MSAnomaly: Time Series Anomaly Detection with Multi-Scale Augmentation and Fusion

Tao Yin[1], Zhibin Zhang[1], Shikang Hou[1], Huan Zhao[2], Lijiao Zheng[2], Yifei Zhou[2], Jin Xie[1*], and Meng Yan[1*]

[1] School of Big Data & Software Engineering, Chongqing University, China
{yintao,hsk}@stu.cqu.edu.cn, {zbinz,xiejin,mengy}@cqu.edu.cn
[2] China Satellite Network Exploration CO,.LTD, China

Abstract. In real-world industrial scenarios, unsupervised time series anomaly detection for massive multi-dimensional sensor data is a pressing research topic. While existing unsupervised anomaly detection methods have achieved significant progress in anomaly detection performance, they still face two limitations. First, existing methods primarily model and analyze data on a single time scale, ignoring the rich dependencies between features at different time scales. Second, traditional methods struggle to capture features across different time scales, failing to represent the temporal structure of the data comprehensively. To address these challenges, based on a multi-scale data augmentation approach and multi-scale Fusion block, we propose an unsupervised anomaly detection model, MSAnomaly, to improve the ability to learn sequential patterns at various time scales. Specifically, MSAnomaly transforms the original sequence into multiple time-scale sequences using a multi-scale data augmentation approach, fusing different time resolution features by multi-scale fusion block to model the time series effectively. Our proposed MSAnomaly enables effective model training with limited data. Extensive experiments demonstrate that MSAnomaly achieves state-of-the-art performance across multiple real-world benchmark datasets for anomaly detection.

Keywords: Time series anomaly detection · multi-scale · feature fusion.

1 Introduction

Time series anomaly detection is critical in domains such as industrial monitoring and medical diagnosis. This process involves identifying deviations from normal data distributions within large temporal datasets by modeling the typical patterns of time series data. The core task is to model the time series to uncover its structure and dynamic changes, which involves analyzing trends, periodicity, seasonality, and irregular fluctuations.

With the advancement of deep learning techniques, deep neural networks have become widely used for modeling time series. Deep neural networks in

[2] * corresponding authors

sequence modeling include three main models: (i) Recurrent Neural Networks (RNNs), which handle time dependence in sequence data but are slow and computationally intensive for long sequences due to their step-by-step processing. (ii) Transformer-based models, which capture long-distance dependencies using a self-attentive mechanism, but face rising computational and memory requirements as sequence length increases, complicating training and inference [26]. These models also have numerous parameters and require substantial data for efficient training. (iii) Temporal Convolutional Networks (TCNs) , which are more lightweight compared to RNNs and Transformers, making them suitable for resource-constrained scenarios. TCNs expand the receptive field by stacking multiple convolutional layers to capture both local and global information in sequence data. However, TCNs may still be limited by local information, necessitating additional mechanisms to integrate global information effectively.

Traditional time series analysis models often overlook learned representations across different time scales. Recent deep learning methods address this by incorporating multiresolution frameworks. TimesNet [22] decomposes complex temporal variations into multiple time scales based on predominant frequency-domain patterns, extracting variations over multiple cycles and inter-week periods. MSGNet [3] utilizes frequency-domain analysis and adaptive graph convolution to capture correlations between different sequences across multiple time scales, enabling effective long-term and short-term forecasting. However, this frequency decomposition approach may result in the loss of some time-domain information, particularly for non-periodic or irregular time series data, which may not be fully captured by frequency-domain representations.

Certain areas require simultaneous focus on short-term fluctuations at high resolution and long-term trends at low resolution. A single focus may distort interpretation and decision-making, especially in domains requiring a nuanced understanding of temporal dynamics. For instance, in finance, events may cause dramatic short-term and medium-term stock fluctuations while the long-term trend remains stable [27]. Similarly, for heart patients, sudden short-term changes in heart rate may signal potential issues, while long-term trends reflect treatment effects or deterioration [15]. However, Traditional transform-based, convolutional-based, and RNN-based deep learning methods [9, 26] typically fuse features from a fixed and single time scale when processing time series data. This limits their ability to capture the correlation characteristics of the series fully.

To address the shortcomings of existing models that overlook correlations between time scales and series stability, inspired by PatchTST [12], which decomposes series data into multiple patches, and we consider the characteristics of time series data, most of the trends and seasonal temporal relationships can be preserved while extending the data after decomposing it into two subsequences by parity downsampling. We introduce MSAnomaly, a novel multi-scale augmentation approach for improved time series learning. MSAnomaly aims to provide a more robust framework for understanding and predicting complex temporal anomaly patterns. The main contributions of this paper are as follows:

- We propose a novel anomaly detection model, MSAnomaly, based on a multi-scale data augmentation approach to accurately and efficiently model time series data by capturing intricate sequence patterns across multiple temporal resolutions.
- To tackle the challenge of integrating multi-scale time series features, we propose a multi-scale fusion block that effectively fusion feature dependencies from various time scales, improving the predictability of time-series representation learning.
- Extensive experiments on real-world datasets demonstrate that MSAnomaly outperforms 14 baseline methods, establishing state-of-the-art performance in most anomaly detection datasets.

