TIMEHSG: HIERARCHICAL LARGE LANGUAGE MODEL SEMANTIC GUIDANCE FOR MULTIVARIATE TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Accurate multivariate time-series forecasting requires jointly modeling variable-specific local dynamics, cross-variable influences, and long-range periodic structures. However, existing forecasting methods often fail to capture these complementary factors within a unified framework. Recent large language model (LLM)-based approaches typically rely on model fine-tuning or direct feature fusion without assigning distinct forecasting roles to frozen LLM embeddings. We use frozen LLM-derived embeddings as hierarchical guidance rather than as prediction features for direct fusion. Based on this insight, we propose **TimeHSG**, a multivariate forecasting framework that combines embedding-guided local and relational modeling with frequency-domain residual learning. Specifically, Semantic-Conditioned Adaptive Modulation (SCAM) uses the embeddings for feature-level modulation of local temporal representations, while the Dynamic Semantic Graph (DSG) uses them to construct interaction topologies for dynamic cross-variable dependencies. A frequency-domain residual branch further complements local encoding by capturing long-range periodic and low-frequency patterns. Experiments on multiple real-world benchmarks show that TimeHSG achieves competitive or superior forecasting performance. By keeping the LLM frozen and reusing cached embeddings, TimeHSG also provides a favorable balance between forecasting accuracy and computational efficiency.

1 INTRODUCTION

Time-series forecasting (TSF) is a fundamental machine learning task that supports data-driven decision-making in traffic flow prediction (Zou et al., 2026), weather prediction (Lam et al., 2023), and public health surveillance (Ali & Cowling, 2021). Given historical observations, TSF aims to estimate future values over a predefined horizon. Despite substantial progress, accurate multivariate long-term forecasting remains challenging because future evolution is jointly governed by multiple forms of temporal structure. Each variable exhibits local dynamics that may contain transient fluctuations and variable-specific changes. Different variables interact through complex dependencies, while long-range periodic patterns provide global context beyond local observations. Capturing these complementary factors is essential for robust forecasting across diverse datasets and prediction horizons.

Classical statistical methods, such as autoregressive integrated moving average (ARIMA) and its variants, provide interpretable forecasting mechanisms but have limited ability to model nonlinear dynamics in high-dimensional sequences (Box et al., 2015). Deep neural networks improve representational capacity by learning temporal patterns directly from data (Salinas et al., 2020; Challu et al., 2023). Transformer-based models further advance long-term forecasting through efficient attention, patching, decomposition, and inverted modeling (Zhou et al., 2021; Nie et al., 2023; Liu et al., 2024b). Graph neural networks explicitly characterize cross-variable dependencies (Wu et al., 2019; 2020; Cao et al., 2020), whereas frequency-domain methods capture periodic and global temporal structures (Zhou et al., 2022; Wu et al., 2023; An et al., 2026). However, these methods primarily derive representations and variable relations from numerical observations. Local encoders may underrepresent long-range periodicity, while dependency structures learned from numerical co-movement can be sensitive to transient or spurious correlations. Moreover, local dynamics, variable interactions,

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.

and global periodicity are often handled within a shared representation or by isolated mechanisms. This treatment makes it difficult to preserve their distinct contributions. Consequently, variables with similar observations may be treated as strongly related even when their underlying temporal behaviors differ.

Recently, large language models (LLMs) have offered a complementary source of contextual prior embeddings. When numerical histories are rendered through fixed textual templates, pretrained language models can encode high-level characteristics such as trend direction, variation, periodicity, and distributional change. These embeddings provide contextual information beyond pointwise numerical observations, helping forecasting models distinguish sequences with similar recent values but different underlying temporal behaviors. They can therefore complement numerical representations by providing higher-level priors for interpreting local dynamics and identifying meaningful relationships among variables. Recent studies have shown that incorporating pretrained LLM representations can improve forecasting performance through prompting, reprogramming, fine-tuning, cross-modal alignment, and semantic fusion (Gruver et al., 2023; Jin et al., 2024; Zhou et al., 2023; Liu et al., 2025; Hu et al., 2025).

However, two limitations remain in how existing methods use these representations. First, adapting LLMs or coupling them deeply with forecasting backbones can introduce substantial training, memory, and inference overhead. Second, many methods inject LLM embeddings through direct concatenation, alignment, or fusion. This strategy treats the embeddings as additional forecasting features but does not assign them distinct roles in temporal and cross-variable modeling. As a result, the contribution of the embeddings to individual forecasting components can become unclear, and heterogeneous LLM and numerical representations may be excessively mixed. As illustrated in Figure 1, these common integration patterns differ from our role-specific view. In our role-specific view, frozen LLM embeddings guide distinct forecasting components rather than serve as numerical features for direct fusion.

Motivated by this distinction, we propose TimeHSG, a Hierarchical LLM Semantic Guidance framework for multivariate long-term forecasting. TimeHSG keeps the LLM frozen and uses variable-level embeddings as non-intrusive guidance. Semantic-Conditioned Adaptive Modulation (SCAM) calibrates local temporal representations, while a Dynamic Semantic Graph (DSG) guides sample-specific cross-variable interactions. Because long-range periodicity is intrinsic to the numerical sequence, a separate frequency-domain residual branch preserves periodic and low-frequency information. This role-specific design applies the embeddings only at the feature and topology levels, avoiding LLM fine-tuning, reprogramming, and deep cross-modal coupling.

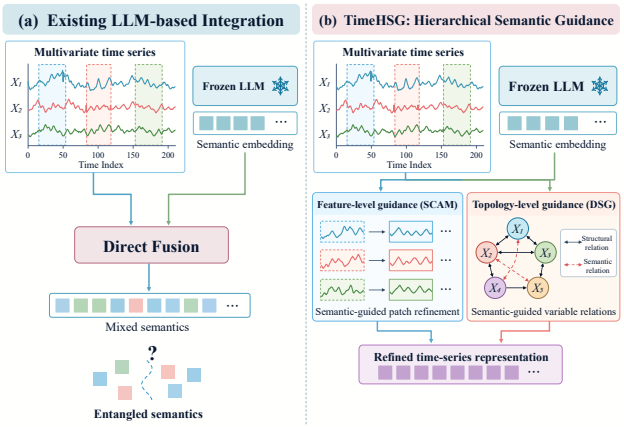


Figure 1: Comparison between existing LLM-based TSF methods and TimeHSG.

Experiments on eight widely used multivariate forecasting benchmarks show the effectiveness of TimeHSG. Compared with both LLM-enhanced and non-LLM forecasting models, TimeHSG achieves competitive overall performance. It obtains a total of 10 first-place results in mean squared error (MSE) and 12 in mean absolute error (MAE) across the two groups of comparisons. Ablation studies verify the benefits of LLM-derived embeddings, feature-level semantic modulation, topology-level semantic interaction, and frequency-domain residual modeling. Visualizations further illustrate selective feature calibration and structured semantic interactions among variables in representative forecasting settings. The efficiency analysis shows that the frozen, guidance-based design maintains a relatively small parameter count and low memory and inference costs compared with representative LLM-based methods.

The main contributions are summarized as follows:

- We propose TimeHSG, a multivariate forecasting framework that uses frozen LLM-derived embeddings to guide local representation learning and cross-variable dependency modeling.
- We introduce SCAM and DSG to use these embeddings at two complementary levels: feature-level adaptive modulation of local temporal representations and topology-level graph construction for dynamic variable interaction.
- We develop a time-frequency decoupled architecture in which a frequency-domain residual branch complements local temporal encoding with long-range periodic and low-frequency information.
- Experiments on eight benchmarks show that TimeHSG achieves competitive forecasting accuracy against both LLM-based and non-LLM baselines while maintaining relatively low parameter, memory, and inference costs.

2 Related Work

TSF Models Time-series forecasting has evolved from statistical models based on stationarity assumptions (Box et al., 2015) to deep neural architectures that capture nonlinear temporal patterns (Lai et al., 2018). For long-horizon forecasting, Informer improves attention efficiency, whereas Autoformer incorporates trend–seasonal decomposition (Zhou et al., 2021; Wu et al., 2021). PatchTST and iTransformer show the effectiveness of patch-based and inverted modeling designs (Nie et al., 2023; Liu et al., 2024b), while GTR retrieves global temporal information to capture long-range periodic patterns (Cao et al., 2026). Despite their different architectures, these methods mainly learn temporal patterns and variable dependencies from historical observations alone. In contrast, TimeHSG uses frozen LLM-derived semantic embeddings as hierarchical guidance. This design regulates both local temporal features and cross-variable interactions while preserving a compact numerical forecasting backbone.

Graph Structures and Frequency-Domain Modeling Graph-based TSF models explicitly capture cross-variable dependencies by learning adaptive or dynamic structures (Wu et al., 2019; 2020; Cao et al., 2020). However, when graph construction relies only on numerical co-movement, the learned topology may reflect spurious correlations rather than meaningful dependencies. Frequency-domain modeling provides another useful inductive bias for long-term forecasting by capturing periodicity and low-frequency trends (Zhou et al., 2022; Wu et al., 2023; Cao et al., 2020). FreDN further uses learnable frequency decomposition to disentangle spectral components (An et al., 2026). Existing graph and frequency methods are effective, but they usually treat structural dependency and spectral modeling as purely numerical operations. In contrast, TimeHSG constructs semantic-aware dynamic graphs from LLM-generated variable representations and uses a decoupled frequency residual branch to complement local time-domain encoding with global periodic information.

TSF Empowered by LLMs Pretrained language models have recently been applied to TSF through prompts that encode historical statistics, metadata, variable descriptions, or forecasting objectives. Early studies adapt language models to numerical series through prompting and reprogramming (Gruver et al., 2023; Jin et al., 2024), while time-series foundation models show potential for zero-shot and cross-domain forecasting (Ansari et al., 2024; Das et al., 2024). A parallel line of work aligns language and numerical sequences for forecasting (Liu et al., 2025; Chowdhury et al., 2026). Context-Alignment (Hu et al., 2025) couples time-series inputs with linguistic components, and DualSG (Ding et al., 2025) injects caption-level guidance. Unlike these methods, TimeHSG is motivated by three complementary forecasting requirements: variable-specific local dynamics, cross-variable influences, and long-range periodic structure. It assigns targeted roles to LLM-derived semantics for the first two requirements through feature-level calibration and topology-level interaction, while a separate numerical frequency branch preserves long-range periodic structure. This factor-oriented formulation enables each component to address a distinct source of predictive information without indiscriminately coupling linguistic semantics and numerical temporal patterns.

