SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2026, Vol. 69, Iss. 3, 132105:1–132105:16
https://doi.org/10.1007/s11432-024-4577-6

A gradual coarse-to-fine framework for irregularly sampled multivariate time series analysis

Jiexi LIU, Meng CAO & Songcan CHEN*

MIIT Key Laboratory of Pattern Analysis and Machine Intelligence,
College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Received 30 November 2024/Revised 17 April 2025/Accepted 29 August 2025/Published online 21 January 2026

Abstract Irregularly sampled multivariate time series (ISMTS) are prevalent in reality. Most existing methods treat ISMTS as synchronized regularly sampled time series with missing values, neglecting that the irregularities are primarily attributed to variations in sampling rates. In this paper, we introduce a novel perspective that irregularity is essentially relative in some sense. With sampling rates artificially determined from low to high, an irregularly sampled time series can be transformed into a hierarchical set of relatively regular time series from coarse to fine. We observe that additional coarse-grained, relatively regular time series not only mitigate the irregularly sampled challenges but also incorporate broad-view temporal information, thereby serving as a valuable asset for representation learning. Therefore, following the philosophy of learning that sees the big picture first, then delving into the details, we present the multi-scale and multi-correlation attention network (MuSiCNet), combining multiple scales to iteratively refine the ISMTS representation. Specifically, within each scale, we explore time attention and frequency correlation matrices to aggregate intra- and inter-series information, naturally enhancing the representation quality with richer and more intrinsic details. Across adjacent scales, we employ a representation rectification method containing contrastive learning and reconstruction results adjustment to further improve representation consistency. Experimental results demonstrate that MuSiCNet consistently achieves competitive performance with state-of-the-art methods across four key ISMTS tasks: classification, interpolation, forecasting, and anomaly detection.

Keywords irregularly sampled multivariate time series, time series analysis, attention mechanism, multi-scale learning, representation learning

Citation Liu J X, Cao M, Chen S C. A gradual coarse-to-fine framework for irregularly sampled multivariate time series analysis. Sci China Inf Sci, 2026, 69(3): 132105, https://doi.org/10.1007/s11432-024-4577-6

1 Introduction

Irregularly sampled multivariate time series (ISMTS) are ubiquitous in realistic scenarios, ranging from scientific explorations to societal interactions [1–7]. The causes of irregularities in time series collection are diverse, including sensor malfunctions, transmission distortions, cost-reduction strategies, and various external forces or interventions. Such ISMTS data exhibit distinctive features, including intra-series irregularity, characterized by inconsistent intervals between consecutive data points, and inter-series irregularity, marked by a lack of synchronization across multiple variables. The above characteristics typically result in the lack of alignment and uneven count of observations [8], invalidating the assumption of a coherent fixed-dimensional feature space for most traditional time series analysis models.

Recent studies have attempted to address these challenges by treating ISMTS as synchronized, regularly sampled multivariate time series (RSMTS) data with missing values, focusing on imputation strategies [2, 9–15]. However, direct imputation is difficult, especially when sampling is sparse. Inaccurate imputation results can distort underlying relationships and introduce significant noise, which can greatly reduce the accuracy of analysis tasks [3, 16–18]. The latest developments circumvent imputation and aim to address these challenges by embracing the inherent continuity of time, thus preserving the continuous temporal dynamics dependencies within the ISMTS data. Despite these innovations, most of the methods above are merely solutions for intra-series irregularities, such as recurrent neural networks(RNNs)- [3, 19, 20] and neural ordinary differential equations (neural ODEs)-based methods [21–24] and the unaligned challenges presented by inter-series irregularities in multivariate time series remain unsolved.

Delving into the nature of ISMTS, we discover that the intra- and inter-series irregularities primarily arise from inconsistency in sampling rates within and across variables. We argue that irregularities are essentially relative

[*] Corresponding author (email: s.chen@nuaa.edu.cn)

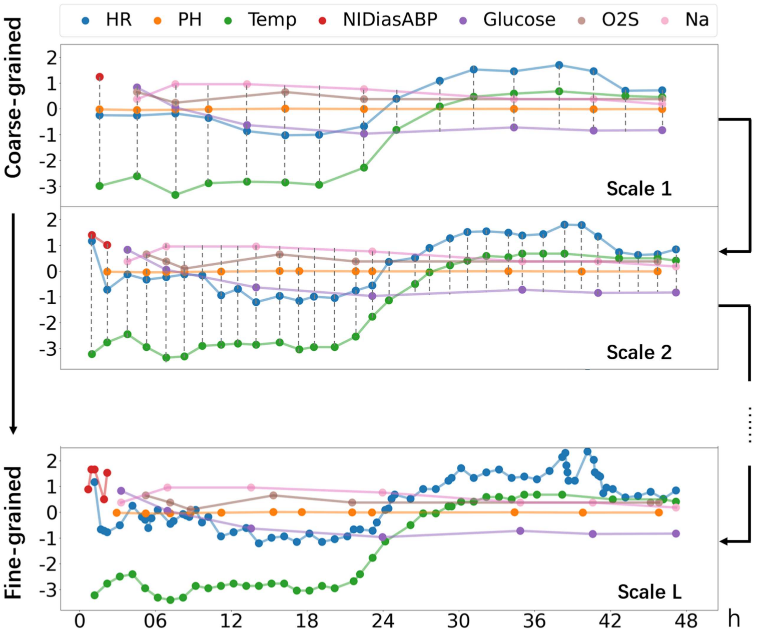

Figure 1 Comparative visualization of multi-scale time series data with various sampling rates. Scale L depicts the original selected representative time series in the P12 Dataset to show the inter- and intra-series irregularities. Scale 1 to scale $L-1$ illustrate the effect of applying different sampling rates from low to high.

in some sense, and by artificially determined sampling rates from low to high, ISMTS can be transformed into a hierarchical set of relatively regular time series from coarse to fine. Taking a broader perspective, setting a lower and consistent sampling rate within an instance can synchronize sampling times across series and establish uniform time intervals within each series. This approach can mitigate both types of irregularity and emphasize long-term dependencies. As shown in Figure 1, the coarse-grained scales 1 and 2 exhibit balanced placement of observation timestamps across all variables in the instance, providing clearer overall trends. However, lower sampling rates may lead to information loss and sacrifice detailed temporal variations. Conversely, with a higher sampling rate as in scale L, more real observations contain rich information and prevent artificially introduced dependencies beyond the original relations during training. Nonetheless, the significant irregularity in fine-grained scales poses a greater challenge for representation learning.

To bridge this gap, we propose MuSiCNet, a multi-scale and multi-correlation attention network, to iteratively optimize ISMTS representations from coarse to fine. Our approach begins by establishing a hierarchical set of coarse-to fine-grained series with sampling rates from low to high. At each scale, we employ a custom-designed encoder-decoder framework called multi-correlation attention network (CorrNet) for representation learning. Different from most existing methods that focus mainly on intra-series relationships, our CorrNet encoder (CorrE) captures embeddings of continuous time values by using an attention mechanism with correlation matrices to aggregate both intra- and inter-series information. This approach is crucial not only because every observation in ISMTS, given the sparse sampling, is valuable for representation learning, but also due to the fact that correlated variables provide deeper insights for a given query timestamp. Specifically, we construct frequency-domain correlation matrices using the Lomb-Scargle periodogram-based dynamic time warping (LSP-DTW), which effectively tackles the challenge of computing correlations in ISMTS. This strategy enables the aggregation of inter-series information, facilitating more effective representation learning. Across scales, we employ a representation rectification operation from coarse to fine to iteratively refine the learned representations with contrastive learning and reconstruction results adjustment methods. This ensures accurate and consistent representation and minimizes error propagation throughout the model. Benefiting from the aforementioned designs, MuSiCNet explicitly learns multi-scale information, enabling good performance on widely used ISMTS datasets, thereby demonstrating its ability to capture proper features for ISMTS analysis. Our main contributions can be summarized as follows.

- We find that irregularities in ISMTS are essentially relative in some sense and multi-scale learning helps balance coarse- and fine-grained information in ISMTS representation learning.
- We introduce CorrNet, an encoder-decoder framework designed to learn fixed-length representations for ISMTS. Notably, our proposed LSP-DTW can mitigate spurious correlations induced by irregularities in the frequency domain and effectively aggregate inter-series information.

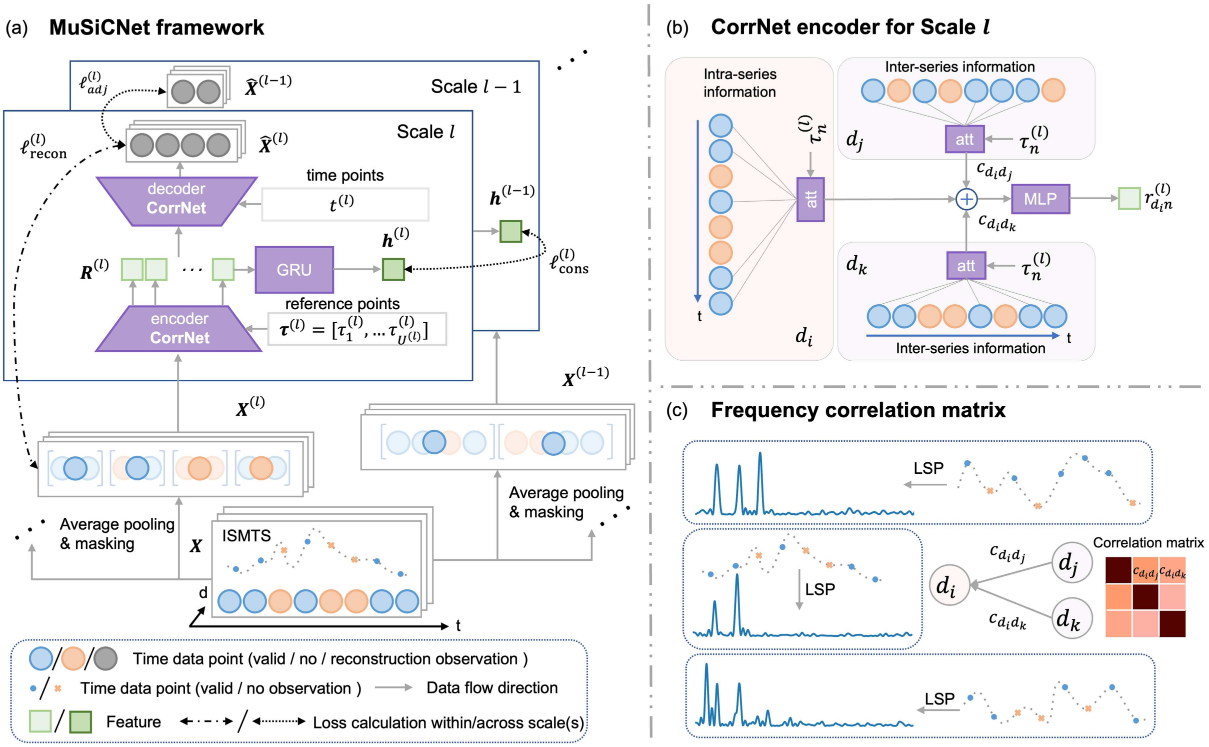

Figure 2 Overview of MuSiCNet framework, as shown in (a), containing three main components for better representation learning, including hierarchical structure $\{X^{(l)}\}_{l=1}^{L}$, representation learning using CorrNet within scale $\ell_{\rm cons}^{(l)}$, and rectification operation across adjacent scales $\ell_{\rm recon}^{(l)}$; (b) visualizes the encoding process in CorrNet for scale l, which relies on $\tau^{(l)}$ to aggregate intra-series information, and then relies on $c_{d_i,(\cdot)}$ to fuse inter-series information from other variables for d_ith dimension; (c) visualizes the calculation process of the correlation matrix, which transfers the time domain into the frequency domain with LSP, and then utilizes DTW to calculate the similarity weight.