2 Related Works

In this section, we present tasks relevant to our approach, including time series anomaly detection, data augmentation, and feature fusion.

2.1 Time Series Anomaly detection

The main purpose of anomaly detection is to isolate anomalies hidden in the data that are not easily detected. Machine learning-based algorithms play a significant role. Density-based methods [2] classify anomalies by learning the density distribution of data points in the feature space, while distance-based methods [1] utilize the distance information between data points to assess anomalies. Classification-based methods [14] treat time series anomaly detection as a classification problem.

Current deep unsupervised learning techniques for detecting time series anomalies can be broadly classified into two main categories: reconstruction-based methods and prediction-based methods. Reconstruction-based methods use the reconstruction error of the time series as the anomaly score, such as VAE [21] and LSTM-VAE [13]. Reconstruction methods incorporating Transformer [20] have likewise emerged in recent years. However, reconstruction-based methods typically provide after-the-fact alarms, as they capture anomalies post-occurrence.

Prediction-based methods, on the other hand, can predict anomalies before they occur, providing advance warnings in real industrial environments. These methods extract anomalies by comparing the error between predicted values and actual observations. MTAD-GAT [25] uses graph attention to construct relationships between features and time dependencies, distinguishing normal data from outliers in multi-dimensional time series. Anomaly detection methods using TCNs [6] employ stochastic convolution and dilation to adapt to data with temporal and large acceptance domains. TCNs use the same convolution kernel for all time steps in each layer, which limits its ability to represent complex time series patterns efficiently. To mitigate this limitation, we consider using a rich set of convolutional filters to extract diverse features for feature fusion, enhancing the dynamic adaptation of TCN.

2.2 Data Augmentation

Data augmentation aims to learn data features from different perspectives. Learning the characteristics of anomalies across various perspectives is challenging, as normal data points tend to exhibit similar latent patterns, leading to consistent reconstruction representations and prediction performance. This approach amplifies the difference between normal and anomalous points. DCdetector [24] leverages channel independence and uses a patch-wise approach to represent time series data at two scales: patch-wise and patch-in, for contrastive learning. Similarly, PatchTST [12] aggregates time steps into subseries-level patches, enhancing locality and capturing comprehensive semantic information that is not available at the point level. TimesNet [22] employs Fourier transform for frequency domain analysis, converting one-dimensional time series into a set of multi-period two-dimensional tensors, with each period involving intra-period and inter-period variations.

2.3 Feature Fusion

Feature fusion combines features from different layers or sources to improve model performance. DenseNet [7] encourages feature reuse by connecting each layer to all others, enhancing information transfer efficiency. FPN (Feature Pyramid Networks) [11] transversally connects high-level features to low-level features through upsampling for multi-scale feature fusion. GCViT [5] combines global contextual self-attention modules and standard local self-attention to model long-range and short-range spatial interactions. Revisiting VAE [21] fuses global and local frequency features into Conditional Variational Auto-Encoder (CVAE) conditions, significantly improving the accuracy of reconstructed normal data.

3 The Proposed Approach

Problem definition and formulation: In the context of multivariate time series forecasting, we consider a system containing N variables. Where the historical data is provided through a backward-looking window $X_{t-L:t}$ of length L, this matrix includes the observed values of each variable from time point $t-L$ to $t-1$. The task of time series forecasting is to estimate the values of these variables at the next T time steps based on this historical data. The output is the prediction matrix $\hat{X}_{t \text{ to } t+T-1}$, which contains the predicted values of all variables from time point t to $t+T-1$. The anomaly detection problem is defined as follows: given an input time series $X_{t-L:t}$, predict $\hat{X}_{t:t+T}$ for an unknown test sequence X_{test} that is identically distributed to the training sequence with a prediction window of length T. Here, $\hat{X}_{t:t+T}$ consists of a series of predicted values $x_t, x_{t+1}, \ldots, x_{t+T}$, where each x_t denotes the predicted value of a data point. The state of each data point is then labeled by calculating the error between the predicted value and the actual observed value, resulting in the label sequence $Y_{\text{test}} = (y_t, y_{t+1}, \ldots, y_{t+T})$. Each label $y_t \in \{0, 1\}$indicates whether the corresponding data point is normal "0" or abnormal"1".

