TimeCAP: A Channel-Aware Pre-Training Framework for Multivariate Time Series Forecasting

Chuanru Ren[1], Yao Lu[2†], Tianjin Huang[3], Haowen Zheng[1], Hengde Zhu[1], Yunyin Li[4], Hengxiao Li[1], Lu Liu[3†]

[1]Tongji University, Shanghai, China
[2]Anhui University, Hefei, China
[3]University of Exeter, Exeter, U.K.
[4]China University of Petroleum (East China), Qingdao, China
rencr@tongji.edu.cn, l.liu3@exeter.ac.uk

Abstract

Amid recent advances for multivariate time series forecasting, self-supervised learning has emerged as a promising paradigm for deriving transferable knowledge from multi-domain data. Despite its effectiveness, existing approaches exhibit two critical limitations: (1) Underestimating the significance of multivariate dependencies in learning generalizable representations and (2) Failing to reconcile the complementary strengths of autoregressive and one-shot generative paradigms. In this work, we propose TimeCAP, a novel channel-aware pre-training framework that internalizes latent causal relationships among variables inherent in multi-domain data, and effectively transfers the acquired knowledge to downstream applications. Technically, we present a flexible channel-grouping learning approach, complemented by an adaptive meta-routing mechanism, enabling TimeCAP to parallel recognize intra-group local patterns while maintaining global coherence. Intra- and inter-group multivariate dependencies are captured through the self- and cross-attention with channel-aware mask, which strictly confine interactions among time-aligned, fine-grained multivariate tokens. To seamlessly unify two advanced generative paradigms, we propose a novel dynamic dual-head decoding and optimization strategy, empowering TimeCAP to leverage critical dependencies in the output series while avoiding cumulative errors over time. In the few-shot evaluation, TimeCAP achieves average MSE and MAE reductions of 11.8% and 6% over leading baselines, while also outperforming state-of-the-art models in full-shot and zero-shot settings by large margins.

Code — https://github.com/RCR-LYY/TimeCAP

Introduction

Time series data extensively permeate diverse domains, including energy, transportation, meteorology, and finance, where accurately forecasting dynamic temporal evolution remains fundamental to enabling informed decision-making (Wu et al. 2023; Wang et al. 2024; Liu et al. 2025b). Motivated by significant progress in large language models (LLMs), multivariate time series modeling has recently transitioned from supervised, task-specific framework (Liu et al. 2024b; Han et al. 2024) toward self-supervised pre-training paradigm (Wang et al. 2025a,b). This advanced approach acquires transferable knowledge from multi-domain unlabeled data, thereby effectively enhancing downstream task performance. Among prevailing self-supervised methods, masked modeling (Zerveas et al. 2021; Dong et al. 2023) and contrastive learning (Luo et al. 2023; Liu and Chen 2024) have been well-developed for time series, which respectively focus on reconstructing missing data and maximizing representation similarity. In contrast, generative pre-training remains insufficiently explored in this field, despite growing evidence (Shi et al. 2024; Liu et al. 2025a) underscoring its essential role in enhancing scalability and generalization.

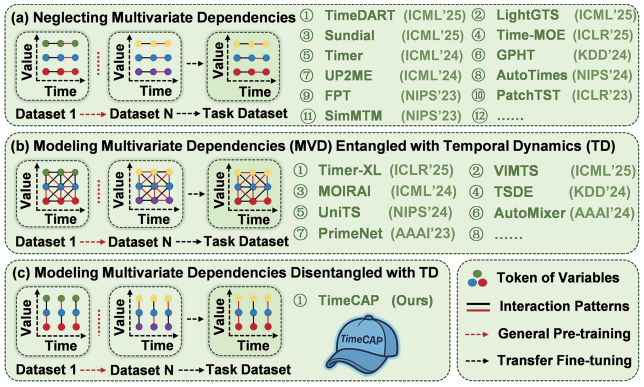

Figure 1: Strategies for modeling multivariate dependencies (MVD) in self-supervised pre-training: (a) neglecting MVD, (b) learning MVD entangled with temporal dynamics (TD), and (c) modeling MVD disentangled from TD.

Given that word-level dependencies in language are analogously manifested as temporal correlations in time series, the time-oriented (channel-independent) pre-training paradigm (Nie et al. 2023; Zhou et al. 2023; Liu et al. 2024c; Wang et al. 2025b) emerges as a logically coherent strategy that transforms multivariate inputs into parameter-sharing univariate series. Despite learning generalizable temporal patterns, this strategy neglects inter-channel interactions, resulting in the complete discarding of critical multivariate dependencies. In contrast, the channel-temporal pre-

†Corresponding author.

training paradigm (Woo et al. 2024; Liu et al. 2025a) models inter-channel interactions and temporal dynamics in an intertwined manner. Although this strategy facilitates modeling spatiotemporal dynamics, recognizing intricate patterns also introduces substantial model complexity and increased optimization difficulty. Moreover, in real-world scenarios with limited temporal patterns, incorporating temporal interactions may lead to additional noise or heightened risk of overfitting. As presented in Figure 1, these observations underscore a critical oversight: **the significance of multivariate dependencies in learning generalizable representations remains largely unrecognized.** Building on the foregoing, leveraging multivariate dependencies for generalizable representation learning faces two outstanding challenges. Firstly, directly modeling global dependencies among all variables (Liu et al. 2024b) significantly complicates knowledge discovery by overemphasizing irrelevant dependencies or missing critical interactions. Recent methods (Hu et al. 2025; Qiu et al. 2025) employ channel clustering algorithms for extracting local pattern. Although they alleviate learning complexity, these models undermine the preservation of global coherence across the variable space. Accordingly, **effectively recognizing local patterns while maintaining global coherence remains a critical challenge**. Additionally, although patch-wise series tokenization (Liu et al. 2024a, 2025a) facilitates capturing fine-grained multivariate correlations, this strategy incurs uncontrollable computational overhead when scaling the model to datasets containing numerous variables. Therefore, **achieving controllable computational overhead in fine-grained modeling constitutes a pressing challenge**.

