# Adaptive Refinement of Time Series Foundation Models via Pattern and Context-Awareness

**Anonymous Author(s)**
Affiliation
Address
email

## Abstract

Time-series forecasting in domains such as ITOps and IoT faces two major challenges: data are non-stationary and multivariate, and state-of-the-art Time-Series Foundation Models (TSFMs) rely on fixed-size windows that miss transient phenomena (e.g., spikes, drifts) and their historical context. Prior efforts address this with seasonal-trend decompositions or frequency-aware pre-training, but these require retraining and offer limited adaptability.

We propose a dynamic two-stream framework that augments any pre-trained TSFM with frequency pattern awareness and contextual retrieval as the two streams. Each input window is decomposed via Fast Fourier Transforms (FFT) and Discrete Wavelet Transforms (DWT) to extract key low- and high-frequency patterns, which are fused into a TSFM through lightweight adapters and gated embedding augmentation. In parallel, frequency pattern signatures are used to retrieve semantically similar historical sequences, enriching long-range context. This approach enhances forecasting robustness of deployed TSFMs without retraining, achieving consistent improvements over baselines on both standard and zero-shot forecasting benchmarks, particularly with abrupt data fluctuations and complex temporal dynamics.

## 1 Introduction

Time-series forecasting underpins critical decisions in finance, energy, healthcare, and IoT. The emergence of time-series foundation models (TSFMs) such as Chronos [1], TimesFM [4], MOMENT [9], and Time-MoE [13] has demonstrated the promise of large-scale pre-training: by training on massive corpora, these models capture general temporal patterns, enable zero- or few-shot forecasting across diverse tasks, and have the potential to become the default backbone for forecasting. However, deploying TSFMs in practice faces two persistent challenges. First, real-world time series are non-stationary: they exhibit abrupt spikes, irregular cycles, and regime shifts that a fixed sliding window often fails to capture. Second, once a TSFM is deployed in production Fine-tuning or retraining is impractical due to strict latency, compute, and data-sharing constraints. Organizations typically treat foundation models as frozen inference engines, requiring adaptation methods that enrich their forecasts without altering the model weights. Existing solutions fall short: lightweight adaptation (e.g., adapters, LoRA, prompts) stays confined to the time domain and struggles with transients, while frequency-aware models (e.g., FEDformer[19], CoST, LaST) require training new architectures and are expensive to adapt post-deployment. This leaves a crucial gap: how do we dynamically enhance a pre-trained TSFM at inference to handle real-world temporal dynamics?

We propose a two-stream refinement framework that augments any pre-trained TSFM [1, 4] with frequency pattern awareness and context awareness with historical retrieval, without modifying its core parameters. In the first stream, we apply analytic transforms (FFT and DWT) to decompose the input window into frequency components (details, approx, or trends). This approach exposes transient oscillations and periodic signatures overlooked by time-domain-only encodings. In the second stream, we retrieve semantically similar historical sequences by matching frequency signatures, capturing

Under review at the NeurIPS 2025 Workshop on Recent Advances in Time Series Foundation Models (BERT$^2$S). Do not distribute.

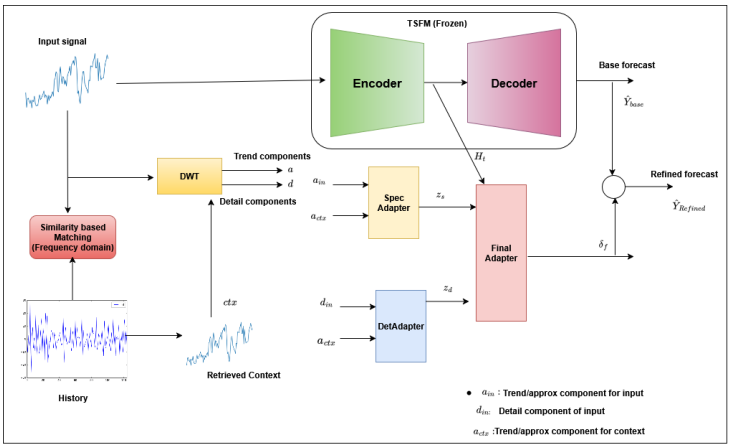



Figure 1: Dynamic two-stage adaptation: (1) decompose the input window into spectral component and detail components; (2) retrieve matching historical windows by frequency signature and integrate them with both components; (3) fuse both streams into a frozen TSFM via adapters

long-range dependencies and rare patterns beyond the fixed receptive field. Both streams are fused into the pre-trained TSFM through lightweight adapters, enriching its representation with the core TSFM untouched. This design offers three deployment-oriented benefits. (i) Plug-and-play adaptability: Our method augments deployed TSFMs without retraining them or additional compute overhead. (ii) Frequency-driven robustness: by conditioning dynamically on the frequency spectrum of the input windows, our framework captures sharp fluctuations and non-stationary patterns that static windows miss. (iii) Scalability across models: large TSFMs gain improved robustness, while compact TSFMs (e.g., TTM) benefit especially from long-range retrieval. Empirical results across cross domain datasets (zero-shot) and long horizon forecasting benchmarks confirm consistent improvements, with notable gains in spiky, irregular, and high-frequency regimes. By coupling frequency decomposition and context retrieval in the two-stream framework, we provide a dynamic, efficient, and generalizable way to refine deployed TSFMs, enabling reliable real-world forecasting.

## 2 Related Work

**TSFMs and Adaptation**. TSFMs aim to learn general-purpose representations from massive corpora. Chronos [1] tokenizes series into discrete tokens and trains T5-style transformers, while Moirai [17] employs a masked encoder trained on 27B observations. TimesFM [5] introduces a decoder-only architecture with input patching, and Time-MoE [13] scales this idea via sparse mixture-of-experts. Tiny Time Mixers (TTMs) [6] demonstrate that even lightweight MLP-Mixer–style models (<1M parameters) can achieve strong zero-/few-shot results. Despite their success, these models operate primarily in the time domain with fixed input windows, limiting their ability to capture transient bursts or rare frequency patterns. Adaptation methods have explored parameter-efficient fine-tuning (adapters, LoRA) [10, 8], in-context learning [3], and curriculum-based strategies such as CCL [11]. While effective, these approaches still rely on time-domain adjustments. Some recent works leverage decomposition signals (e.g., trends, seasonality) in diffusion models [12], however involve training models from scratch rather than augmenting existing TSFMs. TSPulse [7] employs a dual-space masked reconstruction strategy, learning from time and frequency domains to capture complementary signal structures in a unified embedding space; however, it is pre-trained from scratch, and does not consume historical context with varying lengths.