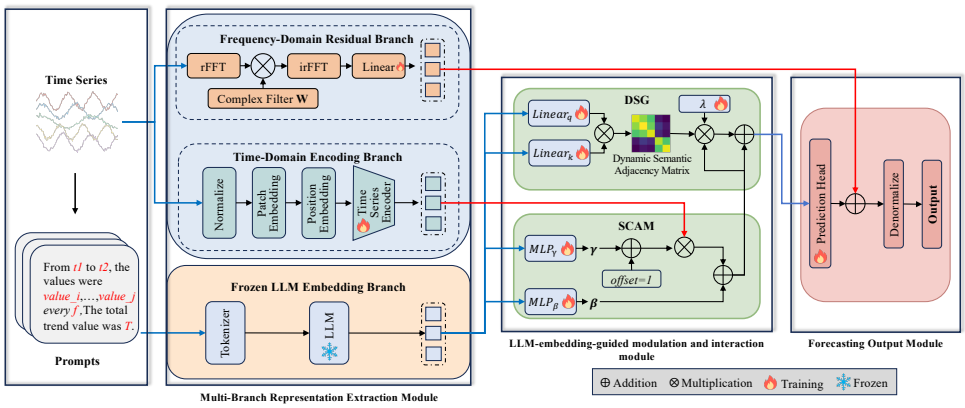


Figure 2: Overall architecture of TimeHSG. A time-domain backbone extracts patch-level temporal features, while a frozen LLM produces variable-level embeddings. SCAM and DSG use these embeddings for feature-level modulation and topology-level interaction, and a frequency residual branch complements global periodic patterns.

3 Methodology

3.1 Problem Statement

Given a historical input sequence $\mathbf{X} \in \mathbb{R}^{B\times L\times N}$ (batch size B, input length L, number of variables N), long-term TSF aims to predict the next H-step future values $\hat{\mathbf{Y}} \in \mathbb{R}^{B\times H\times N}$.

3.2 Overall Architecture

As illustrated in Figure 2, TimeHSG consists of a multi-branch representation extraction module, an LLM-embedding-guided modulation and interaction module, and a forecasting head. In the extraction module, the time-domain branch captures local temporal patterns, the frozen LLM produces variable-level embeddings, and the frequency-domain branch preserves global periodic structure.

TimeHSG keeps the LLM frozen and encodes a fixed variable-level template containing the time range, historical values, sampling frequency, and total trend value. The resulting embeddings guide only SCAM and DSG, while the time-domain and frequency-domain branches remain decoupled to model local fluctuations and global trends separately.

3.3 Multi-Branch Representation Extraction Module

The time-domain branch applies patch embedding, positional encoding, and a Transformer encoder. The LLM branch encodes the history of each variable through a fixed template to produce embeddings for subsequent guidance. The frequency branch operates on the full sequence and provides a residual forecast.

Time-Domain Encoding Branch The time-domain branch models local variations. To reduce scale differences across variables, the input is first normalized:

$$\begin{aligned} \bar{x}_{b,n} &= \frac{1}{L}\sum_{t=1}^{L} X_{b,t,n}, \\ \rho_{b,n} &= \sqrt{\frac{1}{L}\sum_{t=1}^{L}(X_{b,t,n}-\bar{x}_{b,n})^2+\epsilon}, \\ X^{\text{norm}}_{b,t,n} &= \frac{X_{b,t,n}-\bar{x}_{b,n}}{\rho_{b,n}}. \end{aligned} \tag{1}$$

The normalized input is then transposed as $\mathbf{X}_{\text{time}} \in \mathbb{R}^{B\times N\times L}$.

For each variable, TimeHSG applies an independent patch embedding. Let the patch length be $l_p = 8$, the stride be $s = 4$, and the hidden dimension be C. The number of patches is $P = \left\lfloor \frac{L-l_p}{s} \right\rfloor + 1$.

Each patch is projected by

$$U_{b,n,p,c} = \sum_{i=0}^{l_p-1} W_{c,i}^{\text{patch}} X_{\text{time},b,n,(p-1)s+i+1} + a_c, \tag{2}$$

where $\mathbf{U} \in \mathbb{R}^{B\times N\times P\times C}$, W^{patch} denotes the patch projection kernel, and a_c is a learnable bias.

Learnable position embeddings are added: $\mathbf{U}' = \mathbf{U} + \mathbf{P}_{\text{pos}}$.

$\mathbf{U}'$ is encoded by a Transformer: $\mathbf{H} = f_{\text{enc}}(\mathbf{U}') \in \mathbb{R}^{B\times N\times P\times C}$.

This yields compact temporal features for subsequent modulation and variable interaction.

Frequency-Domain Residual Branch Patch encoding may weaken long-range periodicity and low-frequency trends. TimeHSG therefore adds a frequency-domain residual branch that acts on the full normalized sequence.

Given $\mathbf{X}_{\text{time}} \in \mathbb{R}^{B\times N\times L}$, each variable is transformed into the frequency domain:

$$\begin{aligned} \mathbf{X}_f &= \mathcal{F}_r(\mathbf{X}_{\text{time}}), \\ X_{f,b,n,k} &= \sum_{t=0}^{L-1} X_{\text{time},b,n,t+1} e^{-2\pi \mathrm{i} kt/L}. \end{aligned} \tag{3}$$

where $k = 0, \ldots, \lfloor L/2 \rfloor$ and $\mathrm{i}^2 = -1$.

A learnable complex-valued filter is applied: $\mathbf{X}'_f = \mathbf{X}_f \odot \mathbf{W}$. where $\mathbf{W}$ is learnable.

The filtered representation is transformed back: $\mathbf{X}_{\text{global}} = \mathcal{F}_r^{-1}(\mathbf{X}'_f)$.

A linear map projects length L to horizon H:

$$\mathbf{Y}_{\text{freq}} = f_{\text{freq}}(\mathbf{X}_{\text{global}}) \in \mathbb{R}^{B\times N\times H}. \tag{4}$$

This residual predictor captures spectral patterns that may be lost during patch encoding.

Frozen LLM Embedding Branch The frozen LLM provides variable-level embeddings without parameter updates; throughout this paper, "semantic" refers exclusively to the information represented in these outputs, rather than to template-based injection. For each sample and variable, TimeHSG converts the historical window into a fixed textual template. Let $\mathbf{x}_{b,n} = \mathbf{X}^{\text{norm}}_{b,:,n} \in \mathbb{R}^L$ denote the normalized historical series of variable n in sample b. We define the total trend value as $T_{b,n} = x_{b,n,L} - x_{b,n,1}$. Together with the start time t_1, end time t_2, and sampling frequency f, the observations are inserted into the following deterministic template:

From t_1 to t_2, the values were $x_{b,n,1}, \ldots, x_{b,n,L}$ every f. The total trend value was $T_{b,n}$.

Thus, the prompt can be written as $p_{b,n} = \mathcal{T}(t_1, t_2, \mathbf{x}_{b,n}, f, T_{b,n})$. The frozen LLM encoder encodes the fixed template as $\mathbf{e}_{b,n} = g_\psi(p_{b,n}) \in \mathbb{R}^{d_{\text{llm}}}$, where g_ψ is the frozen LLM encoder and $d_{\text{llm}} = 768$ by default. The embedding tensor is

$$\mathbf{E}_{b,n,:} = \mathbf{e}_{b,n}, \quad \mathbf{E} \in \mathbb{R}^{B\times N\times d_{\text{llm}}}. \tag{5}$$

The LLM parameters remain frozen during training. $\mathbf{E}$ is used only by SCAM and DSG, avoiding LLM fine-tuning and deep cross-modal coupling.

3.4 LLM-Embedding-Guided Modulation and Interaction Module

This module uses $\mathbf{E}$ in two ways: SCAM modulates temporal features, and DSG models variable topology.

Semantic-Conditioned Adaptive Modulation (SCAM) Instead of concatenating LLM and temporal features, SCAM performs feature-level conditional modulation.

Given $\mathbf{E}$, SCAM uses two multilayer perceptron (MLP) branches to generate scaling and shifting parameters:

$$\boldsymbol{\gamma} = \phi(f_\gamma(\mathbf{E})), \quad \phi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}, \quad \boldsymbol{\beta} = f_\beta(\mathbf{E}), \tag{6}$$

where $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^{B\times N\times C}$.

$\boldsymbol{\gamma}$ and $\boldsymbol{\beta}$ are broadcast to the patch dimension and applied to $\mathbf{H}$: $\tilde{\mathbf{H}} = \mathbf{H} \odot (1 + \boldsymbol{\gamma}) + \boldsymbol{\beta}$, where $\odot$ denotes element-wise multiplication.

SCAM conditions local temporal features on LLM embeddings without cross-modal attention.

Dynamic Semantic Graph (DSG) To capture cross-variable dependencies, TimeHSG further introduces a Dynamic Semantic Graph (DSG) parameterized by the LLM embeddings.

The SCAM output is first flattened: $\mathbf{Z}_{b,n,:} = \mathrm{vec}(\tilde{\mathbf{H}}_{b,n,:,:}) \in \mathbb{R}^{PC}$. Here, $\mathrm{vec}(\cdot)$ vectorizes the patch-channel matrix.

Query and key vectors are generated from the embeddings as $\mathbf{Q} = \mathbf{E}\mathbf{W}_Q$ and $\mathbf{K} = \mathbf{E}\mathbf{W}_K$, where $\mathbf{W}_Q, \mathbf{W}_K \in \mathbb{R}^{d_{\text{llm}}\times d_g}$ and $d_g = 64$. The resulting adjacency is

$$A_{b,i,j} = \frac{\exp(\mathbf{Q}_{b,i,:}\mathbf{K}_{b,j,:}^\top/\sqrt{d_g})}{\sum_{j'=1}^{N}\exp(\mathbf{Q}_{b,i,:}\mathbf{K}_{b,j',:}^\top/\sqrt{d_g})}, \tag{7}$$

where $\mathbf{A} \in \mathbb{R}^{B\times N\times N}$.

Variable interaction is then computed by

$$M_{b,i,j} \sim \text{Bernoulli}(1-r), \quad \tilde{\mathbf{A}} = \frac{\mathbf{M}\odot\mathbf{A}}{1-r}, \quad \mathbf{Z}_g = \mathbf{Z} + \lambda\tilde{\mathbf{A}}\mathbf{Z}, \tag{8}$$

where r is the dropout rate and λ is a learnable gating parameter.