On the other hand, autoregressive (Liu et al. 2024d) and one-shot generation (Wang et al. 2025a) represent two advanced decoding paradigms, characterized respectively by iterative and simultaneous output generation. However, the former is susceptible to cumulative error over time, thereby compromising long-term forecasting accuracy. Conversely, the latter inadequately captures critical dependencies in the output series and requires horizon-specific hyperparameter tuning. Therefore, **elegantly unifying the two paradigms remains a largely unexplored research direction.**

To address these challenges, we introduce the TimeCAP framework that explicitly disentangles inter-channel dependencies from temporal interactions, thereby fully unleashing their representational potential. Specifically, TimeCAP partitions all channels into overlapping groups for independent learning intra-group local patterns. To enable inter-group communication, each group is equipped with adaptive meta-routers that dynamically orchestrate cross-group information flow, therefore maintaining global coherence. Intricate intra- and inter-group interactions are modeled via meticulously designed channel-aware self- and cross-attention mechanisms. Finally, the discrete groups are aggregated into a unified representation and then processed through dual decoding heads whose activations and interactions are explicitly tailored for distinct phases. Based on the above techniques, TimeCAP consistently achieves superior performance across eight publicly available datasets on downstream tasks. In summary, the principal contributions of this paper are as follows:

- We propose TimeCAP, a novel channel-aware pre-training framework that internalizes latent causal and semantic relationships among entities (variables) inherent in multi-domain data, and effectively transfers the acquired knowledge to downstream applications.
- We present a flexible channel-grouping learning approach, complemented by an adaptive meta-routing mechanism, which enables controllable computational overhead, and facilitates parallel recognizing intra-group local patterns while maintaining global coherence.
- We propose a novel dynamic dual-head decoding and optimization strategy that seamlessly unifies autoregressive and one-shot generation paradigms, empowering TimeCAP to leverage critical dependencies in the output series while avoiding cumulative errors over time.
- In the few-shot evaluation, TimeCAP achieves average MSE and MAE reductions of 11.8% and 6%, underscoring the robustness and adaptability of the channel-oriented pre-training paradigm in downstream scenarios characterized by scarce temporal patterns.

Related Work

Self-Supervised Pre-Training on Time Series

Self-supervised pre-training across multi-domain data has demonstrated significant success in natural language (Radford et al. 2021), image (Bao et al. 2021), and video (Yan et al. 2021) fields. Motivated by its capacity to enable transferable representations in downstream applications, multivariate time series modeling has recently transitioned from supervised, task-specific frameworks (Liu et al. 2024b) toward the self-supervised pre-training paradigm (Wang et al. 2025a). In the context of this approach, although significant breakthroughs have been achieved through masked modeling (Zerveas et al. 2021; Dong et al. 2023) and contrastive learning (Luo et al. 2023; Liu and Chen 2024), fundamental challenges still remain in the development of generative pre-training models. Accordingly, this work seeks to address existing challenges in the generative pre-training paradigm to improve adaptability across downstream applications.

Channel Strategies in Generative Pre-Training

On the one hand, the channel-independent approaches (Shi et al. 2024; Wang et al. 2025b) omit inter-channel interactions, thereby entirely discarding critical multivariate dependencies. On the other hand, the channel-temporal methods (Woo et al. 2024; Liu et al. 2025a) pose significant challenges for stable and efficient knowledge discovery from intricate and entangled global dependencies. Moreover, under real-world scenarios characterized by scarce temporal patterns, the incorporation of temporal interactions may introduce additional noise or lead to overfitting. Departing from both paradigms, we propose a novel channel-aware pre-training paradigm to demonstrate the critical role of multivariate dependencies in transferable representation learning.

Autoregressive versus One-Shot Generation

Autoregressive generation (Ansari et al. 2024; Shi et al. 2024), rooted in classical time series forecasting frameworks such as ARIMA (Box et al. 2015), has long been foundational in this domain. However, its sequential decoding nature inherently results in cumulative errors over time. In contrast, the one-shot generation strategy (Ma et al. 2025; Qiu et al. 2025), which forecasts all future steps in parallel, has seen widespread adoption. Nevertheless, this strategy neglects multivariate dependencies within the output series and suffers from poor generalization due to horizon-specific configurations. In this paper, we propose a unified framework that seamlessly integrates the strengths of both paradigms.

Methodology

Problem Formulation

Given a multivariate time series $\mathbf{X} \in \mathbb{R}^{C \times L}$, the forecasting task aims to accurately predict future observations $\mathbf{Y} \in \mathbb{R}^{C \times H}$, where C denotes the number of channels, L is the look-back window length, and H represents the forecasting horizon. The self-supervised model is initially pretrained on multi-source datasets to learn generalizable and transferable representations. Then, it is fine-tuned on task-specific datasets to better adapt to downstream applications.

Structure Overview

As presented in Figure 2, TimeCAP incorporates reversible instance normalization and unified forecasting blocks. Each layer-specific block comprises three sequential components: **(1)** Group-Wise Adaptive Representation, which partitions all channels into overlapping groups for independent learning and integration of diverse patterns; **(2)** Channel-Aware Encoding, which models multivariate dependencies through comprehensive interactions among intra- and inter-group fine-grained tokens; and **(3)** Dynamic Dual-Head Decoding and Optimization, which ensures accurate forecasts via phase-specific head activation and corresponding loss function. For brevity, layer indices for these components and blocks are omitted while maintaining sequential architecture compliance with the vanilla Transformer.

Reversible Instance Normalization

Distribution shift denotes changes in the distributional properties of time series, occurring both across different datasets and within the same dataset over time. These shifts significantly hinder model generalization and stability in self-supervised learning. Consistent with state-of-the-art time series models (Wang et al. 2025a; Ma et al. 2025), reversible instance normalization (Kim et al. 2021) is utilized to concentrate the original series to zero mean and scale to unit variance. The predicted series is finally inverse normalized to restore the original data distribution.

