoyj}@zju.edu.cn, zhengkai@uestc.edu.cn

***Abstract*—In the era of observability, massive amounts of time series data have been collected to monitor the running status of the target system, where anomaly detection serves to identify observations that differ significantly from the remaining ones and is of utmost importance to enable value extraction from such data. While existing reconstruction-based methods have demonstrated favorable detection capabilities in the absence of labeled data, they still encounter issues of training bias on abnormal times and distribution shifts within time series. To address these issues, we propose a simple yet effective Temporal-Frequency Masked AutoEncoder (TFMAE) to detect anomalies in time series through a contrastive criterion. Specifically, TFMAE uses two Transformer-based autoencoders that respectively incorporate a window-based temporal masking strategy and an amplitude-based frequency masking strategy to learn knowledge without abnormal bias and reconstruct anomalies by the extracted normal information. Moreover, the dual autoencoder undergoes training through a contrastive objective function, which minimizes the discrepancy of representations from temporal-frequency masked autoencoders to highlight anomalies, as it helps alleviate the negative impact of distribution shifts. Finally, to prevent overfitting, TFMAE adopts adversarial training during the training phase. Extensive experiments conducted on seven datasets provide evidence that our model is able to surpass the state-of-the-art in terms of anomaly detection accuracy.**

***Index Terms*—time series anomaly detection, temporal-frequency analysis, masked autoencoder**

I. Introduction

Time series is a sequence of temporally ordered observations, and the analysis of it has swiftly become a focal task in academic and industrial research, propelled by advances in our capability of time series data collection and storage in the context of sensor networks, cloud computing, and especially the recent emerging concept of observability [1], [2]. Anomaly detection is an indispensable task in time series analysis, determining whether the data conforms to the normal data distribution, and the non-conforming parts are called anomalies. Timely alerts for anomalies can empower system maintainers to proactively conduct maintenance, enabling sustainability and safety in real applications such as fraud detection [3], intrusion detection [4], and energy management [5].

*Corresponding author: Kai Zheng and Yan Zhao. Kai Zheng is with Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China

However, providing accurate time series anomaly detection is a non-trivial task because patterns of time series are intricate and dynamic in various applications, which makes it hard to seek a general manner for defining anomalies accurately. Moreover, with the scarcity of labeled data, the progress of supervised methods for time series anomaly detection is impeded. Unsupervised approaches can detect anomalies without labeled data, which often relies on density-based methods (leveraging discrepancies between neighbors) and clustering-based methods (utilizing distances from cluster centers). As the time series data scale grows larger and deep learning excels in data analysis, recent endeavors turn to reconstructing time series with intricate temporal correlations and multivariate dependencies by using various deep learning models. These models focus on the discrepancy between the reconstructed and original time series. Innovations like OmniAno [6], TimesNet [7], and TranAD [8] introduce the recurrent neural network, convolution neural network, and Transformer network into the time series anomaly detection, respectively.

Despite recent improvements, existing deep reconstruction-based methods still face the following challenges.

Challenge I: Abnormal bias. Deep learning models heavily rely on the learned knowledge from data during the training phase, yet extracting information from time series proves more challenging than language and image data due to its intricate patterns [9], [10]. Therefore, learning a high-quality reconstruction model becomes particularly hard, especially in the presence of knowledge-agnostic abnormal bias. As depicted in the left of Figure 1, TimesNet [7], a typical reconstruction model, is able to well reconstruct normal series, yet overfits abnormal observations, leading to performance degradation. This phenomenon arises from the incorporation of misleading abnormal bias, which can blend into normal patterns through the commonly used temporal modeling. Unfortunately, many existing reconstruction-based methods overlook this abnormal bias, resulting in suboptimal performance.

Challenge II: Time series distribution shift. The distribution shift of time series introduces a crucial factor, wherein patterns learned from training data may become unsuitable or even incorrect for testing data, which results in erroneous reconstructed time series. As shown in the right of Figure 1, the curve of the cumulative score on the testing data goes up

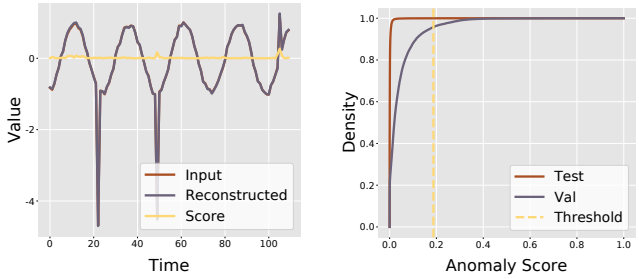


Fig. 1: Left: TimesNet [7] conducts anomaly detection on the synthetic NIPS-TS-Global dataset. Right: Cumulative distribution function (CDF) of the anomaly scores on the real-world SMAP validation and test sets for TimesNet.

faster than those on the validation data, which is attributed to the distribution shift of time series and contributes to poor generalization on the threshold. Despite there are two types of existing techniques that can be incorporated into reconstruction-based methods to mitigate distribution shifts: normalization and decomposition. The static statistics-based normalization [11], [12] and pre-defined parameters-based decomposition [9], [13]–[15] exhibit limitations in generalization. Moreover, methods relying on learned statistics [16] and parameters [17], [18] remain susceptible to the influence of distribution shifts.

This work aims to address the above challenges by designing a pioneering model, called Temporal-Frequency Masked AutoEncoder (TFMAE), which is trained through an adversarial contrastive objective function. For abnormal bias, recognizing the importance of purifying time series, we implement a window-based temporal masking strategy to eliminate potential observation anomalies (*e.g.*, global and contextual observation anomalies) with a large coefficient of variation, and an amplitude-based frequency masking strategy to eliminate potential pattern anomalies (*e.g.*, trend and seasonal anomalies) with a small amplitude. These strategies are beneficial for learning without abnormal bias and reconstructing anomalies with the learned normal knowledge. To tackle the distribution shift issue, we initially devise two Transformer-based autoencoders to generate distinct representations of our temporal and frequency masking-based time series. Then a novel contrastive objective function is introduced to minimize the discrepancy between these representations. Finally, the contrastive discrepancy replaces the conventional reconstruction error for anomaly detection, as the discrepancy between anomalies and their corresponding normal-recovered views exceeds that of normal representations. Breaking free from the reconstruction paradigm, the contrastive criterion leverages the fact that the similarity of different views is distribution-agnostic [19]. Besides, to avoid over-fitting in minimizing discrepancy, adversarial training is integrated into our TFMAE.

Our contributions can be summarized as follows:

- To eliminate potential abnormal observations and patterns before modeling, we present a window-based temporal masking strategy and an amplitude-based frequency masking strategy. Therefore, autoencoders with purified inputs are not misled by observation and pattern anomalies, *i.e.*, TFMAE is an *abnormal bias-resistant model*.
- To the best of our knowledge, this work is the first study that replaces the reconstruction error with the temporal and frequency masking-based contrastive criterion for time series anomaly detection, which is a *distribution shift unaffected model*.
- We conduct extensive experiments on seven public real time series datasets, and the results offer insight into the effectiveness and efficiency of TFMAE.

Section II surveys the related work. The problem statement and the system overview are introduced in Section III. We then present TFMAE in Section IV, followed by the experimental results in Section V. Section VI concludes this paper.

II. Related Work

A. Time Series Anomaly Detection

Numerous studies have been conducted on time series anomaly detection. Based on the manners of detecting anomalies, we can divide the methods into five types: *density*-based methods, *clustering*-based methods, *label*-based methods, *reconstruction*-based methods, and *contrastive*-based methods.

Density-based methods mainly focus on the discrepancy between observations and their neighbors. The density-based local outlier factor (LOF) [20] and connectivity-based connectivity outlier factor (COF) [21] are two typical traditional density-based methods. As deep representational learning gained traction, advanced models such as DAGMM [22] and MPPCACD [23] learn low-dimensional representations of time series and utilize the Gaussian Mixture Model to derive their density, which achieves highly accurate detection results.

Clustering-based methods leverage distances between observations and the cluster center to discern anomalies. Taking clustering into consideration, classic techniques such as support vector data description (SVDD) [24] and one-class support vector machine (OC-SVM) [25] search the hypersphere and hyperplane in the kernel space. Subsequently, DSVDD [26] and THOC [27] are proposed to seek clusters in the deep latent and hierarchical space for anomaly detection.

Label-based methods engage in supervised classification during the training phase. Microsoft [28] pioneered the use of human-generated labels to train a CNN model for anomaly detection. Extending this approach, RobustTAD [29] introduces the assignment of label and value weights to labels and crucial points to improve detection performance.

Reconstruction-based methods have been a cornerstone of research, particularly in the absence of labeled training data. These models are trained in an unsupervised manner, detecting anomalies by discerning discrepancies between original and reconstructed time series. In the earlier years, reconstruction relied on statistical methods like ARIMA [30]. The advent of deep learning gave rise to methods such as Omni-Ano [6], HIFI [31], Interfusion [32], TFAD [13], VQRAE [33], RDAE [34], and TimesNet [7]. They reconstruct time series

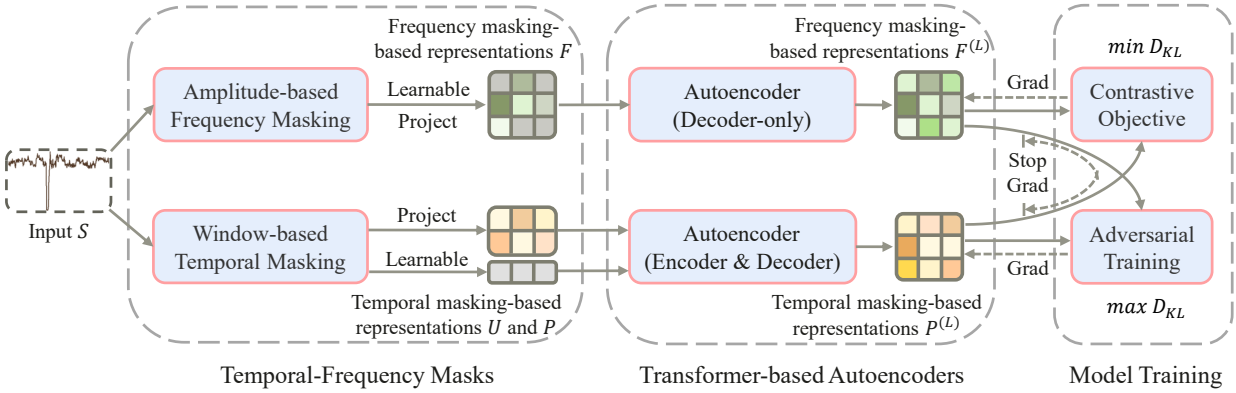


Fig. 2: Overall architecture of the proposed TFMAE. The left part contains temporal-frequency masks, the middle part is transformer-based autoencoders, and the right part shows objective functions. D_{KL} means the Kullback–Leibler divergence.

from the current input. Furthermore, adversarial training has been incorporated into reconstruction-based methods for robust detection. Examples include BeatGAN [35], USAD [36], DAEMON [37], [38], SA-GAN [39], and TranAD [8].