DSG injects embedding-derived structural priors into the forecasting representation through dynamic message passing.

3.5 Forecasting Output

The DSG output is passed to a linear head: $\mathbf{Y}_{\text{main}} = f_{\text{head}}(\mathbf{Z}_g) \in \mathbb{R}^{B\times N\times H}$.

The normalized forecast adds the frequency residual: $\mathbf{Y}^{\text{norm}} = \mathbf{Y}_{\text{main}} + \mathbf{Y}_{\text{freq}}$.

It is then transposed to $[B, H, N]$ and denormalized: $\hat{Y}_{b,h,n} = \rho_{b,n}Y^{\text{norm}}_{b,n,h} + \bar{x}_{b,n}$.

The training objective is MSE:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{BHN}\sum_{b=1}^{B}\sum_{h=1}^{H}\sum_{n=1}^{N}\left(\hat{Y}_{b,h,n} - Y_{b,h,n}\right)^2. \tag{9}$$

Overall, TimeHSG combines local temporal encoding, frozen LLM embedding guidance, dynamic variable interaction, and frequency residual modeling within a unified forecasting framework.

4 Experiments

4.1 Experimental Settings

Datasets We evaluate TimeHSG on eight widely used multivariate TSF benchmarks: ETTm1, ETTm2, ETTh1, and ETTh2 from the Electricity Transformer Temperature (ETT) benchmark (Zhou et al., 2021), together with Weather, Traffic, Influenza-Like Illness (ILI), and Solar (Lai et al., 2018; Wu et al., 2021; Zhou et al., 2022; Nie et al., 2023). These datasets span different temporal granularities and patterns, providing a broad testbed for evaluating multivariate forecasting performance.

Table 1: Comparison with LLM-based time-series forecasting models. Results are averaged over four forecasting horizons. Best results are bolded; second-best results are underlined. Full results are reported in Table A3 of the Appendix.

Dataset	TimeHSG (Ours)		T3Time (2026)		TimeCMA (2025)		Time-LLM (2024)		UniTime (2024)		GPT4TS (2023)	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	**0.372**	**0.391**	0.373	0.393	0.380	0.392	0.410	0.409	0.385	0.399	0.396	0.401
ETTm2	**0.271**	**0.317**	0.281	0.324	0.275	0.323	0.296	0.340	0.293	0.334	0.294	0.339
ETTh1	0.428	**0.427**	**0.418**	0.430	0.423	0.431	0.448	0.443	0.442	0.448	0.457	0.450
ETTh2	**0.344**	**0.384**	0.351	0.392	0.372	0.397	0.381	0.404	0.378	0.403	0.389	0.414
Solar	0.260	0.299	0.290	0.312	0.559	0.470	0.313	0.320	0.252	0.294	**0.247**	**0.284**
Weather	**0.240**	**0.270**	0.244	0.279	0.250	0.276	0.275	0.291	0.253	0.276	0.279	0.297
Traffic	0.504	0.314	0.564	0.374	0.478	0.330	0.535	0.355	**0.450**	**0.291**	0.489	0.318
ILI	1.841	**0.847**	**1.758**	0.848	1.922	0.921	2.432	1.012	2.108	0.929	2.623	1.060
1st Count	**10**		2		0		0		2		2	

Table 2: Comparison with non-LLM time-series forecasting models. Results are averaged over four forecasting horizons. Best results are bolded; second-best results are underlined. Full results are reported in Table A4 of the Appendix.

Dataset	TimeHSG (Ours)		GTR (2026)		FreDN (2026)		SimpleTM (2025)		TQNet (2025)		RAFT (2025)		S-Mamba (2025)		iTransformer (2024)		DLinear (2023)		PatchTST (2023)	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	**0.372**	**0.391**	0.373	0.394	0.382	0.392	0.391	0.403	0.377	0.393	0.381	0.400	0.398	0.405	0.407	0.410	0.403	0.407	0.392	0.402
ETTm2	**0.271**	**0.317**	0.272	**0.317**	0.276	**0.317**	0.283	0.325	0.277	0.323	0.281	0.330	0.288	0.332	0.288	0.332	0.350	0.401	0.285	0.328
ETTh1	**0.428**	**0.427**	0.450	0.439	0.437	0.430	0.442	0.436	0.441	0.434	**0.428**	0.433	0.455	0.450	0.454	0.447	0.456	0.452	0.463	0.449
ETTh2	**0.344**	**0.384**	0.380	0.403	0.365	0.390	0.424	0.413	0.378	0.402	0.382	0.410	0.381	0.405	0.383	0.407	0.559	0.515	0.395	0.414
Solar	0.260	0.299	0.206	**0.255**	0.241	0.262	0.315	0.333	**0.198**	0.256	0.301	0.303	0.240	0.273	0.246	0.272	0.330	0.401	0.250	0.293
Weather	**0.240**	**0.270**	0.242	**0.270**	0.244	**0.270**	0.258	0.282	0.244	**0.270**	0.270	0.309	0.251	0.276	0.258	0.278	0.265	0.317	0.257	0.280
Traffic	0.504	0.314	0.465	0.282	0.471	0.289	0.545	0.347	0.445	**0.276**	**0.414**	0.284	**0.414**	**0.276**	0.423	0.283	0.627	0.388	0.482	0.308
ILI	**1.841**	**0.847**	1.997	0.864	2.254	0.924	1.844	0.857	1.974	0.865	5.089	1.526	2.999	1.181	2.444	1.203	2.616	1.090	2.388	1.011
1st Count	**12**		3		2		0		3		2		2		0		0		0	

Implementation Details All datasets follow the common train/validation/test splits used in the long-term TSF literature. Except for ILI, the input length is set to 96 and the forecast horizon is $H \in \{96, 192, 336, 720\}$. For ILI, the input length is 36 and the forecast horizon is $H \in \{24, 36, 48, 60\}$. The model is trained and evaluated separately for each horizon. To ensure a fair comparison, all baselines use either their public implementations or the experimental protocols reported in the original papers. We adopt the same experimental settings as T3Time (Chowdhury et al., 2026). Experiments are conducted using PyTorch on an NVIDIA GeForce RTX 3090 GPU.

Baselines To verify the effectiveness of TimeHSG, we compare it with two groups of representative forecasting models. The first group contains LLM-based forecasting methods, including T3Time (Chowdhury et al., 2026), TimeCMA (Liu et al., 2025), Time-LLM (Jin et al., 2024), UniTime (Liu et al., 2024a), and GPT4TS (Zhou et al., 2023). The second group contains non-LLM forecasting methods, including GTR (Cao et al., 2026), FreDN (An et al., 2026), SimpleTM (Chen et al., 2025), TQNet (Lin et al., 2025), RAFT (Han et al., 2025), S-Mamba (Wang et al., 2025), iTransformer (Liu et al., 2024b), DLinear (Zeng et al., 2023), and PatchTST (Nie et al., 2023).

4.2 Main Results

Table 1 shows that TimeHSG achieves the best MSE on 4/8 datasets and the best MAE on 6/8 datasets among LLM-based models. Averaged over the eight datasets, TimeHSG obtains an MSE of 0.533 and an MAE of 0.406. Relative to TimeCMA, TimeHSG reduces the average MSE by 8.56% and the average MAE by 8.22%. Relative to Time-LLM, it reduces the average MSE by 16.31% and the

Table 3: Ablation study of TimeHSG. Each cell reports MSE/MAE, averaged over four forecasting horizons; full horizon-level results are provided in Table A5 of the Appendix.

Dataset	w/o S	w/o S+D	w/o S+D+F	CA+F	MLP	RE	TimeHSG
Weather	0.250 0.275	0.252 0.275	0.257 0.279	0.252 0.277	0.251 0.275	0.251 0.274	**0.240** **0.270**
ETTm1	0.376 0.395	0.378 0.397	0.379 0.396	0.852 0.600	0.377 0.395	0.377 0.395	**0.372** **0.391**
ETTm2	0.275 0.320	0.273 0.320	0.276 0.322	0.319 0.360	0.274 0.320	0.273 0.320	**0.271** **0.317**
ILI	1.964 0.881	1.954 0.881	1.979 0.885	1.978 0.901	1.975 0.886	1.926 0.853	**1.841** **0.847**

average MAE by 9.09%. TimeHSG is also competitive with T3Time, reducing the average MAE by 3.07%, although T3Time remains stronger on ETTh1 and ILI in terms of MSE and GPT4TS performs best on Solar.

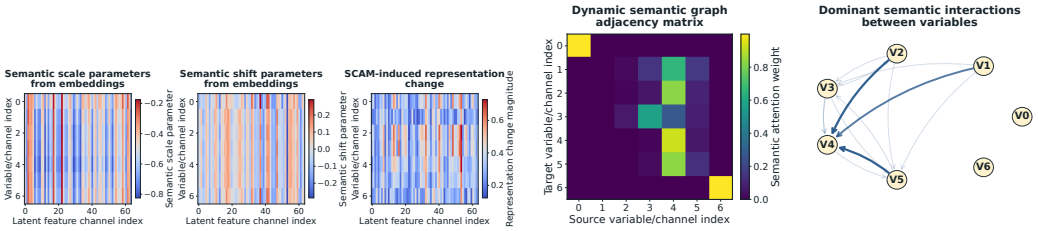
Table 2 further shows that TimeHSG is best or tied for best on 6/8 datasets in terms of MSE and on 6/8 datasets in terms of MAE. Relative to iTransformer, DLinear, and PatchTST, TimeHSG reduces the average MSE by 13.11%, 24.01%, and 13.27%, respectively. For MAE, the corresponding reductions are 10.55%, 18.18%, and 6.77%. The remaining gaps on Solar and Traffic clarify the scope of TimeHSG: it provides general spectral and dependency guidance rather than replacing explicit seasonal or road-topology priors.

We further evaluate result stability using three independent random seeds. Across the 16 dataset–metric combinations, the mean coefficient of variation is 0.276% and the maximum is 0.544%, indicating that the reported results are stable with respect to random initialization. The complete per-seed results and the corresponding mean $\pm$ standard deviation are provided in Table A6 in the Appendix.