Group-Wise Adaptive Representation

Multi-Scale Patch-Wise Series Tokenization For modeling inter-channel interactions, prior advanced methods (Han et al. 2024; Liu et al. 2024b) predominantly employ series-wise tokenization, representing each univariate series as a single token. This coarse-grained approach inherently overlooks fine-grained multivariate relationships. In contrast, TimeCAP divides each univariate series into a sequence of non-overlapping patches, with each patch as a series token. Compared to point-wise tokenization (Ansari et al. 2024), this method mitigates the impact of outliers and random noise, preserving more representative information. Additionally, inspired by multi-scale representation learning (Cheng et al. 2023), tokens across different blocks span varying temporal ranges, enabling TimeCAP to capture multivariate dependencies across multiple time scales and adapt to heterogeneous datasets. The process is formalized as:

$$\mathbf{E}^0 = \text{Tokenization}\left(\mathbf{X}\right), \tag{1}$$

where $\mathbf{E}^0 \in \mathbb{R}^{C \times P \times N}$, with P and N denoting the patch count and temporal span, respectively.

Flexible Channel Grouping Embedding As presented in Figure 2(b), TimeCAP introduces a Flexible Channel Grouping Embedding approach. Specifically, the entire multivariate time series is divided into a sequence of overlapping K groups, each consisting of W channels ($K, W \ll C$), resulting in $\{\mathbf{E}_i^1 \in \mathbb{R}^{W \times P \times N}\}_{i=1}^K$. Then, each group is independently projected into distinct feature spaces in parallel, yielding disentangled embeddings $\{\mathbf{E}_i^2 \in \mathbb{R}^{W \times P \times D}\}_{i=1}^K$, where D represents the projection dimension. This strategy offers the following key benefits: **(1)** Channel disentanglement facilitates capturing local multivariate dependencies and reduces learning complexity, **(2)** Overlapping channels capture diverse feature representations within their respective groups, thereby enhancing model robustness and mitigating overfitting, and **(3)** The attention complexity is reduced from $O((CP)^2D)$ to controllable $O((WP)^2D)$, effectively mitigating the risk of system failure.

Adaptive Meta-Routing Mechanism To address the structural barriers restricting inter-group information exchange, a Adaptive Meta-Routing mechanism is introduced to dynamically orchestrate cross-group information flow through adaptive aggregation and redistribution. Specifically, learnable contextual tokens (meta-routers) are introduced as macroscopic representations for respective groups, enabling dynamic interaction with cross-group tokens. This process is formalized as follows:

$$\mathbf{E}_i^3 = \text{Concat}(\mathbf{E}_i^2, \mathbf{R}_i), \tag{2}$$

where $\{\mathbf{E}_i^3 \in \mathbb{R}^{(W+1) \times P \times D}\}_{i=1}^K$ denotes the concatenated representations, and $\{\mathbf{R}_i \in \mathbb{R}^{1 \times P \times D}\}_{i=1}^K$ represents the meta-routers. As presented in Figure 2(b), fine-grained meta-routers are assigned well-defined temporal positions, ensuring strict adherence to their relative sequencing. This novel mechanism bridges causal information across groups, thereby expanding the receptive field for each channel to cover all others.

Channel-Aware Encoding

To facilitate sequential modeling while maintaining temporal continuity, the 2D token arrangement is flattened

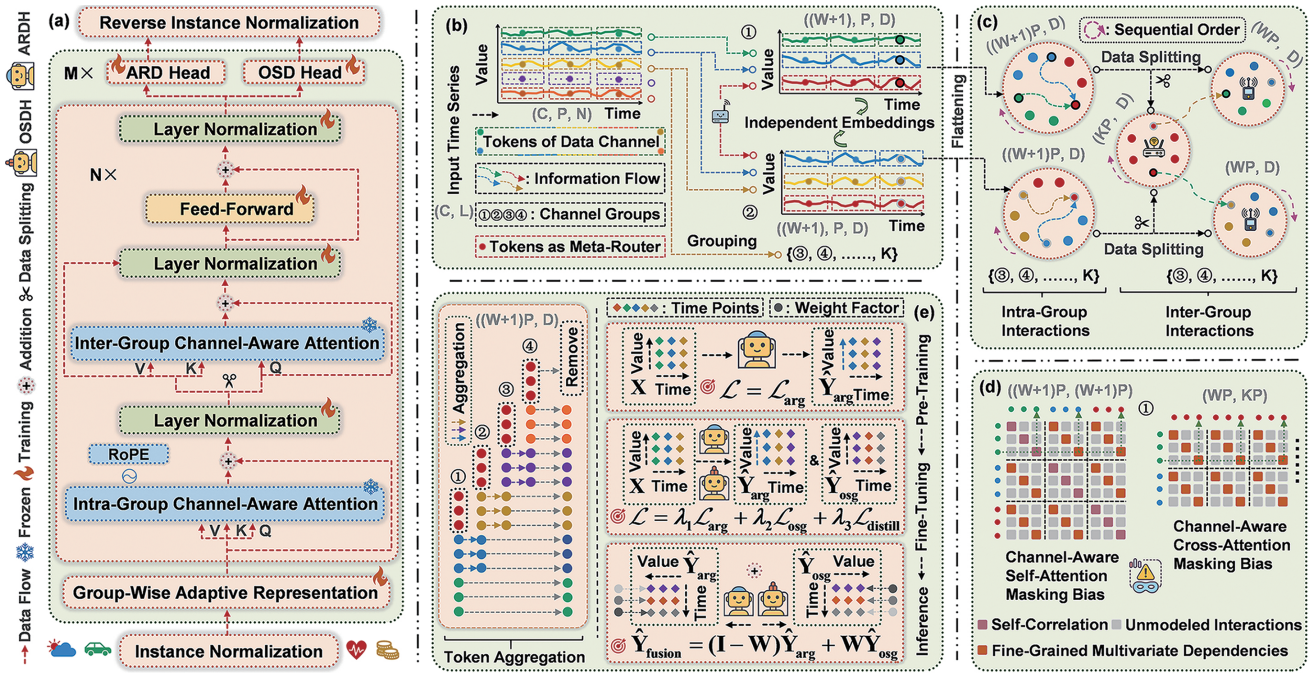

Figure 2: (a) Architectural schematic of the TimeCAP framework. (b) Group-wise adaptive representation component. (c) Intra-group interactions and inter-group communication. (d) Channel-aware masking mechanism. (e) Dynamic dual-head decoding and optimization strategy.