**Frequency Domain Analysis**. Independent of TSFMs, frequency-aware models explicitly exploit spectral structure. FEDformer [19] combines sparse Fourier attention with seasonal-trend decomposition, LaST [16] disentangles latent trends via variational autoencoding, and CoST [18] applies contrastive learning across time and frequency. More integrated designs like TimeMixer++ [15] and TEMPO [2] fuse temporal and spectral patterns through multi-resolution mixing or structured decomposition. While effective, these methods rely on fixed transformation schemes and require full model retraining, limiting their adaptability to accommodate temporal dynamics.

## 3 Proposed Solution

We propose a **context-guided two-stream adaptation framework** that refines forecasts from any frozen TSFM $f_\theta$ without retraining, as demonstrated in Figure 1. Given a query window

$X_{t-k+1:t} \in \mathbb{R}^{k\times C}$, the base TSFM produces

$$\hat{Y}_{\text{base}} = f_\theta(X_{t-k+1:t}), \quad \hat{Y}_{\text{base}} \in \mathbb{R}^{h\times C}. \tag{1}$$

**Frequency Decomposition.** Each window is decomposed into low-frequency trend $a^{(0)}$ and high-frequency details $d^{(1:L)}$ by DWT:

$$[a^{(0)}, d^{(1)}, \ldots, d^{(L)}] = \mathrm{DWT}(X_{t-k+1:t}). \tag{2}$$

**Context Retrieval.** To capture long-range patterns, we compute FFT-based embeddings and retrieve the top-1 most similar historical window $H^*$ as:

$$H^* = \arg\max_{H_i \in \mathcal{H}} \frac{\langle\, |\mathrm{FFT}(X)|, |\mathrm{FFT}(H_i)| \,\rangle}{\|\,\mathrm{FFT}(X)\|\;\|\,\mathrm{FFT}(H_i)\|}. \tag{3}$$

We retrieve the adjacent window of the best-matched context $H^*$, denoted as $ctx$. Using DWT, we then decompose both the input and context signals into approximation and detail components, represented as ($a_{in}$, $d_{in}$ ) and ($a_{ctx}$, $d_{ctx}$ ), respectively.

**Two-Stream Integration.** Spectral Adapter stream encodes trend-level signals:

$$z_s = \mathrm{SpecAdapter}([\,\mathrm{FFT}(a_{\text{in}}^{(0)}), \mathrm{FFT}(a_{\text{ctx}}^{(0)})\,]), \tag{4}$$

while the detail stream encodes fluctuations:

$$z_d = \mathrm{DetAdapter}([\,d_{\text{in}}^{(1:L)}, d_{\text{ctx}}^{(1:L)}\,]). \tag{5}$$

**Fusion with TSFM.** Hidden states of the frozen TSFM are enriched with spectral features:

$$\tilde{H}_t = f_\theta^{\text{enc}}(X_{t-k+1:t}) \tag{6}$$

yielding a refined offset forecast as:

$$\delta_f = FinalAdapter((\tilde{H}_t), z_s, z_t). \tag{7}$$

Then the final refined forecasting is given by:

$$\hat{Y}_{\text{refined}} = \hat{Y}_{\text{base}} + \delta_f \tag{1}$$

**Training Objective.** All Adapters are optimized with a loss:

$$\mathcal{L} = \frac{1}{hc}\|Y - \hat{Y}_{refined}\|_2^2 \tag{8}$$

ensuring both accurate forecasts and alignment of high-frequency details.

# 4 Experimental Setup and Results

We follow the similar setting as TimeMixer++ [15], evaluating four horizons (96, 192, 336, 720). In our fine-tuning setup, we train each frozen foundation model (Chronos-Bolt, TimesFM, TTM) for just three epochs using our two-stream adapters. We present the average MSE and MAE across all horizon sizes here, while the appendix (Table 3) provides the detailed results for each individual horizon. In the zero-shot setup, we fine-tune one series (e.g., ETTh1) and directly test on unseen ones (e.g., ETTh2, ETTm1). To demonstrate the robustness of our approach, we report average results over three independent runs.

Table 1 reports long-term forecasting results, showing consistent improvements over most classical baselines (TimeMixer++, PatchTST, FEDformer) and foundation models (Chronos-bolt, TimesFM). Our two-stream injection framework exploits complementary cues: the frequency stream applies DWT to separate smooth trends from rapid fluctuations, enabling targeted corrections, while the context stream captures long-range temporal dependencies beyond the TSFM's receptive field. Injecting both spectral and context signals into the frozen model's latent space refines its inductive biases, preserving short-range accuracy while enhancing medium- and long-horizon forecasts. For short-term settings, results in Appendix Table 5 confirm similar gains over baselines.

## 4.1 Zero-Shot Forecasting

Table 2 reports zero-shot forecasting under the same TimeMixer++ setup, where our method achieves significant improvements. This cross-dataset success arises because both streams learn abstractions that transfer across domains. This is due to our adapters learn to correct universal frequency motifs (e.g., seasonal cycles vs. noise) and distill recent history into a compact context embedding.

### 4.2 Refining the Foundation Model's Base Forecast

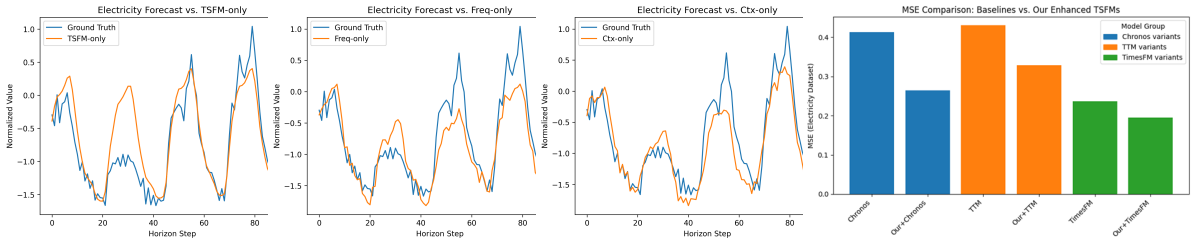
In Tables 1, 2, we report results using the Chronos-Bolt TSFM. To demonstrate the generalizability of our approach, we also evaluated it with TimesFM and TTM, with the corresponding results presented in the Figure 2 and Appendix Table 6. These results clearly show that our method significantly refines the base forecasts produced by pre-trained TSFMs.