• We are not limited to a specific analysis task and attempt to propose a task-general model for ISMTS analysis, including classification, interpolation, forecasting, and anomaly detection.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 presents a detailed introduction to the proposed MuSiCNet. Section 4 presents extensive experiments to demonstrate the effectiveness of MuSiCNet. Section 5 concludes the paper with key findings.

2 Related work

2.1 Irregularly sampled multivariate time series analysis

An effective approach for analyzing ISMTS hinges on the understanding of their unique properties. Most existing methods treat ISMTS as RSMTS with missing values, such as [2, 9–13, 15, 25]. However, most imputation-based methods may distort the underlying relationships, introducing unsuitable inductive biases and substantial noise due to incorrect imputation [3, 16, 17], ultimately compromising the accuracy of downstream tasks. Some other methods treat ISMTS as time series with discrete timestamps, aggregating all sample points of a single variable to extract a unified feature for each variable [16, 26, 27]. These methods can directly accept raw ISMTS data as input but often struggle to handle the underlying relationships within the time series. Recent progress seeks to overcome these challenges by recognizing and utilizing the inherent continuity of time, thereby maintaining the ongoing temporal dynamics present in ISMTS data [19–23, 28].

Despite these advancements, existing methods mainly suffer from two main drawbacks: they primarily address intra-series irregularity while overlooking the alignment issues stemming from inter-series irregularity, and they rely on assumptions tailored to specific downstream tasks [4, 25], hindering their ability to consistently perform well across various ISMTS tasks.

2.2 Multi-scale time series modeling

Multi-scale and hierarchical approaches have demonstrated their utility across various fields, including computer vision (CV) [29,30], natural language processing (NLP) [31,32]. Recent innovations in the time series analysis domain are mostly for RSMTS, with many approaches integrating multi-scale modules into the Transformer architecture or graph neural network (GNN) to enhance analysis capabilities. For Transformer, Scaleformer [33] employs an iterative refinement framework that progressively corrects forecasts across multiple temporal scales, enhancing overall accuracy while keeping computational costs low. Meanwhile, Pathformer [34] introduces adaptive pathways within a Transformer, dynamically routing information at various time scales to simultaneously capture short-term fluctuations and long-term trends for improved forecasting. Moreover, Pyraformer [35] utilizes a pyramidal attention mechanism to efficiently aggregate information across different scales, effectively modeling long-range dependencies with reduced computational complexity. For GNN, MTSF-DG [36] leverages dynamic graph modeling to capture evolving inter-series dependencies over multiple scales, thereby adapting to changing relationships among time series and boosting forecast accuracy. Additionally, MSGNet [37] uses a GNN architecture to learn multi-scale correlations across multiple time series, effectively extracting both global and local dynamics to enhance multivariate forecasting performance.

Nevertheless, while the aforementioned methods are designed for RSMTS, the application of multi-scale modeling for ISMTS data, along with the effective exploitation of cross-scale information, remains less unexplored. As far as we know, Refs. [38, 39] are among the earlier studies on multi-scale ISMTS learning. Ref. [38] addresses multi-resolution signal issues by distributing signals across specialized branches with different resolutions, where each branch employs a flexible irregular time series network (FIT) to process high- and low-frequency data separately. Warpformer [39], on the other hand, is a transformer-based model that stacks multiple Warpformer layers to produce multi-scale representations, combining them via residual connections to support downstream tasks. These studies typically focus on either specific tasks or particular model architectures. In contrast, our design philosophy originates from ISMTS characteristics rather than being tied to a specific feature extraction network structure. Warpformer emphasizes designing a specific network architecture but involves high computational costs and requires manually balancing the trade-off between the number of scales and the dataset. These are challenges that our MuSiCNet avoids entirely.

3 Proposed MuSiCNet framework

As previously mentioned, our work aims to learn ISMTS representations for further analysis tasks by introducing MuSiCNet, a novel framework designed to balance coarse- and fine-grained information across different scales. The overall model architecture illustrated in Figure 2(a) indicates that the effectiveness of MuSiCNet can be guaranteed to a great extent by (1) hierarchical structure, (2) representation learning using cornet within scale, and (3) rectification across adjacent scales. We will first introduce the problem formulation and notations of MuSiCNet, and then discuss key points in the following subsections.

3.1 Problem formulation

Our goal is to learn a nonlinear embedding function f_θ, such that the set of ISMTS data $\boldsymbol{\mathcal{X}} = \{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_N\}$ can map to the best-described representations $\boldsymbol{\mathcal{R}} = \{\boldsymbol{R}_1, \ldots, \boldsymbol{R}_N\}$ for further ISMTS analysis. The nth sample $\boldsymbol{X}_n \in \mathbb{R}^{T_n \times D}$ in $\boldsymbol{\mathcal{X}}$ is a D-dimensional sample, and its real observation length is T_n. We drop the data case index n for brevity when the context is clear.

3.2 Multi-scale data acquisition

By downsampling the original instance, a hierarchical set of relatively regular series is obtained, forming a hierarchical structure ranging from coarse to fine with the original series:

$$\boldsymbol{X}_{\text{multi}} = \{\boldsymbol{M}^{(l)} \odot \text{Avg}_l(\boldsymbol{X})\}_{l=1}^{L-1} \cup \{\boldsymbol{M}^{(L)} \odot \boldsymbol{X}\} = \{\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(L)}\} \tag{1}$$

with corresponding observation time set $T_{\text{multi}} = \{\boldsymbol{t}^{(1)}, \ldots, \boldsymbol{t}^{(L)}\}$. Since MuSiCNet adopts a masked modeling-based reconstruction representation learning approach introduced in the following subsection, a random masking set $\{\boldsymbol{M}^{(1)}, \ldots, \boldsymbol{M}^{(L)}\}$ is applied to the instance at each scale. For simplicity, an improved average pooling operation Avg is selected as the downsampling method, where the average is taken over the observed values based on the real observation counts.

3.3 CorrNet architecture within scale

Drawing inspiration from notable advances in NLP [40, 41] and CV [42–44], masked modeling for time series data aims to learn robust representations for various downstream tasks. CorrNet comprises an encoder–decoder architecture based on continuous-time interpolation. At each scale l, the CorrNet encoder (CorrE) learns a set of latent representations $\boldsymbol{R}^{(l)} = [r_1^{(l)}, \ldots, r_{U^{(l)}}^{(l)}]$ of length $U^{(l)}$ from randomly masked ISMTS data $\boldsymbol{X}^{(l)} \in \mathbb{R}^{T^{(l)} \times D}$ by leveraging intra-series and inter-series correlations via multi-time attention and a correlation matrix $\boldsymbol{C}$. Additionally, CorrNet decoder (CorrD), a simplified version of CorrE without the correlation matrix, generates a reconstructed output $\hat{\boldsymbol{X}}^{(l)}$ from $\boldsymbol{R}^{(l)}$ that matches the dimensions of the input $\boldsymbol{t}^{(l)}$. We apply the same CorrNet iteratively at each scale and emphasize that all scales share a single encoder, which reduces model complexity and ensures consistency in feature extraction across various scales.

Time embedding. The time embedding encoding function **TE** adopts the embedding approach proposed in mTAND [1], which extends the position encoding mechanism used in Transformer [45] models to continuous time. The ith component of the embedding is defined as follows, with each output having a dimension of d_r:

$$\mathbf{TE}_i(t) = \begin{cases} \omega_0 \cdot t + \alpha_0, & \text{if } \ i = 0, \\ \sin\left(\omega_i \cdot t + \alpha_i\right), & \text{if } \ 0 < i < d_r, \end{cases} \tag{2}$$

where ω_i and α_i are learnable parameters.

CorrNet encoder. In CorrE, a multi-correlation attention module is employed to learn representations by integrating both intra-series and inter-series correlations. Prior studies have demonstrated the effectiveness of temporal attention mechanisms in ISMTS learning [1, 26, 28, 46]. However, most existing methods mainly focus on the interaction between observations of a single variable and their corresponding sampling times, thereby overlooking the sparse nature of ISMTS and failing to fully utilize real observations. To address this limitation, we generate fixed-dimensional representations at each query (Q) timestamp by using all real sampling time timestamps and their corresponding observations as keys (K) and values (V). We further introduce a correlation matrix $\boldsymbol{C}$ constructed using the frequency information among variables. This matrix quantifies the similarity between different variables, capturing latent structural information that allows for the aggregation of representations across similar variables, thereby enhancing the utilization of available temporal data and improving representation quality.

