ROUTETS: BUDGETED SPECTRAL COMPONENT ROUTING ACROSS TIME AND FREQUENCY DOMAINS FOR TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Time-domain and frequency-domain predictors offer complementary inductive biases for recurring patterns and localized variations. However, existing architectures generally determine how the two domains are used through fixed design choices, rather than adapting spectral-component allocation to the input spectrum. We propose RouteTS, a lightweight framework that makes this allocation explicit through amplitude-based component routing. For each input window and variable, RouteTS ranks spectral components by amplitude, sends the Top-K components to a shared complex-valued linear predictor, and transforms the complementary spectrum back to the time domain for modeling by a shared MLP. The dataset-specific routing budget K is selected using only validation data, and varying K yields time-only, mixed-domain, and frequency-only configurations within the same architecture. Across diverse benchmarks, RouteTS achieves the best average MSE on the majority of datasets while remaining among the most parameter- and memory-efficient models evaluated. Controlled analyses further show that the preferred allocation varies across datasets. In the mixed-domain cases examined, assigning dominant spectral components to the frequency predictor and the remainder to the temporal predictor improves accuracy over alternative component selections and reversed assignments.

1 INTRODUCTION

Time series forecasting (TSF) supports a broad range of applications, including energy planning (Martinez Alvarez et al., 2011; Lai et al., 2018), financial markets (He et al., 2023), and traffic management (Cirstea et al., 2021; Wu et al., 2021). A central challenge is the diversity of predictive structures: time series may exhibit recurring patterns, localized variations, abrupt events, or a mixture of these behaviors. A predictor well suited to one type of structure may overlook predictive information carried by others, limiting forecasting accuracy. This diversity raises the question of how to match different components of a signal with suitable predictors.

Existing methods typically determine the modeling domain through architectural choices. Single-domain models adopt either temporal or spectral representations for prediction, while hybrid models combine time- and frequency-domain processing through predefined branch roles (Ma et al., 2026). Recent decomposition and expert-routing methods adapt frequency bands, patches, or latent tokens (Liu, 2025; Sun et al., 2025). However, these designs do not explicitly allocate complementary sets of input spectral components to predictors operating in different domains. This leaves a component-level question: which spectral components should be predicted directly in the frequency domain, and which should be reconstructed and modeled in the time domain?

The different modeling biases of the two domains offer a starting point. Frequency-domain predictors operate on global sinusoidal components, providing a compact representation when recurring patterns concentrate energy in a few frequencies. Time-domain predictors operate directly on temporal samples without requiring such spectral concentration. Figure 1 illustrates this distinction. On Traffic, FITS (Xu et al., 2024) preserves prominent spectral peaks, whereas DLinear (Zeng et al., 2023) exhibits deviations in peak timing and amplitude for the same example. On PM2.5, a localized spike has a broad spectrum; FITS attenuates many components and substantially smooths the spike.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

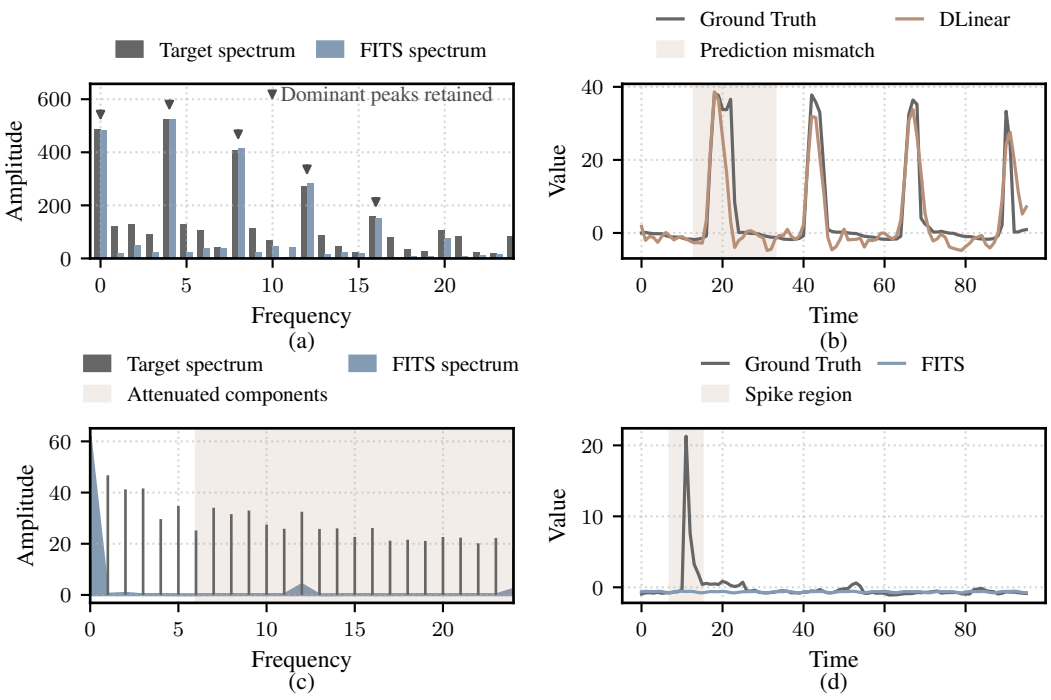


Figure 1: Illustrative forecasting examples. (a) Target and FITS-predicted amplitude spectra on Traffic; arrows mark dominant spectral peaks retained by FITS. (b) DLinear exhibits deviations in peak timing and amplitude for the same example. (c) On PM2.5, FITS attenuates many spectral components associated with a localized spike, producing the smoothed forecast in (d).

These examples motivate predicting dominant spectral components in the frequency domain while retaining the remaining components for temporal modeling. For recurring patterns, spectral amplitude and phase describe the strength and timing of oscillations, making a complex-valued linear mapping a natural choice for predicting the dominant components (Xu et al., 2024). The remaining components can jointly encode localized variations; reconstructing them in the time domain allows an MLP to model their combined temporal structure through a nonlinear mapping. Amplitude ranking makes component selection responsive to the input spectrum, while the number of components assigned to each domain can vary across forecasting settings.

To this end, we propose RouteTS, a lightweight framework for budgeted spectral-component routing. After normalization, RouteTS transforms each input channel into the frequency domain and ranks its components by amplitude. Given a routing budget K, the Top-K components are assigned to a shared complex-valued linear predictor, which jointly transforms their real and imaginary parts. The complementary spectrum is transformed back to the time domain and processed by a shared MLP, providing a nonlinear mapping for the reconstructed remainder. The two forecast contributions are added in the normalized time domain, followed by inverse normalization. Varying K yields time-only ($K = 0$), mixed-domain (intermediate K), and frequency-only ($K = \text{full}$) configurations. For each dataset, we select a single budget by minimizing validation MSE averaged over the forecasting horizons. Although this budget is fixed within a dataset, the selected frequency indices remain input-adaptive because amplitudes are ranked independently for each window and variable.

Our main contribution is an explicit, budget-controlled allocation of spectral components that allows a single forecasting framework to accommodate different dataset preferences for time-domain, frequency-domain, or mixed-domain prediction. Experiments across twelve datasets demonstrate the value of this flexibility: RouteTS achieves the best average MSE on seven datasets, with the winning configurations spanning all three allocation regimes.

2 RELATED WORK

Frequency-Domain Forecasting. Frequency-domain forecasting uses spectral representations to model global dependencies and recurring patterns. Some methods integrate frequency analysis into general backbones, including the Fourier-enhanced blocks of FEDformer (Zhou et al., 2022b) and the Legendre–Fourier projection of FiLM (Zhou et al., 2022a). Others perform prediction directly in spectral space through complex-valued MLPs (FreTS; Yi et al., 2023) or lightweight frequency interpolation (FITS; Xu et al., 2024). Specialized spectral operators include learned frequency-shaping filters (FilterNet; Yi et al., 2024), rotations for evolving periodic patterns (FRNet; Zhang et al., 2024), and cross-variable global filters (FilterTS; Wang et al., 2025). These methods provide diverse spectral representations and operators, but do not explicitly control the allocation of complementary spectral components between prediction domains. Improving spectral modeling alone therefore leaves open whether some components would be better handled by a temporal predictor. RouteTS addresses this question by making component-wise domain allocation an explicit, adjustable choice.

Hybrid Time–Frequency Forecasting. Hybrid methods combine temporal and spectral information through transformations, parallel branches, or structured decomposition. TimesNet (Wu et al., 2023) selects the frequencies with the largest spectral amplitudes to estimate periods, reshapes the complete series into period-specific 2D tensors, and aggregates their representations using amplitude-derived weights. Its selected frequencies guide representation construction rather than being routed to different domain predictors. FBM (Yang et al., 2024) incorporates the discrete Fourier transform into basis-function mappings; CFPT (Kou et al., 2025) combines cross-frequency interaction with periodic-aware timestamp modeling; Amplifier (Fei et al., 2025) enhances low-energy components within a seasonal–trend framework; and MixLinear (Ma et al., 2026) combines segment-based temporal modeling with low-rank spectral filtering.

Adaptive decomposition and expert routing operate at several other levels. FreqMoE (Liu, 2025) learns disjoint spectral bands and uses spectral magnitudes to weight band-specific experts, so its component assignment remains within a frequency-domain expert module. SEMPO (He et al., 2025) separates high- and low-energy components for spectral masking before routing the resulting patch tokens to prompt experts. TFPS (Sun et al., 2025) applies pattern-expert routing separately after parallel time- and frequency-domain encoders, while xCPD (Li et al., 2026) routes channel-patch graphs to frequency-specific graph filters. These methods route frequency bands, latent tokens, or channel dependencies within predefined domain roles. Such adaptivity changes how information is processed within the architecture, but does not directly determine whether both prediction domains are beneficial for a given forecasting setting. RouteTS instead assigns complementary spectral components to predictors operating in different domains and uses a validation-selected budget K to control the allocation. This provides an explicit choice among time-only, mixed-domain, and frequency-only configurations without assuming that combining both domains always improves forecasting.

3 METHOD

In this section, we present RouteTS, which partitions the input spectrum by component amplitude and assigns the resulting streams to predictors with different inductive biases. The two forecast contributions are aggregated in the normalized time domain. Figure 2 gives an overview.

Task Formulation. We consider a regularly sampled multivariate time series with C variables. Given a timestamp t, the model takes a lookback window of length L as input:

$$\mathbf{X} = [\mathbf{x}_{t-L+1}, \mathbf{x}_{t-L+2}, \ldots, \mathbf{x}_t]^\top \in \mathbb{R}^{L \times C}, \tag{1}$$

where $\mathbf{x}_i \in \mathbb{R}^C$ denotes the observations at time step i. The forecasting target is a future window of length T:

$$\mathbf{Y} = [\mathbf{x}_{t+1}, \mathbf{x}_{t+2}, \ldots, \mathbf{x}_{t+T}]^\top \in \mathbb{R}^{T \times C}. \tag{2}$$

The objective is to learn $f : \mathbb{R}^{L \times C} \to \mathbb{R}^{T \times C}$ such that $\hat{\mathbf{Y}} = f(\mathbf{X})$ approximates $\mathbf{Y}$.

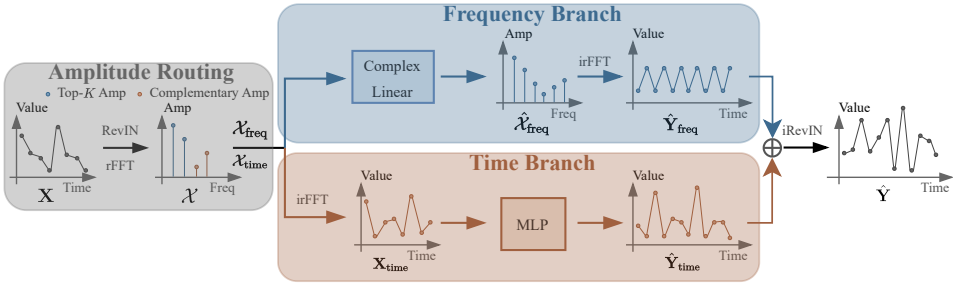


Figure 2: Overall architecture of RouteTS. The input is normalized by RevIN and transformed into the frequency domain, where amplitude routing partitions each variable's spectrum into its Top-K amplitude components and the remaining spectrum. The selected components are processed by a shared complex-valued frequency predictor, while the complementary components are transformed back to the time domain and processed by a shared MLP. The two forecast contributions are added in the normalized time domain, followed by inverse RevIN (iRevIN). The budget K unifies the architecture across regimes: $K = 0$, $0 < K < F$, and $K = \text{full}$ recover time-only, mixed-domain, and frequency-only forecasting, respectively.

3.1 Amplitude Routing

We first apply Reversible Instance Normalization (RevIN) (Kim et al., 2022) to the input $\mathbf{X}$, yielding $\mathbf{X}_{\text{norm}}$, and then apply the real Fast Fourier Transform (rFFT) along the temporal dimension:

$$\mathcal{X} = \text{rFFT}\left(\mathbf{X}_{\text{norm}}, n = L\right) \in \mathbb{C}^{F \times C}, \tag{3}$$

where $F = \lfloor L/2 \rfloor + 1$ is the number of non-redundant frequency components.