Table 1: Long-term forecasting results. We average over horizons {96,192,336,720}. Best results are in red, second best in blue.

| | Proposed (Ours) | | Chronos-Bolt (2024) | | TimesFM (2024) | | TimeMixer++ | | TimeMixer (2024) | | iTransformer (2024) | | PatchTST (2023) | | Crossformer (2023) | | TiDE (2023) | | TimesNet (2023) | | DLinear (2022) | | SCINet (2022) | | FEDformer (2022) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Dataset | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE |
| Electricity | 0.238 | 0.340 | 0.326 | 0.391 | 0.285 | 0.357 | 0.165 | 0.253 | 0.182 | 0.272 | 0.178 | 0.270 | 0.205 | 0.290 | 0.244 | 0.334 | 0.251 | 0.344 | 0.192 | 0.295 | 0.212 | 0.300 | 0.268 | 0.365 | 0.214 | 0.327 |
| ETT (Avg) | 0.236 | 0.283 | 0.276 | 0.325 | 0.427 | 0.389 | 0.349 | 0.399 | 0.367 | 0.388 | 0.383 | 0.377 | 0.381 | 0.397 | 0.685 | 0.578 | 0.482 | 0.470 | 0.391 | 0.404 | 0.442 | 0.444 | 0.689 | 0.597 | 0.408 | 0.428 |
| Exchange | 0.1560 | 0.2185 | 0.173 | 0.235 | 0.470 | 0.480 | 0.357 | 0.391 | 0.391 | 0.453 | 0.378 | 0.360 | 0.403 | 0.404 | 0.940 | 0.707 | 0.370 | 0.413 | 0.416 | 0.443 | 0.354 | 0.414 | 0.750 | 0.626 | 0.519 | 0.429 |
| Traffic | 0.1756 | 0.2480 | 0.212 | 0.272 | 0.718 | 0.372 | 0.416 | 0.264 | 0.484 | 0.297 | 0.428 | 0.282 | 0.481 | 0.304 | 0.550 | 0.304 | 0.760 | 0.473 | 0.620 | 0.336 | 0.625 | 0.383 | 0.804 | 0.509 | 0.610 | 0.376 |
| Weather | 0.091 | 0.073 | 0.098 | 0.0920 | 0.343 | 0.365 | 0.226 | 0.262 | 0.240 | 0.271 | 0.258 | 0.278 | 0.259 | 0.281 | 0.259 | 0.315 | 0.271 | 0.320 | 0.259 | 0.287 | 0.265 | 0.317 | 0.292 | 0.363 | 0.309 | 0.360 |
| Solar-Energy | 0.184 | 0.209 | 0.310 | 0.352 | 0.559 | 0.503 | 0.203 | 0.238 | 0.216 | 0.280 | 0.233 | 0.262 | 0.270 | 0.307 | 0.641 | 0.639 | 0.347 | 0.417 | 0.301 | 0.319 | 0.330 | 0.401 | 0.282 | 0.375 | 0.291 | 0.381 |

Table 2: Zero-shot learning results. Results are averaged over horizons {96,192,336,720}. Best results are in red, second best in blue.

| | Proposed Approach | | Chronos-Bolt (2024) | | TimesFM (2024) | | TimeMixer++ | | TimeMixer (2024) | | LLMTime (2023) | | DLinear (2023) | | PatchTST (2023) | | TimesNet (2023) | | iTransformer (2024) | | Crossformer (2023) | | Time-LLM (2024) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Transfer Task | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE |
| $ETT_{h1\rightarrow h2}$ | 0.260 | 0.322 | 0.297 | 0.416 | 0.446 | 0.454 | 0.367 | 0.391 | 0.427 | 0.424 | 0.992 | 0.708 | 0.493 | 0.488 | 0.380 | 0.405 | 0.421 | 0.431 | 0.481 | 0.474 | 0.555 | 0.574 | 0.353 | 0.387 |
| $ETT_{h1\rightarrow m2}$ | 0.241 | 0.346 | 0.285 | 0.405 | 0.478 | 0.437 | 0.301 | 0.357 | 0.361 | 0.397 | 1.867 | 0.869 | 0.415 | 0.452 | 0.314 | 0.360 | 0.327 | 0.361 | 0.311 | 0.361 | 0.613 | 0.629 | 0.273 | 0.340 |
| $ETT_{h2\rightarrow h1}$ | 0.198 | 0.341 | 0.233 | 0.367 | 0.510 | 0.455 | 0.511 | 0.498 | 0.679 | 0.577 | 1.961 | 0.981 | 0.703 | 0.574 | 0.565 | 0.513 | 0.865 | 0.621 | 0.552 | 0.511 | 0.587 | 0.518 | 0.479 | 0.474 |
| $ETT_{m1\rightarrow h2}$ | 0.318 | 0.407 | 0.356 | 0.416 | 0.431 | 0.428 | 0.417 | 0.422 | 0.452 | 0.441 | 0.992 | 0.708 | 0.464 | 0.475 | 0.439 | 0.438 | 0.457 | 0.454 | 0.434 | 0.438 | 0.624 | 0.541 | 0.381 | 0.412 |
| $ETT_{m1\rightarrow m2}$ | 0.180 | 0.320 | 0.287 | 0.408 | 0.441 | 0.414 | 0.291 | 0.331 | 0.329 | 0.357 | 1.867 | 0.869 | 0.335 | 0.389 | 0.296 | 0.334 | 0.322 | 0.354 | 0.324 | 0.331 | 0.595 | 0.572 | 0.268 | 0.320 |
| $ETT_{m2\rightarrow m1}$ | 0.205 | 0.304 | 0.216 | 0.318 | 0.417 | 0.405 | 0.427 | 0.448 | 0.554 | 0.478 | 1.933 | 0.984 | 0.649 | 0.537 | 0.568 | 0.492 | 0.769 | 0.567 | 0.559 | 0.491 | 0.611 | 0.593 | 0.414 | 0.438 |




(a) Only Chronos-bolt (b) + Frequency (c) + Context (d) Comparison with FMs

Figure 2: Visualization of forecasting results: (a) only Chronos-bolt, (b) with frequency component, (c) with context information, and (d) comparison with other TSFMs.

## 5 Ablation Study

Table 6 shows that frequency adapters reduce local errors, context adapters improve long-horizon coherence, and their combination yields the best overall performance in both zero-shot and fine-tuned settings. Figure 2 shows that while the frozen TSFM drifts, frequency-only captures bursts but misses trends, and context-only preserves slope but ignores spikes. Only the full two-stream model aligns with both global and local patterns, highlighting the benefit of combining frequency and context. We varied context window sizes (96, 192, 336, 720) with a fixed query of 96 and horizon of 192. Figure 3 shows performance drops when the window is much shorter or longer than the horizon, with the best results when both are similar. We report the computational cost analysis in Table 8 of the Appendix.