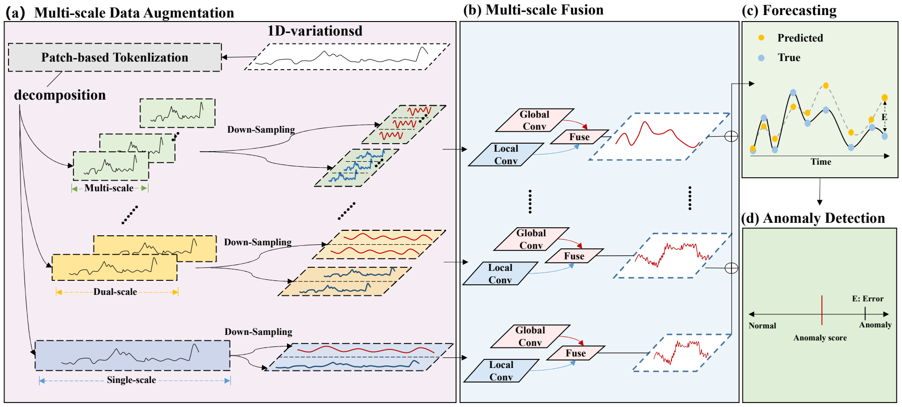

Fig. 1: The overall architecture of MSAnomaly. MSAnomaly consists of a multi-scale data augmentation approach and multi-scale fusion block, which can capture changes in different time scales through fusion convolution blocks and learn based on up-sampling fusion representation.

3.1 Overall Architecture

The overall architecture is illustrated in Fig 1. The process comprises four main components: (a) multi-scale data augmentation, (b) multi-scale fusion block, (c) time series prediction, and (d) time series anomaly detection. (a) MSAnomaly applies a multi-scale augmentation approach to the input data, generating multiple time series at varying resolutions. Each time scale produces two subsequences through continuous downsampling. (b) A parallel fusion convolution block processes these subsequences using diverse convolution filters to extract both local and global features of the time series. The resulting features at different resolutions are then upsampled, reaggregated into a new sequence representation, and incorporated back into the original time series as residuals. (c) The time series is then predicted using a fully connected network acting as a decoder. (d) Finally, data points with large prediction errors are identified as anomalies, completing the anomaly detection process.

3.2 Multi-scale Data Augmentation

Our proposed Multi-scale Data Augmentation approach transforms time series data into multi-scale data inputs. The input data $X_{t-L:t}$ represents observations from time $t-L$ to $t-1$. We process this data using the normalization function RevIN [8] which has been shown to enhance the training efficiency of the model and effectively mitigate data distribution drift. The normalization process is defined as:

$$X_{\text{in}} = \text{RevIN}(X_{t-L:t})$$

Drawing inspiration from PatchTST [22], we employ a patch-based approach to transform the time series into multiple time scales. For a selected set of time scales $\{s_1, \cdots, s_k\}$, we reshape the multivariate time series inputs$X_{\text{in}} \in \mathbb{R}^{L \times N}$into a 3D tensor, creating representations for different time scales using the following equations:

$$X^i = \text{Reshape}_{s_i, f_i}(X_{\text{in}}), \quad s_i = \frac{L}{f_i}$$

Here, $X^i \in \mathbb{R}^{N \times s_i \times f_i}$ denotes the reshaped representation for the time scale s_i. Where L is the length of the sequence, and f_i is the scale partition factor. The f_i factor is then embedded into a vector of size d_{model}, represented as X_{emb} and computed as follows:

$$\mathbf{X}_{\text{emb}} = \text{Conv1D}(\mathbf{X}^i) + \mathbf{PE}$$

We utilize a one-dimensional convolutional filter to project X^i into a d_{model}-dimensional matrix. $\mathbf{PE} \in \mathbb{R}^{d_{\text{model}} \times L}$ represents the positional embedding of the input X^i. The down-sampling decomposition process involves decomposing X^i_{emb} into $X^i_{\text{L}} \in \mathbb{R}^{N \times \frac{s_i}{2} \times d_{\text{model}}}$ and $X^i_{\text{R}} \in \mathbb{R}^{N \times \frac{s_i}{2} \times d_{\text{model}}}$. These decomposed sequences are then used as input matrices for the multi-scale Fusion Block.

3.3 Multi-scale Fusion

We propose a novel multi-scale fusion block to capture global information representations and local dependencies of sequences at different time scales. Compared to TCNs that use a single shared convolution filter at each layer, our fusion convolution block enhances feature extraction by aggregating information from sub-sequences of different time-scale decompositions, each providing a local and global view of the time series at various temporal resolutions. Compared to TCNs using shared convolution, our proposed fused convolutional module not only extracts features at different time scales through a diverse set of convolutional filters but also realizes a larger receptive field similar to extended convolution.

We achieve global information extraction by adjusting the size of spatial pooling to reduce the spatial dimension of each channel of X^i_{L} to a one-dimensional vector with global information. The global channel context is computed as follows:

$$\text{Global}(X^i_{\text{L}}) = \mathcal{B}\left(\text{PWConv}_2\left(\delta\left(\mathcal{B}\left(\text{PWConv}_1(Avg(X^i_{\text{L}}))\right)\right)\right)\right)$$

Here, Avg$(\cdot)$ denotes average pooling, and pointwise convolution PWConv is used for local channel context aggregation. The kernel sizes of PWConv$_1$ and PWConv$_2$ are $(d \times C) \times \frac{C}{r} \times 1$, where r is the channel reduction factor, $\mathcal{B}$ represents the BatchNorm, and δ denotes the Rectified Linear Unit (ReLU). The local channel context branching structure is implemented by PWConv and is computed as follows:

$$\text{Local}(X^i_{\text{R}}) = \mathcal{B}\left(\text{PWConv}_2\left(\delta\left(\mathcal{B}\left(\text{PWConv}_1(X^i_{\text{R}})\right)\right)\right)\right)$$