We compute the amplitude spectrum $\mathcal{A} = |\mathcal{X}| \in \mathbb{R}^{F \times C}$ and, for each variable c, construct a binary routing mask from its Top-K components:

$$\begin{aligned} \mathcal{I}_c &= \text{TopKIndices}\left(\mathcal{A}_{:,c}; K\right), \\ \mathcal{M}_{f,c} &= \begin{cases} 1, & f \in \mathcal{I}_c, \\ 0, & f \notin \mathcal{I}_c, \end{cases} \end{aligned} \tag{4}$$

where K controls the number of dominant components allocated to the frequency predictor. The spectrum is then partitioned into two complementary streams:

$$\mathcal{X}_{\text{freq}} = \mathcal{X} \odot \mathcal{M}, \qquad \mathcal{X}_{\text{time}} = \mathcal{X} \odot (\mathbf{1} - \mathcal{M}). \tag{5}$$

The subscripts denote the destination of each stream rather than an intrinsic categorization: $\mathcal{X}_{\text{freq}}$ is processed directly in the frequency domain, whereas $\mathcal{X}_{\text{time}}$ is reconstructed and processed in the time domain. The budget K spans three regimes: $K = 0$ yields a time-only model, $K = F$ (denoted $K = \text{full}$) a frequency-only model, and $0 < K < F$ activates both predictors. In the standard configuration, predictors are instantiated and evaluated only for active branches: $K = 0$ bypasses the Frequency Branch, whereas $K = \text{full}$ bypasses the Time Branch. The inactive branch therefore makes no contribution to the endpoint prediction.

3.2 Complex-Valued Prediction of Selected Components

Given $\mathcal{X}_{\text{freq}}$, the frequency predictor produces a spectral forecast contribution $\hat{\mathcal{X}}_{\text{freq}} \in \mathbb{C}^{F' \times C}$, where $F' = \lfloor T/2 \rfloor + 1$ is the number of non-redundant frequency components for horizon T.

To respect the algebraic coupling between the real and imaginary parts of the Fourier representation, we employ a complex-valued linear projection. Writing $\mathcal{X}_{\text{freq}} = \mathcal{X}_R + i\mathcal{X}_I$ with $\mathcal{X}_R, \mathcal{X}_I \in \mathbb{R}^{F \times C}$, we introduce a shared learnable weight matrix $\mathbf{W} = \mathbf{W}_R + i\mathbf{W}_I \in \mathbb{C}^{F' \times F}$, applied independently to all C variables. Bias terms are omitted for notational simplicity. The predicted spectrum is

$$\hat{\mathcal{X}}_{\text{freq}} = \mathbf{W}\mathcal{X}_{\text{freq}}, \tag{6}$$

or, in expanded form,

$$\begin{aligned}\hat{\mathcal{X}}_R &= \mathbf{W}_R\mathcal{X}_R - \mathbf{W}_I\mathcal{X}_I,\\ \hat{\mathcal{X}}_I &= \mathbf{W}_R\mathcal{X}_I + \mathbf{W}_I\mathcal{X}_R,\end{aligned} \tag{7}$$

where $\hat{\mathcal{X}}_R, \hat{\mathcal{X}}_I \in \mathbb{R}^{F' \times C}$. This structured projection jointly transforms the real and imaginary components rather than modeling them independently. Finally, after zeroing the imaginary parts of the DC bin and, for even T, the Nyquist bin, $\hat{\mathcal{X}}_{\text{freq}} = \hat{\mathcal{X}}_R + i\hat{\mathcal{X}}_I$ is transformed back to the time domain:

$$\hat{\mathbf{Y}}_{\text{freq}} = \text{irFFT}\left(\hat{\mathcal{X}}_{\text{freq}}, n = T\right) \in \mathbb{R}^{T \times C}. \tag{8}$$

3.3 Temporal Prediction of Complementary Components

The complementary spectrum $\mathcal{X}_{\text{time}}$ contains the components not allocated to the frequency predictor, which may contribute to localized and non-stationary variations. We transform this spectrum back to the time domain:

$$\mathbf{X}_{\text{time}} = \text{irFFT}\left(\mathcal{X}_{\text{time}}, n = L\right) \in \mathbb{R}^{L \times C}, \tag{9}$$

We then obtain a forecast contribution $\hat{\mathbf{Y}}_{\text{time}} \in \mathbb{R}^{T \times C}$ using a shared MLP applied independently to each variable:

$$\hat{\mathbf{Y}}_{\text{time},:,c} = \text{Linear}_2\left(\text{Dropout}\left(\text{GELU}\left(\text{Linear}_1\left(\mathbf{X}_{\text{time},:,c}\right)\right)\right)\right), \qquad c = 1, \ldots, C, \tag{10}$$

where Linear_1 maps L to a hidden dimension and Linear_2 maps it to the horizon T, with parameters shared across variables.

Final Aggregation. The two contributions are added in the normalized time domain, and inverse RevIN restores the location and scale of the input instance:

$$\hat{\mathbf{Y}} = \text{iRevIN}\left(\hat{\mathbf{Y}}_{\text{freq}} + \hat{\mathbf{Y}}_{\text{time}}\right). \tag{11}$$

The budget K thus changes the allocation between the two predictors without modifying their architectures.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate RouteTS on twelve real-world multivariate time series datasets spanning healthcare, environmental monitoring, transportation, energy, and economics. Dataset statistics, data splits, and preprocessing details are provided in Appendix A.1.

Baselines. We compare RouteTS with ten representative forecasting models covering time-domain, frequency-domain, and hybrid architectures: PhaseFormer (Niu et al., 2026), MixLinear (Ma et al., 2026), FilterTS (Wang et al., 2025), CFPT (Kou et al., 2025), Amplifier (Fei et al., 2025), FreqMoE (Liu, 2025), TFPS (Sun et al., 2025), iTransformer (Liu et al., 2024), FITS (Xu et al., 2024), and DLinear (Zeng et al., 2023). All baselines are reproduced using their official implementations and recommended hyperparameters, under matched look-back windows, prediction horizons, and data splits. Detailed baseline descriptions are provided in Appendix A.2.

Implementation Details. RouteTS is implemented in PyTorch 1.11 (Paszke et al., 2019) and optimized using Adam (Kingma & Ba, 2015) with MSE as the training objective. Our model and all reproduced baselines are trained on a single NVIDIA RTX 3090 GPU with 24 GB memory, and we report MSE and MAE as evaluation metrics.

For each dataset, the routing budget is selected from $\{0, 1, 2, 3, 4, 5, 10, \text{full}\}$. Specifically, we train each candidate configuration for all four prediction horizons and select the dataset-level routing budget K that minimizes the average validation MSE. The corresponding test results are then reported without further test-set-based selection. Here, $K = 0$ and $K = \text{full}$ denote the time-domain and frequency-domain endpoints, respectively, while intermediate values activate both branches.

Model	RouteTS (our)		PhaseFormer (2026)		MixLinear (2026)		FilterTS (2025)		CFPT (2025)		Amplifier (2025)		FreqMoE (2025)		TFPS (2025)		iTransformer (2024)		FITS (2024)		DLinear (2023)	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ILI	**1.640**	**0.807**	2.726	1.114	2.516	1.058	2.132	0.936	2.144	0.933	1.952	0.869	3.927	1.438	2.409	1.000	2.579	1.075	2.900	1.190	2.427	1.047
COVID-19	**0.125**	**0.135**	0.211	0.192	1.188	0.470	0.310	0.260	0.416	0.233	0.722	0.369	1.806	0.687	1.243	0.507	0.943	0.431	0.764	0.384	0.182	0.283
PM2.5	0.416	0.424	0.427	0.429	0.426	0.438	0.421	0.427	0.419	0.428	**0.391**	**0.415**	0.424	0.433	0.421	0.434	0.420	0.427	0.422	0.432	0.420	0.467
Weather	0.237	0.274	0.233	0.269	0.257	0.292	0.229	0.265	0.227	0.264	0.226	0.265	0.230	0.268	0.259	0.286	0.237	0.272	**0.224**	**0.261**	0.246	0.299
Traffic	0.402	0.274	0.397	**0.258**	0.441	0.294	0.411	0.286	0.410	0.274	0.422	0.291	0.432	0.303	1.122	0.604	**0.386**	0.274	0.434	0.291	0.454	0.328
Solar	**0.201**	0.249	0.211	**0.245**	0.281	0.297	0.212	0.265	0.210	0.255	0.213	0.263	0.211	0.283	0.243	0.295	0.217	0.267	0.263	0.281	0.254	0.316
ECL	0.164	0.256	0.161	0.250	0.188	0.282	0.165	0.259	**0.155**	**0.249**	0.161	0.253	0.172	0.269	0.213	0.319	0.165	0.260	0.174	0.270	0.166	0.263
ETTh1	**0.407**	**0.419**	0.442	0.429	0.419	0.428	0.422	0.434	0.423	0.432	0.444	0.447	0.567	0.509	0.468	0.473	0.452	0.456	0.413	0.423	0.433	0.445
ETTh2	**0.338**	**0.383**	0.368	0.404	0.350	0.389	0.347	0.389	0.353	0.395	0.354	0.402	0.378	0.414	0.370	0.411	0.385	0.414	0.339	0.384	0.482	0.472
ETTm1	**0.352**	0.380	0.359	0.385	0.376	0.388	0.368	0.391	0.354	0.382	0.355	0.380	0.381	0.396	0.385	0.408	0.366	0.395	0.360	**0.378**	0.359	0.381
ETTm2	**0.257**	**0.315**	0.273	0.326	0.266	0.321	0.272	0.325	0.259	0.316	0.271	0.322	0.286	0.335	0.299	0.341	0.296	0.329	0.258	0.316	0.268	0.332
Exchange	0.458	0.462	0.423	0.443	0.373	0.414	0.404	0.422	0.403	0.422	0.448	0.436	0.407	0.427	0.496	0.488	0.416	0.442	0.382	0.424	**0.333**	**0.392**

Table 1: Multivariate time series forecasting results (averaged over all prediction lengths). For ILI and COVID-19, the look-back window is 48 and prediction lengths are $\{24, 36, 48, 60\}$. For all other datasets, the look-back window is 336 and prediction lengths are $\{96, 192, 336, 720\}$. Lower MSE/MAE indicates better performance. **Bold** denotes the best result, underline denotes the second best. Full horizon-specific results are provided in Table 8.

The selected routing budgets and remaining dataset-specific hyperparameters are provided in Appendix A.3. Unless otherwise stated, each result is obtained from a single training run. For the random component-selection baseline in Section 4.4, we report the mean and standard deviation over three routing-mask seeds.

4.2 Overall Forecasting Performance

Table 1 reports results averaged over four prediction horizons. RouteTS achieves the best MSE on seven datasets and the best MAE on five, ranking among the top two methods on eight and nine datasets, respectively. These wins span all three allocation regimes, indicating that strong performance does not require mixed-domain processing on every dataset. Section 4.3 examines the selected budgets and the contribution of intermediate routing.

The results also expose limitations of the underlying predictors. On Weather and ECL, validation selects $K = 0$, leaving a channel-independent temporal predictor; its limited modeling capacity and lack of cross-variable interaction may partly explain the gap to specialized frequency-domain or cross-variable baselines. On Traffic, RouteTS achieves the second-best MAE but trails models that explicitly exploit inter-variable dependencies in MSE. Domain allocation therefore cannot compensate for all limitations of the branch predictors.

On Exchange, the validation-selected $K = 4$ performs substantially worse on the test split than both the frequency-domain endpoint and DLinear. Because budget selection averages raw validation MSE across horizons, the advantage of $K = 4$ at $T = 720$ outweighs its weaker performance at shorter horizons. This advantage does not transfer to the test split, where the frequency-domain endpoint performs best on average among the candidate budgets. The discrepancy shows how a validation advantage concentrated at one horizon can drive dataset-level selection without generalizing to test data. We report $K = 4$ without test-set retuning; Appendix B.3 provides the horizon-wise analysis.

4.3 Domain-Allocation Behavior across Datasets

Figure 3 shows that nine of the twelve datasets favor an endpoint on validation data: six select $K = 0$, while ETTh1, ETTh2, and ETTm2 select $K = \text{full}$. Solar and Traffic select $K = 1$, and Exchange selects $K = 4$. The preferred allocation thus varies across datasets, with intermediate budgets selected only in a minority of cases.