Contrastive learning-based methods are designed to detect anomalies through discrepancies between representations derived from various aspects of the same input. TimeAutoAD [40] and CTAD [41] construct negative samples by injecting noise, and judge anomalies according to the original-negative discrepancy. AnoTran [42] minimizes distances between prior and association representations to mitigate distribution shifts in anomaly detection. Conversely, DCdetector [43] minimizes distances between different patch size-based association representations to avoid prior knowledge. Notably, previous approaches focus solely on temporal aspects, neglecting the crucial role of frequency features in detecting abnormal patterns.

B. Masked Autoencoder

Building on the success of masked language modeling, exemplified by BERT, the masked autoencoder (MAE) [44] has emerged as a potent self-supervised vision learner for image representations. Specifically, MAE randomly masks tokens within images, allowing its encoder to learn from unmasked tokens and the decoder to recover masked ones. The STMAE [45] extends the MAE paradigm into video modeling, achieving remarkable performance. While MAE has found applications in various domains such as non-Euclidean data modeling (GraphMAE [46]) and time series modeling (STEP [47]), conventional random-based MAE methods lack task-specific adaptability. In response, innovative adaptations have been introduced: MAERec [48] masks tokens with higher structural consistency for robust sequential recommendation, GPT-ST [49] uses cluster-aware masking for cross-cluster knowledge learning, and EMAE [50] masks learned semantic parts in images for fast training. However, these methods can not be directly transferred into time series anomaly detection.

III. Preliminaries

1) Time Series Representations: We define a time series S as a sequence of observations representing numerical values produced by sensors or machines. Formally, $S = \{s_1, \ldots, s_t, \ldots, s_{|S|}\}$, where $s_t \in \mathbb{R}^N$ comprises N numerical values at time t. The length of the time series is denoted as $|S|$ with $S \in \mathbb{R}^{|S| \times N}$. In the special case where $N = 1$, S is a univariate time series. For the given S, we can generate its D-dimensional representation by using the parameterized model $\mathcal{G}(\cdot, \Theta)$, where Θ denotes parameters.

2) Problem Definition: Given a trained model $\mathcal{G}(\cdot, \Theta)$ and a time series $S \in \mathbb{R}^{|S| \times N}$, the objective in time series anomaly detection is to determine whether observations in the series exhibit anomalies. This determination is made through the D-dimensional representation and the labels $Y = \{y_1, \ldots, y_t, \ldots, y_{|S|}\} \in \mathbb{R}^{|S|}$, where $y_t \in \{0, 1\}$ and $y_t = 1$ indicates that the t-th observation is an anomaly.

3) System Overview: Figure 2 depicts the architecture of our TFMAE, which consists of the following components:

- **Temporal-Frequency Masks.** Given an univariate or a multivariate time series, TFMAE adopts a window-based temporal masking strategy and an amplitude-based frequency masking strategy to selectively eliminate potential abnormal observations and patterns, which can avoid abnormal bias and derive two different views of the input time series. After the temporal masking, the masked observations are replaced with learnable parameters, and the unmasked observations are projected to the high-dimensional space, respectively. After the frequency masking, the masked patterns are replaced with learnable parameters and then the whole time series is projected to the high-dimensional space.
- **Transformer-based Autoencoders.** To enhance the recovery of masked patterns and observations, TFMAE incorporates the widely adopted sequential modeling framework, Transformer, for effective temporal information learning. For the frequency masking-based representations, a decoder-only manner is employed for recovering masked patterns due to the mix of frequencies. For the temporal masking-based representations, unmasked parts are initially passed through the encoder to learn normal patterns. Subsequently, all observations are fed into the decoder, facilitating the recovery of masked observations.

TABLE I: Summary of notations.

Notations	Explanations
S, $\|S\|$	input time series and its length
Y, $\hat{Y}$	real and predicted anomaly label
Θ, θ	learnable parameters and filter
V, μ	coefficient of variation and average value
$r^{(T)}$, $r^{(F)}$, r	ratio of temporal masking, frequency masking, and calculating threshold
$idx^{(T)}$, $idx^{(F)}$	masked indices of observations and frequencies
$I^{(T)}$, $I^{(F)}$	number of masked observations and frequencies
X, $\tilde{X}$, A	frequency representation, updated frequency representation, and amplitude
e, j, ω	Euler's number, imaginary unit, and angular frequency
U, P, F	unmasked temporal representation, masked temporal representation, and frequency representation
c	positional encoding
$W^{(F)}$, $b^{(F)}$	learnable parameters and bias in frequency projection
$W^{(T)}$, $b^{(T)}$	learnable parameters and bias in temporal projection
W, L, D	sub-sequence length, layers, and latent dimensions

- **Model Training.** After obtaining learned temporal and frequency masking-based representations, the contrastive objective function is used to minimize the Kullback–Leibler divergence (KLD) of them. The gradient of temporal masking-based representations is halted in the contrastive function to ensure the alignment of frequency masking-based representations. Subsequently, the adversarial training is applied to maximize the KLD between temporal-frequency masking-based representations, with the gradient of frequency masking-based representations being stopped. The adversarial training acts as a safeguard against discrepancy over-fitting.

The details of the above components are shown in Section IV. Moreover, notations utilized in this paper are summarized in Table I.

IV. Methodology

We proceed to provide specifics on each component in TFMAE and the anomaly detection, and then analyze the complexity of TFMAE.

A. Temporal-Frequency Masks

1) Window-based Temporal Masking: Currently, the reconstruction paradigm is popular for time series anomaly detection. However, reconstruction-based methods face the risk of learning misleading information from anomalies when the input time series is not pristine during the training phase, which may lead to recovering anomalies well and normal observations badly. The natural solution to reduce abnormal bias is replacing anomalies with normal patterns. Inspired by the success of masked autoencoder (MAE) [44] in diverse fields and its inherent masking-reconstruction property, we design a novel window-based temporal masking strategy to substitute the random masking strategy in MAE, which can mask potential abnormal observations and recover them with normal information. The details of the window-based temporal masking strategy are shown below.

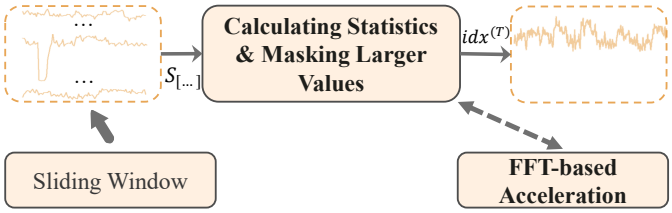


Fig. 3: Overview of window-based temporal masking.

Figure 3 illustrates the pipeline of this masking strategy, which initially utilizes a sliding window to extract sub-sequences, and then employs local statistics as the metric to mask. The reason for sliding a window on time series to extract sub-sequences of each observation is that statistics calculated based on windows are local temporal enriched and more robust to the distribution shift compared with values of observations [51]. Different from directly using the average value of sub-sequences, the coefficient of variation is adopted in our strategy, which can reflect the relative degree of fluctuation of the local sub-sequence [52]. Specifically, given an observation s_t^n of feature n of the input time series S, its sub-sequence $s_{[t-W:t]}^n$ is sampled through the sliding window, where W denotes the length of the sliding window. The coefficient of variation of the t-th sub-sequence is calculated as follows:

$$v_t = \sum_{n=1}^{N} \frac{\frac{1}{W-1}\sum_{k=t-W}^{t}(s_k^n - \mu_t^n)^2}{\mu_t^n} \tag{1}$$

where v_t indicates the summation of coefficient of variation of N features at the t-th sub-sequence, μ_t^n denotes the average value of the n-th feature at t-th sub-sequence, and $V \in \mathbb{R}^{|S|}$ contains the coefficient of variation of all observations. The coefficient of variation will be increased when the window is sliding on fluctuations. A larger coefficient of variation indicates that data are more dispersed, *i.e.*, the local sub-sequence is more abnormal. Therefore, we select a subset of $r^{(T)}\%$ observations with a large coefficient of variation as candidate anomalies to reduce abnormal observation bias. Moreover, our masking strategy is not affected by changes in the scale of the data because the coefficient of variation is normalized by its average value. Mathematically, indices of the masked observations can be formulated as follows:

$$idx^{(T)} = TopIndex(V, I^{(T)}) \tag{2}$$

where $I^{(T)} = \lfloor r^{(T)}\% \times |S| \rfloor$ and $TopIndex(\cdot, I^{(T)})$ returns indices with top $I^{(T)}$ values in the first input, and thus $idx^{(T)} \in \mathbb{R}^{I^{(T)}}$ is a subset of temporal indices. After getting the indices of masked observations, we utilize the learnable parameter-based vector $m^{(T)} \in \mathbb{R}^D$ as masked representations and a linear projection to transform unmasked observations into latent space for better modeling. The temporal unmasked

input $U \in \mathbb{R}^{(|S|-I^{(T)})\times D}$ and the temporal masked input $P \in \mathbb{R}^{I^{(T)}\times D}$ are computed as follows:

$$\begin{aligned} u_t =& W^{(T)} s_t + b^{(T)}, \quad where \quad t \notin idx^{(T)} \\ p_t =& m^{(T)}, \quad where \quad t \in idx^{(T)} \end{aligned} \tag{3}$$

where $W^{(T)} \in \mathbb{R}^{N\times D}$ and $b^{(T)} \in \mathbb{R}^{D}$ are learnable parameters of the linear projection.