We then aggregate global and local scale feature information using the following equations:

$$w = \sigma\left(\text{Local}(X^i_{\text{R}}) \oplus \text{Global}(X^i_{\text{L}})\right),$$

$$\mathcal{X}^{\hat{i}}_{\text{out}} = w \otimes \text{Global}(X^i_{\text{L}}) + (1 - w) \otimes \text{Local}(X^i_{\text{R}}),$$

Here, $\oplus$ denotes the broadcast addition which generates an attentional representation incorporating both local and global context. The function σ is a Sigmoid function. Finally $\mathcal{X}^{\hat{i}}_{\text{out}} \in \mathbb{R}^{N \times \frac{s_i}{2}}$ is used as a fused feature. To advance our model, we need to integrate tensors of different scales $\mathcal{X}^{\hat{1}}_{\text{out}}$, $\mathcal{X}^{\hat{2}}_{\text{out}} \cdots, \mathcal{X}^{\hat{k}}_{\text{out}}$. The FPN (Feature Pyramid Networks) structure, renowned for its ability to capture features at multiple scales, is widely used in target detection and semantic segmentation due to its powerful feature extraction capabilities. Inspired by the FPN, we employ a pyramid structure to aggregate different time scales, enabling our model to integrate and leverage information from various temporal resolutions effectively.

$$\hat{\text{X}}_{\text{out}} = \text{Interp}(\ldots(\text{Interp}(\mathcal{X}^{\hat{1}}_{\text{out}}) + \mathcal{X}^{\hat{2}}_{\text{out}}) + \ldots) + \mathcal{X}^{\hat{k}}_{\text{out}}$$

In this process, $\text{Interp}(\cdot)$ is an interpolation operation where we recover high-resolution features step-by-step by up-sampling through linear interpolation. This method fuses multiple resolution feature layers together, effectively capturing the multi-scale dynamic information of the data. This blending strategy promotes the integration of multi-scale features into the subsequent layers, enhancing the model's ability to utilize diverse temporal information.

3.4 Time Series Forecasting

The model utilizes a linear projection to map $\hat{\mathbf{X}}_{\mathbf{out}} \in \mathbb{R}^{N \times L}$ to the $\hat{\mathbf{X}}_{t:t+T} \in \mathbb{R}^{N \times T}$ for prediction. The projection process is described as follows:

$$\hat{\mathbf{X}}_{t:t+T} = \hat{\mathbf{X}}_{\text{out}} \mathbf{W_t} + \mathbf{b}.$$

Here, $\mathbf{W_t} \in \mathbb{R}^{L \times T}$ and $\mathbf{b} \in \mathbb{R}^{T}$ are learnable parameters. $\hat{\mathbf{X}}_{t:t+T}$ is the final prediction.

3.5 Time Series Anomaly Detection

The selection of anomaly thresholds has a greater impact on the results of anomaly detection. We use the Peak Over Threshold (POT) [17] method, commonly adopted in previous research, to select the thresholds. The core idea of the POT method is to fit the data distribution to a generalized Pareto distribution and use the fitting results to determine the appropriate thresholds. This method more accurately reflects the extremes of the data and provides more robust anomaly detection results.

$$y_i = \{\begin{matrix} 1, \text{ if } e_i \geq \text{POT}(e_i), \\ 0, \quad\quad \text{otherwise}. \end{matrix}$$

Here, e_i denotes the error between the true value $\mathbf{X}_{t:t+T}$ and the predicted value $\hat{\mathbf{X}}_{t:t+T}$, which is used as the anomaly score. Larger errors are more likely to be judged as anomalies. Additionally, if any of the N dimensions is anomalous, we mark the current timestamp as an anomaly by $y = \bigvee_i y_i$.

4 Experiments

In this section, we comprehensively evaluate the performance of multivariate time series prediction models and the effectiveness of anomaly detection to validate the efficacy of MSAnomaly.

4.1 Datasets

To evaluate MSAnomaly's sophistication in time series anomaly detection, the anomaly detection capability was validated on five mainstream anomaly detection datasets: MSL (Mars Science Laboratory), WADI (Water Distribution Dataset), PSM (Pooled Server Metrics Dataset), SWaT (Secure Water Treatment Dataset), and SMD(Server Machine Dataset).

Table 1: Details of Anomaly Detection benchmark datasets. AR (anomaly ratio) represents the abnormal proportion of the whole.

Dataset	Train	Test	Dimensions	Anomalies (%)
SWaT	496800	449919	51	11.98
SMD	708405	708420	38	4.16
MSL	58317	73729	55	10.72
PSM	132,481	87,841	25	27.8
WADI	1048571	172801	127	5.99

4.2 Baselines

To demonstrate the effectiveness of MSAnomaly, we comprehensively compare it with a total of 14 state-of-the-art baseline models for anomaly detection tasks. For anomaly detection, the baseline models include reconstruction-based models: Anomaly Transformer [23], DCdetector [24], TranAD [20], OmniAnomaly [18], GDN [4], InterFusion [10], LSTM-VAE [13]; density-based estimation models: LOF [2], DAGMM [28]; and clustering-based methods: THOC [16], as well as classical methods: OC-SVM [19].