## 6 Conclusion and Limitations

We proposed a dynamic two-stage adaptation framework that augments pre-trained time-series foundation models (TSFMs) with explicit frequency decomposition and retrieval-based long-range context. By integrating short-term spectral features with similar historical patterns via lightweight adapter modules, our method enables zero-shot forecasting without modifying TSFM weights. This design captures both local transients and rare long-range signals.

However, our approach has some limitations. Spectral retrieval may miss semantically relevant patterns in noisy or irregular time series. The added adapters, while efficient, still introduce some inference overhead. Moreover, the framework assumes that informative frequency structures exist—an assumption that may not hold in highly non-periodic or abrupt series. Future work could explore more flexible retrieval objectives and hybrid feature integration to address these challenges.

# References


[1] A. F. Ansari, L. Stella, A. C. Turkmen, X. Zhang, P. Mercado, H. Shen, O. Shchur, S. S. Rangapuram, S. P. Arango, S. Kapoor, et al. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

[2] D. Cao, F. Jia, S. O. Arik, T. Pfister, Y. Zheng, W. Ye, and Y. Liu. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. In *The Twelfth International Conference on Learning Representations*.

[3] A. Das, M. Faw, R. Sen, and Y. Zhou. In-context fine-tuning for time-series foundation models. *arXiv preprint arXiv:2410.24087*, 2024.

[4] A. Das, W. Kong, R. Sen, and Y. Zhou. A decoder-only foundation model for time-series forecasting, 2024.

[5] A. Das, W. Kong, R. Sen, and Y. Zhou. A decoder-only foundation model for time-series forecasting. In *Forty-first International Conference on Machine Learning*, 2024.

[6] V. Ekambaram, A. Jati, P. Dayama, S. Mukherjee, N. Nguyen, W. M. Gifford, C. Reddy, and J. Kalagnanam. Tiny time mixers (ttms): Fast pre-trained models for enhanced zero/few-shot forecasting of multivariate time series. *Advances in Neural Information Processing Systems*, 37:74147–74181, 2024.

[7] V. Ekambaram, S. Kumar, A. Jati, S. Mukherjee, T. Sakai, P. Dayama, W. M. Gifford, and J. Kalagnanam. Tspulse: Dual space tiny pre-trained models for rapid time-series analysis. *arXiv preprint arXiv:2505.13033*, 2025.

[8] M. Germán-Morales, A. Rivera-Rivas, M. del Jesus Díaz, and C. Carmona. Transfer learning with foundational models for time series forecasting using low-rank adaptations. *Information Fusion*, page 103247, 2025.

[9] M. Goswami, K. Szafer, A. Choudhry, Y. Cai, S. Li, and A. Dubrawski. Moment: A family of open time-series foundation models. In *International Conference on Machine Learning*, pages 16115–16152. PMLR, 2024.

[10] D. Gupta, A. Bhatti, and S. Parmar. Beyond lora: Exploring efficient fine-tuning techniques for time series foundational models. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

[11] R. Liang, Y. Deng, D. Xie, F. He, and D. Wang. Enabling time-series foundation model for building energy forecasting via contrastive curriculum learning. *arXiv preprint arXiv:2412.17285*, 2024.

[12] L. Shen, W. Chen, and J. Kwok. Multi-resolution diffusion models for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024.

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

[14] V. Sovrasov. ptflops: a flops counting tool for neural networks in pytorch framework, 2018-2024.

[15] S. Wang, J. Li, X. Shi, Z. Ye, B. Mo, W. Lin, S. Ju, Z. Chu, and M. Jin. Timemixer++: A general time series pattern machine for universal predictive analysis. *arXiv preprint arXiv:2410.16032*, 2024.

[16] Z. Wang, X. Xu, W. Zhang, G. Trajcevski, T. Zhong, and F. Zhou. Learning latent seasonal-trend representations for time series forecasting. *Advances in Neural Information Processing Systems*, 35:38775–38787, 2022.

[17] G. Woo, C. Liu, A. Kumar, C. Xiong, S. Savarese, and D. Sahoo. Unified training of universal time series forecasting transformers. In *International Conference on Machine Learning*, pages 53140–53164. PMLR, 2024.

[18] G. Woo, C. Liu, D. Sahoo, A. Kumar, and S. Hoi. Cost: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. In *International Conference on Learning Representations*, 2022.

[19] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International conference on machine learning*, pages 27268–27286. PMLR, 2022.

# A Appendix

# B Long-term forecasting

Table 3 presents comprehensive results for the long-term forecasting task, comparing a wide range of competitive models across various prediction lengths. The 'Avg' column represents the average performance over all four prediction horizons: 96, 192, 336, and 720. Our proposed approach consistently outperforms existing state-of-the-art methods.

Table 3: Full results for the long-term forecasting task.