4.3 Ablation Study

We conduct ablations on four datasets to test the three modeling requirements that motivate TimeHSG. The w/o S variant removes SCAM, w/o S+D additionally removes DSG, and w/o S+D+F further removes the frequency residual branch, thereby examining local dynamics, cross-variable interaction, and long-range periodic structure, respectively. CA+F replaces SCAM and DSG with standard cross-attention while retaining the frequency branch. MLP replaces the frozen LLM embedding with a learned two-layer MLP embedding, whereas RE replaces it with a random embedding; all remaining modules are unchanged.

The ablations are organized to examine the central design principle from complementary directions. With the guidance mechanism fixed, MLP and RE consistently underperform TimeHSG across all four datasets and both metrics, increasing MSE by 3.58% and 2.82% on average, respectively. This suggests that the gains depend on information carried by the frozen pretrained representations rather than on an arbitrary embedding passed through the same guidance pathway. Conversely, CA+F retains the frozen representations but changes how they are used, and again performs worse throughout, with a particularly large MSE increase of 129.03% on ETTm1. Access to the same pretrained representations is therefore not sufficient, because their effectiveness also depends on how their roles are defined within the forecasting model. The nested component ablations further clarify how these roles contribute to the full model. Removing SCAM increases MSE by 3.35% on average relative to TimeHSG, supporting feature-level modulation for variable-specific local dynamics. When DSG is also removed, MSE on Weather increases by 5.00% relative to TimeHSG, supporting the contribution of topology-level guidance to cross-variable interaction modeling. Removing the frequency branch as well further increases the average MSE degradation to 4.58%, supporting the complementary contribution of long-range periodic modeling. Together, these results support hierarchical guidance as an effective way to assign frozen pretrained representations explicit roles in local representation modulation and cross-variable interaction.



(a) SCAM modulation (b) DSG topology

Figure 3: Visualization of hierarchical semantic guidance on ETTm2 with the 336-step forecasting horizon. See Figures A1 and A2 in the Appendix for larger versions of the figures.

4.4 Visualization

To further examine the proposed semantic components, Figure 3 visualizes the behavior of SCAM and DSG. Figure 3(a) provides a feature-level view of semantic guidance: stronger responses indicate latent channels that receive greater semantic calibration, whereas weaker responses indicate channels that are preserved or only mildly adjusted. This pattern shows that SCAM does not uniformly perturb all temporal features; instead, it selectively amplifies or suppresses channel-wise representations according to the variable-level semantic embedding. Figure 3(b) provides a topology-level view: darker or stronger connections indicate stronger semantic-aware dependencies between variables, while weaker connections indicate limited interaction. The structured graph, rather than a dense uniform matrix, suggests that DSG uses LLM-derived statistical semantics as a general topological prior to guide both self-dependencies and cross-variable message passing.

4.5 Efficiency Analysis

Table 4 reports the efficiency comparison with representative LLM-based baselines. Because semantic embedding generation is shared across the compared methods, inference time is measured using precomputed and cached embeddings for all models. TimeHSG uses only 4.78M parameters, substantially fewer than T3Time and TimeCMA. It also achieves the fastest inference speed, requiring 0.087 ms per sample, which corresponds to reductions of 78.52% relative to T3Time and 39.58% relative to TimeCMA. In terms of memory usage, TimeHSG reduces the GPU memory footprint by 66.59% relative to T3Time and 65.62% relative to TimeCMA, while also achieving the lowest peak memory usage.

Table 4: Efficiency comparison with LLM-based baselines on ETTh2 with input length 96 and output length 720.

Metric	TimeHSG	T3Time	TimeCMA
Params (M)	**4.78**	29.80	28.49
Inference Time (ms)	**0.087**	0.405	0.144
GPU Mem. (MB)	**152.52**	456.47	443.65
Peak Mem. (MB)	**581.38**	1320.56	672.07

5 Conclusion

In this paper, we show that hierarchical guidance provides an effective way to leverage frozen pretrained representations for multivariate time-series forecasting. TimeHSG assigns the same frozen LLM embeddings two explicit roles: SCAM modulates local temporal representations, while DSG constructs dynamic variable-interaction topologies. A separate frequency residual branch complements them by modeling long-range periodic structures. Extensive experiments on eight benchmarks demonstrate competitive forecasting performance. More broadly, our results suggest that pretrained representations need not be treated simply as features to be fused with numerical signals; instead, they can serve as structured guidance with explicitly defined roles. We hope this perspective encourages further exploration of role-specific pretrained representation guidance for time-series forecasting.

AI USE STATEMENT

Generative AI tools were used solely as writing assistants to improve the grammar, clarity, and academic presentation of the manuscript. The research ideas, methodological design, model development, experiments, result analysis, and conclusions were independently conducted and determined by the authors. The authors carefully reviewed and revised all AI-assisted text and take full responsibility for the content of the manuscript.

REFERENCES

Sheikh Taslim Ali and Benjamin J. Cowling. Influenza virus: Tracking, predicting, and forecasting. *Annual Review of Public Health*, 42:43–57, 2021. doi: 10.1146/annurev-publhealth-010720-021049.

Zhongde An, Jinhong You, Jiyanglin Li, Yiming Tang, Wen Li, Heming Du, and Shouguo Du. Fredn: Spectral disentanglement for time series forecasting via learnable frequency decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19623–19631, 2026. doi: 10.1609/aaai.v40i24.39042.

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024.

George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel, and Greta M. Ljung. *Time Series Analysis: Forecasting and Control*. Wiley, 5 edition, 2015.

Defu Cao, Yujing Wang, Juanyong Duan, Ce Zhang, Xia Zhu, Congrui Huang, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. Spectral temporal graph neural network for multivariate time-series forecasting. In *Advances in Neural Information Processing Systems*, 2020.

Fanpu Cao, Lu Dai, Jindong Han, and Hui Xiong. Enhancing multivariate time series forecasting with global temporal retrieval. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/7a3284ce53f620d244fca0c79d94c478-Abstract-Conference.html.

Cristian Challu, Kin G. Olivares, Boris N. Oreshkin, Federico Garza Ramirez, Max Mergenthaler Canseco, and Artur Dubrawski. Nhits: Neural hierarchical interpolation for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 6989–6997, 2023. doi: 10.1609/aaai.v37i6.25854. URL https://ojs.aaai.org/index.php/AAAI/article/view/25854.

Hui Chen, Viet Luong, Lopamudra Mukherjee, and Vikas Singh. Simpletm: A simple baseline for multivariate time series forecasting. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/27c546ab1e4f1d7d638e6a8dfbad9a07-Abstract-Conference.html.

Abdul Monaf Chowdhury, Rabeya Akter, and Safaeid Hossain Arib. T3time: Tri-modal time series forecasting via adaptive multi-head alignment and residual fusion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 20597–20605, 2026. doi: 10.1609/aaai.v40i25.39196.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167. PMLR, 2024. URL https://proceedings.mlr.press/v235/das24c.html.

Kuiye Ding, Fanda Fan, Yao Wang, Ruijie Jian, Xiaorui Wang, Luqi Gong, Yishan Jiang, Chunjie Luo, and Jianfeng Zhan. Dualsg: A dual-stream explicit semantic-guided multivariate time series forecasting framework. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 508–517, 2025. doi: 10.1145/3746027.3755458.

Kuiye Ding, Fanda Fan, Chunyi Hou, Zheya Wang, Lei Wang, Zhengxin Yang, and Jianfeng Zhan. Timemosaic: Temporal heterogeneity guided time series forecasting via adaptive granularity patch and segment-wise decoding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 20790–20798, 2026. doi: 10.1609/aaai.v40i25.39218. URL https://ojs.aaai.org/index.php/AAAI/article/view/39218.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G. Wilson. Large language models are zero-shot time series forecasters. In *Advances in Neural Information Processing Systems*, 2023.

Sungwon Han, Seungeon Lee, Meeyoung Cha, Sercan O Arik, and Jinsung Yoon. Retrieval augmented time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21774–21797. PMLR, 2025. URL https://proceedings.mlr.press/v267/han25d.html.

Yuxiao Hu, Qian Li, Dongxiao Zhang, Jinyue Yan, and Yuntian Chen. Context-alignment: Activating and enhancing llm capabilities in time series. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/e1de63ec74f40d3234c4e053f3528e18-Paper-Conference.pdf.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. Time-llm: Time series forecasting by reprogramming large language models. In *International Conference on Learning Representations*, 2024.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long- and short-term temporal patterns with deep neural networks. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 95–104, 2018.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, et al. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023. doi: 10.1126/science.adi2336.

Shengsheng Lin, Haojun Chen, Haijie Wu, Chunyun Qiu, and Weiwei Lin. Temporal query network for efficient multivariate time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 37797–37814. PMLR, 2025. URL https://proceedings.mlr.press/v267/lin25e.html.

Chenxi Liu, Qianxiong Xu, Hao Miao, Sun Yang, Lingzheng Zhang, Cheng Long, Ziyue Li, and Rui Zhao. Timecma: Towards llm-empowered multivariate time series forecasting via cross-modality alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 18780–18788, 2025. doi: 10.1609/aaai.v39i18.34067. URL https://ojs.aaai.org/index.php/AAAI/article/view/34067.

Xu Liu, Junfeng Hu, Yuan Li, Shizhe Diao, Yuxuan Liang, Bryan Hooi, and Roger Zimmermann. Unitime: A language-empowered unified model for cross-domain time series forecasting. In *Proceedings of the ACM Web Conference*, pp. 4095–4106, 2024a.

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*, 2024b.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. Technical report, OpenAI, 2019.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191, 2020.

Zihan Wang, Fanheng Kong, Shi Feng, Ming Wang, Xiaocui Yang, Han Zhao, Daling Wang, and Yifei Zhang. Is mamba effective for time series forecasting? *Neurocomputing*, 619:129178, 2025. doi: 10.1016/j.neucom.2024.129178. URL https://www.sciencedirect.com/science/article/pii/S0925231224019490.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems*, 2021.

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*, 2023.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pp. 1907–1913, 2019.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 753–763, 2020.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317. URL https://ojs.aaai.org/index.php/AAAI/article/view/26317.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021. doi: 10.1609/aaai.v35i12.17325. URL https://ojs.aaai.org/index.php/AAAI/article/view/17325.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286. PMLR, 2022. URL https://proceedings.mlr.press/v162/zhou22g.html.

Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. One fits all: Power general time series analysis by pretrained lm. In *Advances in Neural Information Processing Systems*, 2023.