FFT-based Acceleration. Considering that efficiency stands as a pivotal consideration in anomaly detection, our window-based temporal masking strategy introduces two unavoidable loops (the inner one for calculating statistics and the outer one for sliding the window), significantly amplifying computational time. Recognizing the critical role of efficiency, we explore an alternative derivation for the coefficient of variation, *i.e.*, one that relies on the expected and average values. This alternative form can be simply formulated as follows:

$$v_t = \sum_{n=1}^{N} \frac{\mu_t^{(2)^n} + \mu_t^{n^2}}{\mu_t^n} \tag{4}$$

where $\mu^{(2)}$ denotes the average value of the square of subsequence. Benefit from the average values of each window can be obtained by sliding a filter $\theta \in \mathbb{1}^W$ on S (*i.e.*, employing a convolution with the ones kernel on time series). The acceleration of calculating the coefficient of variation is achieved through Fast Fourier Transforms (FFT) according to the Wiener–Khinchin theorem [53]:

$$\begin{aligned} V &= \sum_{n=1}^{N} \bar{v}^n, \quad where \\ \bar{v}^n &= \frac{\mathcal{F}^{-1}(\mathcal{F}(s^{n^2}) \odot \mathcal{F}(\theta)) + (\mathcal{F}^{-1}(\mathcal{F}(s^n) \odot \mathcal{F}(\theta)))^2}{\mathcal{F}^{-1}(\mathcal{F}(s^n) \odot \mathcal{F}(\theta)) \cdot W} \end{aligned} \tag{5}$$

where $\bar{v}^n \in \mathbb{R}^{|S|}$ denotes the coefficient of variation of the n-th feature, and $\sum$ in Eq. (5) is the element-wise addition. Next, $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ indicate the FFT and its inverse operation. Thus, two loops are replaced by the FFT operation, which can be parallel executed with $O(|S|log(|S|))$ complexity.

2) Amplitude-based Frequency Masking: While our window-based temporal masking effectively filters potential abnormal observations, various pattern anomalies persist in time series, including seasonal and trend anomalies. To address this, we introduce a novel frequency masking strategy applied directly to the input time series, aiming to alleviate abnormal pattern bias. Notably, we opt to perform frequency masking on the original time series rather than the time series post-temporal masking. The design of this dual channel allows us to retain abnormal patterns after temporal masking and abnormal observations after frequency masking. This deliberate choice is rooted in contrastive learning needs two distinct representations to detect anomalies.

Figure 4 illustrates the process of frequency masking. Initially, to facilitate the masking of potential abnormal patterns, time series is transformed into frequency-based representations. The frequency domain is chosen for its heightened sensitivity to pattern changes and its ease in updating patterns [13].

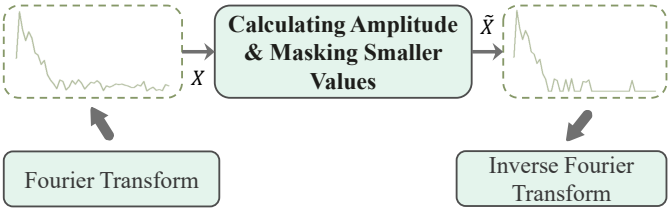


Fig. 4: Overview of amplitude-based frequency masking.

Given the input time series S, its frequency representation $X \in \mathbb{C}^{|S|\times N}$ is obtained by using the Discrete Fourier Transform (DFT) on each feature. Mathematically, the frequency spectrum x_i^n of feature n with angular frequency $\omega_i^n = \frac{2\pi i}{|S|}$ is formulated as follows:

$$x_i^n = \sum_{t=1}^{|S|} (s_t^n e^{-j\omega_i^n t}) \tag{6}$$

where e and j are Euler's number and the imaginary unit, respectively. Previous frequency-based time series modeling methods often discard high frequency components (angular frequency with large i) under the assumption that high frequencies signify noises. However, in this paper, we argue that frequency alone is insufficient as a criterion for judging abnormal patterns. Firstly, certain high frequency operations, *e.g.*, writing logs in server clusters, should be retained. Second, trend and shaplet anomalies may manifest as low frequency components still exist in time series. To address these considerations, we utilize the amplitude to replace frequency as our masking criterion, which allows for a more comprehensive evaluation of frequency importance, considering factors such as existence duration and temporal magnitude [29], like the masked curve in Fig. 4. The amplitude $A \in \mathbb{C}^{|S|\times N}$ of input times series can be calculated as follows:

$$a_i^n = \sqrt{\Re^2[x_i^n] + \Im^2[x_i^n]} \tag{7}$$

where $\Re[x_i^n]$ and $\Im[x_i^n]$ denote real and imaginary parts of the frequency, respectively. Finally, we select a subset of $r^{(F)}\%$ frequencies with small amplitude as masked patterns due to their lower magnitude and shorter existence. Mathematically, indices of these selected frequencies can be formulated as:

$$idx^{(F)^n} = TopIndex(-a^n, I^{(F)}) \tag{8}$$

where $I^{(F)} = \lfloor r^{(F)}\% \times |S| \rfloor$ and $idx^{(F)} \in \mathbb{R}^{I^{(F)}\times N}$.

In contrast to the temporal domain, where masked and unmasked representations can be separated, frequencies are mixed after reverting to the temporal domain. Therefore, the conventional MAE paradigm, which typically models unmasked and all observations sequentially, encounters challenges in the frequency domain. To address this issue, we directly employ a learnable vector $m^{(F)} \in \mathbb{C}^N$ to substitute masked frequencies before converting to the temporal domain. Thus, the replaced representation can be formed as follows:

$$\tilde{x}_i^n = \begin{cases} m^{(F)^n} & [n,i] \in idx^{(F)} \\ x_i^n & [n,i] \notin idx^{(F)} \end{cases} \tag{9}$$

where $\tilde{X} \in \mathbb{C}^{|S| \times N}$ encapsulates information about learnable masked and original unmasked frequencies. To convert the frequency representation into the original temporal domain, the inverse Discrete Fourier Transform (IDFT) is used for $\tilde{X}$. Besides, a linear projection is also utilized to transform the frequency masking-based time series into latent space for better modeling. Mathematically, the representation $F \in \mathbb{R}^{|S| \times D}$ after frequency masking is presented below:

$$f_t = W^{(F)} \sum_{i=1}^{|S|} \left(\frac{\tilde{x}_i e^{j \frac{2\pi}{|S|} it}}{|S|}\right) + b^{(F)} \tag{10}$$

where $W^{(F)} \in \mathbb{R}^{N \times D}$, $b^{(F)} \in \mathbb{R}^{D}$ are learnable parameters of the fully-connected network.

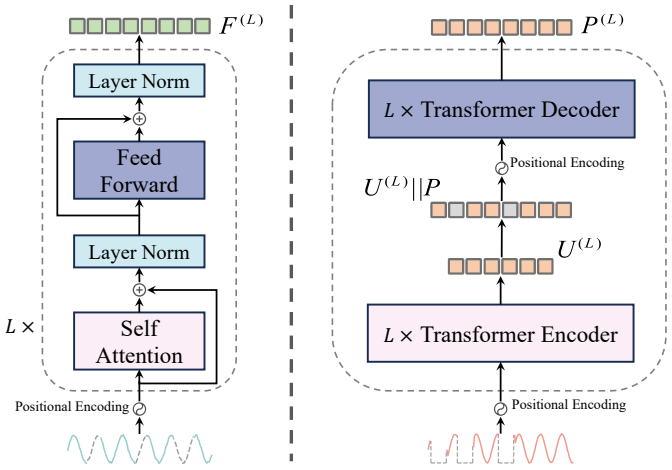


Fig. 5: Left: Transformer-based decoder-only autoencoder for recovering masked frequencies, where $\oplus$ indicates element-wise addition. Right: Transformer-based autoencoder for recovering masked observations, where $||$ denotes inserting masked representations into learned unmasked representations.

B. Transformer-based Autoencoders

1) Frequency Masking-based Autoencoder: In this section, we employ the vanilla Transformer-based decoder to recover masked frequencies, drawing inspiration from the Transformer's efficacy in capturing temporal dependencies [54], as demonstrated by Informer [55], PatchTST [56], AnoTran [42] and TranAD [8] models. The decision to adopt a decoder-only architecture in the frequency masking-based autoencoder arises from the mixing of masked and unmasked frequencies after IDFT. Furthermore, as illustrated in the left part of Figure 5, the vanilla Transformer includes a positional encoding module and the subsequent L-layer self-attention module.

Positional Encoding. Following the paradigm of the vanilla Transformer, we incorporate absolute order information of observations into the representation F using a sinusoidal function to expedite convergence. Specifically, the value of i-th dimension of the t-th representation is denoted as $f_t^{(0)^i}$, obtained by adding the original representation f_t^i and its corresponding positional encoding c_t^i. The details of positional encoding are formulated as follows:

$$\begin{aligned} f_t^{(0)^i} &= f_t^i + c_t^i, \quad where \\ c_t^i &= \begin{cases} sin(t/10000^{i/D}) & i \in Even \\ cos(t/10000^{(i-1)/D} & i \in Odd \end{cases} \end{aligned} \tag{11}$$

where $F^{(0)} \in \mathbb{R}^{|S| \times D}$ denotes the encoding-decorated output.