| **Dataset** | **Len** | **Ours (MSE)** | **Ours (MAE)** | **TimeMixer++** | | **TimeMixer** | | **iTrans** | | **PatchTST** | | **CrossF** | | **TiDE** | | **TimesNet** | | **DLinear** | | **SCINet** | | **FEDf** | | **Stationary** | | **Autof** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE | | |
| Weather | 96 | 0.0049 | 0.0544 | 0.155 | 0.205 | 0.163 | 0.209 | 0.174 | 0.214 | 0.186 | 0.227 | 0.195 | 0.271 | 0.202 | 0.261 | 0.172 | 0.220 | 0.195 | 0.252 | 0.221 | 0.306 | 0.217 | 0.296 | 0.173 | 0.223 | 0.266 | 0.336 |
| | 192 | 0.0102 | 0.0837 | 0.201 | 0.245 | 0.208 | 0.250 | 0.221 | 0.254 | 0.234 | 0.265 | 0.209 | 0.277 | 0.242 | 0.298 | 0.219 | 0.261 | 0.237 | 0.295 | 0.261 | 0.340 | 0.276 | 0.336 | 0.245 | 0.285 | 0.307 | 0.367 |
| | 336 | 0.0095 | 0.0791 | 0.237 | 0.265 | 0.251 | 0.287 | 0.278 | 0.296 | 0.284 | 0.301 | 0.273 | 0.332 | 0.287 | 0.335 | 0.280 | 0.306 | 0.282 | 0.331 | 0.309 | 0.378 | 0.339 | 0.380 | 0.321 | 0.338 | 0.359 | 0.395 |
| | 720 | 0.0089 | 0.0749 | 0.312 | 0.334 | 0.339 | 0.341 | 0.358 | 0.347 | 0.356 | 0.349 | 0.379 | 0.401 | 0.351 | 0.386 | 0.365 | 0.359 | 0.345 | 0.382 | 0.377 | 0.427 | 0.403 | 0.428 | 0.414 | 0.410 | 0.419 | 0.428 |
| | Avg | 0.0084 | 0.0730 | 0.226 | 0.262 | 0.240 | 0.271 | 0.258 | 0.278 | 0.265 | 0.285 | 0.264 | 0.320 | 0.271 | 0.320 | 0.259 | 0.287 | 0.265 | 0.315 | 0.292 | 0.363 | 0.309 | 0.360 | 0.288 | 0.314 | 0.338 | 0.382 |
| Electricity | 96 | 0.2625 | 0.3730 | 0.135 | 0.222 | 0.153 | 0.247 | 0.148 | 0.240 | 0.190 | 0.296 | 0.219 | 0.314 | 0.237 | 0.329 | 0.168 | 0.272 | 0.210 | 0.302 | 0.247 | 0.345 | 0.193 | 0.308 | 0.169 | 0.273 | 0.201 | 0.317 |
| | 192 | 0.2371 | 0.3583 | 0.147 | 0.235 | 0.166 | 0.256 | 0.162 | 0.253 | 0.199 | 0.304 | 0.231 | 0.322 | 0.236 | 0.330 | 0.184 | 0.322 | 0.210 | 0.305 | 0.257 | 0.355 | 0.201 | 0.315 | 0.182 | 0.286 | 0.222 | 0.334 |
| | 336 | 0.2608 | 0.3719 | 0.164 | 0.245 | 0.185 | 0.277 | 0.178 | 0.269 | 0.217 | 0.319 | 0.246 | 0.337 | 0.249 | 0.344 | 0.198 | 0.300 | 0.223 | 0.319 | 0.269 | 0.369 | 0.214 | 0.329 | 0.200 | 0.304 | 0.231 | 0.443 |
| | 720 | 0.3138 | 0.4206 | 0.212 | 0.310 | 0.225 | 0.310 | 0.225 | 0.317 | 0.258 | 0.352 | 0.280 | 0.363 | 0.284 | 0.373 | 0.220 | 0.320 | 0.258 | 0.350 | 0.299 | 0.390 | 0.246 | 0.355 | 0.222 | 0.321 | 0.254 | 0.361 |
| | Avg | 0.2686 | 0.3809 | 0.165 | 0.253 | 0.182 | 0.272 | 0.178 | 0.270 | 0.216 | 0.318 | 0.244 | 0.334 | 0.251 | 0.344 | 0.192 | 0.304 | 0.225 | 0.319 | 0.268 | 0.365 | 0.214 | 0.327 | 0.193 | 0.296 | 0.227 | 0.338 |
| Traffic | 96 | 0.2574 | 0.3102 | 0.392 | 0.253 | 0.462 | 0.285 | 0.395 | 0.268 | 0.526 | 0.347 | 0.644 | 0.429 | 0.805 | 0.493 | 0.593 | 0.321 | 0.650 | 0.396 | 0.788 | 0.499 | 0.587 | 0.366 | 0.612 | 0.338 | 0.613 | 0.388 |
| | 192 | 0.2616 | 0.2956 | 0.402 | 0.258 | 0.473 | 0.296 | 0.417 | 0.276 | 0.522 | 0.332 | 0.665 | 0.431 | 0.756 | 0.474 | 0.617 | 0.336 | 0.598 | 0.370 | 0.789 | 0.505 | 0.604 | 0.373 | 0.613 | 0.340 | 0.616 | 0.382 |
| | 336 | 0.2363 | 0.2925 | 0.428 | 0.263 | 0.498 | 0.296 | 0.433 | 0.283 | 0.517 | 0.334 | 0.674 | 0.420 | 0.762 | 0.477 | 0.629 | 0.336 | 0.605 | 0.373 | 0.797 | 0.508 | 0.621 | 0.383 | 0.618 | 0.328 | 0.622 | 0.337 |
| | 720 | 0.2645 | 0.3267 | 0.441 | 0.282 | 0.506 | 0.313 | 0.467 | 0.302 | 0.552 | 0.352 | 0.683 | 0.424 | 0.719 | 0.449 | 0.640 | 0.350 | 0.645 | 0.394 | 0.841 | 0.523 | 0.626 | 0.382 | 0.653 | 0.355 | 0.660 | 0.408 |
| | Avg | 0.25495 | 0.30625 | 0.416 | 0.264 | 0.484 | 0.297 | 0.428 | 0.282 | 0.529 | 0.341 | 0.667 | 0.426 | 0.760 | 0.473 | 0.620 | 0.336 | 0.625 | 0.383 | 0.804 | 0.509 | 0.610 | 0.376 | 0.624 | 0.340 | 0.628 | 0.379 |
| Exchange | 96 | 0.0680 | 0.1968 | 0.085 | 0.214 | 0.090 | 0.235 | 0.086 | 0.206 | 0.088 | 0.205 | 0.256 | 0.367 | 0.094 | 0.218 | 0.107 | 0.234 | 0.088 | 0.218 | 0.267 | 0.396 | 0.148 | 0.278 | 0.111 | 0.237 | 0.197 | 0.323 |
| | 192 | 0.1000 | 0.2409 | 0.175 | 0.313 | 0.187 | 0.343 | 0.177 | 0.299 | 0.176 | 0.299 | 0.470 | 0.509 | 0.184 | 0.307 | 0.226 | 0.344 | 0.176 | 0.315 | 0.351 | 0.459 | 0.271 | 0.315 | 0.219 | 0.335 | 0.300 | 0.369 |
| | 336 | 0.2429 | 0.3837 | 0.316 | 0.420 | 0.353 | 0.473 | 0.331 | 0.417 | 0.301 | 0.397 | 1.268 | 0.883 | 0.349 | 0.431 | 0.367 | 0.448 | 0.313 | 0.427 | 1.324 | 0.853 | 0.460 | 0.427 | 0.421 | 0.476 | 0.509 | 0.524 |
| | 720 | 0.4022 | 0.5270 | 0.851 | 0.689 | 0.934 | 0.761 | 0.847 | 0.691 | 0.901 | 0.714 | 1.767 | 1.068 | 0.852 | 0.698 | 0.964 | 0.746 | 0.839 | 0.695 | 1.058 | 0.797 | 1.195 | 0.695 | 1.092 | 0.769 | 1.447 | 0.941 |