The overall workflow of this module is illustrated in Figure 2(b). Given ISTS data $\boldsymbol{X}^{(l)}$ as input, the CorrNet encoder CorrE($\cdot$) generates multi-time attention embeddings that incorporate both intra-variable and inter-variable relationships, as detailed in the following computation process.

$$\begin{aligned} &\mathrm{CorrE}(\boldsymbol{Q}_T^{(l)}, \boldsymbol{K}_T^{(l)}, \boldsymbol{X}^{(l)}) = \boldsymbol{A}_T^{(l)} \boldsymbol{X}^{(l)} \boldsymbol{C}, \\ &\boldsymbol{A}_T^{(l)} = \mathrm{softmax}(\boldsymbol{Q}_T^{(l)} \boldsymbol{K}_T^{(l)} / d_r), \end{aligned} \tag{3}$$

in which $\boldsymbol{A}_T^{(l)}$ is computed based on the temporal attention mechanism. The query matrices $\boldsymbol{Q}_T^{(l)}$ and the key-value matrix $\boldsymbol{K}_T^{(l)}$ are the time embedding matrices. Since, for a given query, the relevant variables should receive higher weights to provide more valuable information, different input dimensions are assigned distinct temporal embedding weights via the correlation matrix $\boldsymbol{C}$, providing extra information on the correlation between multiple variables.

Because the continuous functions defined in the CorrE module are incompatible with neural network architectures designed for fixed-dimensional vectors, we follow the approach in [1] to discretize the output at a predefined set of reference time points, $\boldsymbol{\tau}^{(l)} = [\tau_1^{(l)}, \ldots, \tau_{U^{(l)}}^{(l)}]$, thereby generating the final representation. This process converts the continuous output into a fixed-dimension vector, making it suitable for subsequent neural network processing. Accordingly, we have $\boldsymbol{Q}_T^{(l)} = \mathbf{TE}(\boldsymbol{\tau}^{(l)})$ and $\boldsymbol{K}_T^{(l)} = \mathbf{TE}(\boldsymbol{t}^{(l)})$ with learnable parameters.

Correlation matrix extraction. The correlation matrix is essential for deriving reliable and consistent correlations between multiple variables within ISMTS, which must be robust to the inherent challenges of variable sampling rates and inconsistent observation counts at each timestamp in ISMTS. Most existing distance measures, such as Euclidean distance, dynamic time warping (DTW) [47], and optimal transport/Wasserstein distance [48], risk generating spurious correlations in the context of ISMTS. This is due to their dependence on the presence of both data points for the similarity measurement, and the potential for imputation to introduce unreliable information before calculating similarity, which will be explored further in Subsection 4.6 of our experiments.

At an impasse, the Lomb-Scargle periodogram (LSP) [49, 50] provides enlightenment to address this issue. LSP is a well-known algorithm for generating a power spectrum and detecting the periodic component in irregularly sampled time series. It extends the Fourier periodogram approach to accommodate irregularly sampled scenarios [51], eliminating the need for interpolation or imputation. This makes LSP a great tool for simplifying ISMTS analysis. Compared to existing methods that measure similarity between discrete raw observations, LSP-DTW, an implicit continuous method, utilizes inherent periodic characteristics and provides global information to measure the similarity.

As demonstrated in Figure 2(c), we first convert the ISMTS into the frequency domain using LSP and then apply DTW to evaluate the distance between variables. The correlation between $\boldsymbol{X}_{d_i}$ and $\boldsymbol{X}_{d_j}$ is

$$c_{d_i d_j} = \mathrm{DTW}\left(\mathrm{LSP}(\boldsymbol{X}_{d_i}), \mathrm{LSP}(\boldsymbol{X}_{d_j})\right) = \min_{\pi} \sum\nolimits_{(m,n)\in\pi} \left(\mathrm{LSP}(\boldsymbol{X}_{d_i})[m] - \mathrm{LSP}(\boldsymbol{X}_{d_j})[n]\right)^2, \tag{4}$$

where π is the search path of DTW. We calculate the correlation matrix $\boldsymbol{C}$ by iteratively performing the aforementioned step for an instance. Notably, we compute the correlation matrix using LSP-DTW only once per instance, without iteratively applying it, and it is not calculated in model training or inference.

CorrNet decoder and reconstruction loss. CorrD is a simplified version of CorrE without the correlation matrix, intending to reconstruct the input sequence based on the learned representation. Its computation is defined as follows:

$$\begin{aligned} &\mathrm{CorrD}(\boldsymbol{Q}_T'^{(l)}, \boldsymbol{K}_T'^{(l)}, \boldsymbol{R}^{(l)}) = \boldsymbol{A}_T'^{(l)} \boldsymbol{R}^{(l)}, \\ &\boldsymbol{A}_T'^{(l)} = \mathrm{softmax}(\boldsymbol{Q}_T'^{(l)} \boldsymbol{K}_T'^{(l)} / d_r), \end{aligned} \tag{5}$$

where $\boldsymbol{Q}_T'^{(l)} = \mathbf{TE}(\boldsymbol{t}^{(l)})$ and $\boldsymbol{K}_T'^{(l)} = \mathbf{TE}(\boldsymbol{\tau}^{(l)})$ with learnable parameters.

The reconstruction loss is computed as the mean squared error (MSE) between the reconstructed values and the original values at the masked time points. For the nth sample, the loss function is expressed as

$$\ell_{\mathrm{recon}}^{(l)} = \|\boldsymbol{M}^{(l)} \odot (\hat{\boldsymbol{X}}^{(l)} - \boldsymbol{X}^{(l)})\|_2^2. \tag{6}$$

By aggregating over N samples, the overall reconstruction loss for the lth layer is obtained as $\mathcal{L}_{\mathrm{recon}}^{(l)}$.

3.4 Rectification strategy across scales

Following the principle that adjacent scales exhibit similar representations and coarse-grained scales contain more long-term information, the rectification strategy is a key component of our MuSiCNet framework. While the coarse-grained series ignores detailed variations for high-frequency signals and focuses on much clearer broad-view temporal information, the fine-grained series retains detailed variations for frequently sampled series. As a result, iteratively using coarse-grained information for fine-grained series as a strong structural prior can benefit ISMTS learning.

Therefore, we implement a dual rectification strategy across adjacent scales to enhance representation learning. First, the reconstruction result at scale l is designed to align closely with the result at the $(l-1)$th scale, which means the reconstruction results at scale $(l-1)$ can be used to adjust the results at scale l using MSE,

$$\ell_{\mathrm{adj}}^{(l)} = \|\mathrm{Avg}_l(\hat{\boldsymbol{X}}^{(l)}) - \hat{\boldsymbol{X}}^{(l-1)}\|_2^2. \tag{7}$$

By calculating N times, the overall reconstruction loss for the lth layer is obtained as $\mathcal{L}_{\mathrm{adj}}^{(l)}$.

Second, contrastive learning is leveraged to ensure coherence between adjacent scales. Pulling these two representations between adjacent scales together and pushing other representations within the batch $\mathcal{B}$ apart, not only facilitates the learning of within-scale representations but also enhances the consistency of cross-scale representations. Taking into consideration that the dimensions of $\boldsymbol{R}^{(l)}$ and $\boldsymbol{R}^{(l-1)}$ are different, we employ a gated recurrent unit (GRU) as a decoder to uniform dimension as $\boldsymbol{h}^{(l)}$ and $\boldsymbol{h}^{(l-1)}$ before contrastive learning.

$$\ell_{\mathrm{cons}}^{(l)} = -\sum\nolimits_{i=1}^{|\mathcal{B}|} \log \frac{\exp(\boldsymbol{h}_i^{(l)} \cdot \boldsymbol{h}_i^{(l-1)})}{\sum_{j=1}^{|\mathcal{B}|} (\exp(\boldsymbol{h}_i^{(l)} \cdot \boldsymbol{h}_j^{(l-1)}) + \mathbb{I}_{[i\neq j]} \exp(\boldsymbol{h}_i^{(l)} \cdot \boldsymbol{h}_j^{(l)}))}, \tag{8}$$

where the $\mathbb{I}$ is the indicator function. After computing this over all samples in the batch, we obtain the overall contrastive loss $\mathcal{L}_{\mathrm{cons}}^{(l)}$ for the lth layer.

The advantage of the two operations lies in their ability to ensure a consistent and accurate representation of the data at different scales. This strategy significantly improves the model's ability to learn representations from ISMTS data, which is essential for tasks requiring detailed and accurate time series analysis. Last but not least, this method ensures that the model remains robust and effective even when dealing with data at varying scales, making it versatile for diverse applications.

3.5 ISMTS analysis tasks

The overall loss is defined as (9), incorporating an optional task-specific loss component,

$$\mathcal{L} = \frac{1}{L}\sum_{l=1}^{L}\mathcal{L}_{\text{recon}}^{(l)} + \frac{1}{L-1}\sum_{l=2}^{L}(\lambda_1\mathcal{L}_{\text{adj}}^{(l)} + \lambda_2\mathcal{L}_{\text{cons}}^{(l)}). \tag{9}$$

Supervised learning. We augment the encoder-decoder CorrNet by integrating a supervised learning component that utilizes the latent representations for feature extraction. In this paper, we specifically concentrate on classification tasks as a representative example of supervised learning. $\mathcal{Y} = \{1, 2, \dots\}$ denotes the label space. The loss function is

$$\mathcal{L}_{\text{cls}} = \frac{1}{|Y|}\sum_{y=1}^{|Y|}\frac{1}{n^y}\sum_{i=1}^{n^y}\ell_{CE}(\text{CLS}(\boldsymbol{h}_i^{(L)}), y), \tag{10}$$

where $|Y|$ denotes the number of classes, n^y denotes the number of samples in yth class, $\text{CLS}(\cdot)$ denotes the projection head for classification, and $\ell_{CE}(\cdot)$ denotes the cross-entropy loss.

Unsupervised learning. For our unsupervised learning example, we choose interpolation, forecasting, and anomaly detection. The loss function for interpolation and anomaly detection is defined as

$$\mathcal{L}_{\text{int}} = \mathcal{L}_{\text{anom}} = \sum_{n=1}^{N}\|\boldsymbol{M}^{(L)} \odot ((\hat{\boldsymbol{X}}_{\text{recon}}^{(L)})_n - \boldsymbol{X}_n^{(L)})\|_2^2. \tag{11}$$

This equation essentially represents the reconstruction outcome at the finest scale as $\mathcal{L}_{\text{recon}}^{(L)}$ in (6), making the interpolation task fit seamlessly into our model with minimal modifications. Therefore, it is unnecessary to incorporate an additional loss function into our overall loss function (9).