Self-Attention. As depicted in Figure 5, each attention layer comprises a dot-product self-attention mechanism and a feed-forward network. In the l-th layer, the input $F^{(l-1)}$ is initially projected into a query $Q^{(l)} \in \mathbb{R}^{|S| \times D}$, key $K^{(l)} \in \mathbb{R}^{|S| \times D}$, and value $V^{(l)} \in \mathbb{R}^{|S| \times D}$ through three linear projections. Subsequently, attention weights, *i.e.*, correlations between each observation, are calculated through the multiplication of the query $Q^{(l)}$ and key $K^{(l)}$. Finally, the value $V^{(l)}$ is temporal enhanced based on their attention weights. Mathematically, the representation of self-attention for the t-th observation in the l-th layer can be expressed as follows:

$$\tilde{f}_t^{(l)} = \frac{\sum_{i=1}^{|S|} \left(e^{\frac{q_t^{(l)} k_i^{(l)}}{\sqrt{D}}} v_i^{(l)}\right)}{\sum_{i=1}^{|S|} e^{\frac{q_t^{(l)} k_i^{(l)}}{\sqrt{D}}}} \tag{12}$$

where $\tilde{F}^{(l)} \in \mathbb{R}^{|S| \times D}$ indicates the representation of l-th self-attention. To ensure stable training and improve the generalization of self-attention, we incorporate a residual connection and a feed-forward network. The formulation is presented below:

$$\begin{aligned} \bar{F}^{(l)} &= LN(F^{(l-1)} + \tilde{F}^{(l)}) \\ F^{(l)} &= LN(\bar{F}^{(l)} + MLP(\bar{F}^{(l)})) \end{aligned} \tag{13}$$

where $LN(\cdot)$ and $MLP(\cdot)$ indicate the layer normalization and multilayer perceptron, respectively. $\bar{F}^{(l)} \in \mathbb{R}^{|S| \times D}$ and $F^{(l)} \in \mathbb{R}^{|S| \times D}$ are representations after the residual connection and feed-forward network.

2) Temporal Masking-based Autoencoder: In this section, we elaborate on the process of modeling the representation after temporal masking. As illustrated in the right part of Figure 5, the unmasked observations are initially processed by a Transformer-based encoder to learn normal temporal patterns. Subsequently, the entire set of observations is input into the Transformer-based decoder to recover masked observations through the learned normal patterns.

Encoder. Differing from the decoder-only design in the frequency domain, the unmasked temporal input U is initially processed by the encoder to acquire normal temporal patterns. Concretely, U is initially decorated with the sinusoidal positional encoding and then modeled through the L-layer encoder, facilitating the propagation of temporal information from the context to the current state. After the encoder, the learned representation $U^{(L)} \in \mathbb{R}^{(|S|-I^{(T)}) \times D}$ of unmasked observations is obtained.

Decoder. Similar to the frequency view, the full set of observations (*i.e.*, encoded unmasked and learnable masked observations) are used in the Transformer-based decoder. Prior to their input into the decoder, the sinusoidal positional

encoding is added to the masked observations based on their locations in the original time series, as the learnable vector-based masked representation lacks location information. Subsequently, the decorated representation of masked observations is inserted into the encoded representation of unmasked observations. Both are then fed into the decoder to recover masked observations through normal patterns learned in the encoder. Finally, following the L-layer decoder, the representation $P^{(L)} \in \mathbb{R}^{|S| \times D}$ for all observations is obtained.

C. Model Training

In this section, we formulate the adversarial training enhanced contrastive objective function, which uses representations from two distinct perspectives, namely the temporal and frequency view, to train our TFMAE.

1) Contrastive Objective Function: While the reconstruction error has been a staple loss function in anomaly detection tasks, the persistent challenge of time series distribution shift hampers its efficacy. Time series distribution shift denotes alterations in the distribution of test data compared to the training data, posing difficulty in reconstructing unseen data from previously learned patterns. In other words, observations with high anomaly scores may not necessarily be abnormal but rather unseen. To mitigate the negative influence of time series distribution shift, we propose a novel contrastive objective function to train our TFMAE, which replaces the minimization of the discrepancy between original and reconstructed time series with the minimization of the discrepancy between temporal and frequency masking-based representations. This is because the temporal-frequency masking-based representations of the same time series remain consistent according to their temporal-frequency consistency [57], even when the input has not been seen. Moreover, only positive samples are employed in our contrastive objective function to minimize the discrepancy of representations from temporal-frequency masking views, in contrast to classic contrastive learning that utilizes positive-negative pairs [58]. During the training phase, anomalies will be progressively highlighted because the discrepancy between normal-recovered and original-abnormal representations of anomalies in different views is more resistant to reduction compared to consistent normal observations. Therefore, observations with higher discrepancies will be detected as anomalies in the inference stage. Mathematically, the contrastive objective function is calculated as follows:

$$\mathcal{L} = D_{KL}(P^{(L)}, F^{(L)}) + D_{KL}(F^{(L)}, P^{(L)}) \tag{14}$$

where $D_{KL}(\cdot, \cdot)$ indicates the Kullback–Leibler divergence, which calculates the distance between two representations and is used to reflect the discrepancy between temporal-frequency masking-based representations.

2) Adversarial Training: As depicted in Table IV, the utilization of contrastive learning alone does not yield optimal results, primarily due to the emergence of the over-fitting phenomenon, *e.g.*, the representation of abnormal frequencies in the temporal view tends to be proximate to its corresponding normal-recovered representation in the frequency view after the contrastive training. This phenomenon contradicts our intended objective of minimizing the discrepancy between normal observations while maximizing the discrepancy between anomalies. To overcome this issue, we introduce an adversarial mechanism into our objective function, which enhances the robustness and generalization in most deep learning tasks. In TFMAE, adversarial training aims to increase the similarity between normal representations while maintaining a significant discrepancy between abnormal representations and their corresponding normal views. Mathematically, our adversarial contrastive function is expressed below:

$$\mathcal{L} = \min_{F^{(L)}} \max_{P^{(L)}} (D_{KL}(P^{(L)}, F^{(L)}) + D_{KL}(F^{(L)}, P^{(L)})) \tag{15}$$

where max and min denote increasing and decreasing the discrepancy between temporal-frequency masking-based representations. During the training phase, the gap between the discrepancy of abnormal representations and the discrepancy of normal representations gradually increases because the discrepancy of abnormal representations is more difficult to reduce and easier to increase. Moreover, the gradient of temporal and frequency masking-based representations is respectively stopped in the minimizing and maximizing stage because temporal masking-based representations reserve more original information and are thus more suitable for acting as labels in the adversarial training. As illustrated in Figure 8, the adversarial contrastive objective function can output large discrepancies in anomalies.

D. Anomaly Detection

In our contrastive design, the magnitude of discrepancy between temporal-frequency masking-based representations directly corresponds to the likelihood of an anomaly. Consequently, we employ the contrastive discrepancy as the anomaly score during the inference stage. The score for the input observation $s_t \in \mathbb{R}^N$ can be calculated as:

$$Score(s_t) = D_{KL}(p_t^{(L)} || f_t^{(L)}) + D_{KL}(f_t^{(L)} || p_t^{(L)}) \tag{16}$$

Ultimately, observations are assessed based on the anomaly score and the pre-determined threshold δ, *i.e.*, an anomaly is detected when the score surpasses the threshold. The detected labels $\hat{Y} \in \mathbb{R}^{|S|}$ are shown below:

$$\hat{y}_t = \begin{cases} 1 & Score(s_t) \geq \delta \\ 0 & Score(s_t) < \delta \end{cases} \tag{17}$$

E. Complexity Analysis

In this section, we discuss the complexity of TFMAE, which is composed of window-based temporal masking, amplitude-based frequency masking, and Transformer-based autoencoders. The time complexity of the window-based temporal masking is optimized from $O(N|S|W)$ to $O(N|S|log(|S|))$ by the FFT operation, and the DFT algorithm in the amplitude-based frequency masking is also can be fasted by the FFT operation with the complexity $O(N|S|log(|S|))$. Therefore, the time complexity of our TFMAE is dominated by $O(LD|S|^2)$ due to the quadratic self-attention in the Transformer, which is comparable to or even surpasses state-of-the-art methods.

TABLE II: Dataset statistics. The term "AR" indicates anomaly ratio.

Datasets	Sources	Type	Dimension	#Training	#Validation	#Inference	AR(%)
MSL	NASA Space	Multivariate	55	46653	11664	73729	10.5
PSM	eBay Server	Multivariate	25	105984	26497	87841	27.8
SMD	Internet Server	Multivariate	38	566724	141681	708420	4.2
SWaT	Water Treatment	Multivariate	51	396000	99000	449919	12.1
SMAP	NASA Space	Multivariate	25	108146	27037	427617	12.8
NIPS-TS-Global	Synthetic	Univariate	1	40000	10000	50000	5.0
NIPS-TS-Seasonal	Synthetic	Univariate	1	40000	10000	50000	5.0

V. Experiments

To validate the effectiveness and efficiency of our TFMAE, we conduct comprehensive experiments on seven datasets to answer the following six research questions:

- **RQ1:** Does our TFMAE demonstrate superior performance compared to baselines across diverse datasets?
- **RQ2:** How do components within TFMAE, *e.g.*, autoencoders, impact the performance of anomaly detection?
- **RQ3:** How effective are the temporal-frequency masking strategies designed in TFMAE?
- **RQ4:** How does adjusting various hyperparameters, particularly the temporal-frequency masking ratio, impact the performance of TFMAE?
- **RQ5:** Does TFMAE output reasonable anomaly scores?
- **RQ6:** How efficient is TFMAE in anomaly detection?

A. Experimental Setup

1) Benchmark Datasets: In this paper, we selected seven widely used time series anomaly detection datasets to assess the effectiveness of TFMAE. These datasets include five real-world and two synthetic datasets: **MSL** (Mars Science Laboratory rover) and **SMAP** (Soil Moisture Active Passive satellite) are both released by NASA [59], where the time series and anomaly alarms in them are recorded by the Incident Surprise Anomaly reports of spacecraft monitoring systems. **PSM** (Pooled Server Metrics) is released by eBay [60], containing time series from multiple server nodes at eBay. **SMD** (Server Machine Dataset) [6] is a larger dataset compared to the aforementioned ones, encompassing a five-week-long time series of internet server nodes. **SWaT** (Secure Water Treatment) records data from the critical infrastructure system under continuous operations [61]. **NIPS-TS-Global** and **NIPS-TS-Seasonal** are synthetic datasets generated by well-designed rules [62], representing global observation anomalies and seasonal anomalies, respectively. Detailed information about these datasets is presented in Table II.