Yiqing Zou, Hanning Yuan, Qianyu Yang, Ziqiang Yuan, Shuliang Wang, and Sijie Ruan. Meta dynamic graph for traffic flow prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 16584–16592, 2026. doi: 10.1609/aaai.v40i19.38699.

APPENDIX

A EXPERIMENTAL SETUP

A.1 DATASET DETAILS

Datasets. We evaluate TimeHSG on eight real-world multivariate long-term time-series forecasting datasets. These datasets cover diverse domains, including electricity transformer temperature, meteorology, transportation, renewable energy, and public health. Their different sampling intervals, variable dimensions, and temporal patterns provide a broad testbed for evaluating both local temporal modeling and cross-variable dependency learning. The details are as follows:

- **Electricity Transformer Temperature (ETT).** The ETT benchmark contains four subsets collected from two electricity transformers. ETTh1 and ETTh2 are recorded hourly, while ETTm1 and ETTm2 are recorded every 15 minutes. Each subset contains seven variables related to power load and oil temperature.
- **Weather.** Weather contains 21 meteorological indicators collected every 10 minutes at the Weather Station of the Max Planck Biogeochemistry Institute in Germany. It includes variables such as temperature, humidity, pressure, and wind-related measurements.
- **Traffic.** Traffic records hourly road occupancy rates from 862 sensors on freeways in the San Francisco Bay Area. This dataset is high-dimensional and contains strong spatiotemporal dependencies among sensors.
- **Solar-Energy.** Solar-Energy contains solar power production records from 137 photovoltaic plants sampled every 10 minutes. The dataset exhibits strong periodicity and clear daily patterns, making it useful for evaluating long-range seasonal modeling.
- **Influenza-Like Illness (ILI).** ILI contains weekly influenza-like illness statistics from public health surveillance records. Compared with the other datasets, ILI has a much lower sampling frequency and shorter input length, testing the model's robustness under sparse temporal observations.

Table A1: Dataset statistics. Dim denotes the number of variables in each dataset. Dataset size denotes the number of time points in the training, validation, and test splits, respectively. Frequency denotes the sampling interval between time points.

Dataset	Dim	Dataset Size	Frequency
ETTh1, ETTh2	7	(8545, 2881, 2881)	Hourly
ETTm1, ETTm2	7	(34465, 11521, 11521)	15 min
Weather	21	(36792, 5271, 10540)	10 min
Traffic	862	(12185, 1757, 3509)	Hourly
Solar-Energy	137	(36601, 5161, 10417)	10 min
ILI	7	(617, 74, 170)	Weekly

These datasets emphasize different forecasting challenges. The ETT subsets contain relatively low-dimensional industrial signals whose load and temperature variables exhibit both local fluctuations and long-range trends. Weather and Solar-Energy contain strong periodic components, making them useful for evaluating whether the frequency residual branch can complement the patch-based temporal encoder. Traffic is much higher-dimensional and stresses cross-variable dependency modeling, while ILI has sparse weekly observations and a shorter input length, testing robustness under limited temporal context. This diversity helps assess whether TimeHSG can provide general embedding guidance rather than overfit to a single domain or sampling frequency.

A.2 FORECASTING PROTOCOL

All datasets except ILI use an input length of 96 and $H \in \{96, 192, 336, 720\}$; ILI uses an input length of 36 and $H \in \{24, 36, 48, 60\}$. Each horizon is trained and evaluated independently, and each dataset average is over its four horizons. Baseline results are drawn from published studies (Chowdhury et al., 2026; Cao et al., 2026; Ding et al., 2026) when available; otherwise, we train and evaluate the corresponding models under the same experimental configuration used in this study.

A.3 Evaluation Metrics

We report mean squared error (MSE) and mean absolute error (MAE). Given ground-truth values y_i, predictions $\hat{y}_i$, and N forecasted points, the metrics are

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2, \tag{10}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|. \tag{11}$$

Lower values indicate better forecasting performance.

B Implementation Details

B.1 Fixed Textual Template and Frozen LLM Embeddings

For each variable in each sample, TimeHSG converts the historical window into a compact, deterministic textual template. The template contains the start and end times, the historical values in the input window, the sampling frequency, and the total trend value, following the form "From t_1 to t_2, the values were $x_1, \ldots, x_L$ every f. The total trend value was T." In our implementation, the frozen LLM is instantiated using GPT-2 (Radford et al., 2019), whose parameters remain fixed throughout training and evaluation. The frozen LLM encoder outputs an embedding that subsequently guides Semantic-Conditioned Adaptive Modulation (SCAM) and the Dynamic Semantic Graph (DSG). Because the LLM remains frozen, training updates only the forecasting network and the lightweight SCAM and DSG modules.

B.2 Model Configuration

Unless otherwise specified, the LLM embedding dimension is 768, the patch length is 8, the patch stride is 4, the Transformer encoder has one layer, and the number of attention heads is 8. The hidden channel size is tuned for each dataset, as summarized in Table A2. The patch branch captures local temporal variations, the frequency branch learns global spectral residuals through real-valued fast Fourier transform (FFT) filtering, and the LLM branch supplies variable-level embeddings. SCAM applies feature-level scale-and-shift modulation, while DSG constructs embedding-guided variable interactions through a dynamic adjacency matrix.

B.3 Training Details

TimeHSG is trained with the AdamW optimizer. The learning rate is set to 0.0001. Table A2 summarizes the dataset-specific hyperparameter configurations used in the experiments, including model depth, hidden dimension, dropout rate, optimizer settings, batch size, and training epochs.

Table A2: Experimental hyperparameter configurations for the forecasting horizon (H=96).

Dataset	Model Hyperparameter							Training Process				
	Encoder Layer	Patch Len	Stride	Graph Dim	Channel Dim	Heads	Dropout	Learning Rate	Weight Decay	Loss	Batch Size	Epochs
ETTm1	1	8	4	64	128	8	0.60	1e-4	1e-3	MSE	64	150
ETTm2	1	8	4	64	64	8	0.60	1e-4	1e-3	MSE	16	150
ETTh1	1	8	4	64	256	8	0.40	1e-4	1e-3	MSE	256	150
ETTh2	1	8	4	64	64	8	0.25	1e-4	1e-3	MSE	256	150
Weather	2	8	4	64	64	8	0.10	1e-4	1e-3	MSE	32	200
Solar	1	8	4	64	256	8	0.40	1e-4	1e-3	MSE	16	150
Traffic	1	8	4	64	256	8	0.40	1e-4	1e-3	MSE	8	150
ILI	1	8	4	64	32	8	0.10	1e-4	1e-3	MSE	16	100

B.4 Complexity Measurement

For the efficiency analysis, we measure trainable parameters, total parameters, inference time, GPU memory footprint, and peak memory usage using the same model invocation as in testing. Inference time is averaged after warm-up runs, and memory statistics are collected from CUDA peak-memory counters.

B.5 Reproducibility Notes

For a fair comparison, each horizon is trained independently and evaluated using the same train/validation/test split protocol adopted in the long-term forecasting literature. The LLM parameters are frozen throughout training, and only the forecasting backbone, SCAM, DSG, frequency residual branch, and prediction head are optimized. Unless otherwise specified, the random seed is fixed to 2024. To assess seed stability, we additionally repeat the experiments with three different seeds; Table A6 reports the resulting dataset-level averages. Because TimeHSG uses a compact variable-level template containing the historical values, sampling frequency, and total trend value, the LLM branch preserves direct temporal information while avoiding LLM fine-tuning.

C Detailed Comparison Results

Table A3 provides the horizon-level comparison with LLM-based forecasting models. TimeHSG obtains the largest number of first-place results in this group, especially for MAE, indicating that its predictions remain close to the ground truth across horizons. The gains are particularly clear on ETTm1, ETTm2, ETTh2, and Weather, where SCAM modulation and frequency-domain residual modeling jointly help capture local variation and longer-range periodic structure. Compared with methods that rely on stronger cross-modal coupling, TimeHSG keeps the LLM frozen and uses its outputs as lightweight guidance, while remaining competitive across most horizons.

The detailed rows also show where TimeHSG is not dominant. The relative gaps on Solar and Traffic indicate that variable-level guidance from a frozen LLM is best viewed as general complementary information rather than a substitute for strong task-specific structural priors. On ILI, TimeHSG obtains the best MAE on average but does not always achieve the best MSE, which is reasonable because sparse weekly health signals can contain abrupt changes that incur larger squared errors. These observations suggest that TimeHSG provides a general-purpose semantic guidance framework that remains competitive across horizons and can be complemented by domain knowledge.

Table A4 compares TimeHSG with recent non-LLM forecasting models. TimeHSG achieves the highest first-place count for both MSE and MAE, indicating that LLM-derived embeddings can improve a compact numerical forecasting architecture without requiring end-to-end LLM tuning. The improvements on ETT, Weather, and ILI indicate that embedding guidance is useful not only for high-frequency industrial and meteorological signals but also for low-frequency sparse observations.

The comparison also clarifies the complementary roles of different inductive biases. TimeHSG's frequency branch provides general long-range spectral compensation, while DSG constructs dynamic interaction priors from variable-level semantics. These mechanisms strengthen the general forecasting representation but do not aim to encode explicit daily seasonal rules or physical road topology. The Solar and Traffic results therefore delimit the intended role of TimeHSG: its semantic and frequency guidance provides a general foundation that can be further combined with domain-specific structural priors.

D Detailed Ablation Results

In the following table, the multilayer perceptron (MLP) variant replaces the frozen LLM encoder with a learned numerical embedding, while RE denotes a random embedding. CA+F denotes cross-attention with the frequency residual branch retained.

Table A5 gives the detailed ablation results. The full model achieves the highest first-place count on Weather, ETTm1, and ETTm2, indicating that LLM-derived embeddings usually provide more informative guidance than a generic learned projection. Removing SCAM weakens feature-level

Table A3: Full horizon-level comparison with LLM-based time-series forecasting models. The table follows the same formatting convention as the main comparison tables. The input length is 96 for all datasets except ILI, whose input length is 36. Bold indicates the best result and underline indicates the second best result.