While the loss function for forecasting is defined as

$$\mathcal{L}_{\text{fore}} = \sum_{n=1}^{N}\|(\boldsymbol{M}_{\text{fore}})_n \odot ((\hat{\boldsymbol{X}}_{\text{fore}}^{(L)})_n - (\boldsymbol{X}_{\text{fore}})_n)\|_2^2. \tag{12}$$

As observations might be missing in the groundtruth data, to measure forecasting accuracy, we average an element-wise loss function $\mathcal{L}_{\text{fore}}$ over only valid values using $(\boldsymbol{M}_{\text{fore}})_n$.

3.6 Pseudo code for MuSiCNet

The pseudo code is provided in Algorithm 1 using classification as an example. The interpolation and anomaly detection task can be obtained by removing the projection head f_{cls} and the classification loss term $\mathcal{L}_{\text{cls}}$ from the total loss in line #20. For forecasting tasks, the projection head will be replaced with f_{fore} and the task loss will be changed to $\mathcal{L}_{\text{fore}}$ as in (12).

4 Experiment

In this section, we demonstrate the effectiveness of the MuSiCNet framework for time series classification, interpolation, and forecasting. Notably, for each dataset, the window size is initially set to 1/4 of the time series length and then halved iteratively until the majority of the windows contain at least one observation. Our results are based on the mean and standard deviation values computed over 5 independent runs. Bold indicates the best performer, while underline represents the second best.

4.1 Summary of benchmarks

We adopt the data processing approach used in Raindrop [16] for the classification task, mTANs [1] for the interpolation task, and GraFITi [4] for the forecasting task. The aforementioned processing methods serve as the usual setup, which our method also follows for fair comparison. However, it's important to note that we do not incorporate static attribute vectors (such as age, gender, time from hospital to ICU admission, ICU type, and length of stay in ICU) in our processing. This decision is based on the fact that our model, MuSiCNet, is not specifically designed for clinical datasets. Instead, it is designed as a versatile, general model capable of handling various types of datasets, which may not always include such static vectors. The detailed information of baselines is in Table 1.

Algorithm 1 MuSiCNet algorithm for classification as an example.

```
Input: Training set $\boldsymbol{\mathcal{X}}$, the corresponding timestamps $\boldsymbol{\mathcal{T}}$, the number of scale layers $L$, max reference point $U^{(L)}$, hyper-parameters $\lambda_1$, $\lambda_2$, $\lambda_3$.
Parameters: Encoder model $f_{\mathrm{CorrE}}$, decoder model $f_{\mathrm{CorrD}}$, GRU model $f_{\mathrm{GRU}}$, projection head $f_{\mathrm{cls}}$.
Output: Encoder model $f_{\mathrm{CorrE}}$, GRU model $f_{\mathrm{GRU}}$, projection head $f_{\mathrm{cls}}$.
$\boldsymbol{C} \leftarrow$ (4) with $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{T}}$;
for $\boldsymbol{X}$, $\boldsymbol{t}$ in $\boldsymbol{\mathcal{X}}$, $\boldsymbol{\mathcal{T}}$ do
    $\left\{\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(L)}\right\} \leftarrow \left\{\boldsymbol{M}^{(l)} \odot \mathrm{Avg}_l\left(\boldsymbol{X}\right)\right\}_{l=1}^{L-1} \cup \{\boldsymbol{M}^{(L)} \odot \boldsymbol{X}\}$;
    $\{\boldsymbol{t}^{(1)}, \ldots, \boldsymbol{t}^{(L)}\} \leftarrow \left\{\boldsymbol{M}^{(l)} \odot \mathrm{Avg}_l\left(\boldsymbol{t}\right)\right\}_{l=1}^{L-1} \cup \{\boldsymbol{M}^{(L)} \odot \boldsymbol{t}\}$;
    $\ell_{\mathrm{recon}} \leftarrow 0$;
    for $l \leftarrow 1$ to $L$ do
        $U^{(l)} \leftarrow U^{(L)}/2^{(L-l)}$;
        $\boldsymbol{\tau}^{(l)} \leftarrow \left\{\frac{k}{U^{(l)}} \cdot T \;\middle|\; k = 0, 1, \ldots, U^{(l)} - 1\right\}$;
        $\boldsymbol{r}^{(l)} \leftarrow f_{\mathrm{CorrE}}\left(X^{(L)}, \boldsymbol{C}, \boldsymbol{\tau}^{(l)}\right)$;
        $\boldsymbol{h}^{(l)} \leftarrow f_{\mathrm{GRU}}\left(\boldsymbol{R}^{(l)}\right)$;
        $\hat{\boldsymbol{X}}_{\mathrm{recon}}^{(l)} \leftarrow f_{\mathrm{CorrD}}\left(\boldsymbol{R}^{(l)}, \boldsymbol{t}^{(l)}\right)$;
        $\ell_{\mathrm{recon}} \leftarrow \ell_{\mathrm{recon}} +$ (6) with $\boldsymbol{X}^{(l)}$ and $\hat{\boldsymbol{X}}^{(l)}$;
    end for
    $\ell_{\mathrm{adj}}, \ell_{\mathrm{cons}} \leftarrow 0, 0$;
    for $l \leftarrow 2$ to $L$ do
        $\ell_{\mathrm{adj}} \leftarrow \ell_{\mathrm{adj}} +$ (7) with $\hat{\boldsymbol{X}}^{(l-1)}$ and $\hat{\boldsymbol{X}}^{(l)}$;
        $\ell_{\mathrm{cons}} \leftarrow \ell_{\mathrm{cons}} +$ (8) with $\boldsymbol{h}^{(l-1)}$ and $\boldsymbol{h}^{(l)}$;
    end for
    $\mathcal{L}_{\mathrm{cls}} \leftarrow$ (10) with $\boldsymbol{h}^{(L)}$;
    $\mathcal{L}_{\mathrm{overall}} \leftarrow \frac{1}{L}\mathcal{L}_{\mathrm{recon}} + \frac{\lambda_1}{L-1}\mathcal{L}_{\mathrm{adj}} + \frac{\lambda_2}{L-1}\mathcal{L}_{\mathrm{cons}} + \lambda_3\mathcal{L}_{\mathrm{cls}}$;
    Update overall network parameters;
end for
```

Table 1 Statistics of the ISMTS datasets used in our experiments. "#Avg. obs." denotes the average number of observations for each sample.

Tasks	Datasets	#Samples	#Variables	#Avg. obs.	#Classes	Imbalanced	Missing ratio (%)
Classification	P12	11988	36	233	2	True	88.4
	P19	38803	34	401	2	True	94.9
	PAM	5333	17	4048	8	False	60.0
Interpolation	PhysioNet	4000	37	2880	–	–	78.0
Forecasting	USHCN	1100	5	263	–	–	77.9
	MIMIC-III	21000	96	274	–	–	94.2
	PhysioNet12	5333	37	130	–	–	85.7
Anomaly Detection	MSL	1	55	32657	–	–	30.0
	PSM	1	25	74188	–	–	30.0
	SMD	1	38	396706	–	–	30.0
	SWAP	1	25	75702	–	–	30.0

4.2 Time series classification

Datasets and experimental settings. We use real-world datasets from the healthcare and human activity domains to evaluate classification performance. (1) P12 [52] records temporal measurements of 36 sensors of 11988 patients in the first 48-hour stay in ICU, with a missing ratio of 88.4%. (2) P19 [53], with a missing ratio up to 94.9%, includes 38803 patients that are monitored by 34 sensors. (3) PAM [54] contains 5333 segments from 8 activities of daily living that are measured by 17 sensors, and the missing ratio is 60.0%. Importantly, P19 and P12 are imbalanced binary label datasets.

Here, we follow the common setup by randomly splitting the dataset into training (80%), validation (10%), and test (10%) sets, and the indices of these splits are fixed across all methods. Consistent with prior research, we evaluate the performance of our framework on classification tasks using the area under the receiver operating characteristic curve (AUROC) and the area under the precision-recall curve (AUPRC) for the P12 and P19 datasets,

Table 2 Comparison with the baseline methods on the ISMTS classification task.

Method	P12		P19		PAM			
	AUROC	AUPRC	AUROC	AUPRC	Accuracy	Precision	Recall	F1 score
Transformer	83.3 ± 0.7	47.9 ± 3.6	80.7 ± 3.8	42.7 ± 7.7	83.5 ± 1.5	84.8 ± 1.5	86.0 ± 1.2	85.0 ± 1.3
Trans-Mean	82.6 ± 2.0	46.3 ± 4.0	83.7 ± 1.8	45.8 ± 3.2	83.7 ± 2.3	84.9 ± 2.6	86.4 ± 2.1	85.1 ± 2.4
GRU-D	81.9 ± 2.1	46.1 ± 4.7	83.9 ± 1.7	46.9 ± 2.1	83.3 ± 1.6	84.6 ± 1.2	85.2 ± 1.6	84.8 ± 1.2
SeFT	73.9 ± 2.5	31.1 ± 4.1	81.2 ± 2.3	41.9 ± 3.1	67.1 ± 2.2	70.0 ± 2.4	68.2 ± 1.5	68.5 ± 1.8
mTAND	84.2 ± 0.8	48.2 ± 3.4	84.4 ± 1.3	50.6 ± 2.0	74.6 ± 4.3	74.3 ± 4.0	79.5 ± 2.8	76.8 ± 3.4
IP-Net	82.6 ± 1.4	47.6 ± 3.1	84.6 ± 1.3	38.1 ± 3.7	74.3 ± 3.8	75.6 ± 2.1	77.9 ± 2.2	76.6 ± 2.8
DGM^2-O	84.4 ± 1.6	47.3 ± 3.6	86.7 ± 3.4	44.7 ± 11.7	82.4 ± 2.3	85.2 ± 1.2	83.9 ± 2.3	84.3 ± 1.8
MTGNN	74.4 ± 6.7	35.5 ± 6.0	81.9 ± 6.2	39.9 ± 8.9	83.4 ± 1.9	85.2 ± 1.7	86.1 ± 1.9	85.9 ± 2.4
Raindrop	82.8 ± 1.7	44.0 ± 3.0	87.0 ± 2.3	51.8 ± 5.5	88.5 ± 1.5	89.9 ± 1.5	89.9 ± 0.6	89.8 ± 1.0
Warpformer	83.4 ± 0.9	47.2 ± 3.7	88.8 ± 1.7	55.2 ± 3.9	94.3 ± 0.6	95.8 ± 0.8	94.8 ± 1.0	95.2 ± 0.6
ViTST	85.1 ± 0.8	51.1 ± 4.1	**89.2** ± 2.0	**53.1** ± 3.4	95.8 ± 1.3	96.2 ± 1.3	96.1 ± 1.1	96.5 ± 1.2
FPT	84.8 ± 1.1	50.7 ± 3.0	87.3 ± 2.9	51.6 ± 3.6	94.0 ± 1.4	95.3 ± 0.9	94.7 ± 1.1	94.9 ± 1.1
Time-LLM	84.4 ± 1.8	50.2 ± 1.6	85.1 ± 2.6	50.1 ± 3.4	93.4 ± 1.2	94.2 ± 1.3	94.7 ± 1.0	94.4 ± 1.1
PrimeNet	84.9 ± 0.6	49.8 ± 2.7	84.4 ± 1.3	39.7 ± 3.1	95.3 ± 0.5	96.1 ± 0.3	95.5 ± 0.6	95.7 ± 0.4
MuSiCNet	**86.1** ± 0.4	**54.1** ± 2.2	86.8 ± 1.4	45.4 ± 2.7	**96.3** ± 0.7	**96.9** ± 0.6	**96.9** ± 0.5	**96.8** ± 0.5

given their imbalanced nature. For the nearly balanced PAM dataset, we employ Accuracy, Precision, Recall, and F1 Score. For all of the above metrics, higher results indicate better performance.

