H^{2}SCAN: Adaptive Time Series Representation Learning via Heterogeneous Hypergraph Structure-aware Contrasts

Biao Chen[1,*], **Zijie Tang**[1,*], **Junhua Fang**[1,†], **Feng Lu**[2], **Lang Zhang**[2] and **Pengpeng Zhao**[1]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]AISpeech Ltd, Suzhou, China
{bcheny, zjtang1}@stu.suda.edu.cn, {jhfang, ppzhao}@suda.edu.cn, {feng.lu, lang.zhang}@aispeech.com

Abstract

Learning universal representations for time series is fundamental for diverse downstream tasks. However, current approaches largely rely on handcrafted data augmentations, which may distort intrinsic temporal dynamics and structural regularities. In addition, most static representation learning frameworks struggle to cope with the non-stationary nature of real-world time series. To address these issues, we propose Heterogeneous Hypergraph Structure-aware Contrastive Adaptive Network (**H^{2}SCAN**), a novel augmentation-free framework that derives contrastive supervision directly from graph topology. Specifically, H^{2}SCAN constructs a heterogeneous hypergraph with three node types to capture multi-scale temporal characteristics. Building upon this representation, a meta-adaptation network is introduced to dynamically reweight heterogeneous hyperedges, enabling the model to adapt to distribution shifts in real-time. Finally, a structure-aware contrastive learning objective is employed to align latent representation similarity with the intrinsic hypergraph topology. Experiments on multiple benchmarks and cloud Kafka cluster datasets demonstrate that H^{2}SCAN outperforms existing methods by modeling high-order multi-domain dependencies and preserving the semantic integrity of time series data.

1 Introduction

Time series (TS) data pervades diverse domains, including industrial systems, healthcare monitoring, financial markets, and climate science [Yang *et al.*, 2023; Liang *et al.*, 2024]. Learning effective representations from TS is fundamental to enabling downstream tasks such as classification, forecasting, and anomaly detection [Wiliński *et al.*, 2024; Germain *et al.*, 2025]. Nevertheless, labeled TS data remain inherently limited: annotation is often costly, time-consuming, and dependent on specialized domain expertise. This has spurred interest in self-supervised learning (SSL) for learning representations from unlabeled data [Eldele *et al.*, 2023; Zhang *et al.*, 2024; Wang *et al.*, 2024; Cai *et al.*, 2025].

Recent advances in SSL have been dominated by two paradigms: contrastive learning [Chen *et al.*, 2020] and generative modeling [Zerveas *et al.*, 2021; He *et al.*, 2022]. Contrastive methods learn representations by maximizing agreement between differently augmented views of the same sample while distinguishing them from other samples. Generative methods, such as masked reconstruction, learn to predict missing or corrupted portions of the input [Nie, 2022]. While both approaches have shown promise, they face multiple limitations when applied to TS data [Wu *et al.*, 2022].

L1: Non-Stationary Dynamics. TS data often exhibit pronounced non-stationarity with their distinctive multi-scale temporal dynamics and frequency-domain periodicity. As a typical TS example, the electrocardiogram (ECG) signal shown in Figure 1 (a) contains three distinct segments with dramatically different characteristics: *Segment A* exhibits high-frequency oscillations with low variance, *Segment B* displays low-frequency patterns with high amplitude, and *Segment C* shows mixed dynamics with trend shifts. However, existing approaches rely on monolithic or fixed architectures, limiting their adaptability to diverse temporal patterns and dynamic variations in the data.

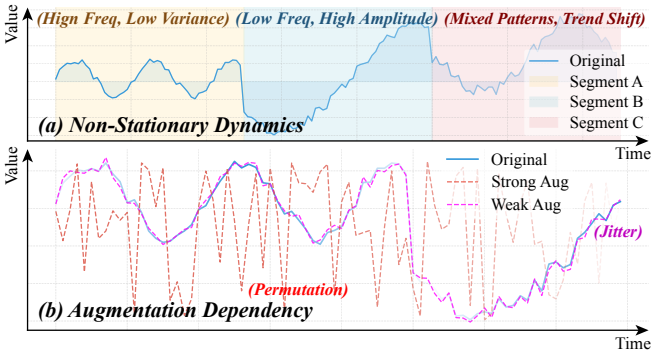

Figure 1: Illustration of major limitations in time series representation learning. (a): Multi-scale temporal dynamics in real-world time series. (b): Effects of different data augmentation strategies.

L2: Augmentation Dependency. Current contrastive methods rely heavily on data augmentation to construct positive/negative pairs. As illustrated in Figure 1 (b), *strong augmentations* (red line), such as temporal permutation, destroy the sequential order and temporal dependencies that define

meaningful patterns. In our ECG example, permuting time steps scrambles the cardiac cycle structure, producing semantically invalid signals; *Weak augmentations* (purple line), such as additive noise, generate views too similar to the original, providing minimal contrastive signal for learning discriminative representations. These imposed artificial representational invariances may not align with the true invariances in the data, limiting generalization [Demirel and Holz, 2025].

Although existing methods have made notable progress in addressing L1 or L2 under specific scenarios, their reliance on fixed modeling paradigms leads to a limited domain perspective [Kara *et al.*, 2025; Cheng *et al.*, 2025]. As a result, global periodic structures in the frequency domain and distributional shifts in the statistical domain are often overlooked. For instance, in cloud Kafka clusters, monitoring models can neither distinguish transient spikes from systemic failures, thus falling into "alert fatigue", nor detect routine periodic features visible only in the frequency domain, such as scheduled batch jobs or log rotations. Moreover, critical distributional shifts, including data skew or hot partitions, often remain undetected due to limited statistical awareness.

To overcome these intertwined challenges, we propose H^{2}SCAN, a unified framework that naturally captures the multi-faceted nature of TS while providing structural similarity for contrastive learning. Specifically, H^{2}SCAN begins by constructing a heterogeneous hypergraph that integrates time, frequency, and statistics-aware nodes to capture multi-domain characteristics, with hyperedges modeling higher-order interactions beyond standard pairwise relationships. For ***L1***, we introduce a meta-learning network that leverages segment-level statistics to adaptively reweight heterogeneous hyperedges, allowing the model to emphasize relevant cross-domain information under non-stationary dynamics without manual intervention. Moreover, these components are unified through a message passing mechanism that facilitates effective information propagation across the hypergraph, enabling rich multi-scale representation. For ***L2***, we eliminate augmentation dependency by deriving soft similarity targets directly from hypergraph topology, where samples with similar structural patterns are naturally aligned, enabling the encoder to be trained with continuous similarity scores instead of hard positive/negative pairs.