4.3 Experimental Setups

The experiments are conducted using NVIDIA GeForce RTX 3090 24GB GPUs with mean square error (MSE) as the training loss function.

For the anomaly detection task, we use a backtracking window of 48 and the widely used single-step prediction method for the anomaly prediction window. For POT parameters, the default coefficient is 10^{-4} for all data sets, following the implementation of OmniAnomaly [18]. Model training may be terminated early if applicable. For the baseline, relevant data from the paper DCdetector [24] or the official code were used.

4.4 Evaluation Metrics

In the anomaly detection task, MSAnomaly and all baseline models adopt the widely used point-adjusted F1 score evaluation strategy [23, 24]. According to this strategy, if an anomaly occurs over a period of time and any of these time points is recognized as anomalous by the model, the entire time period is considered abnormal. This approach is justified in practice, as detecting an anomaly at a single time point typically indicates that the entire continuous segment is abnormal.

Recent research has sparked intense discussions on fair evaluation methods for anomaly detection algorithms. To address this, we complement our experiments with additional metrics, including the Area Under the ROC Curve (AUC), Affiliation metric Precision (Aff-P), and Affiliation metric Recall (Aff-R). AUC measures the performance of classification models by assessing the area under the Receiver Operating Characteristic (ROC) curve. Aff-P and Aff-R evaluate the precision and recall of the membership metric, respectively, providing insight into the model's performance regarding anomaly membership.

Table 2: Comparison of anomaly detection performance of MSAnomaly on five real-world datasets. P, R, and F1 represent precision, recall, and F1 score, respectively. All results are in %. Bold represents the best performance, and underline indicates the second best.

Method	SWaT			PSM			MSL			SMD			WADI		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
IForest	96.2	73.15	83.11	76.09	92.45	83.48	47.72	85.25	61.18	56.34	39.86	46.68	62.41	61.55	61.98
LOF	72.15	65.43	68.62	57.89	90.49	70.61	47.72	85.25	61.18	56.34	39.86	46.68	5.63	88.39	10.58
ITAD	63.13	52.08	57.08	72.8	64.02	68.13	69.44	84.09	76.07	86.22	73.71	79.48	**92.11**	58.79	70.25
MMPCACD	82.52	68.29	74.73	76.26	78.35	77.29	81.42	61.31	69.95	71.2	79.28	75.02	88.61	75.84	81.73
CL-MPPCA	82.52	68.29	74.73	76.26	78.35	77.29	81.42	61.31	69.95	71.2	79.28	75.02	88.61	75.84	81.73
Deep-SVDD	80.42	84.45	82.39	95.41	86.49	90.73	91.92	76.63	83.58	78.54	79.67	79.1	83.7	47.88	60.03
LSTM	86.26	83.37	84.79	76.95	89.64	82.81	85.51	82.53	83.99	78.67	85.34	81.87	72.41	27.93	40.31
DAGMM	89.92	57.84	70.4	93.49	70.03	80.08	89.6	63.93	74.62	67.3	49.89	57.3	22.28	19.76	20.94
OmniAnomaly	81.42	84.3	82.83	88.39	74.46	80.83	89.02	86.37	87.67	83.68	86.82	85.22	31.58	65.41	42.46
InterFusion	80.59	85.58	83.01	83.61	83.45	83.52	81.28	92.7	86.62	87.02	85.43	86.22	85.44	84.62	85.03
GDN	96.91	69.57	81.01	42.16	73.33	53.56	77.51	**100**	87.33	71.7	**99.74**	83.42	29.12	79.31	42.6
TranAD	**97.6**	69.97	81.51	86	89.86	87.89	89.51	92.97	91.15	89.06	89.82	87.85	81.18	83.01	82.08
AnomalyTrans	89.1	99.28	94.22	96.94	97.81	97.37	91.92	96.03	93.93	88.68	89.10	88.89	66.45	**100**	79.84
DCdetector	93.11	**99.77**	**96.33**	97.14	**98.74**	97.94	**92.22**	97.48	**94.77**	83.59	91.1	87.18	85.69	99.12	**91.91**
Ours	96.95	95.15	96.24	**97.75**	98.24	**98.02**	84.88	94.17	89.06	**90.55**	93.0	**91.73**	89.71	91.25	90.47

4.5 Anomaly Detection Performance

In terms of anomaly detection, prediction-based anomaly detection is considered a classical approach for unsupervised point-by-point representation learning in