Dataset	Horizon	TimeHSG (Ours)		T3Time (2026)		TimeCMA (2025)		Time-LLM (2024)		UniTime (2024)		GPT4TS/OFA (2023)	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	0.310	0.352	**0.309**	0.354	0.312	**0.351**	0.359	0.381	0.322	0.363	0.335	0.369
	192	**0.355**	**0.378**	0.357	0.382	0.361	**0.378**	0.383	0.393	0.366	0.387	0.374	0.385
	336	**0.382**	**0.400**	0.384	0.401	0.392	0.401	0.416	0.414	0.398	0.407	0.407	0.406
	720	**0.442**	0.435	**0.442**	**0.434**	0.453	0.438	0.483	0.449	0.454	0.440	0.469	0.442
	Avg	**0.372**	**0.391**	0.373	0.393	0.380	0.392	0.410	0.409	0.385	0.399	0.396	0.401
ETTm2	96	**0.169**	**0.252**	0.173	0.253	0.173	0.258	0.193	0.280	0.183	0.266	0.190	0.275
	192	**0.235**	**0.296**	0.244	0.303	0.238	0.301	0.257	0.318	0.251	0.310	0.253	0.313
	336	**0.293**	**0.333**	0.308	0.341	0.297	0.338	0.317	0.353	0.318	0.350	0.321	0.360
	720	**0.390**	**0.390**	0.401	0.400	0.393	0.394	0.419	0.411	0.420	0.410	0.411	0.406
	Avg	**0.271**	**0.317**	0.281	0.324	0.275	0.323	0.296	0.340	0.293	0.334	0.294	0.339
ETTh1	96	**0.371**	**0.388**	**0.371**	0.398	0.373	0.391	0.398	0.410	0.397	0.418	0.398	0.424
	192	0.420	**0.420**	**0.411**	0.422	0.427	0.421	0.451	0.440	0.434	0.439	0.449	0.427
	336	0.465	**0.437**	**0.449**	0.441	0.458	0.448	0.473	0.451	0.468	0.457	0.492	0.466
	720	0.457	0.466	**0.443**	0.462	0.449	**0.460**	0.469	0.470	0.469	0.477	0.487	0.483
	Avg	0.428	**0.427**	**0.418**	0.430	0.423	0.431	0.448	0.443	0.442	0.448	0.457	0.450
ETTh2	96	0.277	0.338	**0.274**	0.338	0.286	**0.336**	0.295	0.345	0.296	0.345	0.312	0.360
	192	**0.347**	**0.379**	0.366	0.398	0.363	0.387	0.386	0.399	0.374	0.394	0.387	0.405
	336	**0.353**	**0.392**	0.361	0.399	0.406	0.421	0.419	0.429	0.415	0.427	0.424	0.437
	720	**0.402**	**0.430**	0.404	0.433	0.417	0.438	0.425	0.442	0.425	0.444	0.433	0.453
	Avg	**0.344**	**0.384**	0.351	0.392	0.372	0.397	0.381	0.404	0.378	0.403	0.389	0.414
Solar	96	0.230	0.279	0.254	0.288	0.367	0.378	0.266	0.296	**0.218**	**0.269**	0.221	0.271
	192	0.259	0.299	0.309	0.323	0.440	0.428	0.311	0.296	0.250	0.294	**0.245**	**0.282**
	336	0.285	0.318	0.296	0.316	0.736	0.565	0.344	0.352	0.269	0.307	**0.261**	**0.291**
	720	0.266	0.303	0.302	0.324	0.693	0.510	0.332	0.334	0.269	0.304	**0.261**	**0.291**
	Avg	0.260	0.299	0.290	0.312	0.559	0.470	0.313	0.320	0.252	0.294	**0.247**	**0.284**
Weather	96	**0.161**	**0.207**	0.164	0.219	0.167	0.211	0.198	0.235	0.171	0.214	0.203	0.244
	192	**0.206**	**0.249**	0.211	0.254	0.212	0.253	0.240	0.269	0.217	0.254	0.247	0.277
	336	**0.261**	**0.289**	0.267	0.294	0.270	0.292	0.295	0.308	0.274	0.293	0.297	0.311
	720	**0.332**	**0.336**	0.335	0.349	0.350	0.348	0.368	0.353	0.351	0.343	0.368	0.356
	Avg	**0.240**	**0.270**	0.244	0.279	0.250	0.276	0.275	0.291	0.253	0.276	0.279	0.297
Traffic	96	0.478	0.308	0.508	0.332	0.449	0.311	0.528	0.353	**0.427**	**0.281**	0.469	0.311
	192	0.484	0.307	0.573	0.382	0.458	0.322	0.520	0.346	**0.436**	**0.284**	0.475	0.311
	336	0.506	0.310	0.546	0.359	0.477	0.331	0.533	0.355	**0.451**	**0.290**	0.488	0.316
	720	0.550	0.332	0.628	0.422	0.527	0.355	0.559	0.365	**0.485**	**0.309**	0.522	0.334
	Avg	0.504	0.314	0.564	0.374	0.478	0.330	0.535	0.355	**0.450**	**0.291**	0.489	0.318
ILI	24	2.021	0.880	**1.595**	**0.801**	1.996	0.998	2.383	1.004	2.346	0.954	2.732	1.100
	36	1.809	0.837	**1.609**	**0.822**	1.906	0.915	2.390	0.993	1.998	0.912	2.664	1.063
	48	**1.582**	**0.794**	1.915	0.873	1.867	0.868	2.394	1.003	1.979	0.912	2.617	1.041
	60	1.952	**0.879**	**1.915**	0.899	1.920	0.904	2.562	1.049	2.109	0.938	2.478	1.035
	Avg	1.841	**0.847**	**1.758**	0.848	1.922	0.921	2.432	1.012	2.108	0.929	2.623	1.060
1st Count	–	**20**	**24**	12	3	0	4	0	0	6	6	4	4

Table A4: Full horizon-level comparison with non-LLM time-series forecasting models. Each model reports MSE and MAE in adjacent columns. The input length is 96 for all datasets except ILI, whose input length is 36. Best results are bolded; second-best results are underlined. "iTrans." denotes iTransformer. The dashes for iTransformer on ILI at $H = 24$ indicate an unverifiable metric inconsistency in the cited results.