Our main contributions are summarized as follows:

- We introduce a hypergraph architecture that simultaneously models temporal, frequency, and statistical characteristics of time series.
- We propose a meta-adaptation mechanism that generates instance-specific weights for hypergraph message passing, enabling real-time adaptation to distribution shifts.
- We design a soft contrastive objective that uses hypergraph topological similarity as a natural supervision signal, bypassing the pitfalls of manual data augmentation.
- Extensive experiments on various datasets demonstrate H^{2}SCAN outperforms state-of-the-art methods, showcasing the benefits of multi-domain structural modeling. Additionally, we construct four cloud cluster datasets to explore TSRL in elastic message-queue scaling.

2 Related Work

Time Series Representation Learning. Traditional methods focused on learning instance-level representations using recurrent architectures, but recent advances have shifted toward self-supervised contrastive learning paradigms. TS-TCC [Eldele *et al.*, 2021] proposes a temporal and contextual contrasting framework that learns robust representations by utilizing strong and weak augmentations. To capture multi-scale features, TS2Vec [Yue *et al.*, 2022] introduces a hierarchical contrastive framework capable of learning improved representations across various semantic levels. Addressing the frequency domain, CoST [Woo *et al.*, 2022] and TF-C [Zhang *et al.*, 2022] leverage spectral information to disentangle seasonal and trend components, thereby preserving complex temporal dynamics. Beyond contrastive methods, generative approaches inspired by masked image modeling have gained traction. Ti-MAE [Li *et al.*, 2023] applies a masked autoencoder architecture to TS, demonstrating strong generalization capabilities. To mitigate the difficulty of reconstruction in high-noise environments, SimMTM [Dong *et al.*, 2023] proposes a manifold-aware masked modeling strategy that reconstructs the original series by aggregating information from multiple neighbors, effectively guiding the learning process in a low-dimensional manifold.

Using Graphs for Time Series. Modeling TS as graphs captures complex inter-dependencies between variables. Early works like DCRNN [Li *et al.*, 2017] and STGCN [Yu *et al.*, 2017] integrate Graph Neural Networks (GNNs) with recurrent or convolutional layers to model spatial and temporal dependencies simultaneously, primarily relying on pre-defined graph structures. Recent research focuses on capturing dynamic correlations. GTS [Han *et al.*, 2021] utilizes a discrete graph structure learning mechanism to infer probabilistic dependencies between TS online.

3 Problem Definition

Consider a TS dataset $\mathcal{D} = \{\mathcal{T}_i\}_{i=1}^{N}$, where each sample $\mathcal{T}_i \in \mathbb{R}^{T\times C}$ consists of C variables recorded over T timestamps. To capture the complex interactions across different domains, H^{2}SCAN formally map each sample $\mathcal{T}_i$ to a heterogeneous hypergraph $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i, \mathcal{W}_i)$, where $\mathcal{V}_i$ denotes the set of multi-domain nodes, $\mathcal{E}_i$ represents the set of multi-type hyperedges, and $\mathcal{W}_i$ contains the adaptive hyperedge weights predicted by a meta-learner.

Our goal is to learn a universal encoder $f_\theta : \mathbb{R}^{T\times C} \to \mathbb{R}^d$ that maps an input series $\mathcal{T}_i$ to a d-dimensional latent representation $\mathcal{Z}_i$. This encoder is optimized via a two-stage paradigm: ❶ In the *pre-training* phase, we aim to optimize f_θ by minimizing a joint loss $\mathcal{L} = \mathcal{L}_{cl} + \lambda\mathcal{L}_{rec}$. Here, $\mathcal{L}_{rec}$ is a reconstruction loss, $\mathcal{L}_{cl}$ is a structural contrastive loss that aligns the representation similarity $S_{repr}(\mathcal{Z}_i, \mathcal{Z}_j)$ with the hypergraph structural similarity $S_{struct}(\mathcal{G}_i, \mathcal{G}_j)$:

$$\arg\min_{\theta} \sum_{i\neq j} KL(P_{struct}(j|i)||P_{repr}(j|i)), \tag{1}$$

where $P_{struct/repr}(j|i)$ is the normalized similarity distribution. ❷ In the *fine-tuning* phase, the pre-trained encoder f_θ is adapted to specific tasks using a task-specific head g_ϕ.

4 Methodology

Figure 2 shows the framework of H^{2}SCAN, which consists of the following four major modules:

- **Multi-domain Heterogeneous Hypergraph Modeling**, which represents TS across complementary information spaces by explicitly encoding time, frequency, and statistical domains as heterogeneous nodes connected via multi-type hyperedges (Section 4.1).
- **Dynamic Non-stationary Adaptation**, which learns data-driven hyperedge weighting functions to model diverse temporal dynamics beyond fixed architectures, enabling the hypergraph structure to adjust to evolving TS characteristics (Section 4.2).
- **Heterogeneous Message Passing**, which propagates information across multi-domain hypergraph structures through structure-aware attention and cross-type interactions (Section 4.3).
- **Structure-aware Contrastive Learning**, which replaces augmentation-dependent similarity construction with topology-driven supervision by deriving soft similarity targets directly from the intrinsic multi-domain hypergraph structure (Section 4.4).

4.1 Multi-Domain Heterogeneous Hypergraph Construction

This work proposes a heterogeneous hypergraph architecture that explicitly models TS across multiple information domains, as shown in Figure 2 (a).