into a 1D sequence in a temporal-first manner, formulated as $\mathbf{Z}_i = \text{Flatten}(\mathbf{E}_i^3)$. Here, $\mathbf{Z}_i \in \mathbb{R}^{(W+1)P \times D}$ represents the resulting sequential representation. Unlike channel-independent approaches (Wang et al. 2025b) and channel-temporal (Woo et al. 2024) methods, TimeCAP explicitly disentangles inter-channel dependencies from temporal patterns through a Channel-Aware Attention module, thereby fully unleashing their representational capacity. Building on this perspective, the TimeCAP encoder employs channel-aware self-attention to capture intra-group correlations and cross-attention to model inter-group interactions, thereby enabling comprehensive multivariate dependency learning, as shown in Figure 2(c).

Intra-Group Channel-Aware Attention To mitigate the inherent permutation invariance of attention, positional encodings must be introduced to preserve the temporal order of channel-aligned tokens. Unlike prior methods (Wang et al. 2025a; Liu et al. 2024d) that utilize absolute positional encodings, TimeCAP adopts Rotary Positional Encoding (RoPE) (Su et al. 2024), enabling effective relative position modeling. Formally, let $\mathbf{Z}_i = \{\mathbf{z}_j \in \mathbb{R}^D\}_{j=1}^{(W+1)P}$ denote the sequence of token embeddings. The similarity score between tokens $\mathbf{z}_u$ and $\mathbf{z}_v$ is computed as:

$$\mathbf{S}_{u,v}^{\text{self}} = \mathbf{z}_u^\top \mathbf{W}_Q \mathbf{R}_{u-v} \mathbf{W}_K^\top \mathbf{z}_v, \tag{3}$$

where $\mathbf{S}^{\text{self}} \in \mathbb{R}^{(W+1)P \times (W+1)P}$ denotes the self-attention scores, $\mathbf{W}_Q$, $\mathbf{W}_K \in \mathbb{R}^{D \times d}$ are learnable projection matrices for the queries and keys, and $\mathbf{R}_{u-v} \in \mathbb{R}^{d \times d}$ represents the rotary matrix encoding the relative displacement between positions u and v.

As presented in Figure 2(d), TimeCAP introduces a Channel-Aware Masking mechanism to strictly confine interactions among time-aligned, fine-grained multivariate tokens. These interactions comprise two types: **(1)** among data tokens across channels, capturing explicit multivariate dependencies; and **(2)** between data tokens and meta-routers, aggregating patch-level information within each group into the corresponding meta-routers. Formally, the self-attention mask is defined as:

$$\mathbf{M}_{u,v}^{\text{self}} = \begin{cases} -\infty, & \text{if } |u-v| \in \{0, P, \ldots, WP\}, \\ 0, & \text{otherwise}, \end{cases} \tag{4}$$

where $\mathbf{M}^{\text{self}} \in \mathbb{R}^{(W+1)P \times (W+1)P}$ enforces a constraint on multivariate interactions exclusively. The complete channel-aware self-attention (CASA) is expressed as follows:

$$\text{CASA}(\mathbf{Z}_i) = \text{Softmax}\left(\frac{\mathbf{S}^{\text{self}}}{\sqrt{\text{d}}} + \mathbf{M}^{\text{self}}\right)(\mathbf{Z}_i \mathbf{W}_V), \tag{5}$$

where $\mathbf{W}_V \in \mathbb{R}^{D \times d}$ represents the learnable value projection matrix. Following the canonical Transformer, layer normalization and residual connections are applied, resulting in $\mathbf{Z}_i' \in \mathbb{R}^{(W+1)P \times D}$.

Inter-Group Channel-Aware Attention Cross-attention has proven effective in multi-modal learning (Li et al. 2021) for adaptively modeling token-level dependencies across heterogeneous modalities. Building on this foundation, TimeCAP first separates out meta-routers from each group. This produces two individual representations: **(1)** $\mathbf{Z}_i'' \in \mathbb{R}^{WP \times D}$, representing the refined token embeddings

within each group; and **(2)** $\{\mathbf{R}'_i \in \mathbb{R}^{P\times D}\}_{i=1}^{K}$, denoting the set of meta-routers, each aggregating fine-grained information from its corresponding group. Then, all meta-routers are concatenated following the equation below:

$$\mathbf{R}'' = \mathrm{Concat}(\mathbf{R}'_1, \mathbf{R}'_2, ..., \mathbf{R}'_K), \tag{6}$$

where $\mathbf{R}'' \in \mathbb{R}^{KP\times D}$ denotes a macroscopic representation aggregating information across groups. Following this, a channel-aware cross-attention (CA^2) is introduced, in which data tokens $\mathbf{Z}''_i$ serve as queries and the concatenated meta-routers $\mathbf{R}''$ function as both keys and values. This module enables adaptive routing of intra-group information from meta-routers to cross-group tokens, thereby enhancing inter-group communication. The process is formalized as:

$$\mathrm{CA}^2\left(\mathbf{Z}''_i, \mathbf{R}''\right) = \mathrm{Softmax}\left(\frac{\mathbf{S}^{\mathrm{cr}}}{\sqrt{\mathrm{d}}} + \mathbf{M}^{\mathrm{cr}}\right)(\mathbf{Z}''_i\mathbf{W}_V), \tag{7}$$

$$\mathbf{M}^{\mathrm{cr}}_{u,v} = \begin{cases} -\infty, & \text{if } |u-v| \in \{0, P, \ldots, GP\}, \\ 0, & \text{otherwise}, \end{cases} \tag{8}$$

where $\mathbf{S}^{\mathrm{cr}} = (\mathbf{Z}''_i\mathbf{W}_Q)(\mathbf{Z}''_i\mathbf{W}_K)^\top$ and $\mathbf{M}^{\mathrm{cr}} \in \mathbb{R}^{WP\times KP}$ denote the cross-attention score and masking matrix, respectively, where $G = \max\{K-1, W-1\}$ determines the interaction range. Similarly, the CA^2 output undergoes residual addition and layer normalization, resulting in $\mathbf{Z}^{\mathrm{out}}_i \in \mathbb{R}^{WP\times D}$. Notably, except for the final encoder layer, enhanced representations are reintegrated with their corresponding meta-routers by inverting the original separation process, and then passed to the subsequent encoder layer.