2) Metrics: To compare time series anomaly detection performance, we employ three widely used evaluation metrics: precision (P), recall (R), and F1-score (F1). Consistent with literature settings, we apply the point adjustment strategy to obtain detection results, where continuous anomalies are identified if a single observation in the segment is detected.

3) Baselines: We extensively compare our proposed TFMAE against the following six categories-based 14 baselines:

- Density-based methods: **LOF** [20] computes the local density deviation and observations with lower density are anomalies. **DAGMM** [22] further utilizes the Gaussian Mixture Model to compute the density of data.
- Tree-based methods: **IForest** [63] performs anomaly detection using isolation trees with linear time complexity.
- Clustering-based methods: **DSVDD** [26] uses deep networks to derive representation and detect anomalies through distances to clusters. **THOC** [27] extracts hierarchical information through the dilated recurrent neural network and trains the model using one-class objective on clustered hyperspheres for anomaly detection.
- Reconstruction-based methods: **OmniAno** [6] utilizes the normalizing flow enhanced LSTM to reconstruct time series for detecting anomalies. **TimesNet** [7] transforms 1D time series into multiple 2D tensors and then uses convolution backbones to reconstruct time series. **GPT4TS** [64] combines large language models with the time series anomaly detection task.
- Adversarial reconstruction-based methods: **USAD** [36] proposes a two decoders-based autoencoder and uses the adversarial manner for fast training. **BeatGAN** [35] utilizes adversarial enhanced convolution models to detect time series anomalies. **DAEMON** [37] utilizes two discriminators to adversarially train an autoencoder and then derive the robust reconstructed time series. **TranAD** [8] utilizes the Transformer framework to encode time series and then introduces adversarial training into the two decoders-based model to enhance robustness.
- Contrastive-based methods: **AnoTran** [42] encodes time series by the prior association and the series association, and then uses the discrepancy between two representations to distinguish anomalies. **DCdetector** [43] utilizes positive contrastive learning on the representations of time series with different patch sizes to detect anomalies.

We utilize the configuration of best performance in baselines to run their official codes on the same machine that running our model.

4) Hyper-Parameter Settings: TFMAE is trained by the Adam optimizer [65] with an initial learning rate of 0.0001, an epoch of 1, and a batch size of 64. In autoencoders, we set the number of transformer layers as 3 with 128 hidden feature dimensions. During the window-based temporal masking process, the length of the sliding window is set to 10 to calculate local statistical features. Moreover, we set different temporal-frequency masking ratios for different datasets due to their characteristics. The detailed settings can be seen in Figure 6. Besides, the threshold δ is pre-determined by detecting $r\%$

TABLE III: Main results on five time series anomaly detection datasets. The precision (P), recall (R), and F1-score (F1) are in %. The term "Average" refers to the mean value across these five datasets. Grey: Best result, **Bold**: Second best result.

Dataset		SWaT			PSM			SMD		
Model	Venue	P	R	F1	P	R	F1	P	R	F1
LOF	MOD-00	15.37	94.15	26.42	68.77	93.86	79.38	39.52	10.57	15.72
IForest	ICDM-08	80.31	81.90	81.09	95.74	85.83	90.51	47.18	72.66	57.21
DSVDD	ICML-18	91.26	80.49	85.54	72.88	88.99	80.13	56.06	72.56	63.25
DAGMM	ICLR-18	26.19	87.41	40.31	92.07	90.09	91.07	65.39	86.17	74.36
BeatGAN	IJCAI-19	92.46	79.06	85.23	96.58	90.16	93.26	77.11	77.60	77.36
OmniAno	KDD-19	71.65	83.76	77.23	95.71	90.09	92.82	72.58	83.67	77.73
USAD	KDD-20	57.76	83.76	68.37	97.63	98.08	97.86	**90.96**	**90.04**	**90.99**
THOC	NeurIPS-20	83.10	83.54	83.32	78.33	88.60	83.15	69.08	77.30	72.96
DAEMON	ICDE-21	90.86	78.50	84.23	97.57	86.22	91.55	78.08	77.91	77.99
AnoTran	ICLR-22	83.85	**100.0**	91.22	96.74	97.73	97.23	90.90	81.20	85.78
TranAD	VLDB-22	**94.23**	94.36	94.29	97.44	98.19	97.92	74.30	81.65	77.80
TimesNet	ICLR-23	81.83	97.32	88.90	**97.64**	**98.22**	**97.93**	77.67	81.58	79.58
DCdetector	KDD-23	93.25	**100.0**	**96.51**	97.40	98.16	97.78	85.37	82.85	84.09
GPT4TS	NeurIPS-23	90.13	95.60	92.79	97.39	94.13	95.73	89.60	81.13	85.16
TFMAE	-	96.77	100.0	98.36	98.06	99.06	98.56	91.41	91.07	91.24

Dataset		MSL			SMAP			Average		
Model	Venue	P	R	F1	P	R	F1	P	R	F1
LOF	MOD-00	63.46	90.05	74.45	79.62	85.26	82.34	53.35	74.78	55.66
IForest	ICDM-08	76.95	90.77	83.29	88.60	56.36	68.90	77.76	77.50	76.20
DSVDD	ICML-18	87.93	86.93	87.43	87.34	58.78	70.27	79.09	77.55	77.32
DAGMM	ICLR-18	90.05	86.93	88.46	87.58	56.31	68.55	72.26	81.38	72.55
BeatGAN	IJCAI-19	91.44	85.42	88.33	93.91	55.41	69.70	90.30	77.53	82.78
OmniAno	KDD-19	90.82	86.47	88.59	92.34	56.18	69.85	84.62	80.03	81.24
USAD	KDD-20	91.24	94.73	92.96	92.02	65.78	76.71	85.92	86.48	85.38
THOC	NeurIPS-20	90.30	75.99	82.53	90.08	55.50	68.69	82.18	76.19	78.13
DAEMON	ICDE-21	91.47	87.37	89.37	84.95	56.49	67.86	88.59	77.30	82.20
AnoTran	ICLR-22	91.95	**96.50**	**94.17**	**94.01**	86.72	90.22	91.49	92.43	91.72
TranAD	VLDB-22	90.72	94.73	92.68	93.12	71.33	80.78	89.96	88.05	88.69
TimesNet	ICLR-23	87.32	85.42	86.36	86.47	65.48	74.53	86.19	85.60	85.46
DCdetector	KDD-23	**92.08**	94.44	93.24	93.18	**98.84**	**95.93**	**92.26**	**94.86**	**93.51**
GPT4TS	NeurIPS-23	82.08	85.45	83.73	88.78	64.72	74.86	89.60	84.21	86.45
TFMAE	-	92.83	97.59	95.15	94.71	99.19	96.90	94.76	97.38	96.04

data as anomalies. Specifically, we set $r = 0.9\%$ for MSL and PSM, 0.75% for SMAP, 0.45% for SMD, and 0.3% for SWaT.

5) Implementation Details: The experiments of TFMAE and all baselines are conducted on a machine with one Intel(R) Xeon(R) Gold 6230R CPU @ 2.10GHz and a NVIDIA GeForce RTX 3090 GPU card with PyTorch 1.12.1. The source code of TFMAE is available at: https://github.com/LMissher/TFMAE. For a fair comparison, thresholds of all methods are calculated through the validation set.

B. Performance Comparison (RQ1)

Table III showcases the performance of time series anomaly detection across all methods on five datasets, measured in terms of precision, recall, and F1-score. To ensure a fair comparison, we adhere to the original configurations of baselines, with the only adjustment being a fixed input length of 100, in line with [42]. Consequently, the reported performance of baselines may exhibit slight variations compared to the results in the original literature. The salient findings are as follows:

Advantages of Temporal Learning. Among the baseline models considered, deep learning-based approaches, exemplified by DAGMM, demonstrate significant advantages over its classical counterpart LOF. Moreover, the performance of clustering-based DSVDD and THOC suggests that leveraging advanced temporal modeling techniques leads to more accurate results. This phenomenon underscores the effectiveness of learning temporal dependencies.

Advantages of Adversarial Training. In contrast to reconstruction-based methodologies, adversarial reconstruction-based approaches (*e.g.*, TranAD and USAD) exhibit distinct advantages, attributable to the efficacy of adversarial training, *i.e.*, the ability to circumvent the reconstruction of abnormal patterns.

Advantages of Contrastive Learning. As illustrated in Table III, AnoTran and DCdetector leverage contrastive learning on representations obtained through a dual channel model, showcasing notable superiority, particularly on the SWaT dataset when compared to reconstruction-based methods. This phenomenon is attributed to the distribution shift, wherein the multi-view representations of the same time series in the test set exhibit proximity, while the reconstructed time series may diverge significantly from the unseen test data.

Advantages of Frequency Learning. The results of TimesNet demonstrate that using features in the frequency domain can significantly improve the performance of detecting anomalies in time series compared to temporal features alone methods, *e.g.*, GPT4TS. This is because anomalies are not only produced in single times but also patterns and the frequency domain is more sensitive to pattern anomalies.

Consistent Performance Superiority. Drawing on the aforementioned techniques, we introduce TFMAE, a novel ap-

TABLE IV: Ablation results of TFMAE. The precision (P), recall (R), and F1-score (F1) are in %. **Bold**: Ours.