Main results. We compare MuSiCNet against 10 state-of-the-art methods for classification, including Transformer [45], Trans-Mean, GRU-D [2], SeFT [26], mTAND [1], IP-Net [55], DGM^2-O [17], MTGNN [56], Raindrop [16], and ViTST [27], Warpformer [39]. In addition, we include comparisons with pre-trained language model (PLM)-based approaches, such as FPT [57] and Time-LLM [58], as well as the CRL-based ISMTS model, PrimeNet [28]. Since mTAND is proven superior over various RNN-based models, such as RNNImpute [2], Phased-LSTM [59], and ODE-based models like LATENT-ODE and ODE-RNN [60], we primarily focus on mTAND and omit detailed results for those earlier models.

As indicated in Table 2, MuSiCNet demonstrates good performance across three benchmark datasets, underscoring its effectiveness in typical time series classification tasks. Notably, in binary classification scenarios, MuSiCNet surpasses the best-performing baselines on the P12 dataset by an average of 1.0% in AUROC and 3.0% in AUPRC. For the P19 dataset, while our performance is competitive, MuSiCNet stands out due to its lower time and space complexity compared to ViTST. ViTST converts 1D time series into 2D images, potentially leading to significant space inefficiencies due to the introduction of extensive blank areas, especially problematic in ISMTS. In the more complex task of 8-class classification on the PAM dataset, MuSiCNet surpasses current methodologies, achieving a 0.5% improvement in accuracy and a 0.7% increase in precision.

Notably, the consistently low standard deviation in our results indicates that MuSiCNet is a reliable model. Its performance remains steady across varying data samples and initial conditions, suggesting strong potential for generalizing well to new, unseen data. This stability and predictability in performance enhance the confidence in the model's predictions, which is particularly crucial in sensitive areas such as medical diagnosis in clinical settings.

4.3 Time series interpolation

Datasets and experimental settings. PhysioNet [61] contains 37 variables recorded during the first 48 h after ICU admission. We use all 8000 instances for the interpolation task, where the overall missing ratio is 78.0%.

We randomly split the dataset into a training set, encompassing 80% of the instances, and a test set, comprising the remaining 20% of instances. Additionally, 20% of the training data is reserved for validation purposes. The performance evaluation is conducted using MSE, where lower values indicate better performance.

Main results. For the interpolation task, we compare it with RNN-VAE, L-ODE-RNN [60], L-ODE-ODE [21], mTAND-full, NIERT [62], and TimeCHEAT [63].

For the interpolation task, models are trained to predict or reconstruct values for the entire dataset based on a selected subset of available points. Experiments are conducted with varying observation levels, ranging from 50% to 90% of observed points. During test time, models utilize the observed points to infer values at all time points in each test instance.

As illustrated in Table 3, MuSiCNet demonstrates superior performance, highlighting its effectiveness in time series interpolation. This can be attributed to its ability to interpolate progressively from coarse to fine, aligning

Table 3 Comparison with the baseline methods on the ISMTS interpolation task on PhysioNet, reported as MSE $\times 10^{-3}$.

Method	50%	60%	70%	80%	90%
RNN-VAE	13.418 ± 0.008	12.594 ± 0.004	11.887 ± 0.005	11.133 ± 0.007	11.470 ± 0.006
L-ODE-RNN	8.132 ± 0.020	8.140 ± 0.018	8.171 ± 0.030	8.143 ± 0.025	8.402 ± 0.022
L-ODE	6.721 ± 0.109	6.816 ± 0.045	6.798 ± 0.143	6.850 ± 0.066	7.142 ± 0.066
mTAND-Full	4.139 ± 0.029	4.018 ± 0.048	4.157 ± 0.053	4.410 ± 0.149	4.798 ± 0.036
NIERT	2.868 ± 0.021	2.794 ± 0.030	2.656 ± 0.041	2.577 ± 0.086	2.709 ± 0.157
TimeCHEAT	4.185 ± 0.030	3.981 ± 0.016	3.657 ± 0.022	3.642 ± 0.036	3.686 ± 0.009
MuSiCNet	**0.918** ± 0.025	**0.919** ± 0.064	**0.938** ± 0.014	**0.992** ± 0.008	**0.965** ± 0.008

Table 4 Experimental results for forecasting the next three time steps. – indicates no published results.

Method	USHCN	MIMIC-III	PhysioNet12
DLinear+	0.347 ± 0.065	0.691 ± 0.016	0.380 ± 0.001
NLinear+	0.452 ± 0.101	0.726 ± 0.019	0.382 ± 0.001
Informer+	0.320 ± 0.047	0.512 ± 0.064	0.347 ± 0.001
FedFormer+	2.990 ± 0.476	1.100 ± 0.059	0.455 ± 0.004
NeuralODE-VAE	0.960 ± 0.110	0.890 ± 0.010	–
GRUSimple	0.750 ± 0.120	0.820 ± 0.050	–
GRU-D	0.530 ± 0.060	0.790 ± 0.060	–
T-LSTM	0.590 ± 0.110	0.620 ± 0.050	–
mTAND	0.300 ± 0.038	0.540 ± 0.036	0.315 ± 0.002
GRU-ODE-Bayes	0.430 ± 0.070	0.480 ± 0.480	0.329 ± 0.004
Neural Flow	0.414 ± 0.102	0.490 ± 0.004	0.326 ± 0.004
CRU	0.290 ± 0.060	0.592 ± 0.049	0.379 ± 0.003
GraFITi	0.272 ± 0.047	**0.396** ± 0.030	**0.286** ± 0.001
MuSiCNet	**0.268** ± 0.038	0.475 ± 0.031	0.312 ± 0.000

with the intuition of multi-resolution signal approximation [64].

4.4 Time series forecasting

Datasets and experimental settings. (1) USHCN [65] is an artificially preprocessing dataset containing measurements of 5 variables from 1280 weather stations in the USA. The missing ratio is 78.0%. (2) MIMIC-III [66] is a dataset that rounds the recorded observations into 96 variables, 30-minute intervals, and only uses observations from the 48 h after admission. The missing ratio is 94.2%. (3) PhysioNet12 [61] comprises medical records from 12000 ICU patients. It includes measurements of 37 vital signs recorded during the first 48 h of admission, and the missing ratio is 80.4%. We use MSE to measure forecasting performance, with lower values indicating better performance.

Main results. We evaluate our approach against a range of ISMTS forecasting models, including the graph-based method GraFITi [4]; ODE- and RNN-based models such as GRU-ODE-Bayes [19], Neural Flows [67], CRU [20], NeuralODE-VAE [60], as well as GRUSimple, GRU-D, and TLSTM [68]. Additionally, we compare our results with attention-based models, including mTAND. Moreover, it is of significant interest to assess the performance of well-established multivariate time series forecasting models under the ISMTS. To achieve this, we introduce missing value indicators as additional channels, thereby enabling the joint processing of both the time series and the corresponding missing data information. Therefore, we compare our method with variants of Informer [69], Fedformer [70], DLinear, and NLinear [71], denoted as Informer+, Fedformer+, DLinear+, and NLinear+, respectively. This experiment is conducted following the setting of GraFITi, where for the USHCN dataset, the model observes for the first 3 years and forecasts the next 3 time steps, and for other datasets, the model observes the first 36 h in the series and predicts the next 3 time steps.

As shown in Table 4, MuSiCNet consistently achieves competitive performance across all datasets, maintaining accuracy within the top two among baseline models. While GraFITi excels by explicitly modeling the relationship between observation and prediction points, making it superior in certain scenarios, MuSiCNet remains competitive without imposing priors for any specific task.

Table 5 Experimental results of anomaly detection.

Method	MSL	PSM	SMD	SWAP
Transformer	78.68	76.07	79.56	69.70
DLinear	**84.88**	93.55	77.10	69.26
mTAND	82.12	92.90	78.83	69.10
t-patchGNN	74.34	92.45	59.61	65.72
MuSiCNet	82.70	**94.09**	**80.12**	**69.87**

4.5 Time series anomaly detection

Datasets and experimental settings. We use 4 widely-used anomaly detection benchmarks from the service monitoring and space and earth exploration applications. (1) SMD [72] is a five-week dataset collected from a large Internet company, featuring 38 dimensions. (2) PSM [73] comprises data collected internally from multiple application server nodes at eBay, with 26 dimensions. (3) Both MSL (Mars Science Laboratory rover) and SMAP (Soil Moisture Active Passive satellite) are publicly available NASA datasets [74] with 55 and 25 dimensions respectively, containing telemetry anomaly data derived from spacecraft monitoring systems as reported in incident surprise anomaly (ISA) reports.

Our experiments adhere to the configuration described in Anomaly Transformer [75] and TimesNet [76]. It is important to note that, due to the lack of dedicated anomaly detection benchmarks for ISMTS, research in this area is limited, and the datasets typically employed are not naturally sourced ISMTS data but are generated by masking datasets. In our study, we follow this practice by applying a random 30% mask to the selected datasets. We partition the dataset into consecutive, non-overlapping segments using a sliding window approach. Prior studies have utilized reconstruction as a classical task for unsupervised point-wise representation learning, where the reconstruction error naturally serves as an anomaly criterion. We evaluate the efficacy of anomaly detection using the F1 score, where higher values indicate better performance.