Data	H	TimeHSG (Ours)		GTR (2026)		FreDN (2026)		SimpleTM (2025)		TQNet (2025)		RAFT (2025)		S-Mamba (2025)		iTrans. (2024)		DLinear (2023)		PatchTST (2023)	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	**0.310**	**0.352**	0.322	0.367	0.318	0.354	0.323	0.364	0.311	0.353	0.325	0.369	0.333	0.368	0.334	0.368	0.345	0.372	0.344	0.373
	192	0.355	0.378	**0.354**	**0.377**	0.364	0.380	0.371	0.389	0.356	0.378	0.364	0.388	0.376	0.390	0.377	0.391	0.380	0.389	0.367	0.386
	336	0.382	0.400	**0.379**	**0.397**	0.399	0.400	0.411	0.413	0.390	0.401	0.391	0.407	0.408	0.413	0.426	0.420	0.413	0.413	0.392	0.407
	720	0.442	0.435	**0.437**	**0.434**	0.448	0.435	0.460	0.445	0.452	0.440	0.444	0.438	0.475	0.448	0.491	0.459	0.474	0.453	0.464	0.442
	Avg	**0.372**	**0.391**	0.373	0.394	0.382	0.392	0.391	0.403	0.377	0.393	0.381	0.400	0.398	0.405	0.407	0.410	0.403	0.407	0.392	0.402
ETTm2	96	**0.169**	0.252	0.171	**0.251**	0.173	**0.251**	0.178	0.256	0.173	0.256	0.176	0.264	0.179	0.263	0.180	0.264	0.193	0.292	0.177	0.260
	192	0.235	0.296	0.237	0.294	**0.234**	**0.292**	0.242	0.300	0.238	0.298	0.241	0.306	0.250	0.309	0.250	0.309	0.284	0.362	0.246	0.305
	336	0.293	0.333	**0.291**	**0.332**	0.299	0.334	0.309	0.344	0.301	0.340	0.302	0.345	0.312	0.349	0.311	0.348	0.369	0.427	0.305	0.343
	720	**0.390**	**0.390**	**0.390**	**0.390**	0.397	0.391	0.401	0.399	0.397	0.396	0.404	0.405	0.411	0.406	0.412	0.407	0.554	0.522	0.410	0.405
	Avg	**0.271**	**0.317**	0.272	**0.317**	0.276	**0.317**	0.283	0.325	0.277	0.323	0.281	0.330	0.288	0.332	0.288	0.332	0.350	0.401	0.285	0.328
ETTh1	96	0.371	**0.388**	**0.370**	0.393	0.377	**0.388**	0.379	0.396	0.371	0.393	0.376	0.401	0.386	0.405	0.386	0.405	0.386	0.400	0.404	0.413
	192	0.420	**0.420**	0.424	0.423	0.442	0.432	0.432	0.425	0.428	0.426	**0.417**	0.424	0.443	0.437	0.441	0.436	0.437	0.432	0.454	0.430
	336	0.465	**0.437**	0.479	0.448	0.464	0.440	0.476	0.448	0.476	0.446	**0.454**	0.441	0.489	0.468	0.487	0.458	0.481	0.459	0.497	0.462
	720	**0.457**	0.466	0.525	0.492	0.464	**0.460**	0.481	0.473	0.487	0.470	0.465	0.465	0.502	0.489	0.503	0.491	0.519	0.516	0.496	0.481
	Avg	**0.428**	**0.427**	0.450	0.439	0.437	0.430	0.442	0.436	0.441	0.434	**0.428**	0.433	0.455	0.450	0.454	0.447	0.456	0.452	0.463	0.449
ETTh2	96	**0.277**	0.338	0.297	0.347	0.282	**0.330**	0.308	0.356	0.295	0.343	0.289	0.342	0.296	0.348	0.297	0.349	0.333	0.387	0.312	0.358
	192	**0.347**	0.379	0.373	0.392	0.353	**0.378**	0.383	0.404	0.367	0.393	0.382	0.400	0.376	0.396	0.380	0.400	0.477	0.476	0.397	0.408
	336	**0.353**	**0.392**	0.425	0.432	0.398	0.414	0.422	0.436	0.417	0.427	0.424	0.439	0.424	0.431	0.428	0.432	0.598	0.541	0.435	0.440
	720	**0.402**	**0.430**	0.424	0.442	0.425	0.440	0.442	0.454	0.433	0.446	0.433	0.457	0.426	0.444	0.427	0.445	0.831	0.657	0.436	0.449
	Avg	**0.344**	**0.384**	0.380	0.403	0.365	0.390	0.424	0.413	0.378	0.402	0.382	0.410	0.381	0.405	0.383	0.407	0.559	0.515	0.395	0.414
Solar	96	0.230	0.279	0.181	0.237	0.212	0.236	0.252	0.299	**0.173**	**0.233**	0.278	0.278	0.205	0.244	0.221	0.250	0.290	0.378	0.223	0.284
	192	0.259	0.299	**0.196**	**0.245**	0.239	0.259	0.306	0.332	0.199	0.257	0.297	0.297	0.237	0.270	0.249	0.283	0.320	0.398	0.251	0.293
	336	0.285	0.318	**0.210**	**0.260**	0.256	0.276	0.372	0.360	0.211	0.263	0.311	0.311	0.258	0.288	0.255	0.276	0.352	0.415	0.263	0.301
	720	0.266	0.303	0.235	0.279	0.256	0.275	0.331	0.339	**0.209**	**0.270**	0.318	0.325	0.260	0.288	0.257	0.279	0.357	0.413	0.263	0.292
	Avg	0.260	0.299	0.206	**0.255**	0.241	0.262	0.315	0.333	**0.198**	0.256	0.301	0.303	0.240	0.273	0.246	0.272	0.330	0.401	0.250	0.293
Weather	96	0.161	0.207	**0.157**	**0.202**	0.163	0.205	0.175	0.219	0.160	0.203	0.185	0.241	0.165	0.210	0.174	0.214	0.196	0.255	0.177	0.218
	192	0.206	0.249	**0.205**	0.246	0.208	0.246	0.217	0.257	0.206	**0.245**	0.237	0.287	0.214	0.252	0.221	0.254	0.237	0.296	0.222	0.259
	336	**0.261**	0.289	0.262	0.289	0.262	**0.287**	0.282	0.302	0.264	0.289	0.290	0.327	0.274	0.297	0.278	0.296	0.283	0.335	0.277	0.297
	720	**0.332**	**0.336**	0.342	0.343	0.342	0.341	0.357	0.351	0.344	0.342	0.367	0.380	0.350	0.345	0.358	0.349	0.345	0.381	0.352	0.347
	Avg	**0.240**	**0.270**	0.242	**0.270**	0.244	**0.270**	0.258	0.282	0.244	**0.270**	0.270	0.309	0.251	0.276	0.258	0.278	0.265	0.317	0.257	0.280
Traffic	96	0.478	0.308	0.438	0.265	0.455	0.280	0.531	0.337	0.413	**0.261**	0.388	0.265	**0.382**	**0.261**	0.393	0.269	0.652	0.400	0.459	0.298
	192	0.484	0.307	0.450	0.280	0.459	0.284	0.530	0.342	0.432	0.271	0.400	0.278	**0.396**	**0.267**	0.411	0.277	0.601	0.375	0.469	0.301
	336	0.506	0.310	0.466	0.283	0.470	0.286	0.542	0.347	0.450	0.277	**0.411**	0.287	0.417	**0.276**	0.425	0.283	0.608	0.377	0.483	0.307
	720	0.550	0.332	0.504	0.301	0.500	0.304	0.576	0.360	0.486	**0.295**	**0.456**	0.305	0.460	0.300	0.461	0.302	0.648	0.398	0.517	0.326
	Avg	0.504	0.314	0.465	0.282	0.471	0.289	0.545	0.347	0.445	**0.276**	**0.414**	0.284	**0.414**	**0.276**	0.423	0.283	0.627	0.388	0.482	0.308
ILI	24	2.021	0.880	1.906	0.828	2.465	0.947	**1.853**	0.842	1.910	**0.827**	5.056	1.482	2.899	1.134	–	–	2.398	1.040	2.335	0.989
	36	1.809	0.837	2.030	0.854	2.167	0.902	**1.732**	**0.830**	1.939	0.844	5.155	1.560	2.767	1.111	2.468	0.998	2.646	1.088	2.561	1.035
	48	**1.582**	**0.794**	2.044	0.872	2.105	0.898	1.831	0.862	1.924	0.851	5.461	1.609	3.140	1.231	2.489	1.016	2.614	1.086	2.465	1.022
	60	**1.952**	**0.879**	2.007	0.902	2.280	0.947	1.959	0.894	2.124	0.936	4.684	1.454	3.188	1.247	2.471	1.065	2.804	1.146	2.189	0.997
	Avg	**1.841**	**0.847**	1.997	0.864	2.254	0.924	1.844	0.857	1.974	0.865	5.089	1.526	2.999	1.181	2.444	1.203	2.616	1.090	2.388	1.011
1st Count	–	**18**	**16**	10	12	1	9	2	1	3	8	6	0	3	4	0	0	0	0	0	0

calibration, so the temporal encoder receives less variable-specific guidance. Removing both SCAM and DSG further reduces the model's ability to use the embeddings for variable interaction, which is especially important for multivariate datasets in which different channels may share related but non-identical temporal patterns.

The variant without SCAM, DSG, and the frequency branch corresponds to a much weaker embedding-free and frequency-free backbone. Its degradation shows that TimeHSG does not rely on a single component alone: SCAM adjusts local patch representations, DSG guides cross-variable message passing, and the frequency branch supplies global periodic residuals. The stepwise results further show that frozen LLM embeddings are most effective when used through explicit hierarchical constraints. SCAM transforms each variable-level embedding into scale and shift factors, allowing the model to calibrate local patch features according to trend, volatility, and distributional information. DSG then reuses the same embeddings at the topology level, where the resulting adjacency determines dynamic cross-variable interactions. Therefore, the embeddings are not simply appended to the temporal representation; they are assigned two distinct roles: feature-level modulation and graph-level dependency guidance.

The CA+F variant provides a more direct test of this design choice than a simple removal study. In this variant, SCAM and DSG are replaced by a standard cross-attention module, while the LLM embeddings and the frequency residual branch are retained. Thus, if the improvement arose only from adding LLM embeddings or introducing an additional attention operation, CA+F should achieve comparable performance. However, CA+F is consistently weaker than the full model and is especially unstable on ETTm1 and ETTm2. This result indicates that ordinary attention-based fusion cannot fully exploit LLM embedding guidance in multivariate forecasting.

Standard cross-attention performs dense token-level mixing between LLM and temporal features but does not distinguish what the embeddings should control. It may attend to useful cues, but it may also overmix unrelated LLM and numerical patterns or collapse variable-specific guidance into a generic fusion representation. By contrast, TimeHSG restricts the effect of LLM embeddings to two interpretable pathways: SCAM controls how the temporal features of each variable are locally recalibrated, and DSG controls how variables exchange information through its dynamic graph. This separation makes the embedding guidance more stable, reduces the risk of spurious feature fusion, and explains why the proposed hierarchical design is more effective than ordinary attention.

E Hyperparameter Sensitivity

Table A7 evaluates the sensitivity of TimeHSG to the patch configuration and DSG graph dimension on four ETT datasets. Increasing the patch length and stride to 16/8 consistently degrades performance, indicating that overly coarse patches discard useful local dynamics. The smaller 4/2 configuration is competitive on ETTh1 and ETTm1 but does not improve performance on all datasets; the default 8/4 setting matches the best result on ETTh2 and performs best on ETTm2. For the graph dimension, the default dimension of 64 achieves results that are best or tied for best on ETTm1 and ETTm2 while remaining competitive on ETTh1 and ETTh2. These results show that the default configuration provides a stable trade-off across datasets, so we use a patch length/stride of 8/4 and a graph dimension of 64 throughout the main experiments.

F Enlarged Visualizations

The following figures provide enlarged module-level visualizations on ETTm2 for the 336-step forecasting horizon. Each figure occupies the full two-column width to make the local patterns, semantic guidance signals, and forecasting curves easier to inspect.

The visualizations are intended to complement the quantitative results by showing how semantic guidance changes intermediate representations. Figure A1 focuses on feature-level modulation, Figure A2 shows topology-level semantic interaction, Figure A3 illustrates the spectral residual branch, and Figure A4 shows how the main and residual predictions are combined.

Table A5: Full horizon-level ablation results of TimeHSG. Bold indicates the best result and underline indicates the second best result.

Dataset	Horizon	w/o S		w/o S+D		w/o S+D+F		CA+F		MLP		RE		TimeHSG	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	96	0.167	0.209	0.167	0.209	0.175	0.215	0.164	0.208	0.167	0.209	0.167	0.209	**0.161**	**0.207**
	192	0.214	0.251	0.216	0.253	0.223	0.257	0.212	0.252	0.216	0.253	0.216	0.252	**0.206**	**0.249**
	336	0.271	0.293	0.274	0.294	0.278	0.297	0.267	0.292	0.270	0.292	0.271	0.292	**0.261**	**0.289**
	720	0.347	0.345	0.350	0.345	0.352	0.346	0.365	0.355	0.351	0.345	0.350	0.345	**0.332**	**0.336**
	Avg	0.250	0.275	0.252	0.275	0.257	0.279	0.252	0.277	0.251	0.275	0.251	0.274	**0.240**	**0.270**
ETTm1	96	0.319	0.360	0.320	0.360	0.319	0.357	0.376	0.383	0.316	0.359	0.317	0.360	**0.310**	**0.352**
	192	0.358	0.381	0.358	0.382	0.358	0.381	1.340	0.785	0.358	0.380	0.359	0.381	**0.355**	**0.378**
	336	0.385	0.404	0.388	0.406	0.390	0.408	0.774	0.583	0.386	0.403	0.386	0.403	**0.382**	**0.400**
	720	**0.441**	**0.435**	0.448	0.439	0.450	0.440	0.916	0.648	0.446	0.438	0.447	0.438	0.442	**0.435**
	Avg	0.376	0.395	0.378	0.397	0.379	0.396	0.852	0.600	0.377	0.395	0.377	0.395	**0.372**	**0.391**
ETTm2	96	0.170	0.254	0.170	0.253	0.173	0.257	0.216	0.300	0.170	0.253	0.170	0.253	**0.169**	**0.252**
	192	0.236	0.297	**0.235**	0.297	0.238	0.300	0.267	0.326	**0.235**	0.297	**0.235**	0.297	**0.235**	**0.296**
	336	0.296	0.335	0.294	0.334	0.297	0.337	0.388	0.409	0.294	0.335	0.294	0.335	**0.293**	**0.333**
	720	0.397	0.395	0.394	0.393	0.397	0.396	0.406	0.407	0.395	0.393	0.392	0.395	**0.390**	**0.390**
	Avg	0.275	0.320	0.273	0.320	0.276	0.322	0.320	0.360	0.274	0.320	0.273	0.320	**0.271**	**0.317**
ILI	24	2.137	0.910	2.164	0.923	2.166	0.923	**1.978**	**0.877**	2.250	0.942	2.241	0.919	2.021	0.880
	36	1.982	0.882	1.975	0.878	1.976	0.878	1.821	0.844	1.967	0.876	1.959	0.842	**1.809**	**0.837**
	48	1.632	0.809	1.566	0.806	1.660	0.822	2.182	1.004	1.594	0.811	**1.482**	**0.770**	1.582	0.794
	60	2.107	0.923	2.112	0.918	2.113	0.918	**1.931**	**0.879**	2.088	0.915	2.020	**0.879**	1.952	**0.879**
	Avg	1.964	0.881	1.954	0.881	1.979	0.885	1.978	0.901	1.975	0.886	1.926	0.853	**1.841**	**0.847**
1st Count	–	1	1	1	0	0	0	2	2	1	0	2	2	**16**	**18**