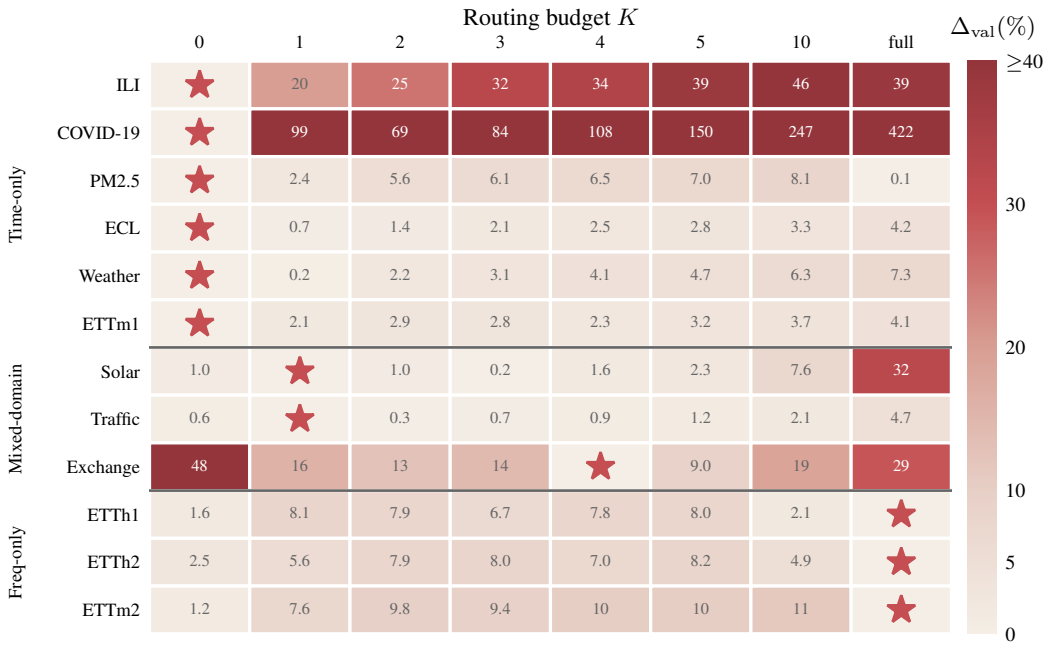


Figure 3: Validation sensitivity of RouteTS to the routing budget K across twelve datasets. Each cell reports the relative increase in validation MSE (averaged over the four horizons) from the per-dataset minimum; stars mark the selected budgets and rows are grouped by the resulting operating regime. Colors are clipped at 40%; annotated values are unclipped.

Sensitivity to K also varies. Deviating from the optimum increases validation MSE by up to 46% on ILI and 422% on COVID-19, whereas Traffic and the ETT datasets exhibit comparatively flat landscapes around their selected budgets. PM2.5 nearly ties at its two endpoints (0.1%). These patterns suggest interpreting K as a validation-dependent operating point rather than an intrinsic dataset property, consistent with the lack of a simple one-to-one correspondence between the selected budgets and spectral or autocorrelation profiles (Appendix B.1).

Table 2 tests whether intermediate routing improves over endpoint-only selection restricted to $K \in \{0, \text{full}\}$. Both strategies select budgets using average validation MSE and report the corresponding test results. On Traffic, $K = 1$ improves MSE by 0.2% and MAE by 0.7% over the selected endpoint; on Solar, it reduces MSE by 1.5% with unchanged MAE. These modest gains show that intermediate allocation can improve forecasting on some datasets.

On Exchange, however, the validation-selected $K = 4$ increases test MSE and MAE by 23.1% and 13.2% relative to the frequency-domain endpoint. This discrepancy illustrates that a budget favored on validation data may not generalize to the test split; the horizon-wise analysis is provided in Appendix B.3.

4.4 Controlled Analysis of Routing Decisions

We examine component selection and branch assignment on Solar and Traffic, where intermediate routing improves over both endpoints on the test set. Qualitative examples on Solar are provided in Appendix B.2.

Effect of spectral-component selection. Table 3 compares amplitude, low-frequency, and random selection at $K = 1$, keeping the allocation size fixed and varying only the selected component. On Solar, amplitude routing reduces average MSE by 6.6% and 7.5% relative to low-frequency and random selection, respectively, while MAE differences remain small. On Traffic, MSE differs by less than 0.4% across strategies, but amplitude routing reduces MAE by about 1.2% and 1.0% relative to the two alternatives. Amplitude ranking thus provides a useful selection bias, with a pronounced benefit on Solar and a modest one on Traffic.

Table 2: Comparison between validation-selected amplitude routing and endpoint-only selection. $K^*_{\text{end}} \in \{0, \text{full}\}$ denotes the endpoint selected using average validation MSE, while K^*_{route} is selected from the complete routing-budget set. The table reports their corresponding average test performance without test-set retuning. Improvement is computed relative to the endpoint result, with positive values indicating an advantage for amplitude routing.

Dataset	Metric	Endpoint K^*_{end}	Endpoint Test	RouteTS K^*_{route}	RouteTS Test	Improvement
Traffic	MSE	0	0.403	1	0.402	0.2%
	MAE	0	0.276	1	0.274	0.7%
Solar	MSE	0	0.204	1	0.201	1.5%
	MAE	0	0.249	1	0.249	0.0%
Exchange	MSE	full	0.372	4	0.458	-23.1%
	MAE	full	0.408	4	0.462	-13.2%

Table 3: Effect of frequency-component selection under the fixed routing budget $K = 1$. Amplitude routing is compared with two strategies that select the same number of components: *Low-frequency*, which selects the K lowest non-DC frequency components, and *Random*, which randomly samples K components. Random results are reported as the mean and standard deviation over three fixed routing-mask seeds while keeping the training configuration unchanged. All results are averaged over four prediction horizons. Best results are in **bold**. Full horizon-specific results for these component-selection strategies are reported in Table 7.

	Solar		Traffic	
Strategy	MSE	MAE	MSE	MAE
Low-frequency	0.216	0.250	0.404	0.277
Random	0.218±0.0002	0.250±0.0008	0.403±0.0001	0.276±0.0002
Amplitude (ours)	**0.201**	**0.249**	**0.402**	**0.274**

Effect of routing selectivity and branch assignment. Figure 4 compares two structural variants. *Broadcast* feeds the complete input to both branches, increasing MSE at every horizon on Solar by at most 1.1% but marginally reducing it on Traffic. Selective partitioning therefore helps on Solar but is not uniformly superior to full-input processing. *Swap* retains the decomposition but reverses branch assignments, increasing MSE at every horizon, by 27.8% on Solar and 8.0% on Traffic on average. This consistent degradation indicates that the predictors are not interchangeable under the same decomposition. Together, these results support the importance of component selection and branch assignment, while showing that the benefit of selective partitioning is dataset-dependent. Predictor architecture ablations are provided in Appendix C.

4.5 Model Efficiency Analysis

We finally examine whether the accuracy of RouteTS comes at a practical cost. Figure 5 compares prediction error, parameter count, and peak allocated GPU memory on ETTm2 and Solar with a look-back window and prediction horizon of 336. Each model is profiled under the official training configuration used to obtain its corresponding MSE, including its model-specific batch size. The comparison therefore characterizes practical accuracy–resource operating points rather than a batch-size-controlled measurement of intrinsic architectural memory. Under these settings, attention-based architectures such as TFPS and PhaseFormer incur heavy memory costs, whereas minimalist models such as MixLinear sacrifice accuracy. RouteTS attains the lowest MSE on both datasets while retaining a small resource footprint: on Solar it uses 70.55 MB, and on ETTm2 it matches CFPT with about 4% of its parameters and 13% of its peak allocated memory. RouteTS also remains robust under moderate input noise and improves monotonically with longer look-back windows, as shown in Appendix D.

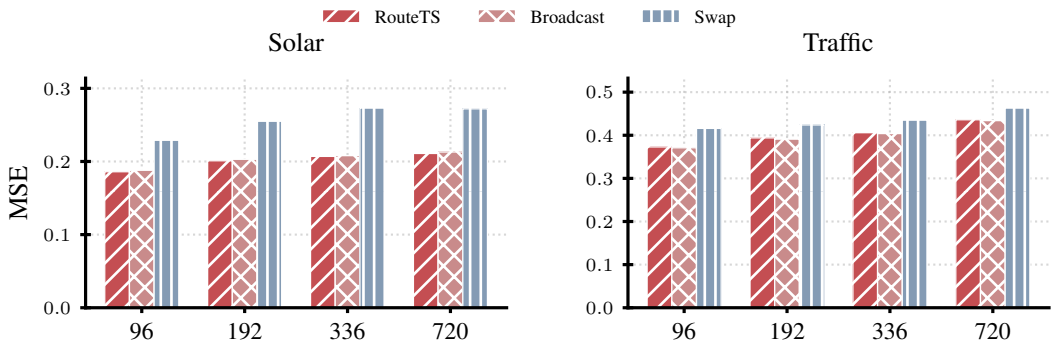


Figure 4: Controlled analysis of routing selectivity and branch assignment on Solar and Traffic using MSE. All variants use look-back 336 and $K = 1$. *Broadcast* removes routing selectivity by providing the complete input to both the Time and Frequency Branches. *Swap* retains the amplitude-based decomposition of RouteTS but reverses the branch assignments of the two component groups.

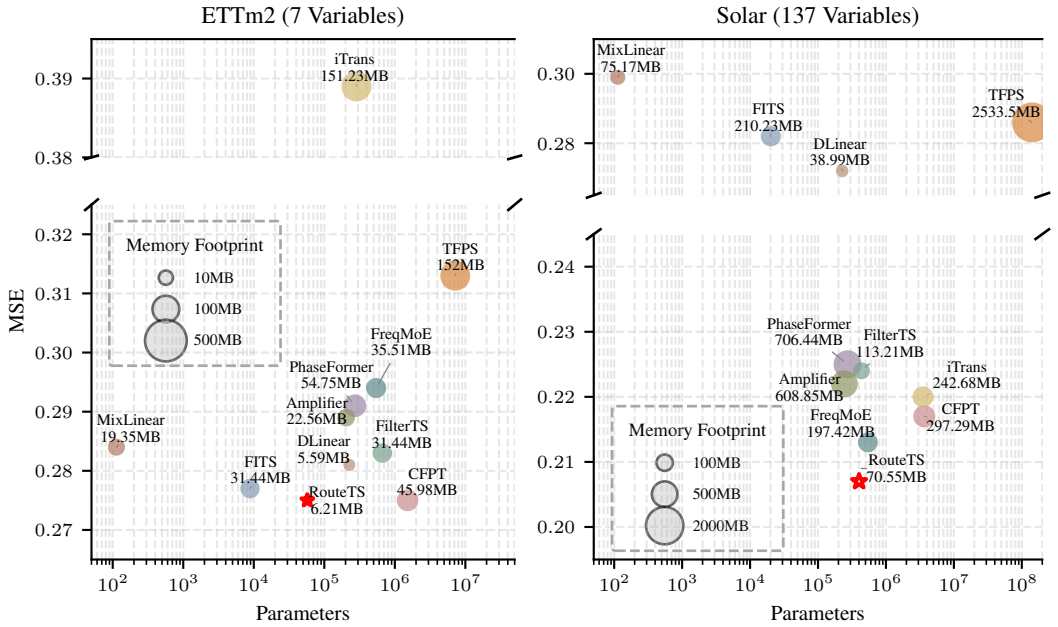


Figure 5: Model effectiveness and efficiency under the look-back 336 and prediction-horizon 336 setting. Each model is profiled during training using its official configuration, including its model-specific batch size; bubble size denotes peak allocated GPU memory. RouteTS (red star) achieves the lowest MSE on both datasets with small parameter and memory budgets.

5 CONCLUSION

We presented RouteTS, a lightweight framework that explicitly allocates amplitude-ranked spectral components between time-domain and frequency-domain predictors. Varying the routing budget K yields time-only, mixed-domain, and frequency-only configurations. With dataset-specific budgets selected exclusively on validation data, RouteTS achieves the lowest average MSE on seven of twelve datasets while remaining among the most parameter- and memory-efficient models evaluated. The results show that no allocation regime is uniformly preferred: intermediate routing benefits some datasets, whereas the two endpoint configurations perform better on others. Controlled analyses show that amplitude-based selection outperforms the alternatives and that reversing the branch assignments consistently degrades accuracy. The Exchange discrepancy nevertheless shows that a dataset-level budget chosen on validation data may not remain the best choice on the test split. Future work will explore per-instance budget selection and lightweight cross-variable modeling.

AI USE STATEMENT

In this work, we used generative AI tools to draft and edit portions of the manuscript, improve readability, clarity, and flow, and refine LaTeX formatting. We also used these tools in a limited advisory capacity to obtain feedback on the organization of selected controlled experiments and on the interpretation and presentation of selected experimental results. These tools assisted with code implementation based on methods specified by the authors, who independently reviewed and validated the resulting code and performed manual debugging and corrections as needed.

The research problem, core methodology, primary experimental design, experimental execution, and final scientific conclusions were developed and determined by the authors. Suggestions produced by generative AI were treated as non-authoritative and incorporated only after independent evaluation by the authors. All AI-assisted text and technical suggestions were checked against the authors' code and experimental records. Generative AI was not used to generate synthetic datasets or to formulate or prove mathematical claims. We take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We provide comprehensive details to facilitate the reproduction of our results. The proposed RouteTS architecture and its data flow are fully specified in Section 3.1– 3.3. Dataset statistics, preprocessing steps, and baseline descriptions are documented in Appendix A.1 and Appendix A.2. The main training settings for RouteTS, including dataset-specific batch sizes, learning rates, and routing budgets K, are reported in Appendix A.3. Detailed implementation and experimental configurations are provided in the code submitted as supplementary material. The main experimental protocol is detailed in Section 4.1. Complete per-horizon results are provided in Appendix E.