Heterogeneous Node Types. Instead of embedding TS into a single latent space, we project each sample $\mathcal{T}$ into three distinct feature spaces. ***i) Time-Domain Nodes.*** For the capture of local temporal patterns, we encode each observation using a convolutional encoder:

$$\mathcal{V}^{time} = \phi_{time}(\mathcal{T}) \in \mathbb{R}^{L\times d}, \tag{2}$$

where ϕ_{time} consists of 1D convolutions with batch normalization. This produces L time-domain nodes v_t^{time} for $t = 1, \ldots, L$. ***ii) Frequency-Domain Nodes.*** In order to capture global periodic patterns, we extract frequency components via the FFT: $\mathcal{F} = FFT(\mathcal{T})$. Complex coefficients are then decomposed into real and imaginary parts and encoded:

$$\mathcal{V}^{freq} = \phi_{freq}([Re(\mathcal{F}), Im(\mathcal{F})]) \in \mathbb{R}^{N_f\times d}, \tag{3}$$

where $N_f = L/2 + 1$ represents the number of frequency modes, and $\phi_{freq} : \mathbb{R}^2 \to \mathbb{R}^d$ is a linear projection.

iii) Statistical Nodes. Summary statistics are computed to capture distributional properties and high-level characteristics: $\mathcal{S} = [\mu(\mathcal{T}), \sigma(\mathcal{T}), \min(\mathcal{T}), \max(\mathcal{T})]$, where μ and σ denote mean and standard deviation across the entire series. Statistical nodes are encoded as:

$$\mathcal{V}^{stat} = \phi_{stat}(\mathcal{S}) \in \mathbb{R}^{4\times d}. \tag{4}$$

The complete heterogeneous node set is: $\mathbf{V} = \mathcal{V}^{time} \cup \mathcal{V}^{freq} \cup \mathcal{V}^{stat}$ with total $N = L + N_f + 4$ nodes.

Multi-Type Hyperedge Design. Hypergraphs model higher-order interactions where multiple hyperedges participate simultaneously. ***i) Temporal Hyperedges.*** To capture short-term dependencies, temporal hyperedges $\mathcal{E}^{temp}$ connect time-domain nodes within local temporal neighborhoods: $e_t^{temp} = \{v_i^{time} : |i - t| \le w\}$, where w is the temporal window size. The hyperedge representation is then computed via attention-based aggregation: $h_t^{temp} = \phi^{temp}\left(\sum_{v\in e_t^{temp}} \alpha_{t,v} v\right)$, where $\alpha_{t,v}$ are learnable attention weights. ***ii) Frequency Hyperedges.*** Frequency hyperedges $\mathcal{E}^{freq}$ connect frequency-domain nodes with coherent spectral characteristics: $e_k^{freq} = \{v_i^{freq} : i \in \mathcal{N}_k\}$, where $\mathcal{N}_k$ contains frequency indices with similar magnitudes or harmonically related frequencies. This hyperedge embedding is: $h_k^{freq} = \phi^{freq}\left(\sum_{v\in e_k^{freq}} \beta_{k,v} v\right)$.

iii) Cross-Modal Hyperedges. Cross-modal hyperedges $\mathcal{E}^{cross}$ bridge different node types to enable information flow across domains: $e_j^{cross} = \{v_i^{time}, v_k^{freq}, v_m^{stat}\}$. Each $\mathcal{E}^{cross}$ connects nodes from all three domains, with representation: $h_j^{cross} = \phi^{cross}\left(\oplus_{v\in e_j^{cross}} \gamma_{j,v} v\right)$, where $\oplus$ denotes concatenation, and $\gamma_{j,v}$ are domain-specific attention weights.

The hypergraph structure is then formally captured by incidence matrices $\mathcal{H}^{(type)} \in \{0,1\}^{|E^{(type)}|\times N}$ for each type of hyperedge:

$$H_{e,v}^{(type)} = \begin{cases} 1 & \text{if } v \in e \\ 0 & \text{otherwise} \end{cases}. \tag{5}$$

The complete hypergraph is represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E}^{temp} \cup \mathcal{E}^{freq} \cup \mathcal{E}^{cross}, \{\mathcal{H}^{(type)}\})$.

4.2 Dynamic Non-Stationary Adaptation

A dynamic adaptation mechanism that reconfigures the hypergraph's focus based on the real-time characteristics of the input data is introduced as shown in Figure 2 (b).

Meta-Adaptation Network. A meta-adaptation network $\mathcal{M}_\theta$ acts as a *hyper-controller* for the model, predicting instance-specific weights for each hyperedge type conditioned on the input's statistical signature.

For an input segment $\mathcal{T} \in \mathbb{R}^{T\times C}$, we first extract a statistical feature vector $s \in \mathbb{R}^4$ that encapsulates the segment's distributional and spectral properties:

$$s = Concat\left(\mu(\mathcal{T}), \sigma(\mathcal{T}), \mathcal{H}_{spec}(\mathcal{T}), \beta_{trend}(\mathcal{T})\right), \tag{6}$$

where $\mu(\mathcal{T})$ and $\sigma(\mathcal{T})$ are the mean and standard deviation, capturing basic distribution properties. $\mathcal{H}_{spec}(\mathcal{T})$ is the spectral entropy, computed from the normalized power spectrum of the FFT, quantifying the signal's complexity and periodicity. $\beta_{trend}(\mathcal{T})$ is the linear trend coefficient, capturing the directional evolution of the series.

The meta-network processes s through a multi-layer perceptron to generate the adaptive weight vector $\boldsymbol{\alpha}$:

$$\boldsymbol{\alpha} = [\alpha_{time}, \alpha_{freq}, \alpha_{cross}] = Softmax(MLP(s)), \tag{7}$$

which determines the relative importance of temporal, frequency, and cross-modal information.

Segment-Level Dynamically Reweighting. The learned adaptive weights $\boldsymbol{\alpha}$ are applied directly during the message