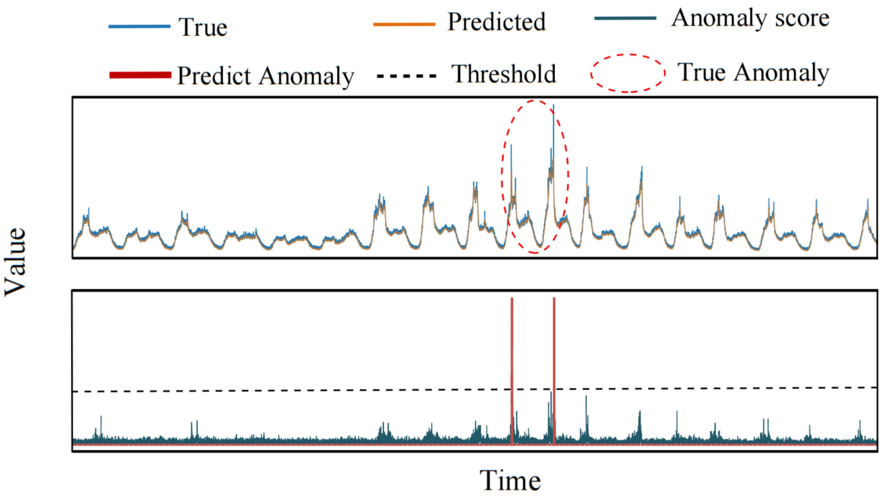

Fig. 2: Visualization of Predictive Anomalies Using MSAnomaly Models.

previous studies. In this approach, prediction error naturally serves as one of the criteria for anomaly judgments, and we adopt this classical prediction error as our anomaly scoring criterion. We use the Peak Over Threshold (POT) method, commonly used in previous research such as TranAD and OmniAnomaly, to account for local peaks in the sequence and automatically select the threshold as the anomaly score. As shown in Fig 2, we visualize the anomaly score in dimension 6 of the SMD dataset, marking large prediction errors as anomalies. Results in Table 2 compare MSAnomaly with 14 baseline models across five real-world multivariate datasets using various application metrics such as precision, recall, and F1 scores. Specifically, MSAnomaly achieves the best performance on all three datasets.

MSAnomaly and DCdetector achieved the highest F1 scores on the WADI dataset, with scores of 90.47% and 91.9% respectively. In contrast, other baseline models performed poorly on this dataset due to the large data volume and long sequence length, which traditional models struggle to handle effectively. The success of DCdetector and MSAnomaly highlights the importance of multi-scale time series analysis in anomaly detection. However, our model is less effective on the MSL dataset. This is because the MSL dataset contains a large number of discrete values, and multi-scale downsampling may result in the loss of key information, affecting MSAnomaly's performance. Additionally, we provide supplementary metrics such as affiliation metric precision, affiliation metric recall, and AUC to evaluate model performance comprehensively. In this comparison, we choose the DCdetector and Anomaly Transformer as benchmarks due to their strong anomaly detection performance. As shown in Table 3, MSAnomaly outperforms both DCdetector and Anomaly Transformer in several metrics across the three selected datasets.

Table 3: Multi-metrics results are compared, all results are expressed as percentages, and the best results are marked in bold.

Dataset	Model	Aff-P	Aff-R	F1	AUC
SWaT	AnomalyTrans	53.03	**98.08**	94.22	98.32
	DCdetector	52.40	97.67	**96.33**	**99.5**
	Ours	**89.29**	93.36	96.24	97.06
PSM	AnomalyTrans	55.35	80.28	97.37	98.42
	DCdetector	54.71	82.93	97.94	98.74
	Ours	**79.84**	**96.02**	**98.02**	**99.51**
SMD	AnomalyTrans	69.96	89.15	88.89	97.03
	DCdetector	**85.96**	84.82	85.39	92.1
	Ours	83.59	**99.91**	**91.73**	**97.88**

Notably, as shown in Table 4, our approach significantly reduces training time for most datasets, with the reduction being especially pronounced compared to DCdetector. On the SMD dataset, our method took 10 seconds longer than AnomalyTrans, but overall performance remained superior. Our method reduces average training time by approximately 42.15% compared to Anomaly Trans and about 98.88% compared to DCdetector. These results demonstrate that MSAnomaly achieves efficient training times while maintaining excellent performance.

In summary, the MSAnomaly model demonstrates exceptional capability in anomaly detection. It effectively identifies rare anomalous temporal patterns from different time scale perspectives and data granularities, underscoring its rapidity and effectiveness in diverse scenarios.

Table 4: Model efficiency comparison experimental results, the minimum model training time is highlighted in bold.

Training Time (s/epoch)	AnomalyTrans	DCdetector	Ours
SWaT	600	10000	**250**
PSM	120	1800	**60**
MSL	80	100	**20**
SMD	**10**	500	20
WADI	400	50000	**350**
Avg	242	12480	**140**

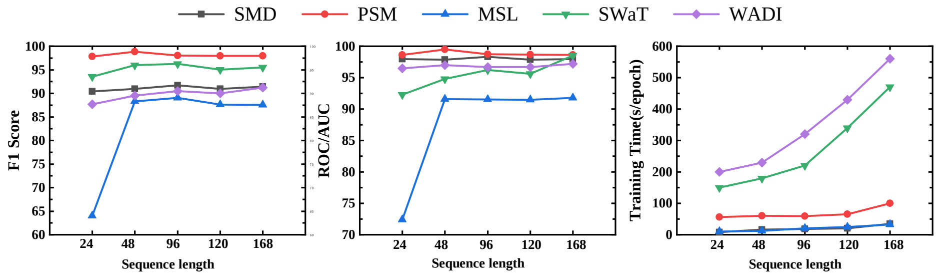

Fig. 3: F1 score, ROC/AUC, F1 score and training times with different Sequence length.