Dataset	SWaT			PSM			SMD			MSL			SMAP		
Metric	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
w/o $\mathcal{L}_{adv}$	49.93	92.93	64.96	99.41	90.80	94.91	88.63	79.64	83.89	93.03	86.33	89.55	91.17	54.83	68.48
w/ $\mathcal{L}_{radv}$	95.73	100.0	97.82	97.44	98.21	97.82	90.23	79.86	84.73	92.07	88.82	90.42	95.22	98.43	96.80
w/o Fre	90.36	96.22	93.20	97.59	98.47	98.03	91.96	89.36	90.65	93.06	93.18	93.12	91.69	66.02	76.76
w/o FD	37.40	95.44	53.74	98.72	90.77	94.58	89.03	82.51	85.65	92.74	82.28	87.20	91.01	65.30	76.04
w/o Tem	87.91	96.22	91.88	98.51	95.86	97.17	93.77	87.13	90.33	92.60	87.21	89.83	92.15	65.44	76.53
w/o TE	94.35	98.52	96.39	97.58	99.01	98.29	90.37	88.44	89.39	92.64	94.31	93.47	94.64	98.66	96.61
w/o TD	35.21	90.43	50.69	99.45	69.28	81.67	74.56	51.01	60.58	90.43	79.86	84.82	93.31	88.75	90.97
TFMAE	**96.77**	**100.0**	**98.36**	**98.06**	**99.06**	**98.56**	**91.41**	**91.07**	**91.24**	**92.83**	**97.59**	**95.15**	**94.71**	**99.19**	**96.90**

TABLE V: Ablation results of masking strategies. The precision (P), recall (R), and F1-score (F1) are in %. **Bold**: Ours.

Dataset	SWaT			PSM			SMD			MSL			SMAP		
Metric	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
w/o MT	94.25	100.0	97.04	97.48	98.27	97.87	91.51	90.06	90.78	92.88	94.90	93.88	95.10	89.54	92.24
w/ SMT	95.79	99.10	97.42	97.65	98.45	98.05	91.48	88.01	89.71	92.49	95.40	93.92	94.24	98.26	96.21
w/ RMT	94.72	99.57	97.09	97.54	98.40	97.97	90.28	88.93	89.60	93.00	90.84	91.91	93.25	99.08	96.08
w/o MF	95.28	99.06	97.13	97.70	98.61	98.15	91.31	89.21	90.03	92.45	96.29	94.33	94.15	97.52	95.81
w/ HMF	95.88	98.88	97.35	97.62	98.87	98.25	91.26	85.01	88.09	92.67	95.20	93.92	93.53	98.31	95.86
w/ RMF	96.08	98.17	96.11	97.72	98.45	98.08	89.92	90.00	89.95	92.50	93.51	93.00	93.69	97.82	95.71
TFMAE	**96.77**	**100.0**	**98.36**	**98.06**	**99.06**	**98.56**	**91.41**	**91.07**	**91.24**	**92.83**	**97.59**	**95.15**	**94.71**	**99.19**	**96.90**

proach that initially employs temporal and frequency masked autoencoders to acquire pristine representations devoid of abnormal patterns and observations. Subsequently, it incorporates temporal-frequency masking-based contrastive objective function as the anomaly criterion and employs adversarial training to mitigate the potential adverse effects of over-fitting. Consequently, as evidenced in Table III, TFMAE consistently attains state-of-the-art performance across all datasets.

C. Model Ablation Study (RQ2)

In this section, we conduct experiments on five datasets using seven variants of TFMAE. These experiments aim to showcase the effectiveness of the model design. The details of these variants are outlined below:

- "w/o $\mathcal{L}_{adv}$": This variant excludes the adversarial objective during the training phase.
- "w/ $\mathcal{L}_{radv}$": This version involves swapping the positions of $F^{(L)}$ and $P^{(L)}$ in Equations 15.
- "w/o Fre": TFMAE eliminates the frequency view.
- "w/o FD": It no longer equips the frequency decoder.
- "w/o Tem": TFMAE removes the temporal view.
- "w/o TE": It no longer equips the temporal encoder.
- "w/o TD": It no longer equips the temporal decoder.

Based on the ablation results presented in Table IV, the following findings can be made.

Benefits Brought by Adversarial Training. The results of "w/o $\mathcal{L}_{adv}$" show a significant drop in performance, suggesting that the inclusion of the adversarial objective guides TFMAE training in a more accurate direction, *i.e.*, adversarial training may help prevent over-fitting. Moreover, the performance of "w/ $\mathcal{L}_{radv}$" further validates that temporal masking-based representations store more original information.

Effectiveness of Frequency Masked Autoencoder. The experiments of "w/o Fre" reveal that, in most scenarios, removing the frequency view leads to a significant decline in performance. This observation underscores the effectiveness of our contrastive objective function. Furthermore, the performance of "w/o FE" is lower than that of "w/o Fre," indicating that deriving correct representations specific to a view is more crucial than merely introducing this view.

Effectiveness of Temporal Masked Autoencoder. Similar to the frequency view, the temporal view is indispensable in our TFMAE as the performance of "w/o Tem". Furthermore, as evidenced by the impaired performance in "w/o TD," the decoder is crucial in the temporal view to convey normal temporal information into masked observations.

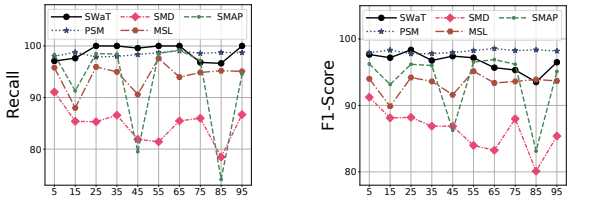
D. Investigation on Temporal-Frequency Masks (RQ3)

In this section, we seek to validate the effectiveness of our proposed temporal-frequency masking strategies. To achieve this, we design six variants of TFMAE and perform experiments on five time series anomaly detection datasets. The details of these variants are as follows:

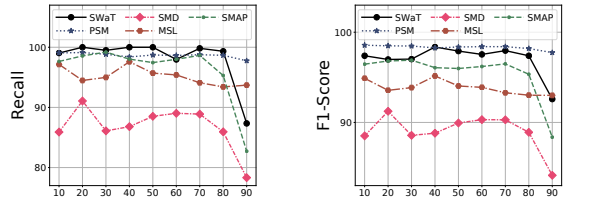
- "w/o MT": It no longer equips temporal masking.
- "w/ SMT": This version only uses the standard deviation to mask observations.
- "w/ RMT": This version randomly masks observations.
- "w/o MF": It no longer equips frequency masking.
- "w/ HMF": It masks high frequencies.
- "w/ RMF": This variant randomly masks frequencies.

According to the results presented in Table V, the following findings can be drawn.

Impact of Window-based Temporal Masking. The results of "w/ RMT" consistently indicate that our window-based temporal masking surpasses the random masking across all datasets. Additionally, the performance of "w/o MT" is comparable and even outperforms "w/ RMT," suggesting that the key factor in enhancing performance lies not in the "Masking" itself but in "Masking Anomalies." Notably, the coefficient of variation demonstrates superior performance compared to "w/ SMT" as it is less sensitive to the change of data scale.



(a) Temporal Masking Ratio (b) Temporal Masking Ratio



(c) Frequency Masking Ratio (d) Frequency Masking Ratio

Fig. 6: Hyper-parameter study of masking strategies.

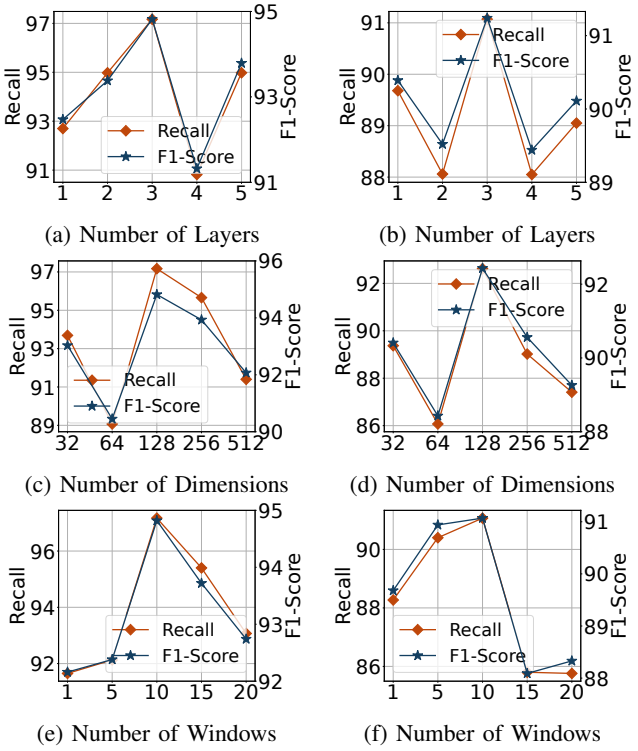
Impact of Amplitude-based Frequency Masking. Similar to temporal masking, the performance of removing frequency masking "w/o MF" is comparable and even outperforms the random masking "w/ RMF." Moreover, to further validate the effectiveness of our masking strategy, we compare our amplitude-based frequency masking to a variant that utilizes high frequency-based masking. As depicted in Table V, "w/ HMF" performs inferiorly to TFMAE in most tasks, suggesting that high frequencies do not exclusively represent anomalies, instead, short-lived temporal patterns that deviate from historical data are more likely to be anomalies.

E. Hyper-Parameter Sensitivity Analysis (RQ4)

Figure 6 and Figure 7 depict the results of varying hyper-parameters. Specifically, we explore layers of Transformer, dimensions of hidden features, the window length of the temporal masking strategy, and the temporal-frequency masking ratio from search spaces of $[1, 2, 3, 4, 5]$, $[32, 64, 128, 256, 512]$, $[1, 5, 10, 15, 20]$, 5 to 95 with an interval of 10, and 10 to 90 with an interval of 10. As depicted in Figure 7, the performance of TFMAE initially improves with increasing layers and then decreases when the number of layers exceeds three. Additionally, when the dimensions of the hidden feature are set to 128, TFMAE achieves optimal performance. This is attributed to the fact that an excessive number of hidden features may impede convergence. Furthermore, additional insights can be gleaned from figures as follows:

Window Length W**.** This hyper-parameter governs the local information considered at each observation. Notably, setting $L = 10$ yields the optimal performance, suggesting that short subsequences overlook crucial information, while long subsequences may diminish the impact of current fluctuations. Furthermore, the performance of $L = 1$, equivalent to masking with original values, indicates that absolute values do not effectively mask abnormal observations, particularly under the potential issue of time series distribution shift.