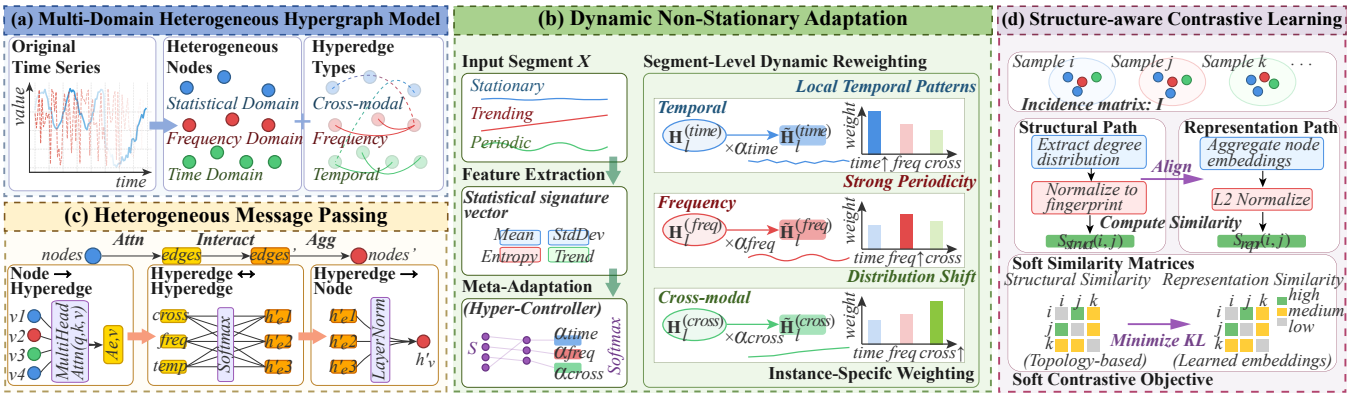

Figure 2: The framework of H^{2}SCAN.

passing phase (detailed in Section 4.3) to modulate information flow. This allows the model to dynamically shift its inductive bias: effectively paying attention to frequency coherence when periodicities are strong (high α_{freq}), or focusing on statistical stability during distribution shifts (high α_{cross}).

Formally, let $\mathcal{H}_l^{(type)}$ denote the aggregated hyperedge representations for a specific type at layer l. The dynamic reweighting is applied prior to node updates as:

$$\tilde{\mathcal{H}}_l^{(type)} = \alpha_{type} \cdot \mathcal{H}_l^{(type)}, \tag{8}$$

where *type* is constrained to the set $\{time, freq, cross\}$.

4.3 Heterogeneous Message Passing

For the propagation of information from the cross-hypergraph, a multi-stage message transmission mechanism is designed as shown in Figure 2 (c).

Node-to-Hyperedge Attention. First, we aggregate information from nodes to their connected hyperedges. A multi-head attention mechanism that dynamically weighs the contribution of each node based on its semantic relevance is employed. For a hyperedge $e \in \mathcal{E}$ and its connected nodes $\mathcal{N}(e)$, the hyperedge representation $h_e \in \mathbb{R}^d$ is updated as:

$$h_e = MultiHeadAttn\left(q = h_e^{(0)}, k = \mathcal{H}_\mathcal{V}, v = \mathcal{H}_\mathcal{V}\right), \tag{9}$$

where $h_e^{(0)}$ is the initial hyperedge embedding serving as the query, and $\mathcal{H}_\mathcal{V}$ represents the matrix of node features serving as keys and values.

To ensure that only structurally connected nodes contribute to the hyperedge representation, the attention mechanism then utilizes the incidence matrix $\mathcal{I}$ to mask irrelevant nodes:

$$\mathcal{A}_{e,v} = Softmax\left(\frac{(\mathcal{W}_Q h_e)(\mathcal{W}_K v)^\top}{\sqrt{d_k}}\right) \cdot \mathbb{I}(v \in \mathcal{N}(e)). \tag{10}$$

Hyperedge-to-Hyperedge Interaction. To capture higher-order dependencies beyond direct node connections, we introduce an interaction layer among hyperedges.

Representations from all hyperedge types are concatenated into a unified set $\mathcal{H}_\mathcal{E} = [\mathcal{H}_{time}; \mathcal{H}_{freq}; \mathcal{H}_{cross}]$. We then apply an irregularity-aware attention mechanism to model the dependencies between these hyperedges:

$$\mathcal{H}'_\mathcal{E} = Softmax\left(\frac{\mathcal{Q}_\mathcal{E}\mathcal{K}_\mathcal{E}^\top}{\sqrt{d}}\right)\mathcal{V}_\mathcal{E}, \tag{11}$$

where $\mathcal{Q}_\mathcal{E}, \mathcal{K}_\mathcal{E}, \mathcal{V}_\mathcal{E}$ are linear projections of $\mathcal{H}_\mathcal{E}$.

Hyperedge-to-Node Aggregation. Finally, we propagate the refined hyperedge information back to the nodes to update their representations. For a node v, its updated representation h'_v is computed by aggregating information from all hyperedges e connected to it, weighted by the incidence matrix $\mathcal{I}$:

$$m_v = \sum_{e \in \mathcal{E}} \mathcal{I}(v, e) \cdot \mathcal{W}_{agg} h'_e. \tag{12}$$

The node update incorporates a residual connection and layer normalization for training stability:

$$h'_v = LayerNorm\left(h_v + Dropout(ReLU(m_v))\right). \tag{13}$$

This completes one layer of heterogeneous message passing [Li *et al.*, 2025], fusing multi-domain information into a unified representation for downstream tasks.

4.4 Structure-aware Contrastive Learning

As shown in Figure 2 (d), a structure-aware contrastive learning framework leverages the intrinsic connectivity of the heterogeneous hypergraph to define pairwise similarities.

Hypergraph Structural Similarity. The structural equivalence between samples is quantified based on their connectivity patterns within the hypergraph topology. Two TS samples are similar if they exhibit similar connectivity patterns.

A *structural fingerprint* $f_i \in \mathbb{R}^{|\mathcal{V}|}$ for the i-th sample is defined based on its node degree distribution across all hyperedge types. Let $\mathcal{I} \in \mathbb{R}^{|\mathcal{E}| \times |\mathcal{V}|}$ be the aggregated incidence matrix. The degree vector d_i for sample i is computed by summing connections across all hyperedges:

$$d_i = \sum_{e \in \mathcal{E}} \mathcal{I}_i(e, \cdot). \tag{14}$$

The structural fingerprint is obtained by normalizing the degree vector to form a probability distribution:

$$f_i = \frac{d_i}{\|d_i\|_1 + \epsilon}. \tag{15}$$

The structural similarity $S_{struct}(i, j)$ between sample i and sample j is then computed as the cosine similarity between their fingerprints:

$$S_{struct}(i, j) = \frac{f_i \cdot f_j}{\|f_i\|_2 \|f_j\|_2}. \tag{16}$$

Representation Similarity. Similarity in the learned latent space is computed concurrently. Let $z_i \in \mathbb{R}^d$ be the global representation of sample i, obtained by aggregating its constituent node embeddings:

$$z_i = \textit{Concat}\left(\bar{h}_{i,time}, \bar{h}_{i,freq}, \bar{h}_{i,stat}\right). \tag{17}$$

The representation similarity $S_{repr}(i,j)$ is defined as the cosine similarity between the normalized embedding vectors:

$$S_{repr}(i,j) = \frac{z_i \cdot z_j}{\|z_i\|_2 \|z_j\|_2}. \tag{18}$$

Soft Contrastive Objective. To align the learned representation space with the structural topology, we employ a soft contrastive objective. The structural similarity distribution serves as a soft target rather than binary labels.