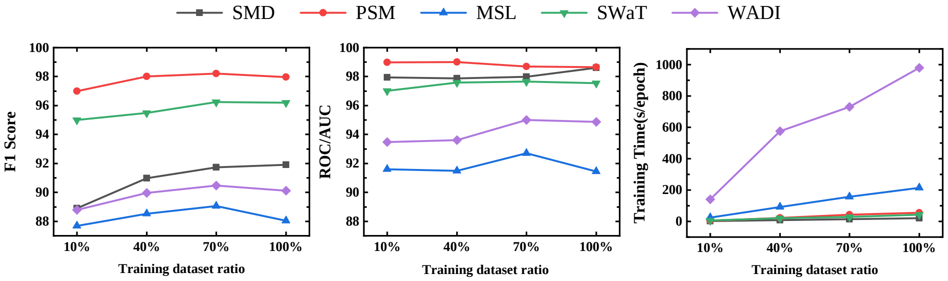

Fig. 4: F1 score, ROC/AUC, F1 score and training times with different dataset size.

4.6 Model Analysis

Sensitivity to sequence length sequence length is a critical factor influencing the detection performance of MSAnomaly models. We extensively investigate the impact of sequence length on anomaly detection in multivariate time series data. Figure 3 illustrates the anomaly detection performance of MSAnomaly with varying sequence lengths. For most datasets (black, red, green, blue, and purple lines), the F1 score generally stabilizes as sequence length increases, maintaining consistent detection performance. Specifically, the F1 score increases significantly at a sequence length of 48 and stabilizes at larger sizes. When the sequence length is 24, the F1 scores of the MSL dataset (blue line) are notably lower, indicating that the multi-scale information captured by MSAnomaly is insufficient for accurate anomaly detection due to the small sequence length. Overall, the F1 scores of MSAnomaly under different sequence lengths vary minimally and remain high, suggesting that MSAnomaly is less sensitive to sequence length changes and has stable performance. Additionally, our experiments show that MSAnomaly can respond and detect anomalies faster with smaller sequence lengths because a smaller input window reduces inference time. However, if the window is too small, the model may not fully capture the necessary local context information, affecting detection accuracy. Conversely, if the window is too large, although the model can utilize richer contextual information, short-term anomalies may be masked by normal data, reducing sensitivity to small-scale anomalies.

To balance response speed and accuracy, our experiments determine that a sequence length of 48 is optimal. This sequence length ensures fast inference speed and enables efficient anomaly detection without losing crucial contextual information.

Sensitivity to training data set size We investigated the impact of training dataset size on MSAnomaly's performance across five anomaly detection datasets, including PSM and SwaT. Figure 4 illustrates variations in average F1 and AUC scores, as well as training time, with training data proportions ranging from 10% to 100%. On the PSM and SwaT datasets, MSAnomaly consistently achieved high and stable F1 and AUC scores. For the SMD dataset, the F1 score peaked at 40% training data, with minor fluctuations at other proportions. The MSL and WADI datasets showed minimal variability, indicating consistent performance across different training data proportions. Training time significantly increased with larger training data proportions, especially for the WADI dataset due to its size and high feature dimensionality. However, training times for the other datasets remained relatively low, demonstrating efficient training capabilities. Overall, MSAnomaly performed exceptionally well, maintaining efficiency and stability even with smaller training datasets. To balance efficiency and performance, we use 40%-70% of the data for training, ensuring high F1 and AUC scores while keeping training time reasonable.

5 Conclusion

We propose MSAnomaly, a multi-scale data augmentation approach and multi-scale fusion block for time series modeling and anomaly detection. MSAnomaly enhances the precision of detecting deviations from normal data distributions by fusing intricate sequence patterns across multiple temporal resolutions. Our findings emphasize the importance of fusing features across different time scales in time series data analysis. Extensive experiments and validations on various real-world datasets demonstrate that MSAnomaly not only outperforms existing models in terms of performance but also effectively captures complex multi-scale dependencies across different time scales. Specifically, Compared with state-of-the-art models, MSAnomaly significantly reduces training time while maintaining anomaly detection performance, enabling fast and accurate anomaly prediction and analysis. We plan to enhance MSAnomaly further to handle data collected at irregular time intervals and improve its robustness against noisy and missing data. Additionally, we will explore other approaches to further improve the accuracy of MSAnomaly in time series anomaly detection, making it more effective in a wider range of practical applications.