REFERENCES

Razvan-Gabriel Cirstea, Tung Kieu, Chenjuan Guo, Bin Yang, and Sinno Jialin Pan. EnhanceNet: Plugin neural networks for enhancing correlated time series forecasting. In International Conference on Data Engineering (ICDE), 2021.

Jingru Fei, Kun Yi, Wei Fan, Qi Zhang, and Zhendong Niu. Amplifier: Bringing attention to neglected low-energy components in time series forecasting. In AAAI Conference on Artificial Intelligence (AAAI), 2025.

Hui He, Kun Yi, Yuanchi Ma, Qi Zhang, Zhendong Niu, and Guansong Pang. SEMPO: Lightweight foundation models for time series forecasting. In Advances in Neural Information Processing Systems (NeurIPS), 2025.

Qi-Qiao He, Shirley Weng In Siu, and Yain-Whar Si. Instance-based deep transfer learning with attention for stock movement prediction. Applied Intelligence, 53(6):6887–6908, 2023.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In International Conference on Learning Representations (ICLR), 2022.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In International Conference on Learning Representations (ICLR), 2015.

Feifei Kou, Jiahao Wang, Lei Shi, Yuhan Yao, Yawen Li, Suguo Zhu, Zhongbao Zhang, and Junping Du. CFPT: Empowering time series forecasting through cross-frequency interaction and periodic-aware timestamp modeling. In International Conference on Machine Learning (ICML), 2025.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2018.

Dongyuan Li, Shun Zheng, Chang Xu, Jiang Bian, and Renhe Jiang. Routing channel-patch dependencies in time series forecasting with graph spectral decomposition. arXiv preprint arXiv:2603.13702, 2026.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In International Conference on Learning Representations (ICLR), 2024.

Ziqi Liu. FreqMoE: Enhancing time series forecasting through frequency decomposition mixture of experts. In International Conference on Artificial Intelligence and Statistics (AISTATS), 2025.

Aitian Ma, Dongsheng Luo, and Mo Sha. MixLinear: Extreme low resource multivariate time series forecasting with $0.1K$ parameters. In International Conference on Learning Representations (ICLR), 2026.

Henrik Madsen. Time Series Analysis. CRC Press, 2007.

Francisco Martinez Alvarez, Alicia Troncoso, Jose C. Riquelme, and Jesus S. Aguilar Ruiz. Energy time series forecasting based on pattern sequence similarity. IEEE Transactions on Knowledge and Data Engineering, 23(8):1230–1243, 2011.

Yiming Niu, Jinliang Deng, and Yongxin Tong. PhaseFormer: From patches to phases for efficient and effective time series forecasting. In International Conference on Learning Representations (ICLR), 2026.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems (NeurIPS), 2019.

Yanru Sun, Zongxia Xie, Emadeldeen Eldele, Dongyue Chen, Qinghua Hu, and Min Wu. Learning pattern-specific experts for time series forecasting under patch-level distribution shift. In Advances in Neural Information Processing Systems (NeurIPS), 2025.

Yulong Wang, Yushuo Liu, Xiaoyi Duan, and Kai Wang. FilterTS: Comprehensive Frequency Filtering for Multivariate Time Series Forecasting. In AAAI Conference on Artificial Intelligence (AAAI), 2025.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In International Conference on Learning Representations (ICLR), 2023.

Xinle Wu, Dalin Zhang, Chenjuan Guo, Chaoyang He, Bin Yang, and Christian S. Jensen. Autocts: Automated correlated time series forecasting – extended version. arXiv preprint arXiv:2112.11174, 2021.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with 10k parameters. In International Conference on Learning Representations (ICLR), 2024.

Runze Yang, Longbing Cao, JIE YANG, and li jianxun. Rethinking Fourier transform from A basis functions perspective for long-term time series forecasting. In Advances in Neural Information Processing Systems (NeurIPS), 2024.

Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Ning An, Defu Lian, Longbing Cao, and Zhendong Niu. Frequency-domain MLPs are more effective learners in time series forecasting. In Advances in Neural Information Processing Systems (NeurIPS), 2023.

Kun Yi, Jingru Fei, Qi Zhang, Hui He, Shufeng Hao, Defu Lian, and Wei Fan. FilterNet: Harnessing frequency filters for time series forecasting. In Advances in Neural Information Processing Systems (NeurIPS), 2024.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are Transformers Effective for Time Series Forecasting? In AAAI Conference on Artificial Intelligence (AAAI), 2023.

Xinyu Zhang, Shanshan Feng, Jianghong Ma, Huiwei Lin, Xutao Li, Yunming Ye, Fan Li, and Yew Soon Ong. FRNet: Frequency-based rotation network for long-term time series forecasting. In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2024.

Tian Zhou, Ziqing Ma, Xue Wang, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, and Rong Jin. FiLM: Frequency improved legendre memory model for long-term time series forecasting. In Advances in Neural Information Processing Systems (NeurIPS), 2022a.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In International Conference on Machine Learning (ICML), 2022b.

A Experimental Details

A.1 Datasets

Table 4: Statistics of the twelve benchmark datasets.

Dataset	Channels	Sampling Interval	Total Time Steps
ILI	7	1 week	966
COVID-19	8	1 day	1,143
PM2.5	184	3 hours	11,688
Weather	21	10 min	52,696
Traffic	862	1 hour	17,544
Solar	137	10 min	52,560
ECL	321	1 hour	26,304
ETTh1	7	1 hour	17,420
ETTh2	7	1 hour	17,420
ETTm1	7	15 min	69,680
ETTm2	7	15 min	69,680
Exchange	8	1 day	7,588

We provide a comprehensive overview of the twelve benchmark datasets utilized in our experiments. Table 4 summarizes the statistical properties of these datasets, including the number of input channels (variables), temporal sampling interval, and total available timesteps.

All datasets are split chronologically. ETTh1, ETTh2, ETTm1, and ETTm2 use the first 12 months for training, the subsequent 4 months for validation, and the final 4 months for testing. The remaining eight datasets use a 70%/10%/20% train/validation/test split. Each variable is standardized using the mean and standard deviation computed from the training split only, and the same transformation is applied to the validation and test splits before the model-level RevIN operation. At split boundaries, validation and test input windows may include the immediately preceding historical observations, while their forecasting targets remain entirely within the corresponding split. All reported MSE and MAE values are computed directly in this standardized space without applying the dataset-level inverse transformation.

ILI Dataset[1]**.** The Influenza-like Illness (ILI) dataset comprises weekly reports on influenza-like cases monitored by the Centers for Disease Control and Prevention (CDC) in the United States. Spanning 966 timesteps with 7 clinical and demographic channels, it records the ratio of patients exhibiting influenza symptoms to the total patients treated. Modeling this dataset is crucial for public health decision-making, though its coarse weekly resolution and sparse timesteps make long-term forecasting exceptionally challenging.

COVID-19 Dataset[2]**.** This dataset records daily cumulative confirmed cases across major global regions, sourced from the Johns Hopkins University (JHU) Center for Systems Science and Engineering. Rather than applying temporal differencing or smoothing, the dataset retains the raw cumulative time-series observations, preserving the original monotonic growth and non-stationary trends. It captures the top 8 countries with the highest cumulative cases at the end of the observation window. Spanning 1,143 daily timesteps, this dataset evaluates the capacity of forecasting architectures to model continuous monotonic growth patterns and long-term distributional shifts.

PM2.5 Dataset[3]**.** We use the PM2.5 dataset distributed through the data download link in the official TimeBase repository. It contains 11,688 time steps across 184 variables sampled at 3-hour intervals and is used to evaluate multivariate air-quality forecasting.

[1] https://github.com/juyongjiang/TimeSeriesDatasets

[2] https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv

[3] https://github.com/hqh0728/TimeBase

Weather Dataset[3]. The Weather dataset comprises 21 meteorological indicators (e.g., air temperature, humidity, atmospheric pressure) collected from monitoring stations in Germany. Spanning 52,696 timesteps with a 10-minute sampling resolution, it captures pronounced diurnal cycles and seasonal atmospheric fluctuations. Modeling this dataset serves as a critical benchmark for environmental forecasting tasks, while its rich periodic components and inter-variable correlations test the robustness of multivariate temporal modeling.

Traffic Dataset[3]. The Traffic dataset provides hourly road occupancy rates measured by 862 inductive loop sensors deployed along major freeways in the San Francisco Bay Area, maintained by the California Department of Transportation. Comprising 17,544 timesteps, the dataset exhibits strong daily and weekly periodicities intertwined with abrupt, non-stationary patterns caused by rush-hour congestion, making it a standard testbed for capturing spatial-temporal interactions.

Solar Dataset[3]. The Solar-Energy dataset records solar power generation tracked at 10-minute intervals throughout 2006 across 137 photovoltaic (PV) plants in Alabama. It consists of 52,560 timesteps, capturing high-frequency fluctuations in solar irradiance alongside prominent diurnal and seasonal periodicities, which is ideal for validating the model's ability to capture global deterministic periodic components.

ECL Dataset[3]. The Electricity Load (ECL) dataset records hourly electricity consumption of 321 distinct residential and commercial clients. Spanning 26,304 timesteps with 321 channels of consumption data, it features high dimensionality and heterogeneous usage patterns across users. As a widely adopted benchmark for power load forecasting, it evaluates a model's ability to capture both individual user behaviors and aggregate demand trends, presenting notable challenges for long-term multivariate prediction.

ETT Dataset[4]. The Electricity Transformer Temperature (ETT) datasets consist of key operational parameters and oil temperatures collected from multiple electricity substations. In our experiments, we utilize four subsets across two temporal granularities: ETTh1 and ETTh2 are sampled hourly with 17,420 timesteps each, while ETTm1 and ETTm2 are sampled at 15-minute intervals with 69,680 timesteps each. All subsets comprise 7 variables representing transformer load and oil temperature indicators, providing a comprehensive benchmark for testing long-term forecasting generalization across different sites and sampling intervals.

Exchange Dataset[3]. The Exchange-Rate dataset tracks the daily currency exchange rates of eight major foreign economies (including Australia, Canada, Switzerland, China, Japan, New Zealand, Singapore, and the United Kingdom) relative to the US dollar from 1990 to 2016. Spanning 7,588 timesteps, this dataset is characterized by high non-stationarity, market volatility, and a lack of explicit temporal periodicities, serving as a demanding benchmark for economic financial series modeling.

A.2 BASELINES

We select ten representative baselines for comparison, including Transformer-based models, Linear-based models, and Frequency-domain models. We introduce these baselines as follows:

PhaseFormer. PhaseFormer (Niu et al., 2026) is a lightweight model that introduces a phase-based perspective to characterize time-series periodicity. By reframing series into compact phase tokens instead of patches, it maps features into a shared latent space and employs a low-rank routing mechanism for efficient cross-phase interactions. This design successfully alleviates the computational bottlenecks of patch-level processing with only around 1k parameters.

MixLinear. MixLinear (Ma et al., 2026) is an ultra-lightweight forecasting framework that combines segment-based trend extraction in the time domain with adaptive complex-valued low-rank filtering in the frequency domain. It reduces parameter complexity from $\mathcal{O}(n^2)$ to $\mathcal{O}(n)$ by utilizing strategic segmentation and rank-constrained filters, achieving high forecasting accuracy with merely 0.1k parameters.

FilterTS. FilterTS (Wang et al., 2025) is a frequency-domain model utilizing specialized filtering modules to extract complex periodic and trend patterns. It integrates a Static Global Filtering Module to capture stable periodic components and a Dynamic Cross-Variable Filtering Module to empha-

[4] https://github.com/zhouhaoyi/ETDataset

size shared frequency features across variables. By executing operations directly in the frequency domain, it avoids costly time-domain convolutions, boosting efficiency and accuracy.

CFPT. CFPT (Kou et al., 2025) is a dual-branch framework that leverages frequency dynamics and timestamp patterns via a Cross-Frequency Interaction (CFI) branch and a Periodic-Aware Timestamp Modeling (PTM) branch. The CFI branch extracts synergistic relationships among different frequencies through feature fusion, while the PTM branch converts 1D timestamps into 2D tensors to capture intra- and inter-period correlations via 2D convolutions.

Amplifier. Amplifier (Fei et al., 2025) captures often-neglected low-energy components in time series using a novel energy amplification technique. It equalizes energy levels via spectrum flipping to improve learning efficiency for subtle features. Furthermore, it integrates a seasonal-trend forecaster and a semi-channel interaction block to independently model distinct energy peaks and shared temporal patterns across channels.

FreqMoE. FreqMoE (Liu, 2025) is a frequency-domain model utilizing a Frequency Decomposition Mixture-of-Experts architecture to dynamically assign distinct frequency bands to specialized expert networks. A gating mechanism adaptively weights these experts based on the frequency magnitude spectrum. This allows the model to effectively capture dominant frequency patterns while maintaining high parameter efficiency.