Both similarity matrices are converted into probability distributions using a temperature parameter τ:

$$P_{struct/repr}(j|i) = \frac{\exp(S_{struct/repr}(i,j)/\tau)}{\sum_{k \neq i} \exp(S_{struct/repr}(i,k)/\tau)}. \tag{19}$$

The objective is to minimize the Kullback-Leibler divergence between the predicted representation distribution and the target structural distribution:

$$\mathcal{L}_{CL} = \frac{1}{B} \sum_{i=1}^{B} \textit{KL}\left(P_{struct}(\cdot|i) \parallel P_{repr}(\cdot|i)\right), \tag{20}$$

where $P_{struct}(\cdot|i)$ and $P_{repr}(\cdot|i)$ represent the structural and the latent representation similarity distribution, respectively.

5 Experiments

In this section, we conduct comprehensive experiments to evaluate the effectiveness of H^{2}SCAN:

- **RQ1:** How does H^{2}SCAN perform compared with existing TSRL methods?
- **RQ2:** How does every component that we design contribute to the overall performance?
- **RQ3:** How sensitive is H^{2}SCAN to variations in key parameter settings?
- **RQ4:** How does the computational efficiency of H^{2}SCAN compare to the baselines?

5.1 Experimental Setup

Datasets. The experiments are conducted on 12 real-world datasets, four self-built cloud Kafka cluster datasets constructed for both classification and regression tasks across *broker-level* and *topic-level* components, with detailed statistics reported in Table 1. The cloud Kafka cluster datasets are collected from a 5-broker cloud cluster (September–December 2025), comprising KCluster-broker with seven broker-level metrics and KCluster-topic with topic-level traffic, storage, and request-handling metrics. In addition, eight well-established public datasets are used, including Gesture [Liu *et al.*, 2009], FD-B [Lessmeier *et al.*, 2016], EMG [Goldberger *et al.*, 2000], EPI [Andrzejak *et al.*, 2001], HAR [Anguita *et al.*, 2013], and UCR [Dau *et al.*, 2019] for classification, and C-MAPSS [Saxena *et al.*, 2008] and Bearing [Wang *et al.*, 2018] for regression.

Tasks	Datasets	*Samples*	*Length*	*Freq.*	*Classes*
Clas.	**KCluster-bc**	2,163	178	2 Hours	5
	KCluster-tc	2,163	178	2 Hours	7
Reg.	**KCluster-br**	1,384/346/433	35	2 Hours	-
	KCluster-tr	1,384/346/433	35	2 Hours	-

Table 1: Cloud Kafka cluster dataset descriptions.

Baselines. To evaluate the effectiveness of the proposed H^{2}SCAN, we implement in total seven baselines: ***i)*** **FEI** [Fu and Hu, 2025] leverages frequency-masked prompts to model continuous semantics without explicit positive/negative pairs. ***ii)*** **TimeDRL** [Chang *et al.*, 2024] learns dual-level disentangled embeddings for multivariate TS via a token-based strategy. ***iii)*** **TimesURL** [Liu and Chen, 2024] captures segment- and instance-level information by injecting double Universums as hard negatives. ***iv)*** **SimMTM** [Dong *et al.*, 2023] connects masked modeling with manifold learning by reconstructing series from aggregated masked variants. ***v)*** **InfoTS** [Luo *et al.*, 2023] adaptively selects data augmentations using information-theoretic criteria. ***vi)*** **TF-C** [Zhang *et al.*, 2022] enforces time–frequency consistency through cross-space projectors and a joint loss. ***vii)*** **TS2Vec** [Yue *et al.*, 2022] performs hierarchical contrastive learning over augmented views to learn robust timestamp representations.

Implementation Details. All experiments were conducted using PyTorch 2.6.0 on a server with two Tesla V100 GPUs, 128GB RAM, and a 12-core CPU. We optimized the models using the Adam optimizer with a learning rate of 5e-4 and a batch size of 512, training for 150 epochs per data batch. For the specific modules within H^{2}SCAN, the structural similarity threshold in Equation (16) is set to 0.5, and the adaptation function initial weight is set to 0.3.

5.2 Main Results (RQ1)

We compare H^{2}SCAN with all seven baselines across multiple classification and regression benchmarks. Tables 2 and 3 list the results for two tasks.

Classification. The first observation is that H^{2}SCAN achieves the best performance in terms of accuracy across all eight datasets. This confirms the superior capability of our framework in learning discriminative representations by integrating multi-domain nodes during the pre-training phase. Specifically, H^{2}SCAN achieves an average improvement of 2.89% over the strongest baseline FEI. Moreover, we observe that augmentation-based methods like TS2Vec, TimeDRL, and TimesURL perform relatively poorly on datasets with sensitive periodic signatures, such as EMG and KCluster-tc. This is likely because their manual perturbations distort the underlying frequency components and physical laws of the TS. The third observation is that while frequency-aware methods like TF-C and FEI capture periodic patterns, H^{2}SCAN offers a key advantage by using cross-modal hyperedges to fuse these patterns with global statistical context, leading to more robust classifications in complex cloud scenarios where data noise is prevalent.

Datasets	Rand. Init.	TS2Vec	TimeDRL	TF-C	TimesURL	SimMTM	InfoTS	FEI	**H²SCAN**
Gesture	68.33	72.50	73.33	70.00	73.33	76.67	71.67	77.50	**78.33**
FD-B	69.61	48.31	47.97	65.48	54.41	63.49	62.99	70.99	**71.38**
EMG	95.12	78.05	78.05	92.68	73.17	87.80	97.56	97.56	**97.56**
EPI	80.21	95.60	94.05	95.28	96.67	96.22	97.07	97.24	**97.30**
HAR	82.66	64.74	79.88	77.27	77.57	83.61	82.49	84.15	**84.97**
UCR	72.44	67.44	63.36	78.50	79.40	80.42	81.77	82.65	**82.71**
KCluster-bc	72.02	72.02	77.52	66.51	71.04	74.77	72.02	69.27	**83.95**
KCluster-tc	64.68	65.60	65.60	61.93	66.51	61.93	61.01	65.59	**67.43**
Avg.	75.63	70.53	72.47	75.96	74.01	78.11	78.32	80.62	**82.95**

Table 2: The end-to-end fine-tuning accuracy(%) of all methods on classification task. Results of baselines are retrieved from original papers.