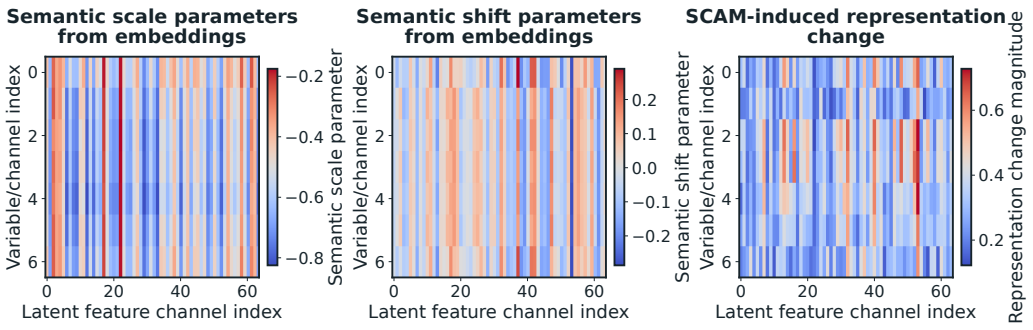


Figure A1: Enlarged visualization of SCAM semantic modulation. The left and middle heatmaps show semantic-conditioned scale and shift parameters generated from LLM embeddings, while the right heatmap shows the induced representation change after modulation. The non-uniform channel responses indicate that SCAM selectively amplifies or suppresses local temporal features instead of applying a global perturbation.

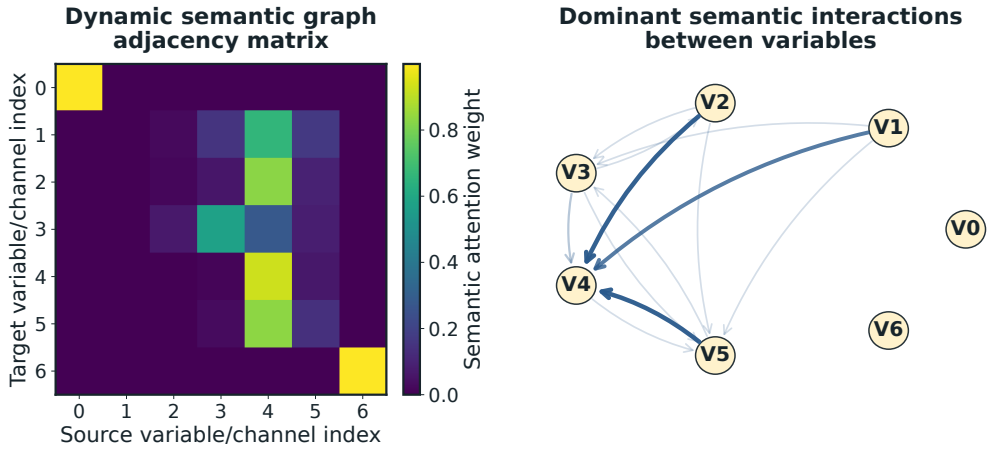


Figure A2: Enlarged visualization of the DSG dynamic semantic graph. The adjacency heatmap shows semantic-aware interaction strengths among variables, and the graph view highlights the dominant cross-variable dependencies. The sparse and structured connections suggest that DSG uses LLM-derived statistical semantics to guide variable interaction rather than relying only on dense numerical co-movement.

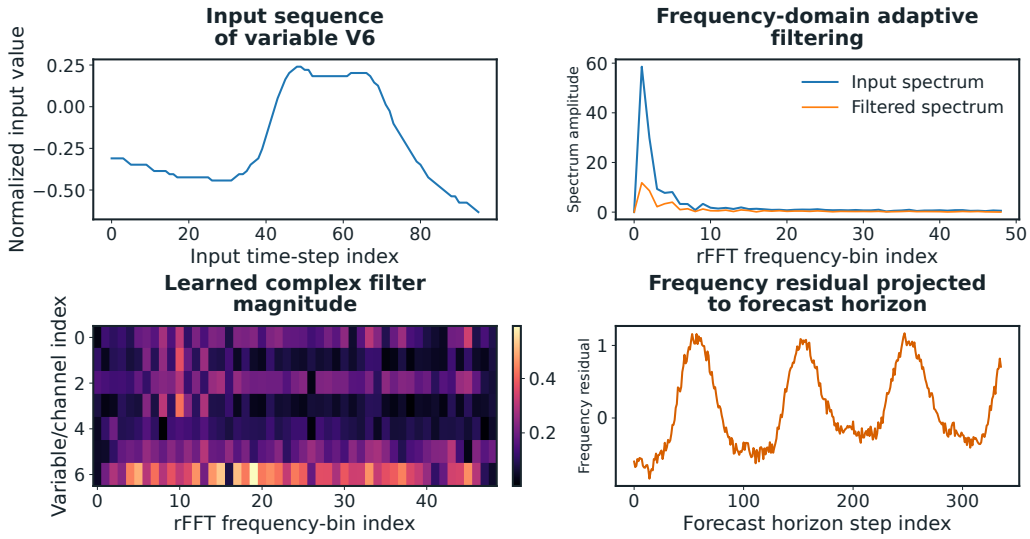


Figure A3: Enlarged visualization of the global frequency residual branch. The figure shows the input sequence, the learned frequency-domain filtering behavior, the complex filter magnitude, and the resulting frequency residual prediction. The branch captures smooth periodic components and low-frequency trends that complement the patch-based time-domain encoder.

Table A6: Random-seed stability of TimeHSG. Results for each seed are averaged over the four forecasting horizons of each dataset. Mean ± Std. uses the sample standard deviation over the three independent seeds.

Dataset	MSE				MAE			
	Seed 1	Seed 2	Seed 3	Mean ± Std.	Seed 1	Seed 2	Seed 3	Mean ± Std.
ETTm1	0.372	0.373	0.373	0.373 ± 0.001	0.391	0.392	0.392	0.392 ± 0.001
ETTm2	0.271	0.273	0.273	0.272 ± 0.001	0.317	0.319	0.319	0.318 ± 0.001
ETTh1	0.428	0.429	0.429	0.429 ± 0.001	0.427	0.428	0.428	0.428 ± 0.001
ETTh2	0.344	0.343	0.343	0.343 ± 0.001	0.384	0.384	0.384	0.384 ± 0.000
Solar	0.260	0.261	0.259	0.260 ± 0.001	0.299	0.300	0.298	0.299 ± 0.001
Weather	0.240	0.242	0.242	0.241 ± 0.001	0.270	0.272	0.271	0.271 ± 0.001
Traffic	0.504	0.505	0.505	0.505 ± 0.001	0.314	0.316	0.315	0.315 ± 0.001
ILI	1.841	1.852	1.832	1.842 ± 0.010	0.847	0.847	0.842	0.845 ± 0.003

Table A7: Hyperparameter sensitivity of TimeHSG. We vary the patch configuration and DSG graph dimension separately; the remaining settings use the default configuration. The default patch length/stride and graph dimension are 8/4 and 64, respectively. Bold indicates the best result.

Dataset	Metric	Patch length / stride			DSG graph dimension		
		16/8	4/2	8/4	32	64	128
ETTh1	MSE	0.430	**0.428**	**0.428**	**0.428**	**0.428**	**0.428**
	MAE	0.429	0.430	**0.427**	0.428	**0.427**	0.428
ETTh2	MSE	0.351	**0.344**	**0.344**	**0.344**	**0.344**	**0.344**
	MAE	0.387	**0.384**	**0.384**	0.385	**0.384**	0.385
ETTm1	MSE	0.375	**0.372**	**0.372**	0.374	**0.372**	0.373
	MAE	0.393	**0.390**	0.391	0.392	**0.391**	0.392
ETTm2	MSE	0.273	0.275	**0.271**	0.273	**0.271**	0.274
	MAE	0.319	0.320	**0.317**	0.319	**0.317**	0.320

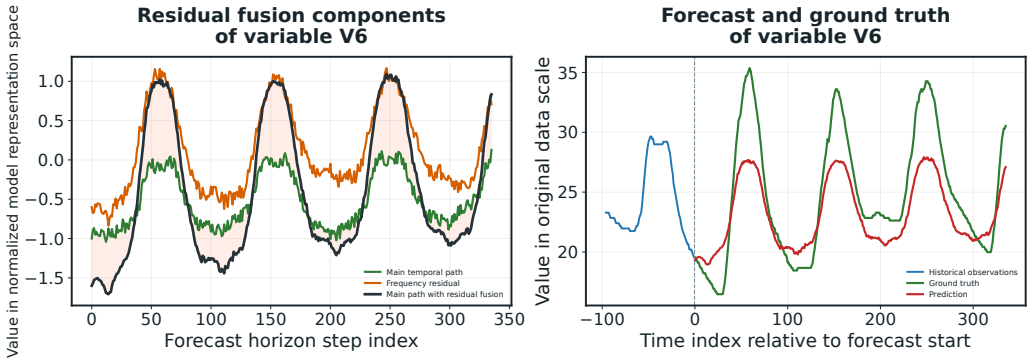


Figure A4: Enlarged visualization of residual fusion and forecasting. The left panel decomposes the main temporal path, frequency residual, and fused prediction, while the right panel compares the final prediction with the ground truth. The visualization shows how the frequency residual branch complements the local temporal path and helps the fused output better follow long-range periodic patterns.