TFPS. TFPS (Sun et al., 2025) addresses patch-level distribution shifts in non-stationary time series via a Time-Frequency Pattern-Specific architecture. It employs a Dual-Domain Encoder for comprehensive representation and a Pattern Identifier based on subspace clustering to dynamically classify patches. These patches are routed to a Mixture of Pattern Experts, enabling highly adaptable, pattern-aware forecasting.

iTransformer. iTransformer (Liu et al., 2024) is a Transformer-based architecture that applies attention and feed-forward networks on inverted dimensions. It embeds the time points of individual series as independent variate tokens, capturing multivariate correlations via self-attention and learning nonlinear representations with MLPs. This inverted design avoids computational explosion and enables the effective utilization of long lookback windows.

FITS. FITS (Xu et al., 2024) is an exceptionally lightweight frequency-domain model that operates by interpolating time series in the complex frequency domain. Using a complex-valued network, it simultaneously captures amplitude and phase information to yield a compact yet comprehensive representation of temporal data, delivering competitive forecasting accuracy with only about 10k parameters.

DLinear. DLinear (Zeng et al., 2023) is a Linear-based model that incorporates a classic decomposition scheme to handle complex temporal dynamics. It splits the input sequence into trend-cyclical and seasonal components via moving average, applying individual linear layers to forecast each component. This simple design drastically reduces architectural complexity while maintaining competitive performance against Transformer-based models.

A.3 Implementation Details

RouteTS was implemented in PyTorch 1.11 and optimized using Adam. All experiments were conducted on NVIDIA RTX 3090 GPUs. Each training run was limited to 100 epochs, with early stopping applied when the validation MSE did not improve for 12 consecutive epochs. Peak allocated GPU memory is measured over four training epochs using `torch.cuda.max_memory_allocated()` after resetting the peak-memory statistics immediately before training. The Time Branch in Eq. 10 uses a hidden dimension of 512 and a dropout rate of 0.05 for all datasets and forecasting horizons. Both linear layers include bias terms.

We used dataset-specific batch sizes and initial learning rates, as summarized in Table 5. For each dataset, the routing budget was selected from $\{0, 1, 2, 3, 4, 5, 10, \text{full}\}$. Because components are ranked by amplitude, the candidate set uses fine-grained increments for small budgets ($K = 0, \ldots, 5$), while $K = 10$ represents a larger intermediate allocation and $K = \text{full}$ the frequency-only endpoint. Specifically, we first averaged the raw validation MSE over all forecasting horizons for each candidate budget and then selected the budget with the lowest average. The selected value

Table 5: Dataset-specific hyperparameters for RouteTS. The routing budget K is selected according to the average validation MSE over all forecasting horizons.

Dataset	Batch size	Learning rate	Selected K
ILI	32	1×10^{-2}	0
COVID-19	32	1×10^{-3}	0
PM2.5	64	5×10^{-4}	0
Weather	16	1×10^{-3}	0
Traffic	16	5×10^{-3}	1
Solar	16	1×10^{-3}	1
ECL	16	1×10^{-3}	0
ETTh1	32	1×10^{-3}	full
ETTh2	32	5×10^{-3}	full
ETTm1	32	1×10^{-4}	0
ETTm2	32	5×10^{-3}	full
Exchange	8	5×10^{-4}	4

was subsequently fixed for all test evaluations on that dataset, without using test performance for model selection.

B Additional Analysis of Domain Allocation

B.1 Spectral and Autocorrelation Characteristics

Spectral characteristics. To characterize the inputs underlying the domain-allocation analysis, we compute the mean amplitude spectrum of each dataset. For every test-set input window, we apply the same RevIN and rFFT operations used by RouteTS and then average the resulting amplitudes across windows and variables. For each dataset, we use the input configuration associated with the longest forecasting horizon considered in our experiments.

As shown in Figure 6, the datasets exhibit substantially different spectral structures. ILI, COVID-19, Weather, and Exchange are dominated by a smooth low-frequency decay, while Solar concentrates most of its energy within a narrow low-frequency region. Traffic, ECL, and ETTh1 exhibit more pronounced spectral peaks and harmonic structure. PM2.5 and several ETT datasets combine a decaying low-frequency envelope with secondary peaks or a broader spectral tail. Because the input lengths and frequency resolutions differ across datasets, the comparison concerns the shape and concentration of each spectrum rather than its absolute amplitude. These observations establish the diversity of spectral structures encountered by RouteTS, but do not imply that the spectrum alone determines the preferred routing budget.

Autocorrelation characteristics.

We further examine the same datasets in the time domain using the Autocorrelation Function (ACF) (Madsen, 2007). For a training sequence $x_{:,c}$ of length N and mean $\bar{x}_c$, the sample autocorrelation of variable c at lag k is defined as

$$\rho_c(k) = \frac{\sum_{t=1}^{N-k}(x_{t,c} - \bar{x}_c)(x_{t+k,c} - \bar{x}_c)}{\sum_{t=1}^{N}(x_{t,c} - \bar{x}_c)^2}. \tag{12}$$

For multivariate datasets, we report the pointwise median across variables,

$$\widetilde{\rho}(k) = \underset{c=1,\ldots,C}{\text{median}}\, \rho_c(k), \tag{13}$$

which reduces the influence of individual variables with atypical autocorrelation patterns. For univariate datasets, $\widetilde{\rho}(k) = \rho_1(k)$. The maximum lag is chosen separately for each dataset according to its sampling interval, allowing the plots to cover multiple plausible temporal cycles.

Figure 7 reveals several forms of temporal dependence. COVID-19 and Exchange exhibit gradual autocorrelation decay, whereas PM2.5 loses most of its dependence within the initial lags. Weather,

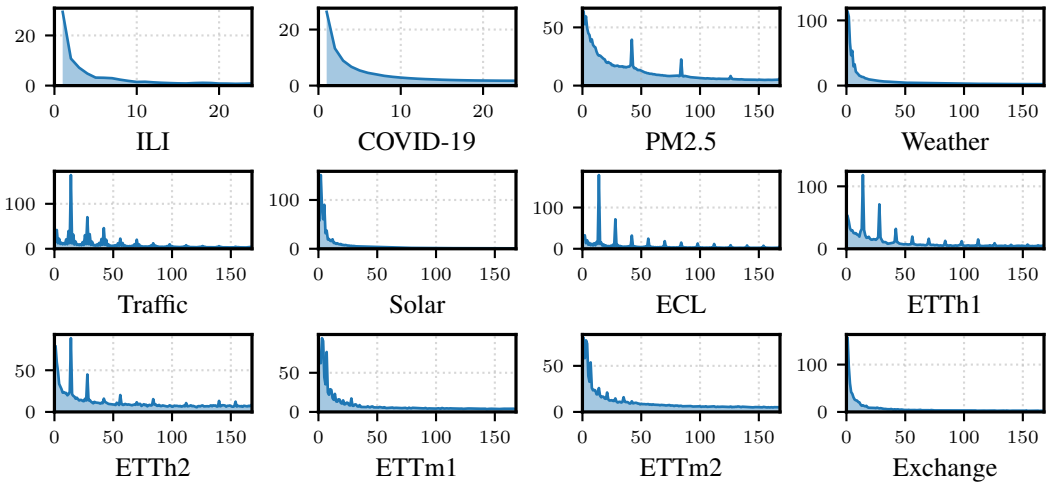


Figure 6: Mean amplitude spectra of the 12 datasets. For each dataset, the spectrum is computed from RevIN-normalized test-set input windows under the longest-horizon configuration and averaged across windows and variables. Horizontal axes denote frequency-bin indices, and each panel uses its own vertical scale. The plots illustrate diverse degrees of low-frequency concentration, harmonic structure, and spectral dispersion; they provide descriptive context for domain allocation rather than a direct predictor of the selected routing budget K.

Traffic, Solar, and ECL display persistent oscillations, while several ETT datasets combine repeated short-period fluctuations with a more slowly decaying baseline. ILI presents a broader alternating pattern over its shorter available lag range. These ACF patterns complement the spectral view by exposing periodic behavior that may be less visible after averaging amplitude spectra across windows and variables.

Importantly, neither diagnostic exhibits a one-to-one correspondence with the validation-selected routing budget. For example, Weather and ECL contain clear periodic autocorrelation but select $K = 0$, whereas Traffic and Solar select $K = 1$. Likewise, ETTm1 and ETTm2 have qualitatively similar spectral and autocorrelation profiles but favor opposite endpoint allocations. The preferred budget therefore reflects the predictive utility of assigning components to the two branch operators under a given forecasting setting, rather than an intrinsic category that can be inferred from a single descriptive statistic.

B.2 Qualitative Domain-Routing Behavior

Solar provides a representative setting for inspecting the behavior of mixed-domain routing, as the validation procedure selects an intermediate routing budget of $K = 1$. We therefore visualize a test example under the prediction horizon $T = 336$. These examples are intended as qualitative illustrations of the learned behavior; the quantitative evidence for the routing mechanism is provided by the controlled analyses in the main text.

Time-domain behavior and branch contributions. As shown in the left panel of Figure 8, RouteTS more closely follows the timing and magnitude of the target peaks in this example, whereas PhaseFormer and DLinear exhibit larger deviations around the periodic transitions. The right panel visualizes the two branch contributions before their additive aggregation. The temporal contribution captures much of the sharp peak structure, while the frequency-domain contribution provides a smoother complementary correction. Their sum forms the normalized RouteTS prediction. Notably, these observations should not be interpreted as a predefined semantic decomposition in which one branch exclusively represents periodicity and the other exclusively represents local variation. The routing mask determines where the selected input components are processed, while the resulting branch contributions are learned jointly for forecasting.

Time–frequency comparison. Figure 9 compares RouteTS with FITS in both the time and frequency domains. In this example, the FITS forecast produces smoother and lower-amplitude tran-

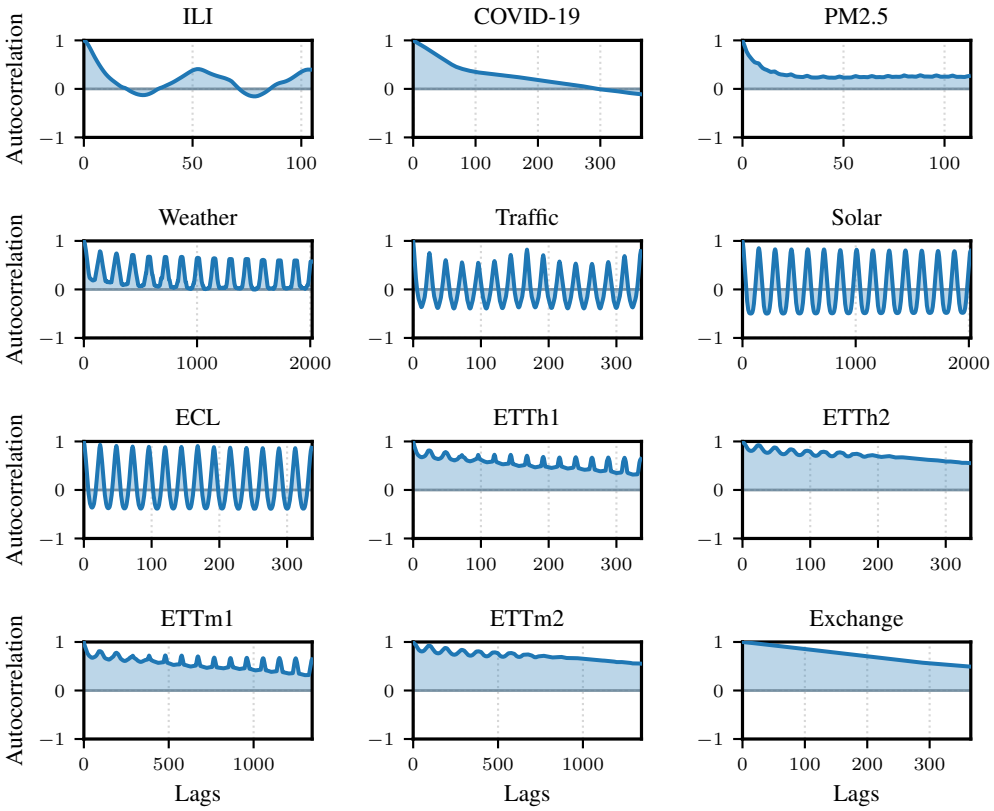


Figure 7: Autocorrelation characteristics computed from the continuous training sequences of the 12 datasets. For multivariate datasets, each curve reports the pointwise median ACF across variables; for univariate datasets, the original ACF is shown. Dataset-specific lag ranges are selected according to the corresponding sampling intervals. Shaded regions indicate the area between each ACF curve and zero. The curves reveal gradual decay, rapid attenuation, and persistent oscillatory dependence, complementing the spectral analysis without implying a one-to-one correspondence with the selected routing budget.