Datasets	Rand. Init.	TS2Vec	TimeDRL	TF-C	TimesURL	SimMTM	InfoTS	FEI	**H²SCAN**
C-MAPSS	0.067 / 0.206	0.061 / 0.207	0.083 / 0.245	0.063 / 0.215	0.060 / 0.205	0.086 / 0.221	0.072 / 0.213	0.079 / 0.214	**0.060 / 0.199**
Bearing	0.073 / 0.184	0.184 / 0.265	0.147 / 0.289	0.604 / 0.645	0.164 / 0.256	0.054 / 0.140	0.073 / 0.178	0.033 / 0.115	**0.030 / 0.109**
KCluster-br	0.063 / 0.177	0.269 / 0.481	0.029 / 0.065	0.529 / 0.700	0.440 / 0.646	0.032 / 0.093	0.030 / 0.078	**0.024** / 0.077	0.028 / **0.055**
KCluster-tr	0.005 / 0.057	0.317 / 0.439	0.004 / 0.054	0.458 / 0.501	0.672 / 0.712	0.014 / 0.109	0.001 / 0.028	0.002 / 0.032	**0.001 / 0.024**
Avg.	0.052 / 0.156	0.208 / 0.348	0.066 / 0.163	0.414 / 0.515	0.334 / 0.455	0.047 / 0.141	0.044 / 0.124	0.035 / 0.110	**0.030 / 0.097**

Table 3: The average end-to-end fine-tuning results (*MSE / MAE*) of all methods on regression task. Bold: Best, Underline: Suboptimal.

Metrics	*MSE*	*MAE*
H²SCAN	**0.0604**	**0.1987**
w/o frequency	0.0618	0.2078
w/o statistical	0.0612	0.2100
w/o freq.+stat.	0.0754	0.2140
w/o meta.	0.0612	0.2093

Table 4: Ablation study results on C-MAPSS dataset.

Regression. H²SCAN also achieves the best regression performance across all datasets. A primary reason for this success is the introduction of the Meta-Adaptation Network, which allows H²SCAN to handle the non-stationary nature of industrial sensor data and cloud traffic. In remaining useful life prediction tasks, where statistical properties evolve as machinery degrades, H²SCAN dynamically reweights its hypergraph interactions to maintain predictive stability. Specifically, our model achieves an average error reduction of 16.89% compared to FEI and InfoTS. The performance advantage is particularly pronounced on the KCluster-tr dataset. This suggests that the high-order relationships captured by our hyperedge design are essential for modeling the intricate dependencies in distributed clusters.

5.3 Ablation Study (RQ2)

In the ablation study, we quantify the contribution of each H²SCAN component. The results on the C-MAPSS dataset for regression are shown in Table 4, confirming the necessity of each module in the framework.

Experimental results of the four variants illustrate that the most significant performance decline occurs when both the frequency and statistical domains (*w/o freq.+stat.*) are excluded. This drop is due to the model's inability to capture global periodicities and distributional properties, forcing it to rely solely on local temporal sequences. Similarly, the individual removal of the frequency domain (*w/o freq.*) or the statistical domain (*w/o stat.*) leads to a noticeable decrease in predictive accuracy, highlighting the importance of multi-domain feature integration for robust representation learning. Furthermore, excluding the meta-adaptation network (*w/o meta.*) results in performance degradation, as it disables the model's ability to dynamically reweight hyperedge interactions in response to the non-stationary nature of industrial sensor data.

5.4 Hyperparameter Sensitivity (RQ3)

Varying Hypergraph Layers. We investigate the impact of model depth by varying the number of hypergraph layers from 1 to 5. As shown in Figures 3 (a) and (b), it can be observed that the performance of H²SCAN remains relatively stable as the depth increases. Specifically, the best results are achieved with 4 layers for both datasets. H²SCAN captures high-order dependencies without requiring excessively deep architectures, demonstrating its robustness across varying model complexities.

Impact of Attention Heads. We analyze the influence of the number of attention heads used in the heterogeneous message passing module. Figures 3 (c) and (d) show results for heads in $\{2, 4, 6, 8, 10\}$. We observe that the choice of attention heads has a noticeable impact on the model's ability to capture multi-domain correlations. Specifically, the optimal

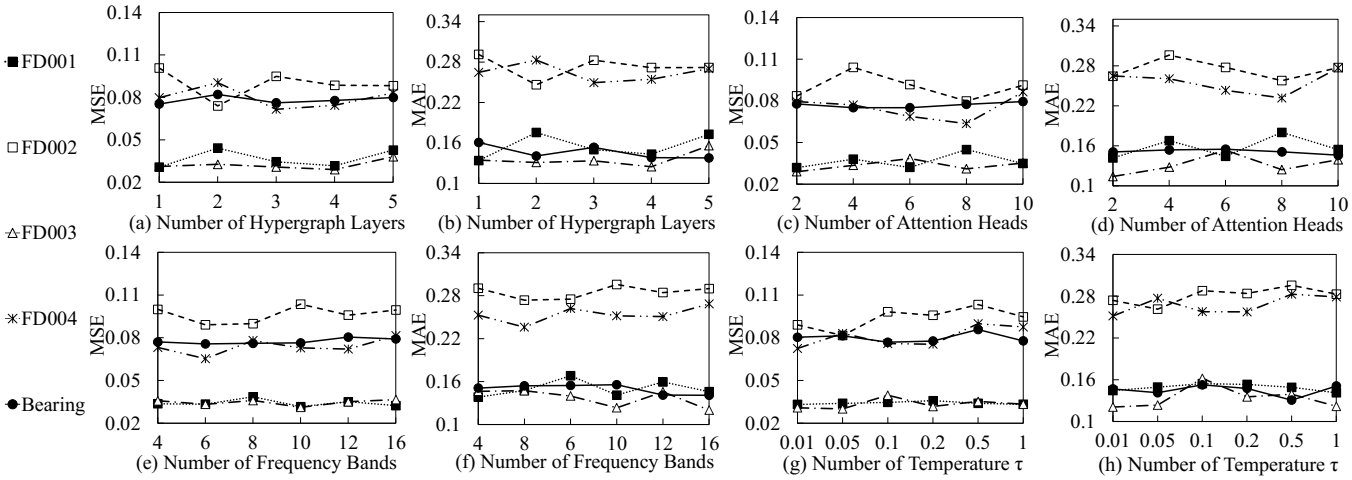

Figure 3: Performance of H^{2}SCAN under varying hyperparameters.