Dynamic Dual-Head Decoding and Optimization

In TimeCAP, overlapping channels across groups are encoded as distinct token embeddings. Although encapsulating diverse and complementary knowledge independently learned within their respective groups, these embeddings remain semantically aligned. As presented in Figure 2(e), TimeCAP aggregates these embeddings to capture more comprehensive and robust multivariate dependencies. This process is formulated as follows:

$$\mathbf{O} = \mathrm{scatter\_add}\left(\{\mathbf{Z}^{\mathrm{out}}_i \in \mathbb{R}^{WP\times D}\}_{i=1}^{K},\ \mathbf{idx}\right), \tag{9}$$

where $\mathbf{O} \in \mathbb{R}^{CP\times D}$ is the aggregated representation. The $\mathrm{scatter\_add}(\cdot)$ operation accumulates and averages embeddings from $\mathbf{Z}^{\mathrm{out}}$ into $\mathbf{O}$ at positions designated by the corresponding index $\{\mathbf{idx}_i \in \mathbb{R}^{WP\times D}\}_{i=1}^{K}$.

To unify the autoregressive and one-shot generation paradigms, TimeCAP introduces a novel Dynamic Dual-Head Decoding and Optimization framework. This design orchestrates dual decoding heads whose activations and interactions are explicitly tailored to distinct phases, including pre-training, fine-tuning, and inference. The subsequent subsections detail the strategies employed at each phase.

Self-Supervised Pre-Training Drawing inspiration from the next-token prediction employed by large language models (Achiam et al. 2023), TimeCAP activates the autoregressive decoding head (ARDH) during the self-supervised pre-training phase to facilitate generalizable representation learning. The classic Mean Squared Error (MSE) is utilized as the loss function, formulated as:

$$\mathcal{L}_{\mathrm{pretrain}} = \| \mathrm{Concat}\left(\mathbf{X}[:, T:],\ \mathbf{Y}\right) - \hat{\mathbf{Y}}_{\mathrm{arg}}\|_F^2, \tag{10}$$

where $\hat{\mathbf{Y}}_{\mathrm{arg}} = \mathrm{ARDH}(\mathbf{O}) \in \mathbb{R}^{C\times L}$ is the autoregressive output, with T denoting the next-token prediction length. At this phase, the forecasting horizon is set to $H = T$.

Mixed-Supervision Fine-Tuning In contrast to state-of-the-art supervised fine-tuning methods (Wang et al. 2025a), TimeCAP introduces a novel mixed-supervision fine-tuning approach. Specifically, both the autoregressive decoding head and the one-shot decoding head (OSDH) are activated, with the multi-task learning loss function formulated as:

$$\mathcal{L}_{\mathrm{finetune}} = \lambda_1 \cdot \mathcal{L}_{\mathrm{arg}} + \lambda_2 \cdot \mathcal{L}_{\mathrm{osg}} + \lambda_3 \cdot \mathcal{L}_{\mathrm{distill}}, \tag{11}$$

$$\mathcal{L}_{\mathrm{arg}} = \| \mathrm{Concat}\left(\mathbf{X}[:, T:],\ \mathbf{Y}[:, :T]\right) - \hat{\mathbf{Y}}_{\mathrm{arg}}\|_F^2, \tag{12}$$

$$\mathcal{L}_{\mathrm{osg}} = \|\mathbf{Y}[:, T:] - \hat{\mathbf{Y}}_{\mathrm{osg}}[:, T:]\|_F^2, \tag{13}$$

$$\mathcal{L}_{\mathrm{distill}} = \|\hat{\mathbf{Y}}_{\mathrm{arg}}[:, (L-T):] - \hat{\mathbf{Y}}_{\mathrm{osg}}[:, :T]\|_F^2, \tag{14}$$

where $\mathcal{L}_{\mathrm{arg}}$, $\mathcal{L}_{\mathrm{osg}}$, and $\mathcal{L}_{\mathrm{distill}}$ denote the autoregressive loss, one-shot loss, and self-distillation loss, respectively. The scaling factors λ_1, λ_2, and λ_3 weight each loss term accordingly. Additionally, $\hat{\mathbf{Y}}_{\mathrm{osg}} = \mathrm{OSDH}(\mathbf{O}) \in \mathbb{R}^{C\times H}$ represents the one-shot output. At this phase, the forecasting horizon is set to $H > T$.

In this phase, the ARDH follows the self-supervised learning paradigm, while the OSDH is fine-tuned through task-specific supervision. The self-distillation loss facilitates effective knowledge transfer from ARDH to OSDH, thereby improving their overall performance.

Task-Specific Inference In the initial stage, both decoding heads are simultaneously activated, enabling autoregressive and one-shot generation. Then, the OSDH is deactivated, allowing the ARDH to continue the iterative generation process. Ultimately, outputs from both heads are dynamically fused using a sigmoid-based function as follows:

$$\hat{\mathbf{Y}}_{\mathrm{fusion}} = (\mathbf{I} - \mathbf{W})\hat{\mathbf{Y}}_{\mathrm{arg}} + \mathbf{W}\hat{\mathbf{Y}}_{\mathrm{osg}}, \tag{15}$$

$$\mathbf{W} = \frac{1}{1 + \exp\left(-\alpha \cdot (\mathbf{x} - \beta)\right)}, \tag{16}$$

where $\hat{\mathbf{Y}}_{\mathrm{fusion}} \in \mathbb{R}^{C\times H}$ represents the final forecasting output, and $\mathbf{W} \in \mathbb{R}^{H}$ denotes the fusion weight derived from the time indices $\mathbf{x} = \{1, ..., H\}$ over the forecasting horizon. This fusion strategy unifies autoregressive and one-shot strengths to ensure forecasting consistency and accuracy.