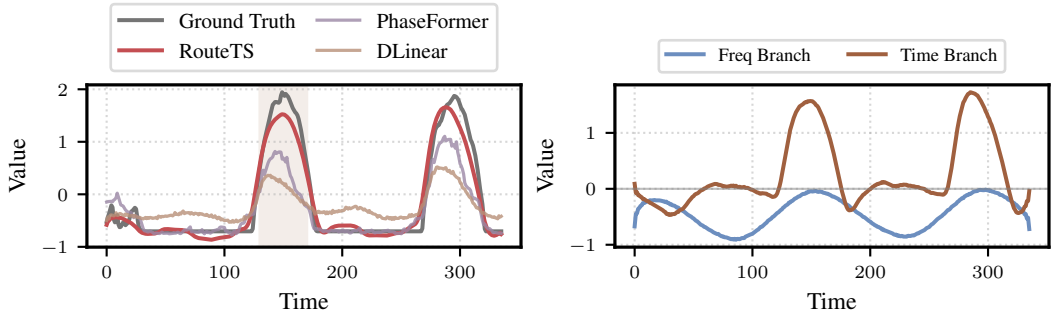


Figure 8: Qualitative domain-routing behavior on a representative Solar test example with prediction horizon $T = 336$ and routing budget $K = 1$. Left: Ground truth and forecasts produced by RouteTS, PhaseFormer, and DLinear. Right: Frequency- and time-branch contributions in the normalized time domain before additive aggregation and inverse RevIN.

sitions than the target. Its spectrum correspondingly underestimates several components outside the most prominent low-frequency modes. RouteTS more closely matches both the dominant spectral peaks and the smaller components of the target, resulting in a more faithful reconstruction of the localized transitions. Together, the two visualizations illustrate how component-wise routing can

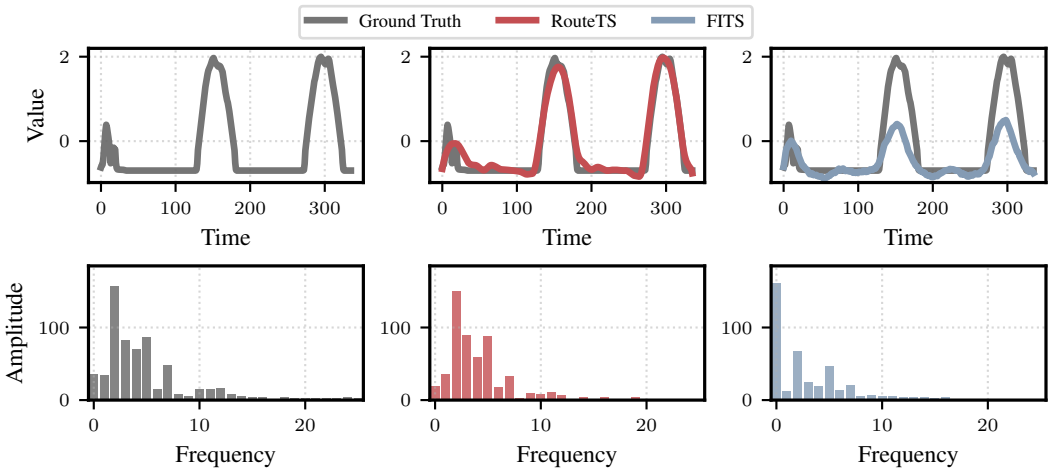


Figure 9: Time- and frequency-domain comparison on a representative Solar test example with prediction horizon $T = 336$. The top row shows the target and the forecasts produced by RouteTS and FITS, while the bottom row presents their corresponding amplitude spectra. In this example, FITS underestimates several spectral components and produces smoother, lower-amplitude transitions, whereas RouteTS more closely follows the target in both domains.

combine distinct domain-specific responses within a single forecast, without assigning the entire signal to either domain.

B.3 Horizon-wise Routing Sensitivity and Validation–Test Discrepancy on Exchange

Figure 10 examines how the routing budget affects Exchange across individual forecasting horizons and data splits. The horizon-wise validation results exhibit substantially different preferences: $K = \text{full}$ performs best at $T = 96$, $K = 10$ is marginally preferred at $T = 192$, $K = 0$ is optimal at $T = 336$, and $K = 4$ achieves the lowest validation error at $T = 720$. Therefore, Exchange does not exhibit a consistent domain preference across forecasting horizons.

Following our dataset-level selection rule, we average the raw validation MSE over the four horizons before comparing the candidate budgets. This procedure selects $K = 4$. Its advantage is primarily associated with the longest horizon: at $T = 720$, the validation MSE increases by 124% and 79% under the time-domain and frequency-domain endpoints, respectively, relative to $K = 4$. This improvement offsets the weaker performance of $K = 4$ at the shorter horizons and determines the minimum of the averaged validation criterion.

The preference does not transfer to the test split. After averaging the raw test MSE over the four horizons, $K = \text{full}$ becomes optimal, whereas the validation-selected $K = 4$ incurs a 23% relative increase. By comparison, $K = 1$ and $K = 0$ are only 4.8% and 7.0% above the test optimum. Following the predefined evaluation procedure, we retain $K = 4$ in the main results rather than selecting a more favorable configuration using test performance.

This discrepancy illustrates a limitation of selecting one dataset-level routing budget when the horizon-wise rankings vary substantially: an advantage concentrated at one validation horizon may influence the averaged selection without generalizing to the test split. Exchange is the only dataset for which the validation-selected and test-optimal budgets differ in our evaluation. More granular horizon-specific or sample-adaptive budget selection may alleviate this sensitivity, but is beyond the scope of the present work.

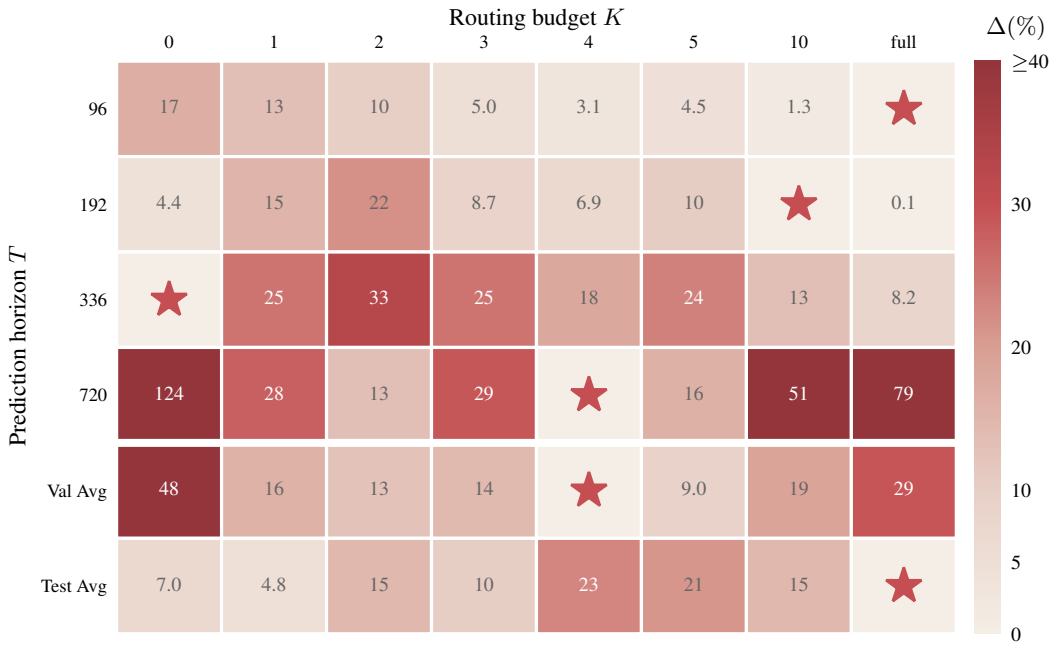


Figure 10: Routing-budget sensitivity on Exchange. Rows corresponding to individual forecasting horizons report the relative increase in validation MSE under each routing budget K. "Val Avg." and "Test Avg." are obtained by first averaging the raw MSE over the four horizons and then computing the relative increase from the minimum within the corresponding row. Stars mark the row-wise minima. The color scale is clipped at 40%, while the annotated values show the original, unclipped percentages.

C Ablation Study

C.1 Time-Domain Predictor Architecture

We examine the architectural choice of the time-domain predictor by replacing the MLP described in Section 3.3 with a shared Linear mapping, while keeping the routing configuration and all other settings unchanged. The comparison covers two mixed-routing datasets, Traffic and Solar ($K = 1$), and two time-domain endpoint datasets, PM2.5 and COVID-19 ($K = 0$).

As shown in Figure 11, the MLP consistently achieves lower MSE than the Linear predictor across all datasets and forecasting horizons, although the magnitude of the improvement varies considerably. Under mixed routing, the advantage is moderate on Traffic but more pronounced on Solar, indicating that the complementary components remaining after amplitude-based selection can require different levels of temporal modeling capacity across datasets.

The contrast is particularly clear between the two $K = 0$ datasets, for which the time-domain predictor receives the complete normalized input. On COVID-19, the gap widens substantially with the forecasting horizon: the MSE of the Linear predictor increases from approximately 0.28 to 1.35, whereas that of the MLP remains below 0.30. On PM2.5, both predictors remain comparatively close, although the MLP retains a consistent advantage, especially at longer horizons. Thus, the routing budget alone does not determine the required complexity of the time-domain predictor; datasets with the same domain allocation can still exhibit markedly different temporal forecasting behavior.

Overall, these results support the MLP as the default time-domain predictor in RouteTS. Its advantage is consistent across both mixed-routing and time-domain endpoint configurations, while becoming particularly substantial on datasets for which a single Linear mapping is insufficient over extended horizons.

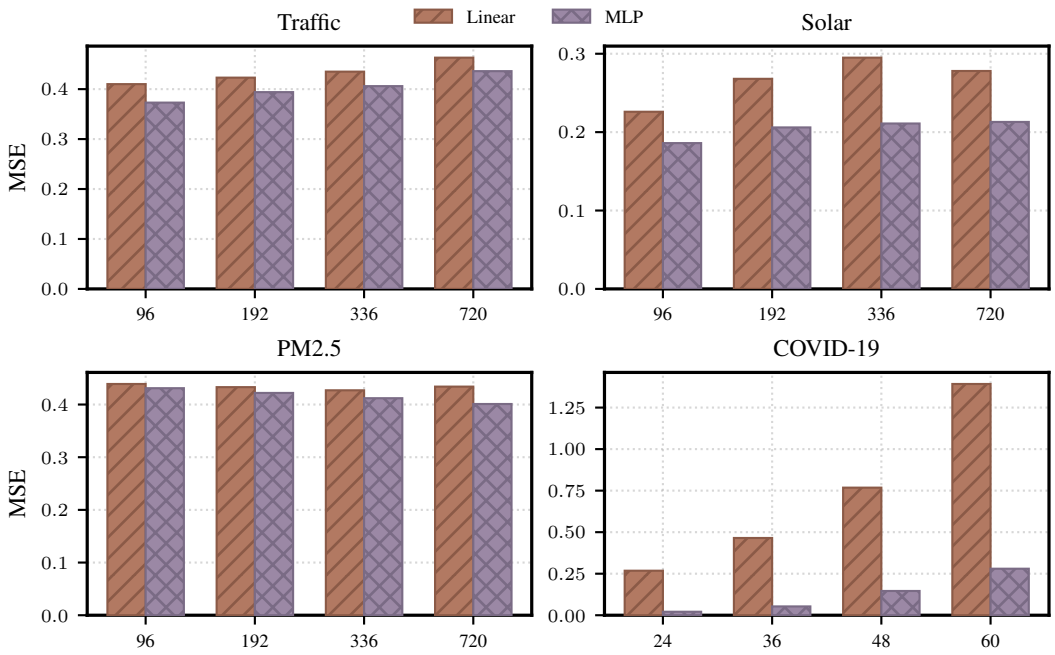


Figure 11: Comparison of Linear and MLP architectures for the time-domain predictor across four datasets and forecasting horizons. Lower MSE indicates better performance.

C.2 Frequency-Domain Predictor Architecture

The complex-valued predictor jointly transforms the real and imaginary parts of the routed spectrum through a shared pair of weight matrices. To assess whether this structure improves forecasting, we compare it with independent real-valued projections for the two parts. For each dataset, we fix the routing budget selected for the default model on validation data and change only the frequency predictor.

The default predictor applies a shared complex weight matrix $\mathbf{W} = \mathbf{W}_R + i\mathbf{W}_I \in \mathbb{C}^{F' \times F}$ to the routed spectrum $\mathcal{X}_{\text{freq}} = \mathcal{X}_R + i\mathcal{X}_I$:

$$\begin{aligned} \hat{\mathcal{X}}_{\text{freq}}^{(\text{complex})} &= (\mathbf{W}_R + i\mathbf{W}_I)(\mathcal{X}_R + i\mathcal{X}_I) \\ &= (\mathbf{W}_R\mathcal{X}_R - \mathbf{W}_I\mathcal{X}_I) + i\,(\mathbf{W}_R\mathcal{X}_I + \mathbf{W}_I\mathcal{X}_R)\,. \end{aligned} \tag{14}$$

The cross-terms allow both input parts to contribute to each output part, with $\mathbf{W}_R$ and $\mathbf{W}_I$ reused across the two mappings.