performance is achieved when the number of heads is set to 8. This indicates that while multiple heads allow the model to attend to different semantic subspaces simultaneously, an excessive number of heads may introduce redundant information that slightly degrades the accuracy.

Selecting Frequency Bands b. We examine the parameter b, which determines the number of top frequency components selected for the frequency-domain nodes. Figures 3 (e) and (f) show the results for b values ranging from 4 to 16. We observe that b is a crucial factor in capturing the periodic signatures of industrial sensors. The C-MAPSS dataset, which contains complex turbofan engine data, benefits from a smaller b (set to 8), whereas the Bearing dataset, characterized by more distinct vibration frequencies, achieves peak performance with a larger b (set to 16). This suggests that H^{2}SCAN can be tailored to the spectral complexity of the specific TS.

Effect of Temperature τ. H^{2}SCAN utilizes a temperature parameter τ to control the sharpness of the soft contrastive distribution. We test various levels of τ, and the results are shown in Figures 3 (g) and (h). The model's performance is sensitive to the scale of similarity scores. A temperature of $\tau = 0.01$ yields the best alignment between the representation space and the hypergraph topology. However, the impact of varying τ is relatively limited, with performance fluctuations controlled within 9.8%. This robustness is attributed to our structure-based contrastive objective, which provides a stable and natural supervision signal compared to traditional augmentation-based methods.

5.5 Efficiency Analysis (RQ4)

We evaluate the efficiency of each measure in both model pre-training time (per epoch) and end-to-end fine-tuning time. The experimental results are shown in Table 5. FEI distinguishes itself with the shortest pre-training time, attributed to its 1D ResNet-based encoder structure and the innovative frequency-masked embedding inference mechanism. In contrast, other models like TF-C and InfoTS consider additional complex contrastive pairs or frequency-time alignments that must be computed during pre-training, resulting in longer processing times. Our model, H^{2}SCAN, although it includes complex components such as heterogeneous hypergraph modeling and a meta-adaptation network, still maintains high efficiency through optimized design. By utilizing sparse hypergraph message passing and efficient FFT-based frequency encoding, its pre-training time is only slightly longer than FEI and TS2Vec but achieves significantly higher accuracy in both classification and regression tasks. This indicates that H^{2}SCAN successfully balances computational complexity and model effectiveness.

Model	*Pre-training time*	*Fine-tuning time*
TS2Vec	171.49 s	4.78 s
TimeDRL	426.95 s	5.63 s
TF-C	2120.15 s	9.99 s
TimesURL	650.71 s	4.45 s
SimMTM	517.57 s	5.85 s
InfoTS	1267.51 s	19.16 s
FEI	120.51 s	7.61 s
H^{2}SCAN	212.09 s	7.53 s

Table 5: Pre-training and fine-tuning efficiency on HAR dataset.

6 Conclusion

In this paper, we propose H^{2}SCAN, a structure-aware framework for time series representation learning. H^{2}SCAN eliminates augmentation dependency by constructing multi-domain heterogeneous hypergraphs, and leverages hypergraph topology as a natural source of similarity for contrastive learning. The meta-adaptation network enables dynamic adjustment to non-stationary patterns, while heterogeneous message passing captures higher-order multi-scale dependencies. Comprehensive empirical analyses validate that H^{2}SCAN achieves state-of-the-art performance on twelve real-world datasets across diverse domains. The robustness and universality of H^{2}SCAN make it a promising candidate for various practical applications.

Acknowledgments

This work was supported by the Natural Science Foundation of the National Natural Science Foundation of China under grant (No. 62572335), the National Key Research and Development Program of China (No.2023YFF0725002), Jiangsu Higher Education Institutions of China (No.23KJA520011), Enhancing Competitiveness through Data-Driven and Intelligent Process Optimization (No. SBC20071).

Contribution Statement

Biao Chen and Zijie Tang contributed equally. Junhua Fang is the corresponding author.

References

[Andrzejak *et al.*, 2001] Ralph G Andrzejak, Klaus Lehnertz, Florian Mormann, Christoph Rieke, Peter David, and Christian E Elger. Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. *PRE*, 64(6):061907, 2001.

[Anguita *et al.*, 2013] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, Jorge Luis Reyes-Ortiz, et al. A public domain dataset for human activity recognition using smartphones. In *Esann*, volume 3, pages 3–4, 2013.

[Cai *et al.*, 2025] Ruichu Cai, Zhifan Jiang, Kaitao Zheng, Zijian Li, Weilin Chen, Xuexin Chen, Yifan Shen, Guangyi Chen, Zhifeng Hao, and Kun Zhang. Learning disentangled representation for multi-modal time-series sensing signals. In *WWW*, pages 3247–3266, 2025.

[Chang *et al.*, 2024] Ching Chang, Chiao-Tung Chan, Wei-Yao Wang, Wen-Chih Peng, and Tien-Fu Chen. Timedrl: Disentangled representation learning for multivariate time-series. In *ICDE*, pages 625–638, 2024.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020.

[Cheng *et al.*, 2025] Rui Cheng, Xiangfei Jia, Qing Li, Rong Xing, Jiwen Huang, Yu Zheng, and Zhilong Xie. Fat: Frequency-aware pretraining for enhanced time-series representation learning. In *SIGKDD*, pages 310–321, 2025.

[Dau *et al.*, 2019] Hoang Anh Dau, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, and Eamonn Keogh. The ucr time series archive. *JAS*, 6(6):1293–1305, 2019.