Main results. Due to the lack of well-established methods specifically designed for anomaly detection in ISMTS, we selected two representative models originally developed for RSMTS, including the canonical Transformer [45] and DLinear [77] as baselines. We also conducted experiments with the influential ISMTS method mTAND and the recently proposed t-patchGNN [78].

As shown in Table 5, MuSiCNet consistently achieves competitive or superior performance across all four datasets. In particular, it surpasses both the classical and state-of-the-art ISMTS models, and performs comparably with the DLinear, the SOTA RSMTS method in 2023, in most cases. These results highlight MuSiCNet's strong capability in anomaly detection without the need for task-specific design.

4.6 Correlation results

In this subsection, we focus on validating the necessity, effectiveness, and efficiency of the correlation matrix in the classification task as an example.

First, we verify the necessity of the correlation matrix using results from all classification datasets in Table 6. Removing the correlation matrix, i.e., w/o Corr (line 3) led to performance drops across all datasets, with P19 showing the largest decline due to its 94.9% missing rate. This highlights the importance of capturing inter-series relationships in irregularly sampled time series, making the correlation matrix essential. Replacing the designed correlation matrix with a learnable one (line 4) also worsened performance, indicating that learning inter-series relationships purely from the network remains highly challenging and that specialized correlation designs are needed.

Second, we evaluate LSP-DTW against other correlation calculation methods (I-GAK [79], I-DTW [47]) on the P12 dataset to verify the effectiveness. Interpolation-based methods (I-GAK, I-DTW) distort correlations, leading to unreliable results as seen in Figure 3. I-GAK shows fictitious correlations based on observation rates, while I-DTW presents uniformly positive correlations, neither of which captures true data characteristics. In contrast, LSP-DTW accurately identifies correlations, verified by Table 7, where it outperforms all baselines, demonstrating the importance of appropriate correlation modeling.

Lastly, we report the computation time for the correlation matrix. The LSP-DTW based correlation matrix is computed per instance in parallel, with acceptable runtimes (0.137 s for P12, 0.127 s for P19, 0.049 s for PAM). It is calculated once, making it efficient for the entire learning process.

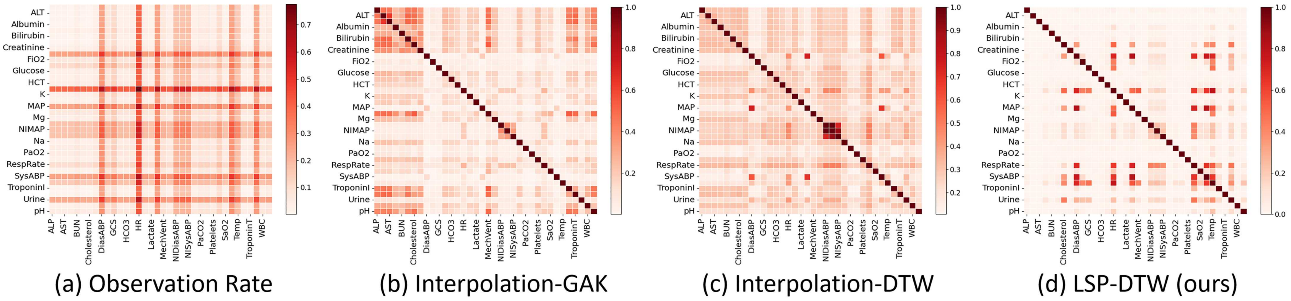

Figure 3 (Color online) Visualization of various methods to extract the correlation matrix from P12 dataset. The darker the color, the more similar the relationship. (a) denotes the average pairwise observation rate (i.e., 1 minus the missing rate), and (b)–(d) denote different correlation matrices.

Table 6 Classification performance of MuSiCNet to verify the necessity of the correlation matrix.

Method	P12		P19		PAM			
	AUROC	AUPRC	AUROC	AUPRC	Accuracy	Precision	Recall	F1 score
w/o Corr	85.5 ± 0.3	53.0 ± 2.1	83.6 ± 0.8	36.7 ± 2.1	95.7 ± 0.9	96.2 ± 0.51	96.5 ± 0.2	96.3 ± 0.3
Learnable Corr	85.7 ± 0.4	53.0 ± 2.0	83.9 ± 0.7	35.8 ± 2.7	96.1 ± 0.5	96.7 ± 0.38	96.5 ± 0.7	96.6 ± 0.5
MuSiCNet	**86.1** ± 0.4	**54.1** ± 2.2	**86.8** ± 0.4	**54.1** ± 2.2	**96.3** ± 0.7	**96.9** ± 0.6	**96.9** ± 0.5	**96.8** ± 0.5

Table 7 Classification performance of MuSiCNet with different correlation matrices on three datasets to verify the effectiveness.

Method	P12		P19		PAM			
	AUROC	AUPRC	AUROC	AUPRC	Accuracy	Precision	Recall	F1 score
Ones	66.7 ± 2.2	25.2 ± 0.3	81.5 ± 2.7	41.0 ± 4.7	63.3 ± 1.7	66.3 ± 1.9	66.7 ± 2.5	65.6 ± 0.3
Rand	84.7 ± 0.8	52.2 ± 3.2	82.3 ± 2.0	34.8 ± 1.9	95.1 ± 0.5	95.8 ± 0.3	95.5 ± 0.7	95.6 ± 0.3
Diag	84.2 ± 0.8	48.2 ± 3.4	83.4 ± 1.1	37.1 ± 2.3	95.5 ± 0.4	95.9 ± 0.3	95.8 ± 0.3	95.8 ± 0.3
I-GAK	85.1 ± 0.6	52.8 ± 3.0	83.7 ± 0.9	33.6 ± 1.7	96.0 ± 0.3	96.5 ± 0.6	96.3 ± 0.4	96.2 ± 0.3
I-DTW	81.9 ± 0.6	46.9 ± 3.0	83.9 ± 1.3	34.3 ± 1.1	95.9 ± 0.5	96.3 ± 0.5	96.4 ± 0.5	96.3 ± 0.5
LSP-DTW	**86.1** ± 0.4	**54.1** ± 2.2	**86.8** ± 0.4	**54.1** ± 2.2	**96.3** ± 0.7	**96.9** ± 0.6	**96.9** ± 0.5	**96.8** ± 0.5

Table 8 Ablation studies on different strategies of MuSiCNet in classification. ✓(×) indicates that the component has (not) been applied.

Component			P12		P19		PAM			
Corr matrix	Adjustment	Contrastive	AUROC	AUPRC	AUROC	AUPRC	Accuracy	Precision	Recall	F1 score
×	×	×	84.0 ± 0.8	47.9 ± 3.4	83.5 ± 1.0	38.9 ± 2.4	95.4 ± 0.2	96.0 ± 0.4	95.9 ± 0.3	95.8 ± 0.3
✓	×	×	85.2 ± 0.6	52.6 ± 2.5	84.3 ± 0.8	37.1 ± 1.1	95.6 ± 0.4	96.5 ± 0.8	96.6 ± 0.7	96.4 ± 0.7
✓	×	✓	85.4 ± 0.4	53.0 ± 2.5	84.6 ± 0.8	37.1 ± 1.0	96.0 ± 0.8	96.4 ± 0.7	96.3 ± 0.5	96.3 ± 0.4
✓	✓	×	85.4 ± 0.6	52.9 ± 2.8	84.3 ± 0.7	37.0 ± 0.8	96.0 ± 0.8	96.8 ± 0.7	96.7 ± 0.7	96.5 ± 0.7
×	✓	✓	85.5 ± 0.3	53.0 ± 2.1	83.6 ± 0.8	36.7 ± 2.1	95.7 ± 0.9	96.2 ± 0.51	96.5 ± 0.2	96.3 ± 0.3
✓	✓	✓	**86.1** ± 0.4	**54.1** ± 2.2	**86.8** ± 0.4	**54.1** ± 2.2	**96.3** ± 0.7	**96.9** ± 0.6	**96.9** ± 0.5	**96.8** ± 0.5

4.7 Ablation analysis and efficiency evaluation

We conduct the ablation study to assess the necessity of two fundamental components of MuSiCNet: the correlation matrix and the multi-scale learning reflected in reconstruction results adjustment and contrastive learning. As shown in Table 8, the complete MuSiCNet framework, incorporating all components, achieves the best performance.

We find that the importance of the two components varies across datasets. For example, for P19, the dataset with the highest missing ratio, we effectively capture the relationships between sequences using the correlation matrix, maximizing the use of real observations and aggregating relevant information across sequences. As a result, with the correlation matrix, the classification performance on P19 only drops by around 2 percentage points compared to the original performance. However, without the correlation matrix, the classification performance on P19 significantly decreases by about 4 percentage points. In contrast, for P12, where multiple sensors record the same physiological signals, with synchronized sampling at high frequency, learning the correlation matrix does not play a decisive role. Instead, multi-scale learning proves to be more important. For PAM, with relatively low missing rates, both components are equally important. Thus, for the diverse ISMTS datasets, both components are indispensable.

To demonstrate computational complexity, taking P12 in the classification task with a batch size of 50 in

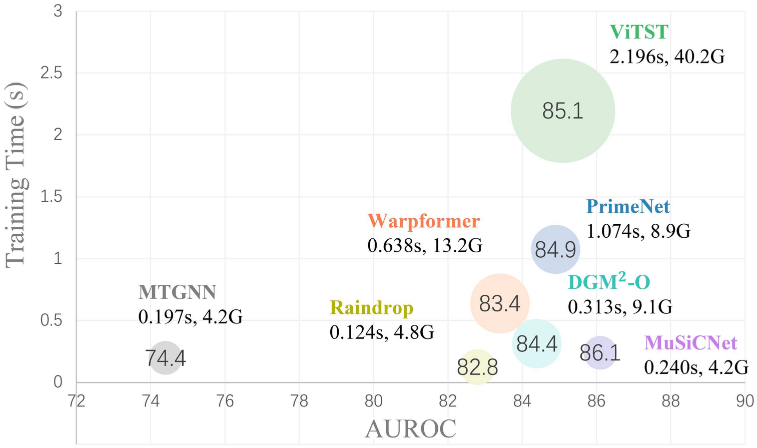

Figure 4 Efficiency comparisons in terms of training time (s) and memory usage (GB) with the latest advanced models on the P12 dataset. The closer a circle is to the bottom-right corner and the smaller its area, the higher the model's classification accuracy, with faster training speed and lower memory usage.