The real-valued variant instead transforms the two parts independently:

$$\hat{\mathcal{X}}_{\text{freq}}^{(\text{real})} = \mathbf{W}_R^{\text{ind}}\mathcal{X}_R + i\mathbf{W}_I^{\text{ind}}\mathcal{X}_I, \tag{15}$$

where $\mathbf{W}_R^{\text{ind}}, \mathbf{W}_I^{\text{ind}} \in \mathbb{R}^{F' \times F}$ are learned independently. Both predictors use two real-valued projection matrices of the same dimensions. The comparison therefore evaluates the structured real–imaginary coupling and weight sharing of the complex predictor without changing the number of learned projection parameters.

Table 6 shows the clearest benefit on Solar: replacing the complex predictor with independent projections increases MSE by 2.2%–4.0% across all four horizons. Complex also achieves lower MSE and MAE at every horizon on ETTh2, with the largest MSE gap at $T = 96$ (4.0%). On Traffic, MSE differences remain within 0.3%, consistent with the small differences among component-selection strategies in Table 3. Together, these observations suggest that overall accuracy is relatively insensitive to the precise frequency-branch design in this setting. On Exchange, the advantage of Real at $T = 96$ may reflect a benefit from learning separate mappings for the real and imaginary inputs

Table 6: Comparison of complex-valued and independent real-valued frequency projections at the validation-selected routing budgets. $\Delta\mathrm{MSE}$ is computed as $(\mathrm{MSE}_{\mathrm{real}} - \mathrm{MSE}_{\mathrm{complex}})/\mathrm{MSE}_{\mathrm{complex}} \times 100\%$ using unrounded MSE; positive values favor the complex-valued predictor.

Dataset	K	Horizon	Complex (MSE / MAE)	Real (MSE / MAE)	ΔMSE (%)
Traffic	1	96	0.373 / 0.259	0.373 / 0.259	+0.0
		192	0.394 / 0.269	0.393 / 0.268	−0.3
		336	0.406 / 0.274	0.406 / 0.274	+0.0
		720	0.436 / 0.292	0.437 / 0.292	+0.2
Solar	1	96	0.186 / 0.236	0.190 / 0.237	+2.2
		192	0.201 / 0.248	0.209 / 0.261	+4.0
		336	0.207 / 0.254	0.214 / 0.267	+3.4
		720	0.211 / 0.258	0.216 / 0.268	+2.4
ETTh2	full	96	0.275 / 0.337	0.286 / 0.353	+4.0
		192	0.336 / 0.377	0.337 / 0.383	+0.3
		336	0.355 / 0.396	0.356 / 0.399	+0.3
		720	0.384 / 0.422	0.385 / 0.424	+0.3
Exchange	4	96	0.101 / 0.224	0.096 / 0.219	−4.7
		192	0.241 / 0.355	0.243 / 0.357	+0.7
		336	0.448 / 0.491	0.450 / 0.494	+0.5
		720	1.040 / 0.778	1.058 / 0.781	+1.8

in this setting. The coupled complex-valued predictor nevertheless achieves lower MSE at all three longer horizons.

Overall, Complex achieves lower MSE in 12 of the 16 comparisons, ties in two at the reported precision, and has higher MSE in two. The gains on Solar and ETTh2 support the value of jointly transforming the real and imaginary spectral components through the complex-valued structure. These results favor the complex-valued projection as the default frequency predictor, improving accuracy in most evaluated settings without additional learned projection parameters.

D Robustness and Input-Length Sensitivity

Beyond forecasting accuracy and computational efficiency, we examine the behavior of RouteTS under noisy observations and different amounts of historical context. We compare against representative baselines covering attention-based (PhaseFormer), linear (DLinear), frequency-domain (FITS), and hybrid time–frequency (Amplifier) architectures.

D.1 Gaussian Noise Injection

To evaluate robustness to measurement noise at the prediction horizon $T = 96$, we perturb the standardized input sequences using zero-mean Gaussian noise with ratios $\sigma_{\mathrm{noise}} \in \{0.0, 0.2, 0.4, 0.6, 0.8\}$. For each input window and variable, the noise standard deviation is set to σ_{noise} times its temporal standard deviation, and the perturbation is applied before RevIN. We evaluate the models on Solar and ETTm2 using their pretrained checkpoints. For RouteTS, the routing budgets selected on clean validation data are kept fixed across all noise levels: $K = 1$ for Solar and $K = \mathrm{full}$ for ETTm2.

Figure 12 shows that all models experience increasing errors as the noise level rises, but their degradation patterns differ substantially. On ETTm2, RouteTS achieves the lowest MSE from the clean setting through $\sigma_{\mathrm{noise}} = 0.4$, while FITS becomes more accurate at the two highest noise levels. On Solar, RouteTS remains the strongest model through $\sigma_{\mathrm{noise}} = 0.6$. PhaseFormer is particularly sensitive on this dataset, with its MSE increasing from 0.192 to 0.853, whereas RouteTS exhibits a considerably smaller increase.

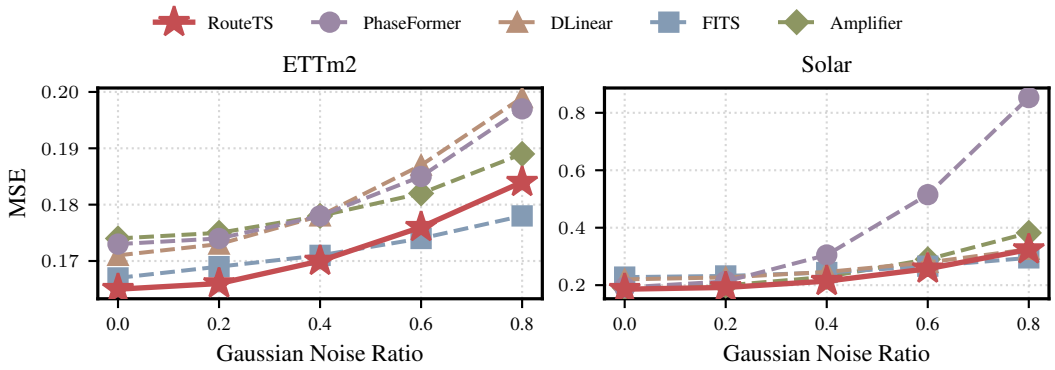


Figure 12: Test MSE under increasing Gaussian noise ratios on ETTm2 (K = full) and Solar (K = 1). RouteTS performs best under clean and moderate-noise conditions, while FITS achieves lower errors at the highest noise levels, consistent with the noise-suppression effect of its compact spectral representation. The routing budgets of RouteTS are fixed across all noise levels.

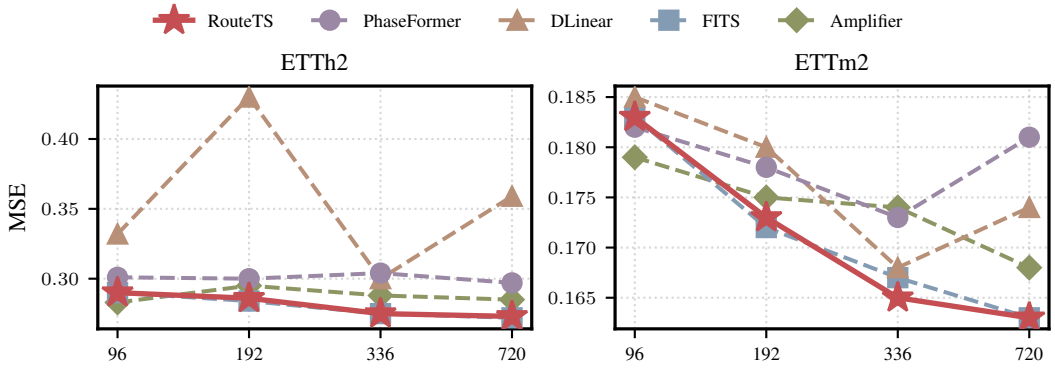


Figure 13: Sensitivity to the look-back window length at the prediction horizon $T = 96$ on ETTh2 and ETTm2, for which RouteTS uses K = full. RouteTS improves monotonically as the historical context increases and closely matches FITS at $L = 720$, while the remaining baselines exhibit varying degrees of sensitivity to the input length.

At $\sigma_{\text{noise}} = 0.8$, FITS obtains the lowest error on both datasets. This behavior is consistent with its compact spectral representation suppressing a substantial portion of the injected broadband noise. The same compression, however, can also remove informative non-dominant components, as reflected by its higher clean-data error on Solar (0.228, compared with 0.186 for RouteTS). These results indicate that RouteTS maintains strong clean- and moderate-noise accuracy while remaining competitive under severe perturbations, whereas more aggressive spectral compression becomes advantageous only at the highest noise levels.

D.2 Sensitivity to Input Length

We further evaluate sensitivity to the look-back window length $L \in \{96, 192, 336, 720\}$ at a fixed prediction horizon $T = 96$ on ETTh2 and ETTm2. Both datasets select the frequency-domain endpoint K = full, allowing us to examine whether the corresponding RouteTS configuration can benefit from increasingly long historical contexts.

As shown in Figure 13, the MSE of RouteTS decreases monotonically with increasing input length on both datasets. At $L = 720$, its performance is nearly identical to that of FITS, with the two models differing by approximately 0.001 MSE on each dataset. The remaining baselines exhibit more varied behavior. DLinear is particularly sensitive on ETTh2, where its MSE increases sharply at $L = 192$, while PhaseFormer does not consistently benefit from longer inputs. Amplifier remains comparatively stable but achieves higher errors than RouteTS at the longest input length.

Table 7: Full results of the routing-strategy comparison on Solar and Traffic for all prediction horizons. Random is reported for three fixed routing-mask seeds (2027–2029). Best results per horizon are in **bold**.

		Solar		Traffic	
Strategy	Horizon	MSE	MAE	MSE	MAE
Low-frequency	96	0.195	**0.233**	0.374	0.262
	192	0.215	0.249	0.395	0.272
	336	0.225	0.257	0.408	0.278
	720	0.227	0.260	0.438	0.295
Random (seed 2027)	96	0.198	**0.233**	**0.373**	0.262
	192	0.218	0.250	0.394	0.271
	336	0.227	0.258	0.407	0.277
	720	0.228	0.260	0.438	0.295
Random (seed 2028)	96	0.197	0.234	**0.373**	0.262
	192	0.218	0.251	0.394	0.272
	336	0.228	0.259	0.407	0.277
	720	0.228	0.261	0.437	0.295
Random (seed 2029)	96	0.197	**0.232**	**0.373**	0.262
	192	0.217	0.249	**0.393**	0.271
	336	0.227	0.257	0.407	0.277
	720	0.228	0.259	0.437	0.295
Amplitude (ours)	96	**0.186**	0.236	**0.373**	**0.259**
	192	**0.201**	**0.248**	0.394	**0.269**
	336	**0.207**	**0.254**	**0.406**	**0.274**
	720	**0.211**	**0.258**	**0.436**	**0.292**

Because K = full on both datasets, these results primarily characterize the long-context behavior of RouteTS's complex-valued frequency predictor rather than mixed-domain allocation. They show that this endpoint configuration can exploit additional historical observations without the instability observed in some competing models.

E FULL RESULTS

This section provides the horizon-specific results corresponding to the averaged results reported in the main text. Table 8 presents the complete forecasting results of RouteTS and all baselines across every prediction horizon, complementing the dataset-level averages in Table 1. Table 7 expands the component-selection comparison in Table 3 by reporting the performance of amplitude, low-frequency, and random routing at each prediction horizon on Solar and Traffic. For random routing, Table 7 lists the results for the three fixed routing-mask seeds separately, whereas Table 3 reports their mean and standard deviation.