Temporal Masking Ratio $r^{(T)}$**.** This hyper-parameter determines the proportion of observations to be masked. As shown in Fig. 6, the optimal ratio for SWaT, SMD, SMAP, PSM, and MSL is 25%, 5%, 65%, 65%, and 55%. Despite achieving optimal performance with a small masking ratio, such as 5% on the SMD dataset, comparable results are also observed when increasing the temporal masking ratio to even 95% on the SWaT dataset. This is attributed to time series exhibiting significant temporal redundancy, making it relatively easy to recover from missing observations.



(a) Number of Layers (b) Number of Layers

(c) Number of Dimensions (d) Number of Dimensions

(e) Number of Windows (f) Number of Windows

Fig. 7: Left: Hyper-parameter study of TFMAE on MSL. Right: Hyper-parameter study of TFMAE on SMD.

Frequency Masking Ratio $r^{(F)}$**.** This hyper-parameter dictates the number of frequencies to be masked. As shown in Fig. 6, the optimal ratio for SWaT, SMD, SMAP, PSM, and MSL is 40%, 20%, 30%, 10%, and 40%. It is observed that a large frequency masking ratio results in inferior performance compared to a large temporal masking ratio. This discrepancy arises from the fact that a single frequency encapsulates more information than a single observation. Consequently, frequency masking-based representations may be difficult to reconstruct and should be aligned with temporal.

F. Case Study (RQ5)

Abnormal Bias. To validate the capability of TFMAE in producing accurate detection results, we conduct a case study using the NIPS-TS-Seasonal and NIPS-TS-Global datasets. As depicted in Figure 8, anomaly scores generated by TFMAE and DCdetector are consistently distinguishable, *i.e.*, scores remain small except in the presence of anomalies. Furthermore, TFMAE can identify seasonal and global observation anomalies, whereas DCdetector fails. This discrepancy suggests that the abnormal bias can mislead the model, and our masked autoencoders effectively address this issue.

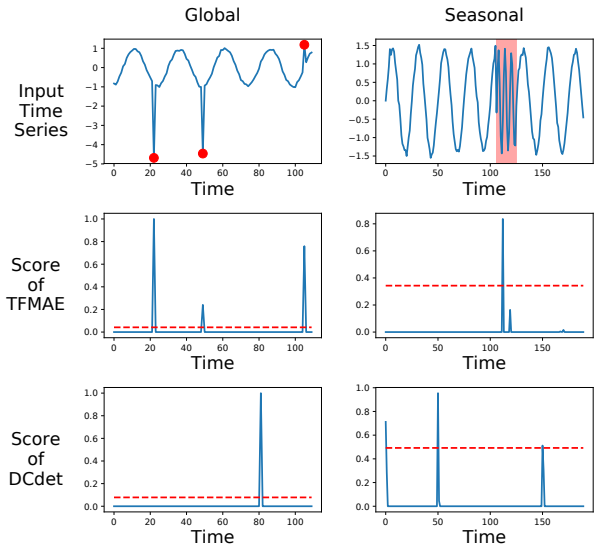


Fig. 8: Visualization of seasonal and global observation anomalies in the NIPS-TS-Seasonal and NIPS-TS-Global datasets. 'DCdet' refers to DCdetector. In the first row, red circles denote global observation anomalies, and the red box indicates a seasonal anomaly. In the second and third rows, red lines denote the threshold for detecting anomalies.

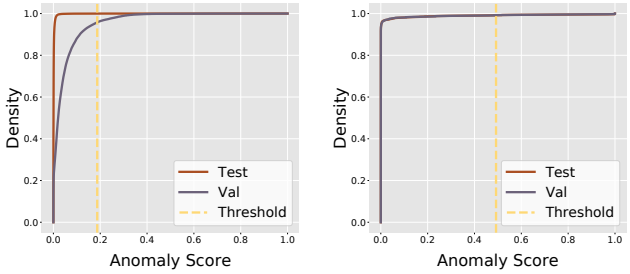


Fig. 9: Left: CDF of anomaly scores on SMAP validation and test sets for TimesNet. Right: CDF of anomaly scores on SMAP validation and test sets for TFMAE.

Time Series Distribution Shift. To ascertain the robustness of our contrastive criterion in the presence of time series distribution shifts, we performed a case study using the SMAP dataset. As illustrated in Figure 9, the cumulative scores of TimesNet on the validation and test set show a clear gap caused by shifts. However, cumulative scores generated by our TFMAE on the validation and test set are always similar. This comparison substantiates that our contrastive criterion can effectively mitigate time series distribution shifts, resulting in higher generalization in the threshold.

G. Model Efficiency Study (RQ6)

To evaluate the effectiveness and efficiency of TFMAE, we present the F1-Score, training speed, and GPU memory size comparisons among TFMAE, the large language model-based GPT4TS, and state-of-the-art baselines including TranAD, AnoTran, TimesNet, and DCdetector. Additionally, we include a variant, denoted as "w/o FFT," of TFMAE in the assessment.

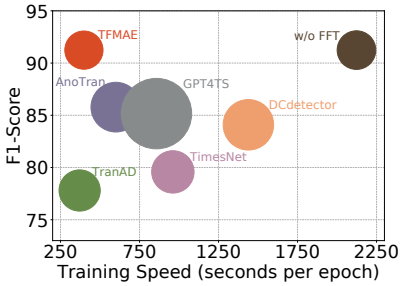


Fig. 10: Performance metrics are compared on the SMD dataset, with F1-Score represented in % on the y-axis, speed on the x-axis, and memory footprint indicated by the size of the circles for each method.

The evaluation is conducted on the SMD dataset, which is the longest and second largest dataset in our paper.

- "w/o FFT": This version omits the use of FFT to expedite the calculation of the coefficient of variation.

As depicted in Figure 10, TFMAE attains the highest F1-Score, boasts the most efficient GPU memory utilization, and ranks second in terms of training speed. While TranAD excels in speed, the larger memory footprint and lower performance compared to TFMAE suggest that TFMAE achieves a superior trade-off between speed and performance, coupled with excellent memory usage. Furthermore, the discernible decrease in training speed of "w/o FFT" underscores the effectiveness of our FFT-based acceleration.

VI. Conclusion

In this paper, we introduce a novel Temporal-Frequency Masked Autoencoder (TFMAE) for time series anomaly detection, which departs from the conventional reconstruction paradigm. TFMAE leverages the discrepancy between temporal-frequency masking-based representations to replace the traditional reconstruction error and mitigate the impact of time series distribution shifts. Additionally, TFMAE incorporates a window-based temporal masking strategy and an amplitude-based frequency masking strategy before Transformer-based autoencoders to reduce abnormal bias in time series. To prevent potential over-fitting during contrastive training, the adversarial objective function is integrated into TFMAE. Experimental results on seven benchmark datasets showcase the superior performance of TFMAE against 14 baselines. Future work will extend TFMAE to other time series tasks, such as time series prediction and classification.

ACKNOWLEDGMENTS

This work is partially supported by NSFC (No. 61972069, 61836007 and 61832017), Shenzhen Municipal Science and Technology R&D Funding Basic Research Program (JCYJ20210324133607021), and Municipal Government of Quzhou under Grant (No. 2022D037, 2023D044), and Key Laboratory of Data Intelligence and Cognitive Computing, Longhua District, Shenzhen.

REFERENCES

[1] J. Li, S. Di, Y. Shen, and L. Chen, "Fluxev: a fast and effective unsupervised framework for time-series anomaly detection," in *Proceedings of WSDM*, 2021, pp. 824–832.

[2] Y. Zhao, L. Deng, X. Chen, C. Guo, B. Yang, T. Kieu, F. Huang, T. B. Pedersen, K. Zheng, and C. S. Jensen, "A comparative study on unsupervised anomaly detection for time series: Experiments and analysis," *arXiv preprint arXiv:2209.04635*, 2022.

[3] Y. Liu, X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He, "Pick and choose: a gnn-based imbalanced learning approach for fraud detection," in *Proceedings of WWW*, 2021, pp. 3168–3177.

[4] G. Andresini, A. Appice, and D. Malerba, "Autoencoder-based deep metric learning for network intrusion detection," *Information Sciences*, vol. 569, pp. 706–727, 2021.

[5] Y. Himeur, K. Ghanem, A. Alsalemi, F. Bensaali, and A. Amira, "Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives," *Applied Energy*, vol. 287, p. 116601, 2021.

[6] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *Proceedings of SIGKDD*, 2019, pp. 2828–2837.

[7] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," in *Proceedings of ICLR*, 2023, pp. 1–23.

[8] S. Tuli, G. Casale, and N. R. Jennings, "Tranad: deep transformer networks for anomaly detection in multivariate time series data," *Proceedings of VLDB*, pp. 1201–1214, 2022.

[9] Y. Fang, Y. Qin, H. Luo, F. Zhao, B. Xu, L. Zeng, and C. Wang, "When spatio-temporal meet wavelets: Disentangled traffic forecasting via efficient spectral graph attention networks," in *Proceedings of ICDE*, 2023, pp. 517–529.

[10] Z. Lai, D. Zhang, H. Li, C. S. Jensen, H. Lu, and Y. Zhao, "Lightcts: A lightweight framework for correlated time series forecasting," *Proceedings of SIGMOD*, vol. 1, no. 2, pp. 1–26, 2023.

[11] T. Kim, J. Kim, Y. Tae, C. Park, J.-H. Choi, and J. Choo, "Reversible instance normalization for accurate time-series forecasting against distribution shift," in *Proceedings of ICLR*, 2021.

[12] J. Lv, Y. Wang, and S. Chen, "Adaptive multivariate time-series anomaly detection," *Information Processing & Management*, p. 103383, 2023.