| | Avg | 0.203275 | 0.3371 | 0.357 | 0.391 | 0.391 | 0.453 | 0.360 | 0.403 | 0.367 | 0.404 | 0.940 | 0.707 | 0.370 | 0.413 | 0.416 | 0.443 | 0.354 | 0.414 | 0.750 | 0.626 | 0.519 | 0.429 | 0.461 | 0.454 | 0.613 | 0.539 |
| ETTh1 | 96 | 0.1722 | 0.3044 | 0.361 | 0.403 | 0.375 | 0.400 | 0.386 | 0.405 | 0.460 | 0.447 | 0.423 | 0.448 | 0.479 | 0.464 | 0.384 | 0.402 | 0.397 | 0.412 | 0.654 | 0.599 | 0.395 | 0.424 | 0.513 | 0.491 | 0.449 | 0.459 |
| | 192 | 0.1827 | 0.3242 | 0.416 | 0.441 | 0.429 | 0.421 | 0.441 | 0.512 | 0.477 | 0.429 | 0.471 | 0.474 | 0.525 | 0.492 | 0.436 | 0.429 | 0.446 | 0.441 | 0.719 | 0.631 | 0.469 | 0.470 | 0.534 | 0.504 | 0.500 | 0.482 |
| | 336 | 0.1956 | 0.3382 | 0.430 | 0.434 | 0.484 | 0.458 | 0.487 | 0.458 | 0.546 | 0.496 | 0.570 | 0.546 | 0.565 | 0.515 | 0.491 | 0.469 | 0.489 | 0.467 | 0.778 | 0.659 | 0.530 | 0.499 | 0.588 | 0.535 | 0.521 | 0.496 |
| | 720 | 0.1980 | 0.3413 | 0.467 | 0.451 | 0.498 | 0.482 | 0.503 | 0.491 | 0.544 | 0.517 | 0.653 | 0.621 | 0.594 | 0.558 | 0.521 | 0.500 | 0.513 | 0.510 | 0.836 | 0.699 | 0.598 | 0.544 | 0.643 | 0.616 | 0.514 | 0.512 |
| | Avg | 0.1871 | 0.3270 | 0.419 | 0.432 | 0.447 | 0.440 | 0.454 | 0.447 | 0.516 | 0.484 | 0.529 | 0.522 | 0.541 | 0.507 | 0.458 | 0.450 | 0.461 | 0.457 | 0.747 | 0.647 | 0.498 | 0.484 | 0.570 | 0.537 | 0.496 | 0.487 |
| ETTh2 | 96 | 0.2144 | 0.3584 | 0.276 | 0.328 | 0.289 | 0.341 | 0.297 | 0.349 | 0.308 | 0.355 | 0.745 | 0.584 | 0.400 | 0.440 | 0.340 | 0.374 | 0.340 | 0.394 | 0.707 | 0.621 | 0.358 | 0.397 | 0.476 | 0.458 | 0.346 | 0.388 |
| | 192 | 0.2509 | 0.3902 | 0.342 | 0.379 | 0.372 | 0.392 | 0.380 | 0.400 | 0.393 | 0.405 | 0.877 | 0.656 | 0.528 | 0.509 | 0.402 | 0.414 | 0.482 | 0.479 | 0.860 | 0.689 | 0.429 | 0.439 | 0.512 | 0.493 | 0.456 | 0.452 |
| | 336 | 0.2675 | 0.4017 | 0.346 | 0.398 | 0.386 | 0.414 | 0.428 | 0.432 | 0.427 | 0.436 | 1.043 | 0.731 | 0.643 | 0.571 | 0.452 | 0.452 | 0.591 | 0.541 | 1.000 | 0.744 | 0.496 | 0.487 | 0.552 | 0.551 | 0.482 | 0.486 |
| | 720 | 0.2806 | 0.3992 | 0.392 | 0.415 | 0.412 | 0.434 | 0.427 | 0.445 | 0.436 | 0.450 | 1.104 | 0.763 | 0.874 | 0.679 | 0.462 | 0.468 | 0.839 | 0.661 | 1.249 | 0.838 | 0.463 | 0.474 | 0.562 | 0.560 | 0.515 | 0.511 |
| | Avg | 0.2534 | 0.3874 | 0.339 | 0.380 | 0.364 | 0.395 | 0.383 | 0.407 | 0.391 | 0.411 | 0.942 | 0.684 | 0.611 | 0.550 | 0.414 | 0.427 | 0.563 | 0.519 | 0.954 | 0.723 | 0.437 | 0.449 | 0.526 | 0.516 | 0.450 | 0.459 |
| ETTm1 | 96 | 0.1079 | 0.2380 | 0.310 | 0.334 | 0.320 | 0.357 | 0.334 | 0.368 | 0.352 | 0.374 | 0.404 | 0.426 | 0.364 | 0.387 | 0.338 | 0.375 | 0.346 | 0.374 | 0.418 | 0.438 | 0.379 | 0.419 | 0.386 | 0.398 | 0.505 | 0.475 |
| | 192 | 0.1562 | 0.2902 | 0.348 | 0.362 | 0.361 | 0.381 | 0.390 | 0.393 | 0.374 | 0.387 | 0.450 | 0.451 | 0.398 | 0.404 | 0.374 | 0.387 | 0.382 | 0.391 | 0.439 | 0.450 | 0.426 | 0.441 | 0.459 | 0.444 | 0.553 | 0.496 |
| | 336 | 0.2206 | 0.3533 | 0.376 | 0.391 | 0.390 | 0.404 | 0.426 | 0.420 | 0.421 | 0.414 | 0.532 | 0.515 | 0.428 | 0.425 | 0.410 | 0.411 | 0.415 | 0.415 | 0.490 | 0.485 | 0.445 | 0.459 | 0.495 | 0.464 | 0.621 | 0.537 |
| | 720 | 0.2395 | 0.3709 | 0.440 | 0.423 | 0.454 | 0.441 | 0.491 | 0.459 | 0.462 | 0.449 | 0.666 | 0.589 | 0.487 | 0.461 | 0.478 | 0.450 | 0.473 | 0.451 | 0.595 | 0.550 | 0.543 | 0.490 | 0.585 | 0.516 | 0.671 | 0.561 |
| | Avg | 0.1811 | 0.3131 | 0.369 | 0.378 | 0.381 | 0.395 | 0.407 | 0.410 | 0.406 | 0.407 | 0.513 | 0.495 | 0.419 | 0.419 | 0.400 | 0.406 | 0.404 | 0.408 | 0.485 | 0.481 | 0.448 | 0.452 | 0.481 | 0.456 | 0.588 | 0.517 |
| ETTm2 | 96 | 0.2131 | 0.3455 | 0.170 | 0.245 | 0.175 | 0.258 | 0.180 | 0.264 | 0.183 | 0.270 | 0.287 | 0.366 | 0.207 | 0.305 | 0.187 | 0.267 | 0.193 | 0.293 | 0.286 | 0.377 | 0.203 | 0.287 | 0.192 | 0.274 | 0.255 | 0.339 |
| | 192 | 0.2014 | 0.3353 | 0.229 | 0.291 | 0.237 | 0.299 | 0.250 | 0.309 | 0.255 | 0.314 | 0.414 | 0.492 | 0.290 | 0.364 | 0.249 | 0.309 | 0.284 | 0.361 | 0.399 | 0.445 | 0.269 | 0.328 | 0.280 | 0.339 | 0.281 | 0.340 |
| | 336 | 0.2983 | 0.4137 | 0.303 | 0.343 | 0.298 | 0.340 | 0.311 | 0.348 | 0.309 | 0.347 | 0.597 | 0.542 | 0.377 | 0.422 | 0.321 | 0.351 | 0.382 | 0.429 | 0.637 | 0.591 | 0.325 | 0.366 | 0.334 | 0.361 | 0.339 | 0.372 |
| | 720 | 0.3098 | 0.4120 | 0.373 | 0.399 | 0.391 | 0.396 | 0.412 | 0.407 | 0.412 | 0.404 | 1.730 | 1.042 | 0.558 | 0.524 | 0.408 | 0.403 | 0.558 | 0.525 | 0.960 | 0.735 | 0.421 | 0.415 | 0.417 | 0.413 | 0.433 | 0.432 |
| | Avg | 0.2557 | 0.3766 | 0.269 | 0.320 | 0.275 | 0.323 | 0.288 | 0.332 | 0.290 | 0.334 | 0.757 | 0.610 | 0.358 | 0.404 | 0.291 | 0.333 | 0.354 | 0.402 | 0.954 | 0.723 | 0.305 | 0.349 | 0.306 | 0.347 | 0.327 | 0.371 |