Figure 4 as an example, our model MuSiCNet achieves a time cost of 0.240 s per batch with 4.2 GB of memory usage. In comparison, ViTST requires 2.196 s and 40.2 GB, Raindrop uses 0.124 s and 4.8 GB, MTGNN takes 0.1967 s and 4.2 GB, and DGM^2-O needs 0.313 s and 9.1 GB. MuSiCNet demonstrates lower time complexity than most other methods and significantly lower memory usage, particularly compared to ViTST, which also performs well on classification tasks.

4.8 MuSiCNet parameters and parameter analysis

We present the training hyperparameters and model parameters here. The maximum epoch is set to 300, and the AdamW optimizer is selected as our optimizer without weight decay. By default, the learning rate is set to 1×10^{-3}, and the learning rate schedule is cosine decay for each epoch. The batch size for all datasets is set to 50, the dimension of the encoder output is set to 256, and the dimension of the hidden representations in the GRU is typically set to 50. The random masking ratio r for each scale is set to 0.1.

Due to inconsistent series lengths, we set the maximum reference point number, K, to 128 for long series, such as P12, PAM, PhysioNet, and USHCN, to 96 for PhysioNet12, and to 48 for short series, such as PAM and MIMIC-III.

Initially, the window size is set to 1/4 of the time series length and then halved iteratively until the majority of the windows contain at least one observation.

According to the observed timestamps on each dataset, the number of scale layers L is set to 6, 5, 7, 6, 8, 4, and 5 for P12, P19, PAM, PhysioNet, USHCN, MIMIC-III, and PhysioNet12, respectively. For example, in classification, for P12, the scales are 4, 8, 16, 32, 64, and raw length. For P19, the scales are $4, 8, 16, 32$ and the raw length. And for PAM, the scales are $4, 8, 16, 32, 64, 128$ and the raw length. In all mainstream tasks involved, the hyperparameters $\lambda_1, \lambda_2, \lambda_3$ are selected in $[1 \times 10^{-3}, 1 \times 10^{-2}, \ldots, 1 \times 10^{2}]$. All the models were experimented using the PyTorch library on a GeForce RTX-2080 Ti GPU.

To analyze the hyperparameter sensitivity, we conducted the experiments for λ_1, λ_2, and λ_3 with grid search. Due to the closer relationship between the hyper-parameters of the adjustment term and the contrastive learning term, i.e., λ_1 and λ_2, we jointly analyzed λ_1 and λ_2 while separately analyzing the hyper-parameter of the downstream task λ_3, as illustrated in Figures 5 and 6.

From Figure 5, in general, when $\lg \lambda_1$ and $\lg \lambda_2$ take values around 2 and -2, respectively, our MuSiCNet can perform well. Specifically, on the one hand, when the hyperparameter of the contrastive term is fixed, increasing the hyperparameter of the adjustment term, i.e., λ_1, from 1×10^{-3} to 1×10^{2} consistently improves performance. This trend highlights the critical role of the adjustment term in enhancing model stability and promoting better generalization. However, further increasing λ_1, e.g., greater than 1×10^{2}, may produce a slight performance degradation, since the model may focus too much on inter-scale consistency, leading to the collapse of the representation space. On the other hand, the contrastive term in the representation space contributes positively when its hyperparameter λ_2 lies around 1×10^{2}. This indicates that the contrastive objective effectively encourages consistency across different representational layers and enhances the expressiveness of representations at the Lth layer.

From Figure 6, we can find that our MuSiCNet becomes effective with a large λ_3. This indicates that more effective representations will be captured when utilizing downstream tasks, matching the general insight. We also

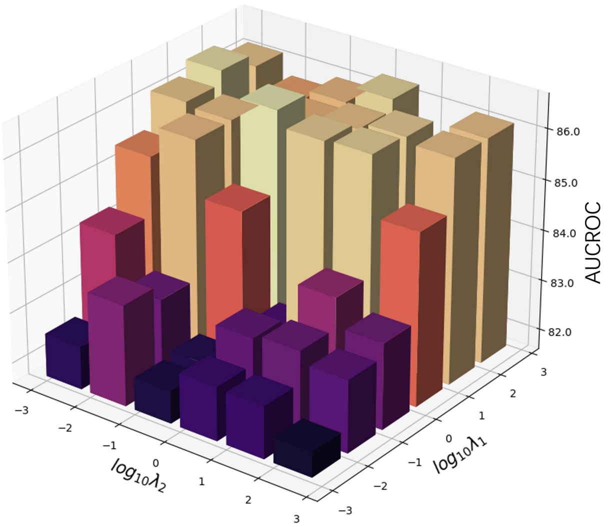

Figure 5 (Color online) AUCROC performance with varying combinations of the hyperparameter of the adjustment term λ_1 and the hyperparameter of the contrastive learning term λ_2 in the logarithmic form on P12.

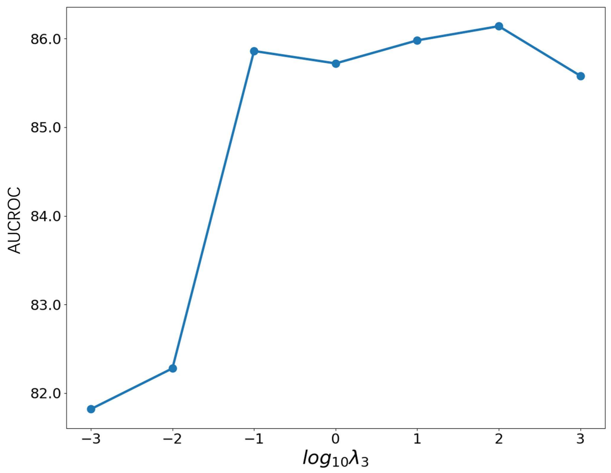

Figure 6 (Color online) AUCROC performance with varying hyperparameters of the downstream task λ_3 in logarithmic form on P12.

note that it becomes less sensitive when $\lg \lambda_3 \geqslant -1$. Its suitable range may be located in $[10, 1 \times 10^2]$.

5 Conclusion

In this study, we introduce MuSiCNet, an innovative framework designed for analyzing ISMTS datasets. MuSiCNet addresses the challenges arising from data irregularities and shows superior performance in both supervised and unsupervised tasks. We recognize that irregularities in ISMTS are inherently relative and accordingly implement multi-scale learning, a vital element of our framework. In this multi-scale approach, the contribution of extra coarse-grained and relatively regular series is important, providing comprehensive temporal insights that facilitate the analysis of finer-grained series. As another key component of MuSiCNet, CorrNet is engineered to aggregate temporal information effectively, employing time embeddings and correlation matrices calculated from both intra- and inter-series perspectives, in which we employ LSP-DTW to develop frequency correlation matrices that not only reduce the burden for similarity calculation for ISMT, but also significantly enhance inter-series information extraction.

Acknowledgements This work was supported by Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. KYCX21_0225), and National Natural Science Foundation of China (Grant Nos. 62076124, 62376126). The authors wish to thank all the donors of the original datasets, and everyone who provided feedback on this study.

References

1 Shukla S N, Marlin B. Multi-time attention networks for irregularly sampled time series. In: Proceedings of International Conference on Learning Representations, 2021
2 Che Z, Purushotham S, Cho K, et al. Recurrent neural networks for multivariate time series with missing values. Sci Rep, 2018, 8: 6085
3 Agarwal R, Sinha A, Vishwakarma A, et al. No imputation needed: a switch approach to irregularly sampled time series. ArXiv:2309.08698
4 Yalavarthi V K, Madhusudhanan K, Scholz R, et al. GraFITi: graphs for forecasting irregularly sampled time series. In: Proceedings of AAAI Conference on Artificial Intelligence, 2024. 16255–16263
5 Cai R C, Wu Y J, Huang X K, et al. Granger causal representation learning for groups of time series. Sci China Inf Sci, 2024, 67: 152103
6 Sun L, Wang Y Y, Ren Y J, et al. Path signature-based XAI-enabled network time series classification. Sci China Inf Sci, 2024, 67: 170305
7 Huang G, Ge C J, Xiong T Y, et al. Large scale air pollution prediction with deep convolutional networks. Sci China Inf Sci, 2021, 64: 192107
8 Shukla S N, Marlin B M. A survey on principles, models and methods for learning from irregularly sampled time series. ArXiv:2012.00168
9 Tashiro Y, Song J, Song Y, et al. CSDI: conditional score-based diffusion models for probabilistic time series imputation. In: Proceedings of Advances in Neural Information Processing Systems, 2021. 24804–24816
10 Chen X, Zhang C, Zhao X L, et al. Nonstationary temporal matrix factorization for multivariate time series forecasting. ArXiv:2203.10651
11 Fan J. Dynamic nonlinear matrix completion for time-varying data imputation. In: Proceedings of AAAI Conference on Artificial Intelligence, 2022. 6587–6596
12 Yoon J, Jordon J, Schaar M. Gain: missing data imputation using generative adversarial nets. In: Proceedings of International Conference on Machine Learning, Stockholm, 2018. 5689–5698
13 Camino R D, Hammerschmidt C A, State R. Improving missing data imputation with deep generative models. ArXiv:1902.10666
14 Zhang Z Y, Zhang S Q, Jiang Y, et al. LIFE: learning individual features for multivariate time series prediction with missing values. In: Proceedings of IEEE International Conference on Data Mining, Auckland, 2021. 1511–1516