[Demirel and Holz, 2025] Berken Utku Demirel and Christian Holz. Learning without augmenting: Unsupervised time series representation learning via frame projections. *NeurIPS*, 2025.

[Dong *et al.*, 2023] Jiaxiang Dong, Haixu Wu, Haoran Zhang, Li Zhang, Jianmin Wang, and Mingsheng Long. Simmtm: A simple pre-training framework for masked time-series modeling. *NeurIPS*, 36:29996–30025, 2023.

[Eldele *et al.*, 2021] Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee Keong Kwoh, Xiaoli Li, and Cuntai Guan. Time-series representation learning via temporal and contextual contrasting. *IJCAI*, 2021.

[Eldele *et al.*, 2023] Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee-Keong Kwoh, Xiaoli Li, and Cuntai Guan. Self-supervised contrastive representation learning for semi-supervised time-series classification. *TPAMI*, 45(12):15604–15618, 2023.

[Fu and Hu, 2025] En Fu and Yanyan Hu. Frequency-masked embedding inference: A non-contrastive approach for time series representation learning. In *AAAI*, volume 39, pages 16639–16647, 2025.

[Germain *et al.*, 2025] Thibaut Germain, Chrysoula Kosma, and Laurent Oudre. Time series representations with hard-coded invariances. In *ICML*, 2025.

[Goldberger *et al.*, 2000] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23):e215–e220, 2000.

[Han *et al.*, 2021] Peng Han, Jin Wang, Di Yao, Shuo Shang, and Xiangliang Zhang. A graph-based approach for trajectory similarity computation in spatial networks. In *SIGKDD*, pages 556–564, 2021.

[He *et al.*, 2022] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, pages 16000–16009, 2022.

[Kara *et al.*, 2025] Denizhan Kara, Tomoyoshi Kimura, Jinyang Li, Bowen He, Yizhuo Chen, Yigong Hu, Hongjue Zhao, Shengzhong Liu, and Tarek F Abdelzaher. Adats: Learning adaptive time series representations via dynamic soft contrasts. In *NeurIPS*, 2025.

[Lessmeier *et al.*, 2016] Christian Lessmeier, James Kuria Kimotho, Detmar Zimmer, and Walter Sextro. Condition monitoring of bearing damage in electromechanical drive systems by using motor current signals of electric motors: A benchmark data set for data-driven classification. In *PHME*, volume 3, 2016.

[Li *et al.*, 2017] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *ICLR*, 2017.

[Li *et al.*, 2023] Zhe Li, Zhongwen Rao, Lujia Pan, Pengyun Wang, and Zenglin Xu. Ti-mae: Self-supervised masked time series autoencoders. *arXiv preprint arXiv:2301.08871*, 2023.

[Li *et al.*, 2025] Boyuan Li, Yicheng Luo, Zhen Liu, Junhao Zheng, Jianming Lv, and Qianli Ma. Hyperimts: Hypergraph neural network for irregular multivariate time series forecasting. *ICML*, 2025.

[Liang *et al.*, 2024] Yuxuan Liang, Haomin Wen, Yuqi Nie, Yushan Jiang, Ming Jin, Dongjin Song, Shirui Pan, and

Qingsong Wen. Foundation models for time series analysis: A tutorial and survey. In *SIGKDD*, pages 6555–6565, 2024.

[Liu and Chen, 2024] Jiexi Liu and Songcan Chen. Timesurl: Self-supervised contrastive learning for universal time series representation learning. In *AAAI*, volume 38, pages 13918–13926, 2024.

[Liu *et al.*, 2009] Jiayang Liu, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. uwave: Accelerometer-based personalized gesture recognition and its applications. *PMC*, 5(6):657–675, 2009.

[Luo *et al.*, 2023] Dongsheng Luo, Wei Cheng, Yingheng Wang, Dongkuan Xu, Jingchao Ni, Wenchao Yu, Xuchao Zhang, Yanchi Liu, Yuncong Chen, Haifeng Chen, et al. Time series contrastive learning with information-aware augmentations. In *AAAI*, volume 37, pages 4534–4542, 2023.

[Nie, 2022] Y Nie. A time series is worth 64words: Long-term forecasting with transformers. *ICLR*, 2022.

[Saxena *et al.*, 2008] Abhinav Saxena, Kai Goebel, Don Simon, and Neil Eklund. Damage propagation modeling for aircraft engine run-to-failure simulation. In *PHM*, pages 1–9, 2008.

[Wang *et al.*, 2018] Biao Wang, Yaguo Lei, Naipeng Li, and Ningbo Li. A hybrid prognostics approach for estimating remaining useful life of rolling element bearings. *IEEE Transactions on Reliability*, 69(1):401–412, 2018.

[Wang *et al.*, 2024] Daoyu Wang, Mingyue Cheng, Zhiding Liu, and Qi Liu. Timedart: A diffusion autoregressive transformer for self-supervised time series representation. *ICML*, 2024.

[Wiliński *et al.*, 2024] Michał Wiliński, Mononito Goswami, Willa Potosnak, Nina Żukowska, and Artur Dubrawski. Exploring representations and interventions in time series foundation models. *ICML*, 2024.

[Woo *et al.*, 2022] Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Cost: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. *ICLR*, 2022.

[Wu *et al.*, 2022] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *ICLR*, 2022.

[Yang *et al.*, 2023] Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. In *SIGKDD*, pages 3033–3045, 2023.

[Yu *et al.*, 2017] Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *IJCAI*, 2017.

[Yue *et al.*, 2022] Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *AAAI*, volume 36, pages 8980–8987, 2022.

[Zerveas *et al.*, 2021] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A transformer-based framework for multivariate time series representation learning. In *SIGKDD*, pages 2114–2124, 2021.

[Zhang *et al.*, 2022] Xiang Zhang, Ziyuan Zhao, Theodoros Tsiligkaridis, and Marinka Zitnik. Self-supervised contrastive pre-training for time series via time-frequency consistency. *NeurIPS*, 35:3988–4003, 2022.

[Zhang *et al.*, 2024] Kexin Zhang, Qingsong Wen, Chaoli Zhang, Rongyao Cai, Ming Jin, Yong Liu, James Y Zhang, Yuxuan Liang, Guansong Pang, Dongjin Song, et al. Self-supervised learning for time series analysis: Taxonomy, progress, and prospects. *TPAMI*, 46(10):6775–6794, 2024.