Table 4: Ablation study: contribution of each component on four datasets (MSE / MAE).

| Dataset | Base FM | | FM + Frequency | | FM + Context | | FM + Frequency + Context | |
|---|---|---|---|---|---|---|---|---|
| | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE |
| Electricity | 0.295 | 0.379 | 0.232 | 0.351 | 0.235 | 0.353 | 0.227 | 0.346 |
| Traffic | 0.237 | 0.261 | 0.220 | 0.290 | 0.249 | 0.305 | 0.201 | 0.223 |
| Exchange | 0.202 | 0.347 | 0.0912 | 0.228 | 0.0887 | 0.225 | 0.069 | 0.198 |
| Weather | 0.021 | 0.0934 | 0.0054 | 0.0614 | 0.0067 | 0.0679 | 0.0102 | 0.0837 |

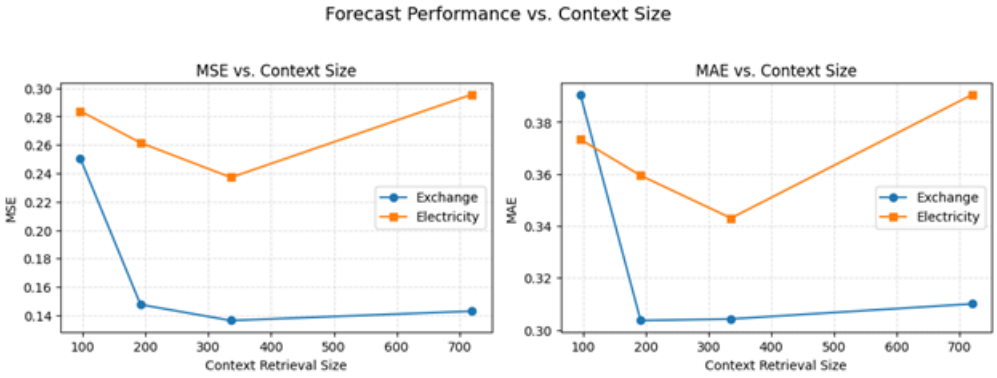



Figure 3: Effect of context size in the forecasting.

# C Short-term forecasting

Table 5 presents the experimental results on the M4 dataset for short-term forecasting, demonstrating that our proposed approach leads to significant improvements in forecasting performance.

Table 5: Short-term forecasting results on the M4 dataset (single variate, prediction lengths in [6, 48]). Lower SMAPE and MASE indicate better performance.

| Frequency | Metric | Ours | TimeMixer++ (Ours) | TimeMixer (2024b) | iTrans (2024) | TiDE (2023a) | TimesNet (2023) | N-HiTS (2023) | N-BEATS⋆ (2019) | PatchTST (2023) | MICN (2023a) | FiLM (2022a) | LightTS (2022a) | DLinear (2023) | FED. (2022b) | Stationary (2022c) | Auto. (2021) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Yearly | SMAPE | **11.675** | 13.179 | 13.206 | 13.923 | 15.320 | 13.387 | 13.418 | 13.436 | 16.463 | 25.022 | 17.431 | 14.247 | 16.965 | 13.728 | 13.717 | 13.974 |
| | MASE | **2.147** | 2.934 | 2.916 | 3.214 | 3.540 | 2.996 | 3.045 | 3.043 | 3.967 | 7.162 | 4.043 | 3.109 | 4.283 | 3.048 | 3.078 | 3.134 |
| Quarterly | SMAPE | **3.3665** | 9.755 | 9.996 | 10.757 | 11.830 | 10.100 | 10.202 | 10.124 | 10.644 | 15.214 | 12.925 | 11.364 | 12.145 | 10.792 | 10.958 | 11.338 |
| | MASE | **1.3325** | 1.159 | 1.166 | 1.283 | 1.410 | 1.182 | 1.194 | 1.169 | 1.278 | 1.963 | 1.664 | 1.328 | 1.520 | 1.283 | 1.325 | 1.365 |
| Monthly | SMAPE | **1.4036** | 12.432 | 12.605 | 13.796 | 15.180 | 12.670 | 12.791 | 12.677 | 13.399 | 16.943 | 15.407 | 14.014 | 13.514 | 14.260 | 13.917 | 13.958 |
| | MASE | **0.8780** | 0.904 | 0.919 | 1.083 | 1.190 | 0.933 | 0.969 | 0.937 | 1.031 | 1.442 | 1.298 | 1.053 | 1.037 | 1.102 | 1.097 | 1.103 |
| Others | SMAPE | **2.980** | 4.698 | 4.564 | 5.569 | 6.120 | 4.891 | 5.061 | 4.925 | 6.558 | 41.985 | 7.134 | 15.880 | 6.709 | 4.954 | 6.302 | 5.485 |
| | MASE | **2.150** | 2.931 | 3.115 | 3.940 | 4.330 | 3.302 | 3.216 | 3.391 | 4.511 | 62.734 | 5.090 | 11.434 | 4.953 | 3.264 | 4.064 | 3.865 |