15 Du W, Côté D, Liu Y. SAITS: self-attention-based imputation for time series. Expert Syst Appl, 2023, 219: 119619
16 Zhang X, Zeman M, Tsiligkaridis T, et al. Graph-guided network for irregularly sampled multivariate time series. In: Proceedings of International Conference on Learning Representations, 2021
17 Wu Y, Ni J, Cheng W, et al. Dynamic Gaussian mixture based deep generative model for robust forecasting on sparse multivariate time series. In: Proceedings of AAAI Conference on Artificial Intelligence, 2021. 651–659
18 Sun C, Li H, Song M, et al. Time pattern reconstruction for classification of irregularly sampled time series. Pattern Recogn, 2024, 147: 110075
19 De Brouwer E, Simm J, Arany A, et al. GRU-ODE-Bayes: continuous modeling of sporadically-observed time series. In: Proceedings of Advances in Neural Information Processing Systems, Vancouver, 2019. 7377–7388
20 Schirmer M, Eltayeb M, Lessmann S, et al. Modeling irregular time series with continuous recurrent units. In: Proceedings of International Conference on Machine Learning, Baltimore, 2022. 19388–19405
21 Rubanova Y, Chen R T Q, Duvenaud D K. Latent ordinary differential equations for irregularly-sampled time series. In: Proceedings of Advances in Neural Information Processing Systems, Vancouver, 2019. 5321–5331
22 Kidger P, Morrill J, Foster J, et al. Neural controlled differential equations for irregular time series. In: Proceedings of Advances in Neural Information Processing Systems, 2020. 6696–6707
23 Jhin S Y, Lee J, Jo M, et al. Exit: extrapolation and interpolation-based neural controlled differential equations for time-series classification and forecasting. In: Proceedings of ACM Web Conference, 2022. 3102–3112
24 Jin M, Zheng Y, Li Y F, et al. Multivariate time series forecasting with dynamic graph neural ODEs. IEEE Trans Knowl Data Eng, 2022, 35: 9168–9180
25 Wang J, Du W, Cao W, et al. Deep learning for multivariate time series imputation: a survey. ArXiv:2402.04059
26 Horn M, Moor M, Bock C, et al. Set functions for time series. In: Proceedings of International Conference on Machine Learning, 2020. 4353–4363
27 Li Z, Li S, Yan X. Time series as images: vision transformer for irregularly sampled time series. In: Proceedings of Advances in Neural Information Processing Systems, New Orleans, 2024
28 Chowdhury R R, Li J, Zhang X, et al. Primenet: pre-training for irregular multivariate time series. In: Proceedings of AAAI Conference on Artificial Intelligence, Washington, 2023. 7184–7192
29 Fan H, Xiong B, Mangalam K, et al. Multiscale vision transformers. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2021. 6824–6835
30 Zhang P, Dai X, Yang J, et al. Multi-scale vision longformer: a new vision transformer for high-resolution image encoding. In: Proceedings of IEEE/CVF International Conference on Computer Vision, 2021. 2998–3008
31 Nawrot P, Tworkowski S, Tyrolski M, et al. Hierarchical transformers are more efficient language models. ArXiv:2110.13711
32 Zhao Y, Luo C, Zha Z J, et al. Multi-scale group transformer for long sequence modeling in speech separation. In: Proceedings of International Conference on International Joint Conferences on Artificial Intelligence, 2021. 3251–3257
33 Shabani M A, Abdi A H, Meng L, et al. Scaleformer: iterative multi-scale refining transformers for time series forecasting. In: Proceedings of International Conference on Learning Representations, Kigali, 2022
34 Chen P, Zhang Y, Cheng Y, et al. Pathformer: multi-scale transformers with adaptive pathways for time series forecasting. In: Proceedings of International Conference on Learning Representations, Vienna, 2024
35 Liu S, Yu H, Liao C, et al. Pyraformer: low-complexity pyramidal attention for long-range time series modeling and forecasting. In: Proceedings of International Conference on Learning Representations, 2021
36 Zhao K, Guo C, Cheng Y, et al. Multiple time series forecasting with dynamic graph modeling. In: Proceedings of VLDB Endowment, Vancouver, 2023. 753–765
37 Cai W, Liang Y, Liu X, et al. Msgnet: learning multi-scale inter-series correlations for multivariate time series forecasting. In: Proceedings of AAAI Conference on Artificial Intelligence, Vancouver, 2024. 11141–11149
38 Singh B P, Deznabi I, Narasimhan B, et al. Multi-resolution networks for flexible irregular time series modeling (multi-fit). ArXiv:1905.00125
39 Zhang J, Zheng S, Cao W, et al. Warpformer: a multi-scale modeling approach for irregular clinical time series. In: Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Long Beach, 2023. 3273–3285
40 Kenton J D M W C, Toutanova L K. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2019. 4171–4186
41 Yang Z, Dai Z, Yang Y, et al. XLNet: generalized autoregressive pretraining for language understanding. In: Proceedings of Advances in Neural Information Processing Systems, Vancouver, 2019
42 He K, Chen X, Xie S, et al. Masked autoencoders are scalable vision learners. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, 2022. 16000–16009
43 Baboshina V, Lyakhov P, Lyakhova U, et al. Bidirectional encoder representation from image transformers for recognizing sunflower diseases from photographs. Computer, 2025, 49: 3
44 Xie Z, Zhang Z, Cao Y, et al. Simmim: a simple framework for masked image modeling. In: Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, 2022. 9653–9663
45 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Proceedings of Advances in Neural Information Processing Systems, Long Beach, 2017. 5998–6008
46 Yu Z, Chu X, Ma L, et al. Imputation with inter-series information from prototypes for irregular sampled time series. ArXiv:2401.07249
47 Berndt D J, Clifford J. Using dynamic time warping to find patterns in time series. In: Proceedings of International Conference on Knowledge Discovery and Data Mining, Seattle, 1994. 359–370
48 Villani C. Optimal Transport: Old and New. Berlin-Heidelberg: Springer, 2009
49 Lomb N R. Least-squares frequency analysis of unequally spaced data. Astrophys Space Sci, 1976, 39: 447–462
50 Scargle J D. Studies in astronomical time series analysis. II-Statistical aspects of spectral analysis of unevenly spaced data. Astrophys J, 1982, 263: 835–853
51 VanderPlas J T. Understanding the Lomb-Scargle periodogram. Astrophys J Suppl Ser, 2018, 236: 16
52 Goldberger A L, Amaral L A N, Glass L, et al. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. Circulation, 2000, 101: e215–e220
53 Reyna M A, Josef C S, Jeter R, et al. Early prediction of sepsis from clinical data: the PhysioNet/Computing in Cardiology Challenge 2019. Critical Care Medicine, 2020, 48: 210–217
54 Reiss A, Stricker D. Introducing a new benchmarked dataset for activity monitoring. In: Proceedings of International Symposium on Wearable Computers, Newcastle, 2012. 108–109
55 Shukla S N, Marlin B. Interpolation-prediction networks for irregularly sampled time series. In: Proceedings of International Conference on Learning Representations, New Orleans, 2018
56 Wu Z, Pan S, Long G, et al. Connecting the dots: multivariate time series forecasting with graph neural networks. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020. 753–763
57 Zhou T, Niu P, Sun L, et al. One fits all: power general time series analysis by pretrained LM. In: Proceedings of Advances in Neural Information Processing Systems, New Orleans, 2023. 43322–43355
58 Jin M, Wang S, Ma L, et al. Time-LLM: time series forecasting by reprogramming large language models. In: Proceedings of the Twelfth

International Conference on Learning Representations, Vienna, 2024

59 Neil D, Pfeiffer M, Liu S C. Phased LSTM: accelerating recurrent network training for long or event-based sequences. In: Proceedings of Advances in Neural Information Processing Systems, Barcelona, 2016. 3882–3890

60 Chen R T Q, Rubanova Y, Bettencourt J, et al. Neural ordinary differential equations. In: Proceedings of Advances in Neural Information Processing Systems, Montreal, 2018. 6572–6583

61 Silva I, Moody G, Scott D J, et al. Predicting in-hospital mortality of ICU patients: the PhysioNet/computing in cardiology challenge 2012. 2012 Comput Cardiology, 2012, 39: 245–248

62 Ding S, Xia B, Ren M, et al. NIERT: accurate numerical interpolation through unifying scattered data representations using transformer encoder. IEEE Trans Knowl Data Eng, 2024, 36: 6731–6744

63 Liu J, Cao M, Chen S. TimeCHEAT: a channel harmony strategy for irregularly sampled multivariate time series analysis. In: Proceedings of AAAI Conference on Artificial Intelligence, Philadelphia, 2025. 18861–18869

64 Mallat S G. A theory for multiresolution signal decomposition: the wavelet representation. IEEE Trans Pattern Anal Machine Intell, 1989, 11: 674–693

65 Menne M J, Durre I, Vose R S, et al. An overview of the global historical climatology network-daily database. J Atmos Ocean Tech, 2012, 29: 897–910

66 Johnson A E W, Pollard T J, Shen L, et al. MIMIC-III, a freely accessible critical care database. Sci Data, 2016, 3: 1–9

67 Biloš M, Sommer J, Rangapuram S S, et al. Neural flows: efficient alternative to neural ODEs. In: Proceedings of Advances in Neural Information Processing Systems, 2021. 21325–21337

68 Baytas I M, Xiao C, Zhang X, et al. Patient subtyping via time-aware LSTM networks. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, 2017. 65–74

69 Zhou H, Zhang S, Peng J, et al. Informer: beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of AAAI Conference on Artificial Intelligence, 2021. 11106–11115

70 Zhou T, Ma Z, Wen Q, et al. Fedformer: frequency enhanced decomposed transformer for long-term series forecasting. In: Proceedings of International Conference on Machine Learning, Baltimore, 2022. 27268–27286

71 Zeng A, Chen M, Zhang L, et al. Are transformers effective for time series forecasting? In: Proceedings of AAAI Conference on Artificial Intelligence, Washington, 2023. 11121–11128

72 Su Y, Zhao Y, Niu C, et al. 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, Anchorage, 2019. 2828–2837

73 Abdulaal A, Liu Z, Lancewicki T. Practical approach to asynchronous multivariate time series anomaly detection and localization. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021. 2485–2494

74 Hundman K, Constantinou V, Laporte C, et al. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, 2018. 387–395

75 Xu J, Wu H, Wang J, et al. Anomaly transformer: time series anomaly detection with association discrepancy. In: Proceedings of International Conference on Learning Representations, 2022

76 Wu H, Hu T, Liu Y, et al. TimesNet: temporal 2D-variation modeling for general time series analysis. In: Proceedings of International Conference on Learning Representations, Kigali, 2023

77 Zeng A, Chen M, Zhang L, et al. Are transformers effective for time series forecasting? In: Proceedings of AAAI Conference on Artificial Intelligence, Washington, 2023. 11121–11128

78 Zhang W, Yin C, Liu H, et al. Irregular multivariate time series forecasting: a transformable patching graph neural networks approach. In: Proceedings of the Forty-first International Conference on Machine Learning, Vienna, 2024

79 Cuturi M. Fast global alignment kernels. In: Proceedings of International Conference on Machine Learning, Bellevue, Washington, 2011. 929–936