[13] C. Zhang, T. Zhou, Q. Wen, and L. Sun, "Tfad: A decomposition time series anomaly detection architecture with time-frequency analysis," in *Proceedings of CIKM*, 2022, pp. 2497–2507.

[14] Y. Fang, Y. Qin, H. Luo, F. Zhao, and K. Zheng, "Stwave+: A multi-scale efficient spectral graph attention network with long-term trends for disentangled traffic flow forecasting," *IEEE Transactions on Knowledge and Data Engineering*, 2023.

[15] Z. Lai, H. Li, D. Zhang, Y. Zhao, W. Qian, and C. S. Jensen, "E2usd: Efficient-yet-effective unsupervised state detection for multivariate time series," *arXiv preprint arXiv:2402.14041*, 2024.

[16] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Deep adaptive input normalization for time series forecasting," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 9, pp. 3760–3765, 2019.

[17] Z. Shao, Z. Zhang, W. Wei, F. Wang, Y. Xu, X. Cao, and C. S. Jensen, "Decoupled dynamic spatial-temporal graph neural network for traffic forecasting," *Proceedings of VLDB*, pp. 2733–2746, 2022.

[18] S. Jiang, T. Syed, X. Zhu, J. Levy, B. Aronchik, and Y. Sun, "Bridging self-attention and time series decomposition for periodic forecasting," in *Proceedings of CIKM*, 2022, pp. 3202–3211.

[19] Y. Xue, S. Joshi, D. Nguyen, and B. Mirzasoleiman, "Understanding the robustness of multi-modal contrastive learning to distribution shift," *arXiv preprint arXiv:2310.04971*, 2023.

[20] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of SIGMOD*, 2000, pp. 93–104.

[21] J. Tang, Z. Chen, A. W.-C. Fu, and D. W. Cheung, "Enhancing effectiveness of outlier detections for low density patterns," in *Proceedings of PAKDD*, 2002, pp. 535–548.

[22] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. Cho, and H. Chen, "Deep autoencoding gaussian mixture model for unsupervised anomaly detection," in *Proceedings of ICLR*, 2018, pp. 1–19.

[23] T. Yairi, N. Takeishi, T. Oda, Y. Nakajima, N. Nishimura, and N. Takata, "A data-driven health monitoring method for satellite housekeeping data based on probabilistic clustering and dimensionality reduction," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1384–1401, 2017.

[24] D. M. Tax and R. P. Duin, "Support vector data description," *Machine learning*, vol. 54, pp. 45–66, 2004.

[25] Z. Ghafoori, S. M. Erfani, S. Rajasegarar, J. C. Bezdek, S. Karunasekera, and C. Leckie, "Efficient unsupervised parameter estimation for one-class support vector machines," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 10, pp. 5057–5070, 2018.

[26] L. Ruff, R. Vandermeulen, N. Goernitz, L. Deecke, S. A. Siddiqui, A. Binder, E. Müller, and M. Kloft, "Deep one-class classification," in *Proceedings of ICML*, 2018, pp. 4393–4402.

[27] L. Shen, Z. Li, and J. Kwok, "Timeseries anomaly detection using temporal hierarchical one-class network," in *Proceedings of NeurIPS*, 2020, pp. 13 016–13 026.

[28] H. Ren, B. Xu, Y. Wang, C. Yi, C. Huang, X. Kou, T. Xing, M. Yang, J. Tong, and Q. Zhang, "Time-series anomaly detection service at microsoft," in *Proceedings of SIGKDD*, 2019, pp. 3009–3017.

[29] J. Gao, X. Song, Q. Wen, P. Wang, L. Sun, and H. Xu, "Robusttad: Robust time series anomaly detection via decomposition and convolutional neural networks," *arXiv preprint arXiv:2002.09545*, 2020.

[30] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–33, 2021.

[31] L. Deng, X. Chen, Y. Zhao, and K. Zheng, "Hifi: Anomaly detection for multivariate time series with high-order feature interactions," in *Proceedings of DASFAA*, 2021, pp. 641–649.

[32] Z. Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, and D. Pei, "Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding," in *Proceedings of SIGKDD*, 2021, pp. 3220–3230.

[33] T. Kieu, B. Yang, C. Guo, R.-G. Cirstea, Y. Zhao, Y. Song, and C. S. Jensen, "Anomaly detection in time series with robust variational quasi-recurrent autoencoders," in *Proceedings of ICDE*, 2022, pp. 1342–1354.

[34] T. Kieu, B. Yang, C. Guo, C. S. Jensen, Y. Zhao, F. Huang, and K. Zheng, "Robust and explainable autoencoders for unsupervised time series outlier detection," in *Proceedings of ICDE*, 2022, pp. 3038–3050.

[35] B. Zhou, S. Liu, B. Hooi, X. Cheng, and J. Ye, "Beatgan: Anomalous rhythm detection using adversarially generated time series." in *Proceedings of IJCAI*, 2019, pp. 4433–4439.

[36] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "Usad: Unsupervised anomaly detection on multivariate time series," in *Proceedings of SIGKDD*, 2020, pp. 3395–3404.

[37] X. Chen, L. Deng, F. Huang, C. Zhang, Z. Zhang, Y. Zhao, and K. Zheng, "Daemon: Unsupervised anomaly detection and interpretation for multivariate time series," in *Proceedings of ICDE*, 2021, pp. 2225–2230.

[38] X. Chen, L. Deng, Y. Zhao, and K. Zheng, "Adversarial autoencoder for unsupervised time series anomaly detection and interpretation," in *Proceedings of WSDM*, 2023, pp. 267–275.

[39] Y. Zhao, X. Chen, L. Deng, T. Kieu, C. Guo, B. Yang, K. Zheng, and C. S. Jensen, "Outlier detection for streaming task assignment in crowdsourcing," in *Proceedings of WWW*, 2022, pp. 1933–1943.

[40] Y. Jiao, K. Yang, D. Song, and D. Tao, "Timeautoad: Autonomous anomaly detection with self-supervised contrastive loss for multivariate time series," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 3, pp. 1604–1619, 2022.

[41] H. Kim, S. Kim, S. Min, and B. Lee, "Contrastive time-series anomaly detection," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–14, 2023.

[42] J. Xu, H. Wu, J. Wang, and M. Long, "Anomaly transformer: Time series anomaly detection with association discrepancy," in *Proceedings of ICLR*, 2022, pp. 1–20.

[43] Y. Yang, C. Zhang, T. Zhou, Q. Wen, and L. Sun, "Dcdetector: Dual attention contrastive representation learning for time series anomaly detection," in *Proceedings of SIGKDD*, 2023, p. 3033–3045.

[44] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of CVPR*, 2022, pp. 16 000–16 009.

[45] C. Feichtenhofer, Y. Li, K. He *et al.*, "Masked autoencoders as spatiotemporal learners," in *Proceedings of NeurIPS*, 2022, pp. 35 946–35 958.

[46] Z. Hou, X. Liu, Y. Cen, Y. Dong, H. Yang, C. Wang, and J. Tang, "Graphmae: Self-supervised masked graph autoencoders," in *Proceedings of SIGKDD*, 2022, pp. 594–604.

[47] Z. Shao, Z. Zhang, F. Wang, and Y. Xu, "Pre-training enhanced spatial-temporal graph neural network for multivariate time series forecasting," in *Proceedings of SIGKDD*, 2022, pp. 1567–1577.

[48] Y. Ye, L. Xia, and C. Huang, "Graph masked autoencoder for sequential recommendation," in *Proceedings of SIGIR*, 2023, pp. 321–330.

[49] Z. Li, L. Xia, Y. Xu, and C. Huang, "Generative pre-training of spatio-temporal graph neural networks," in *Proceedings of NeurIPS*, 2023, pp. 1–18.

[50] Z. Feng and S. Zhang, "Evolved part masking for self-supervised learning," in *Proceedings of CVPR*, 2023, pp. 10 386–10 395.

[51] A. Siffer, P.-A. Fouque, A. Termier, and C. Largouet, "Anomaly detection in streams with extreme value theory," in *Proceedings of SIGKDD*, 2017, pp. 1067–1075.

[52] P. Castagliola, G. Celano, and S. Psarakis, "Monitoring the coefficient of variation using ewma charts," *Journal of Quality Technology*, vol. 43, no. 3, pp. 249–265, 2011.

[53] N. Wiener, "Generalized harmonic analysis," *Acta mathematica*, vol. 55, no. 1, pp. 117–258, 1930.

[54] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of NeurIPS*, vol. 30, 2017.

[55] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of AAAI*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[56] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," *arXiv preprint arXiv:2211.14730*, 2022.

[57] X. Zhang, Z. Zhao, T. Tsiligkaridis, and M. Zitnik, "Self-supervised contrastive pre-training for time series via time-frequency consistency," in *Proceedings of NeurIPS*, 2022, pp. 3988–4003.

[58] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," in *Proceedings of NeurIPS*, vol. 33, 2020, pp. 18 661–18 673.

[59] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Soderstrom, "Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding," in *Proceedings of SIGKDD*, 2018, pp. 387–395.

[60] A. Abdulaal, Z. Liu, and T. Lancewicki, "Practical approach to asynchronous multivariate time series anomaly detection and localization," in *Proceedings of SIGKDD*, 2021, pp. 2485–2494.

[61] A. P. Mathur and N. O. Tippenhauer, "Swat: A water treatment testbed for research and training on ics security," in *2016 international workshop on cyber-physical systems for smart water networks (CySWater)*, 2016, pp. 31–36.

[62] K.-H. Lai, D. Zha, J. Xu, Y. Zhao, G. Wang, and X. Hu, "Revisiting time series outlier detection: Definitions and benchmarks," in *Proceedings of NeurIPS*, 2021, pp. 1–13.

[63] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proceedings of ICDM*, 2008, pp. 413–422.

[64] T. Zhou, P. Niu, X. Wang, L. Sun, and R. Jin, "One Fits All: Power general time series analysis by pretrained lm," in *Proceedings of NeurIPS*, 2023, pp. 1–34.

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.