Experiments

Experimental Setup

Datasets for Pre-Training and Evaluation To enable a comprehensive evaluation, two experimental settings are adopted: single-domain and multi-domain. In the former setting, pre-training is conducted across multi-source energy datasets. The pre-trained model is then evaluated through zero-shot and few-shot forecasting tasks conducted

Portion	Zero-Shot Forecasting (0% training samples)										Few-Shot Forecasting (10% training samples)							
Models	**TimeCAP** (standard)		**TimeCAP**$^{\#}$ (variant)		**Timer-XL** (2025)		**GPHT** (2024)		**Timer** (2024)		**TimeCAP** (standard)		**Timer-XL** (2025)		**GPHT** (2024)		**Timer** (2024)	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh-avg	**0.565**	**0.501**	0.631	0.530	0.582	0.511	0.592	0.515	0.616	0.527	**0.474**	**0.454**	0.566	0.500	0.535	0.476	0.648	0.529
ETTm-avg	**0.544**	**0.460**	0.567	0.476	0.546	0.462	0.553	0.464	0.564	0.474	**0.430**	**0.409**	0.510	0.450	0.501	0.439	0.486	0.432
Weather	**0.285**	**0.307**	0.312	0.332	0.291	0.311	0.292	0.317	0.291	0.317	**0.259**	**0.283**	0.269	0.292	0.267	0.291	0.275	0.296
Exchange	0.359	0.404	0.415	0.450	**0.351**	**0.397**	0.365	0.410	0.354	0.402	**0.339**	**0.397**	0.352	0.397	0.393	0.422	0.398	0.425

Table 1: Zero-shot and few-shot evaluation results for multivariate time series forecasting. The results are averaged over all horizons $H \in \{96, 192, 336, 720\}$, and the best and second-best results are denoted by **bold** and underline, respectively.

on both in-domain and out-of-domain datasets, all exclusive from the pre-training datasets. In the multi-domain setting, aligned with established research practices (Wang et al. 2025a), a diverse collection of datasets spanning energy, environmental, and financial domains is utilized for pre-training. Subsequently, full-shot fine-tuning is conducted on eight widely adopted benchmark datasets, including Electricity, Solar, Weather, Exchange, and four ETT subsets. Dataset details are provided in extended version.

Baselines Four state-of-the-art self-supervised models are selected as baselines: TimeDART (Wang et al. 2025a) (full-shot only), and Timer-XL (Liu et al. 2025a), GPHT (Liu et al. 2024d), and Timer (Liu et al. 2024c) (evaluated across full-shot, few-shot, and zero-shot settings). In addition, four superior supervised models, including SOFTS (Han et al. 2024), iTransformer (Liu et al. 2024b), PatchTST (Nie et al. 2023), and Crossformer (Zhang and Yan 2023), are included for comparison under the full-shot setting only. Supervised models encompass paradigms dominated by channel-wise dependencies, temporal interactions, and their disentangled integration. TimeCAP fills the gap in channel-aware self-supervised learning.

Fair Experiment To ensure fair comparison, the look-back window length is fixed at 96 for all models, with forecasting horizons set to $H \in \{96, 192, 336, 720\}$. The mean squared error (MSE) and mean absolute error (MAE) serve as the evaluation metrics, with lower values signifying superior accuracy. All self-supervised models share identical pre-training datasets to ensure consistency. Building upon the original configurations, hyperparameters are further optimized using the Optuna (Akiba et al. 2019) framework with a budget of 40 trials per setting to maximize performance. Configuration ranges are provided in extended version.

Zero-Shot and Few-Shot Forecasting Evaluation

Table 1 reports the zero-shot and few-shot forecasting results under the single-domain setting. Notably, TimeCAP$^{\#}$ denotes the randomly initialized variant without prior training. The principal findings are summarized as follows. In the zero-shot scenario, TimeCAP effectively acquires generalizable knowledge, as evidenced by average reductions exceeding 9.2% in MSE and 6.7% in MAE compared to the untrained variant. Furthermore, this acquired knowledge exhibits superior transferability, enabling TimeCAP to consistently outperform time-oriented and channel-temporal state-of-the-art models across both in-domain and out-of-domain datasets. In the few-shot scenario, TimeCAP achieves average MSE reductions of 14.8%, 10.6%, and 9.9% compared to Timer, GPHT, and Timer-XL, respectively, along with corresponding MAE reductions of 7.6%, 5%, and 5.3%. These results underscore the robustness and adaptability of channel-aware pre-training in real-world scenarios characterized by scarce temporal patterns.

Full-Shot Forecasting Evaluation

Table 2 presents the full-shot forecasting performance under the multi-domain setting. Notably, TimeCAP* denotes the variant trained from scratch using a unified hyperparameter configuration. The primary insights are summarized as follows. Firstly, TimeCAP consistently demonstrates best overall performance over eight state-of-the-art baselines, attaining the top accuracy across 81.3% of the 16 evaluation metrics. Specifically, it reduces MSE by 5.2% and 2.7% on the Exchange and ETTh2 datasets, respectively, compared to the most competitive models. This improvement arises from the channel grouping strategy enabling local pattern extraction within groups while maintaining global coherence, and from the adaptive fusion of dual decoding heads that integrate complementary generative outputs. Additionally, relative to TimeCAP*, this notable performance gain indicates that TimeCAP internalizes latent causal and semantic relationships among variables inherent in multi-domain data, and effectively transfers the acquired knowledge to downstream applications. Importantly, the learned representations differ fundamentally from those obtained through time-oriented or channel-temporal pre-training paradigms. These findings strongly substantiate the significance of multivariate dependencies in learning generalizable representations.