Table 6: Comparison of Chronos-Bolt, TimesFM, Our+TimesFM, and Our+Chronos on six datasets. Lower MSE/MAE indicate better performance.

| Dataset | Chronos-Bolt | | Our+Chronos | | TimesFM | | Our+TimesFM | |
|---|---|---|---|---|---|---|---|---|
| | MSE | MAE | MSE | MAE | MSE | MAE | MSE | MAE |
| Electricity | 0.3260 | 0.3914 | 0.2319 | 0.3424 | 0.1924 | 0.2650 | 0.1654 | 0.2150 |
| Traffic | 0.2126 | 0.2727 | 0.1722 | 0.2572 | 0.6581 | 0.6890 | 0.5868 | 0.3847 |
| Weather | 0.0017 | 0.0287 | 0.0071 | 0.0718 | 0.1488 | 0.2360 | 0.1028 | 0.2116 |
| Exchange | 0.0680 | 0.1947 | 0.0340 | 0.1445 | 0.0924 | 0.1768 | 0.0524 | 0.1739 |
| ETTm1 | 0.1560 | 0.2750 | 0.1079 | 0.2380 | 0.4900 | 0.6860 | 0.4125 | 0.5621 |
| ETTh1 | 0.2133 | 0.3122 | 0.1722 | 0.3044 | 0.4289 | 0.5150 | 0.3924 | 0.4738 |

We evaluated our proposed approach on different versions of the Chronos-Bolt model—mini, small, and tiny to demonstrate that our method is not limited to a specific foundation model variant. As shown in Table 4, our approach significantly improves forecasting performance across all model versions.

## C.1 Contribution of each component

We present the component-wise performance of our approach in Table 4. In our visual comparisons figure 2, the baseline TSFM produces overly smooth straight-line forecasts that miss both rapid spikes and long-term trend shifts. The Freq-only variant corrects high-frequency oscillations and periodic

Table 7: Performance comparison of Chronos-Bolt (Small, Mini, Tiny) and our approach on four datasets. Lower MSE/MAE indicate better performance.

| Dataset | Small | | Mini | | Tiny | |
|---|---|---|---|---|---|---|
| | Ours (MSE/MAE) | CB (MSE/MAE) | Ours (MSE/MAE) | CB (MSE/MAE) | Ours (MSE/MAE) | CB (MSE/MAE) |
| Electricity | 0.2319 / 0.3424 | 0.3260 / 0.3914 | 0.2305 / 0.3413 | 0.3288 / 0.3963 | 0.2351 / 0.3499 | 0.3379 / 0.4059 |
| Traffic | 0.1722 / 0.2572 | 0.2126 / 0.2727 | 0.1728 / 0.2577 | 0.2147 / 0.2740 | 0.1780 / 0.2597 | 0.2191 / 0.2798 |
| Weather | 0.0071 / 0.0718 | 0.0017 / 0.0287 | 0.0071 / 0.0718 | 0.0017 / 0.0291 | 0.0071 / 0.0718 | 0.0017 / 0.0289 |
| Exchange | 0.0340 / 0.1445 | 0.0680 / 0.1947 | 0.0321 / 0.1395 | 0.0664 / 0.1926 | 0.0298 / 0.1363 | 0.0662 / 0.1919 |

bumps, but, lacking context awareness, its forecasts still drift off the true slow trend over longer horizons. Conversely, the Ctx-only model preserves the overall trajectory and prevents drift, but fails to capture transient spikes and oscillations. Only the full two-stream approach combines both strengths, faithfully tracking global trends while adjusting to every local fluctuation.

### C.2 The improvement in forecasting using our proposed approach with FMs

We evaluated our approach in conjunction with the pre-trained Chronos-Bolt, TimesFM, and TTM foundation models. As shown in Figure 2, integrating our method with these pre-trained models leads to significant improvements in forecasting performance.

### C.3 Effect of context window size

We evaluated our approach using various context window sizes to assess their impact on forecasting performance. For this experiment, we fixed the input query size at 96 and the forecasting horizon at 192, while varying the context window size across 96, 192, 336, and 720. The results, illustrated in Figure 3, show that forecasting performance declines when the context window is either smaller or much larger than the forecasting horizon. Notably, our model achieves optimal performance when the context window size is approximately equal to the forecasting horizon, as the extracted context segment most closely aligns with the prediction interval in this scenario.

## D Computational complexity

In this section, we compare the computational complexity of our adapter layers with computational complexities of original TSFM models. We follow common practices and measure the number of floating point operations (FLOPs) in multiply-accumulate operations (MACs) using batch size of 1 in forward pass. We use `ptflops` [14] utility that estimates FLOPs and number of parameters of PyTorch models. Results of comparison are shown in table 8. For family of Chronos models, the number of parameters in our adapters is only 5.52% of the parameters of the original tiny model, 2.56% of the mini model, 1.27% of the small model and 0.36% of the base model. FLOPs vary from 0.69% for the tiny model to only 0.04% for the base model.

Table 8: Comparison of computation complexity (FLOPs measured in multiply-accumulate operations) of our adaptor layers with original TSFM models. FLOPs and parameters are in millions (M).

| Model | Original Model | | MLP adaptors | | As percents of original models | |
|---|---|---|---|---|---|---|
| | parameters (M) | FLOPs (M) | parameters (M) | FLOPs (M) | parameters (%) | FLOPS (%) |
| chronos-bolt-tiny | 8.65 | 137.83 | 0.48 | 0.95 | 5.52 | 0.69 |
| chronos-bolt-mini | 21.24 | 348.08 | 0.54 | 1.08 | 2.56 | 0.31 |
| chronos-bolt-small | 47.72 | 794.38 | 0.61 | 1.21 | 1.27 | 0.15 |
| chronos-bolt-base | 205.29 | 3473.93 | 0.74 | 1.48 | 0.36 | 0.04 |