Model		**RouteTS (our)**		PhaseFormer (2026)		MixLinear (2026)		FilterTS (2025)		CFPT (2025)		Amplifier (2025)		FreqMoE (2025)		TFPS (2025)		iTransformer (2024)		FITS (2024)		DLinear (2023)	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ILI	24	**1.658**	**0.787**	3.008	1.172	2.759	1.103	2.162	0.917	2.174	0.902	1.805	0.832	4.065	1.483	1.856	0.891	2.539	1.053	3.118	1.269	2.267	1.015
	36	**1.578**	**0.777**	2.618	1.075	2.505	1.052	2.100	0.926	1.981	0.899	2.188	0.908	3.978	1.436	2.203	0.929	2.508	1.055	3.164	1.274	2.339	1.000
	48	**1.557**	**0.796**	2.690	1.100	2.352	1.020	2.145	0.951	2.171	0.953	1.826	0.843	3.937	1.453	2.961	1.121	2.571	1.077	2.674	1.110	2.456	1.050
	60	**1.770**	**0.867**	2.588	1.107	2.447	1.057	2.121	0.951	2.252	0.976	1.990	0.892	3.728	1.379	2.617	1.057	2.699	1.116	2.644	1.108	2.645	1.121
	Avg	**1.640**	**0.807**	2.726	1.114	2.516	1.058	2.132	0.936	2.144	0.933	1.952	0.869	3.927	1.438	2.409	1.000	2.579	1.075	2.900	1.190	2.427	1.047
COVID-19	24	**0.021**	**0.070**	0.062	0.107	0.308	0.251	0.251	0.211	0.031	0.075	0.256	0.202	1.324	0.596	1.399	0.614	0.294	0.234	0.226	0.202	0.062	0.163
	36	**0.053**	**0.096**	0.175	0.185	0.566	0.313	0.257	0.238	0.067	0.110	0.544	0.329	1.910	0.721	0.402	0.306	0.697	0.375	0.500	0.310	0.112	0.225
	48	**0.146**	**0.158**	0.216	0.202	0.781	0.385	0.303	0.271	0.169	0.172	0.956	0.453	1.671	0.653	1.326	0.528	1.147	0.499	0.889	0.439	0.178	0.300
	60	**0.280**	**0.217**	0.390	0.273	3.096	0.933	0.431	0.318	1.396	0.573	1.133	0.492	2.320	0.779	1.843	0.580	1.632	0.615	1.443	0.585	0.376	0.443
	Avg	**0.125**	**0.135**	0.211	0.192	1.188	0.470	0.310	0.260	0.416	0.233	0.722	0.369	1.806	0.687	1.243	0.507	0.943	0.431	0.764	0.384	0.182	0.283
PM2.5	96	0.431	0.428	0.432	0.431	0.431	0.437	0.435	0.434	0.428	0.431	**0.414**	**0.420**	0.430	0.432	0.443	0.444	0.432	0.434	0.421	0.424	0.422	0.448
	192	0.422	0.422	0.437	0.432	0.430	0.436	0.433	0.432	0.427	**0.429**	**0.399**	**0.411**	0.429	0.430	0.437	0.434	0.430	**0.429**	0.426	0.430	**0.418**	0.445
	336	0.412	0.420	0.430	0.430	0.424	0.434	0.436	0.430	0.416	0.424	**0.376**	**0.407**	0.445	0.440	0.418	0.431	0.417	0.426	0.422	0.430	0.401	0.448
	720	0.401	0.425	0.409	0.424	0.420	0.447	0.379	**0.413**	0.406	0.426	**0.375**	0.422	0.392	0.430	0.387	0.426	0.401	0.419	0.419	0.446	0.440	0.525
	Avg	0.416	0.424	0.427	0.429	0.426	0.438	0.421	0.427	0.419	0.428	**0.391**	**0.415**	0.424	0.433	0.421	0.434	0.420	0.427	0.422	0.432	0.420	0.467
Weather	96	0.160	0.212	0.153	0.204	0.188	0.240	0.150	0.200	0.146	0.198	0.150	0.203	0.156	0.207	0.152	0.209	0.163	0.213	**0.145**	**0.196**	0.175	0.236
	192	0.204	0.251	0.199	0.247	0.229	0.274	0.192	0.241	**0.188**	0.239	0.192	0.241	0.198	0.246	0.207	0.258	0.203	0.249	0.189	**0.238**	0.220	0.281
	336	0.253	0.289	0.251	0.284	0.274	0.304	0.248	0.284	**0.241**	0.278	**0.241**	0.279	0.247	0.283	0.300	0.315	0.255	0.290	**0.241**	**0.277**	0.265	0.318
	720	0.331	0.343	0.331	0.342	0.339	0.348	0.326	0.335	0.331	0.341	0.323	0.336	**0.319**	0.334	0.376	0.361	0.326	0.338	**0.319**	**0.333**	0.324	0.363
	Avg	0.237	0.274	0.233	0.269	0.257	0.292	0.229	0.265	0.227	0.264	0.226	0.265	0.230	0.268	0.259	0.286	0.237	0.272	**0.224**	**0.261**	0.246	0.299
Traffic	96	0.373	0.259	0.366	**0.242**	0.416	0.281	0.382	0.272	0.376	0.257	0.400	0.281	0.410	0.292	0.933	0.546	**0.359**	0.260	0.411	0.281	0.429	0.314
	192	0.394	0.269	0.387	**0.252**	0.430	0.286	0.400	0.279	0.403	0.266	0.410	0.283	0.423	0.296	0.736	0.425	**0.375**	0.269	0.424	0.285	0.446	0.324
	336	0.406	0.274	0.398	**0.256**	0.444	0.296	0.414	0.286	0.416	0.279	0.421	0.290	0.432	0.305	1.475	0.750	**0.389**	0.275	0.436	0.291	0.458	0.329
	720	0.436	0.292	0.438	**0.282**	0.474	0.314	0.449	0.306	0.446	0.295	0.456	0.310	0.461	0.321	1.343	0.694	**0.421**	0.291	0.464	0.307	0.484	0.343
	Avg	0.402	0.274	0.397	**0.258**	0.441	0.294	0.411	0.286	0.410	0.274	0.422	0.291	0.432	0.303	1.122	0.604	**0.386**	0.274	0.434	0.291	0.454	0.328
Solar	96	0.186	0.236	0.192	**0.231**	0.253	0.275	**0.183**	0.240	0.190	0.234	0.191	0.243	0.204	0.263	0.215	0.264	0.202	0.254	0.228	0.260	0.220	0.287
	192	**0.201**	0.248	0.212	**0.247**	0.282	0.308	0.217	0.268	0.210	0.254	0.216	0.264	0.213	0.289	0.227	0.267	0.219	0.266	0.260	0.278	0.254	0.315
	336	**0.207**	0.254	0.225	**0.251**	0.299	0.307	0.224	0.276	0.217	0.263	0.222	0.270	0.213	0.289	0.286	0.367	0.220	0.268	0.282	0.292	0.272	0.329
	720	**0.211**	0.258	0.217	**0.252**	0.290	0.298	0.222	0.277	0.221	0.270	0.225	0.274	0.216	0.289	0.245	0.281	0.229	0.279	0.282	0.292	0.272	0.331
	Avg	**0.201**	0.249	0.211	**0.245**	0.281	0.297	0.212	0.265	0.210	0.255	0.213	0.263	0.211	0.283	0.243	0.295	0.217	0.267	0.263	0.281	0.254	0.316
ECL	96	0.134	0.228	0.128	**0.219**	0.172	0.273	0.131	0.225	**0.126**	0.220	0.132	0.226	0.146	0.247	0.190	0.297	0.131	0.227	0.146	0.246	0.140	0.237
	192	0.150	0.243	0.146	**0.236**	0.160	0.254	0.150	0.243	**0.145**	0.238	0.146	0.240	0.157	0.256	0.200	0.307	0.155	0.249	0.160	0.257	0.153	0.250
	336	0.167	0.260	0.165	**0.256**	0.202	0.298	0.172	0.268	**0.160**	0.257	0.165	0.258	0.174	0.272	0.221	0.330	0.169	0.265	0.176	0.273	0.169	0.267
	720	0.204	0.293	0.205	0.289	0.216	0.303	0.209	0.300	**0.190**	**0.282**	0.201	0.290	0.210	0.301	0.242	0.342	0.207	0.299	0.214	0.304	0.203	0.300
	Avg	0.164	0.256	0.161	0.250	0.188	0.282	0.165	0.259	**0.155**	**0.249**	0.161	0.253	0.172	0.269	0.213	0.319	0.165	0.260	0.174	0.270	0.166	0.263
ETTh1	96	**0.373**	0.394	0.419	0.405	0.401	0.412	**0.373**	0.398	0.375	0.396	0.374	0.398	0.378	0.399	0.412	0.436	0.402	0.418	**0.373**	**0.393**	**0.373**	0.395
	192	**0.405**	**0.412**	0.439	0.425	0.424	0.425	0.422	0.428	0.413	0.421	0.441	0.440	0.432	0.430	0.451	0.458	0.448	0.446	0.406	0.413	0.416	0.427
	336	**0.427**	**0.425**	0.461	0.441	0.428	0.431	0.443	0.443	0.432	0.436	0.438	0.447	0.715	0.586	0.504	0.494	0.467	0.464	0.431	0.429	0.460	0.462
	720	**0.421**	**0.443**	0.451	0.447	0.423	0.444	0.449	0.465	0.472	0.478	0.523	0.502	0.742	0.622	0.507	0.503	0.492	0.498	0.442	0.457	0.481	0.495
	Avg	**0.407**	**0.419**	0.442	0.429	0.419	0.428	0.422	0.434	0.423	0.432	0.444	0.447	0.567	0.509	0.468	0.473	0.452	0.456	0.413	0.423	0.433	0.445
ETTh2	96	0.275	**0.337**	0.304	0.361	0.292	0.348	**0.274**	**0.337**	**0.274**	**0.337**	0.288	0.352	0.295	0.353	0.347	0.392	0.304	0.360	0.275	**0.337**	0.300	0.365
	192	0.336	**0.377**	0.371	0.398	0.350	0.380	0.342	0.381	0.342	0.384	0.343	0.390	0.359	0.395	0.381	0.415	0.392	0.413	**0.335**	**0.377**	0.407	0.434
	336	0.355	**0.396**	0.383	0.413	0.369	0.401	0.369	0.404	0.369	0.409	**0.351**	0.407	0.373	0.415	0.360	0.409	0.426	0.438	0.360	0.398	0.469	0.478
	720	**0.384**	**0.422**	0.415	0.443	0.392	0.427	0.403	0.435	0.427	0.450	0.436	0.458	0.485	0.493	0.393	0.430	0.416	0.443	0.386	0.424	0.752	0.609
	Avg	**0.338**	**0.383**	0.368	0.404	0.350	0.389	0.347	0.389	0.353	0.395	0.354	0.402	0.378	0.414	0.370	0.411	0.385	0.414	0.339	0.384	0.482	0.472
ETTm1	96	**0.288**	**0.342**	0.294	0.344	0.333	0.365	0.292	0.346	0.294	0.346	0.295	0.344	0.313	0.353	0.316	0.370	0.302	0.357	0.303	0.345	0.303	0.346
	192	**0.328**	0.366	0.342	0.375	0.352	0.374	0.345	0.378	0.336	0.373	0.331	0.367	0.348	0.376	0.356	0.389	0.345	0.383	0.337	**0.365**	0.337	0.368
	336	0.364	0.387	0.374	0.394	0.386	0.391	0.384	0.402	**0.362**	0.386	0.365	0.387	0.392	0.401	0.400	0.416	0.379	0.403	0.372	**0.385**	0.370	0.388
	720	0.429	0.424	0.425	0.426	0.432	0.420	0.452	0.438	**0.423**	0.421	0.430	0.424	0.471	0.454	0.466	0.458	0.439	0.437	0.428	**0.416**	0.427	0.422
	Avg	**0.352**	0.380	0.359	0.385	0.376	0.388	0.368	0.391	0.354	0.382	0.355	0.380	0.381	0.396	0.385	0.408	0.366	0.395	0.360	**0.378**	0.359	0.381
ETTm2	96	**0.165**	0.255	0.173	0.261	0.178	0.267	0.176	0.260	0.167	**0.254**	0.174	0.258	0.175	0.260	0.184	0.275	0.180	0.272	0.167	0.256	0.168	0.262
	192	**0.221**	**0.292**	0.249	0.308	0.230	0.298	0.232	0.301	0.228	0.295	0.241	0.301	0.252	0.311	0.237	0.306	0.239	0.311	0.222	0.295	0.224	0.303
	336	**0.275**	0.328	0.291	0.340	0.284	0.332	0.283	0.335	**0.275**	**0.327**	0.289	0.335	0.294	0.344	0.313	0.353	0.389	0.341	0.277	0.329	0.281	0.342
	720	0.368	0.383	0.380	0.394	0.373	0.386	0.396	0.402	**0.366**	0.386	0.381	0.395	0.423	0.426	0.460	0.431	0.374	0.392	**0.366**	**0.382**	0.397	0.421
	Avg	**0.257**	**0.315**	0.273	0.326	0.266	0.321	0.272	0.325	0.259	0.316	0.271	0.322	0.286	0.335	0.299	0.341	0.296	0.329	0.258	0.316	0.268	0.332
Exchange	96	0.101	0.224	0.102	0.227	0.101	0.227	0.095	0.217	0.090	0.212	0.092	0.214	**0.085**	**0.206**	0.116	0.249	0.099	0.226	0.096	0.214	0.089	0.208
	192	0.241	0.355	0.220	0.337	0.190	0.313	0.192	0.312	0.191	0.312	0.183	0.308	0.179	0.307	0.250	0.377	0.216	0.337	0.196	0.309	**0.156**	**0.288**
	336	0.448	0.491	0.397	0.462	0.329	0.417	0.387	0.446	0.360	0.434	0.359	0.437	0.338	0.420	0.482	0.520	0.387	0.461	0.380	0.420	**0.272**	**0.394**
	720	1.040	0.778	0.972	0.744	0.871	0.700	0.943	0.714	0.971	0.732	1.159	0.788	1.025	0.779	1.136	0.806	0.962	0.745	0.854	0.752	**0.814**	**0.678**
	Avg	0.458	0.462	0.423	0.443	0.373	0.414	0.404	0.422	0.403	0.422	0.448	0.436	0.407	0.427	0.496	0.488	0.416	0.442	0.382	0.424	**0.333**	**0.392**

Table 8: Full multivariate time series forecasting results for all prediction lengths. Avg means the average of all prediction lengths. The best results are highlighted in **bold**, while the second-best results are underlined.