Model Analysis

Ablation Studies To thoroughly evaluate the performance of the TimeCAP modules, ablation studies are conducted on full-shot forecasting employing five distinct model variants: **w/o-ARDH**, in which the autoregressive decoding head is removed; **w/o-OSDH**, entailing exclusion of the one-shot

Paradigms	Self-Supervised Learning												Supervised Learning							
Models	**TimeCAP** (MVD)		**TimeCAP*** (variant)		**Timer-XL** (M&T)		**TimeDART** (TD)		**GPHT** (TD)		**Timer** (TD)		**SOFTS** (MVD)		**iTrans.** (MVD)		**PatchTST** (TD)		**Cross.** (M&T)	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	**0.378**	**0.394**	0.398	0.407	0.428	0.423	0.382	0.395	0.433	0.422	0.437	0.418	0.393	0.403	0.407	0.410	0.387	0.400	0.513	0.496
ETTm2	**0.278**	**0.323**	0.282	0.325	0.298	0.337	0.287	0.333	0.294	0.333	0.295	0.337	0.287	0.330	0.288	0.332	0.281	0.326	0.757	0.610
ETTh1	0.424	**0.430**	0.432	0.439	0.444	0.444	**0.421**	0.436	0.448	0.437	0.473	0.449	0.449	0.442	0.454	0.447	0.469	0.454	0.529	0.522
ETTh2	**0.363**	**0.391**	0.379	0.403	0.384	0.402	0.378	0.402	0.390	0.408	0.386	0.403	0.373	0.400	0.383	0.407	0.387	0.407	0.942	0.684
Weather	**0.250**	**0.276**	0.252	0.277	0.257	0.280	0.264	0.288	0.263	0.284	0.367	0.337	0.255	0.278	0.258	0.279	0.259	0.281	0.259	0.315
Exchange	**0.326**	**0.387**	0.344	0.397	0.349	0.398	0.357	0.402	0.344	0.393	0.363	0.403	0.358	0.405	0.360	0.403	0.367	0.404	0.940	0.707
Solar	**0.228**	0.263	0.237	0.265	0.299	0.304	0.278	0.314	0.358	0.345	0.550	0.423	0.229	**0.256**	0.233	0.262	0.236	0.266	0.641	0.639
Electricity	**0.171**	0.269	0.244	0.269	0.242	0.310	0.210	0.290	0.263	0.335	0.254	0.319	0.174	**0.264**	0.178	0.270	0.216	0.304	0.244	0.334

Table 2: Full-shot results for multivariate time series forecasting. The results are averaged from all forecasting horizons $H \in \{96, 192, 336, 720\}$, and the best and second-best results are denoted by **bold** and underline, respectively. The MVD, TD, and M&T denote the modeling of multivariate, temporal, and entangled multivariate–temporal dependencies, respectively.

decoding head; **re-SFF**, whereby the sigmoid-based fusion function (SFF) is replaced with a direct weighted averaging mechanism; and **w/o-AMR & CA**^{2}, where the adaptive meta-routers (AMR) and channel-aware cross-attention module are removed.

Dataset	ETT-avg		Exchange		Weather		Electricity	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
w/o-ARDH	0.370	0.399	0.347	0.397	0.250	0.277	0.186	0.274
w/o-OSDH	0.399	0.403	0.342	0.399	0.262	0.286	0.497	0.435
w/o-AMR	0.362	0.386	0.349	0.397	0.253	0.280	0.181	0.276
re-SFF	0.372	0.390	0.339	0.395	0.251	0.280	0.252	0.326
TimeCAP	**0.361**	**0.385**	**0.326**	**0.387**	**0.250**	**0.276**	**0.171**	**0.269**

Table 3: Ablation on key components of TimeCAP, with average results reported across all forecasting horizons.

Table 3 summarizes the ablation results, from which the following insights are drawn. The autoregressive decoding head, one-shot decoding head, and sigmoid-based fusion function collaboratively enhance forecasting performance by capturing essential dependencies within the output series, mitigating cumulative errors over long horizons, and adaptively integrating complementary outputs from both decoding heads. Additionally, the variant without adaptive meta-routers and channel-aware cross-attention incurs average MSE reductions of 6.6% and 5.5% on Exchange and Electricity datasets, respectively, highlighting the critical role of cross-group communication in preserving global coherence.

Model Efficiency Under the zero-shot forecasting setting on ETTh2, with an input length of 96 and a prediction horizon of 720, TimeCAP is quantitatively evaluated against three baselines in terms of parameter scale, inference time, floating-point operations (FLOPs), and maximum memory consumption. As shown in Table 4, TimeCAP is demonstrated to attain lower model complexity and computational overhead, along with reduced memory usage and faster inference, thereby underscoring its efficiency advantages over compared approaches.

Models	TimeCAP	Timer-XL	GPHT	Timer
Parameters	**5.02** M	25.27 M	37.98 M	12.63 M
Inference Time (s)	**3.63**	9.97	7.09	9.75
FLOPs	**9.55** G	22.73 G	34.04 G	34.10 G
Max Mem. (MB)	**76.73**	157.53	227.73	130.27

Table 4: Efficiency comparison of TimeCAP and baselines under zero-shot forecasting on ETTh2 (input length 96, forecast horizon 720), evaluated with same batch size.

Conclusion

This paper presents TimeCAP, emphasizing the pivotal role of multivariate dependencies in transferable representation learning. Technically, the channel-grouping embedding with adaptive meta-routers facilitates the recognition of intra-group local patterns while preserving global coherence. Intra- and inter-group multivariate dependencies are captured through self- and cross-attention with a channel-aware mask, which strictly confines interactions among time-aligned, fine-grained multivariate tokens. Additionally, through the dynamic fusion and phase-specific optimization of the generative outputs from both decoding heads, TimeCAP leverages critical dependencies in the output series while mitigating cumulative errors over time. The superior performance of TimeCAP across diverse settings highlights the potential of channel-aware pre-training in time series modeling and opens promising directions for future research on cross-domain generalization and transferability.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; and Koyama, M. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2623–2631.