6 Acknowledgements

This work was supported in part by the Natural Science Foundation of Chongqing (No. CSTB2023NSCQ-LZX0130), the Chongqing Technology Innovation and Application Development Project (No. CSTB2023TIAD-STX0025), the General Program of Chongqing Natural Science Foundation(cstc2021jcyj-msxmX0309) and the Fundamental Research Funds for the Central Universities (No. 2023CD-JKYJH013).

References

1. Angiulli, F., Pizzuti, C.: Fast outlier detection in high dimensional spaces. In: European conference on principles of data mining and knowledge discovery. pp. 15–27. Springer (2002)
2. Breunig, M.M., Kriegel, H.P., Ng, R.T., Sander, J.: Lof: identifying density-based local outliers. In: Proceedings of the 2000 ACM SIGMOD international conference on Management of data. pp. 93–104 (2000)
3. Cai, W., Liang, Y., Liu, X., Feng, J., Wu, Y.: Msgnet: Learning multi-scale inter-series correlations for multivariate time series forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 11141–11149 (2024)
4. Deng, A., Hooi, B.: Graph neural network-based anomaly detection in multivariate time series. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 4027–4035 (2021)
5. Hatamizadeh, A., Yin, H., Heinrich, G., Kautz, J., Molchanov, P.: Global context vision transformers. In: International Conference on Machine Learning. pp. 12633–12646. PMLR (2023)
6. He, Y., Zhao, J.: Temporal convolutional networks for anomaly detection in time series. In: Journal of Physics: Conference Series. vol. 1213, p. 042050. IOP Publishing (2019)
7. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)
8. Kim, T., Kim, J., Tae, Y., Park, C., Choi, J.H., Choo, J.: Reversible instance normalization for accurate time-series forecasting against distribution shift. In: International Conference on Learning Representations (2021)
9. Kitaev, N., Kaiser, Ł., Levskaya, A.: Reformer: The efficient transformer. arXiv preprint arXiv:2001.04451 (2020)
10. Li, Z., Zhao, Y., Han, J., Su, Y., Jiao, R., Wen, X., Pei, D.: Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding. In: Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining. pp. 3220–3230 (2021)
11. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
12. Nie, Y., Nguyen, N.H., Sinthong, P., Kalagnanam, J.: A time series is worth 64 words: Long-term forecasting with transformers. arXiv preprint arXiv:2211.14730 (2022)

13. Park, D., Hoshi, Y., Kemp, C.C.: A multimodal anomaly detector for robot-assisted feeding using an lstm-based variational autoencoder. IEEE Robotics and Automation Letters **3**(3), 1544–1551 (2018)
14. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural computation **13**(7), 1443–1471 (2001)
15. Shaffer, F., McCraty, R., Zerr, C.L.: A healthy heart is not a metronome: an integrative review of the heart's anatomy and heart rate variability. Frontiers in psychology **5**, 108292 (2014)
16. Shen, L., Li, Z., Kwok, J.: Timeseries anomaly detection using temporal hierarchical one-class network. Advances in Neural Information Processing Systems **33**, 13016–13026 (2020)
17. Siffer, A., Fouque, P.A., Termier, A., Largouet, C.: Anomaly detection in streams with extreme value theory. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 1067–1075 (2017)
18. Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W., Pei, D.: Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 2828–2837 (2019)
19. Tax, D.M., Duin, R.P.: Support vector data description. Machine learning **54**, 45–66 (2004)
20. Tuli, S., Casale, G., Jennings, N.R.: Tranad: Deep transformer networks for anomaly detection in multivariate time series data. arXiv preprint arXiv:2201.07284 (2022)
21. Wang, Z., Pei, C., Ma, M., Wang, X., Li, Z., Pei, D., Rajmohan, S., Zhang, D., Lin, Q., Zhang, H., et al.: Revisiting vae for unsupervised time series anomaly detection: A frequency perspective. arXiv preprint arXiv:2402.02820 (2024)
22. Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., Long, M.: Timesnet: Temporal 2d-variation modeling for general time series analysis. In: The eleventh international conference on learning representations (2022)
23. Xu, J., Wu, H., Wang, J., Long, M.: Anomaly transformer: Time series anomaly detection with association discrepancy. arXiv preprint arXiv:2110.02642 (2021)
24. Yang, Y., Zhang, C., Zhou, T., Wen, Q., Sun, L.: Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. In: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. pp. 3033–3045 (2023)
25. Zhao, H., Wang, Y., Duan, J., Huang, C., Cao, D., Tong, Y., Xu, B., Bai, J., Tong, J., Zhang, Q.: Multivariate time-series anomaly detection via graph attention network. In: 2020 IEEE International Conference on Data Mining (ICDM). pp. 841–850. IEEE (2020)
26. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., Zhang, W.: Informer: Beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 11106–11115 (2021)
27. Zhou, Z., Lin, L., Li, S.: International stock market contagion: A ceemdan wavelet analysis. Economic Modelling **72**, 333–352 (2018)
28. Zong, B., Song, Q., Min, M.R., Cheng, W., Lumezanu, C., Cho, D., Chen, H.: Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In: International conference on learning representations (2018)