Ansari, A. F.; Stella, L.; Turkmen, C.; Zhang, X.; Mercado, P.; Shen, H.; Shchur, O.; Rangapuram, S. S.; Arango, S. P.; Kapoor, S.; et al. 2024. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*.

Bao, H.; Dong, L.; Piao, S.; and Wei, F. 2021. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*.

Box, G. E.; Jenkins, G. M.; Reinsel, G. C.; and Ljung, G. M. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons.

Cheng, M.; Liu, Q.; Liu, Z.; Li, Z.; Luo, Y.; and Chen, E. 2023. FormerTime: Hierarchical Multi-Scale Representations for Multivariate Time Series Classification. In *Proceedings of the ACM Web Conference 2023*, 1437–1445.

Dong, J.; Wu, H.; Zhang, H.; Zhang, L.; Wang, J.; and Long, M. 2023. Simmtm: A simple pre-training framework for masked time-series modeling. *Advances in Neural Information Processing Systems*, 36: 29996–30025.

Han, L.; Chen, X.-Y.; Ye, H.-J.; and Zhan, D.-C. 2024. Softs: Efficient multivariate time series forecasting with series-core fusion. *Advances in Neural Information Processing Systems*, 37: 64145–64175.

Hu, Y.; Zhang, G.; Liu, P.; Lan, D.; Li, N.; Cheng, D.; Dai, T.; Xia, S.-T.; and Pan, S. 2025. TimeFilter: Patch-Specific Spatial-Temporal Graph Filtration for Time Series Forecasting. In *Forty-second International Conference on Machine Learning*.

Kim, T.; Kim, J.; Tae, Y.; Park, C.; Choi, J.-H.; and Choo, J. 2021. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International conference on learning representations*.

Li, J.; Selvaraju, R.; Gotmare, A.; Joty, S.; Xiong, C.; and Hoi, S. C. H. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34: 9694–9705.

Liu, J.; and Chen, S. 2024. Timesurl: Self-supervised contrastive learning for universal time series representation learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, 13918–13926.

Liu, J.; Liu, C.; Woo, G.; Wang, Y.; Hooi, B.; Xiong, C.; and Sahoo, D. 2024a. Unitst: Effectively modeling inter-series and intra-series dependencies for multivariate time series forecasting. *arXiv preprint arXiv:2406.04975*.

Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; and Long, M. 2024b. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. In *The Twelfth International Conference on Learning Representations*.

Liu, Y.; Qin, G.; Huang, X.; Wang, J.; and Long, M. 2025a. Timer-XL: Long-Context Transformers for Unified Time Series Forecasting. In *International Conference on Learning Representations*.

Liu, Y.; Qin, G.; Shi, Z.; Chen, Z.; Yang, C.; Huang, X.; Wang, J.; and Long, M. 2025b. Sundial: A Family of Highly Capable Time Series Foundation Models. In *Forty-second International Conference on Machine Learning*.

Liu, Y.; Zhang, H.; Li, C.; Huang, X.; Wang, J.; and Long, M. 2024c. Timer: Generative Pre-trained Transformers Are Large Time Series Models. In *Forty-first International Conference on Machine Learning*.

Liu, Z.; Yang, J.; Cheng, M.; Luo, Y.; and Li, Z. 2024d. Generative pretrained hierarchical transformer for time series forecasting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2003–2013.

Luo, D.; Cheng, W.; Wang, Y.; Xu, D.; Ni, J.; Yu, W.; Zhang, X.; Liu, Y.; Chen, Y.; Chen, H.; et al. 2023. Time series contrastive learning with information-aware augmentations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4534–4542.

Ma, X.; Ni, Z.; Xiao, S.; and Chen, X. 2025. TimePro: Efficient Multivariate Long-term Time Series Forecasting with Variable-and Time-Aware Hyper-state. *arXiv preprint arXiv:2505.20774*.

Nie, Y.; H. Nguyen, N.; Sinthong, P.; and Kalagnanam, J. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *International Conference on Learning Representations*.

Qiu, X.; Wu, X.; Lin, Y.; Guo, C.; Hu, J.; and Yang, B. 2025. Duet: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, 1185–1196.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Shi, X.; Wang, S.; Nie, Y.; Li, D.; Ye, Z.; Wen, Q.; and Jin, M. 2024. Time-moe: Billion-scale time series foundation models with mixture of experts. *arXiv preprint arXiv:2409.16040*.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063.

Wang, D.; Cheng, M.; Liu, Z.; and Liu, Q. 2025a. TimeDART: A Diffusion Autoregressive Transformer for Self-Supervised Time Series Representation. In *Forty-second International Conference on Machine Learning*.

Wang, Y.; Qiu, Y.; Chen, P.; Shu, Y.; Rao, Z.; Pan, L.; Yang, B.; and Guo, C. 2025b. LightGTS: A Lightweight General Time Series Forecasting Model. *arXiv preprint arXiv:2506.06005*.

Wang, Y.; Wu, H.; Dong, J.; Qin, G.; Zhang, H.; Liu, Y.; Qiu, Y.; Wang, J.; and Long, M. 2024. Timexer: Empowering transformers for time series forecasting with exogenous variables. *Advances in Neural Information Processing Systems*, 37: 469–498.

Woo, G.; Liu, C.; Kumar, A.; Xiong, C.; Savarese, S.; and Sahoo, D. 2024. Unified Training of Universal Time Series Forecasting Transformers. In *Forty-first International Conference on Machine Learning*.

Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; and Long, M. 2023. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *International Conference on Learning Representations*.

Yan, W.; Zhang, Y.; Abbeel, P.; and Srinivas, A. 2021. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*.

Zerveas, G.; Jayaraman, S.; Patel, D.; Bhamidipaty, A.; and Eickhoff, C. 2021. A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2114–2124.

Zhang, Y.; and Yan, J. 2023. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In *The eleventh international conference on learning representations*.

Zhou, T.; Niu, P.; Sun, L.; Jin, R.; et al. 2023. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36: 43322–43355.