SAGE: VARIATE-WISE SEMANTIC AUGMENTATION FOR VISION-LANGUAGE TIME SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Time series forecasting models mainly learn from numerical sequences, while variable meanings, statistical behavior, and temporal dynamics offer additional context. Existing language-assisted approaches often rely on generative models during forecasting or use dataset-level prompts that overlook differences among variates. We propose **SAGE** (**S**eeing and **A**ugmenting with **G**rounded **E**ncoding), a CLIP-based framework centered on variable-specific descriptor augmentation. The CLIP text encoder models temporal and cross-variable information, while gated residual paths inject cached descriptions and statistical descriptors. In the full configuration, frequency-enhanced patches capture temporal structure and a frozen CLIP vision encoder provides training-only contrastive supervision from rendered series. This design combines variable-level context with visual regularization without generative LLM calls or image encoding at forecast time. Across eight long-term benchmarks and M4, SAGE achieves competitive accuracy against the listed baselines. Component ablations show gains associated with variable-level augmentation and visual alignment.

1 INTRODUCTION

Time series forecasting supports energy, transportation, weather, and finance (Zhou et al., 2020; Wu et al., 2021). Transformer forecasters have advanced performance (Zhou et al., 2022; Liu et al., 2024b), while competitive linear baselines (Zeng et al., 2023) emphasize the importance of temporal inductive biases.

Two trends extend this development. **Large-scale pretraining** in TimesFM (Das et al., 2024), Chronos (Ansari et al., 2024), and Time-MoE (Shi et al., 2025) supports zero-shot and few-shot forecasting. **Cross-modal transfer** in GPT4TS (Zhou et al., 2023), Time-LLM (Jin et al., 2024), and TEST (Sun et al., 2024) uses pretrained language representations. However, replacing some language components with simple attention without degrading performance (Tan et al., 2024) questions whether gains reflect temporal knowledge or generic features.

Vision-language models (VLMs) offer representations of multiscale temporal patterns (Ni et al., 2025), building on the value of 2D time-series representations (Wu et al., 2022). Time-VLM (Zhong et al., 2025) combines retrieval, vision, and text; Aurora (Wu et al., 2025) develops multimodal foundation modeling; and OccamVTS (Lyu et al., 2026) reports comparable forecasting after distilling vision models to 1% of their parameters. CiK demonstrates settings where textual context is essential (Williams et al., 2024).

These developments motivate integrating variable-specific descriptions and statistics into a CLIP forecaster. CLIP (Radford et al., 2021) jointly pretrains text and vision encoders; TimesCLIP (Dong et al., 2025) already adapts CLIP's text encoder to numerical patch and variate tokens with visual alignment. SAGE focuses on gated augmentation within this family, using cached metadata and data-derived descriptors. Existing knowledge-injection schemes (Jin et al., 2024; Pan et al., 2024) provide further context. Figure 1 complements aggregate results with selected test windows.

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.

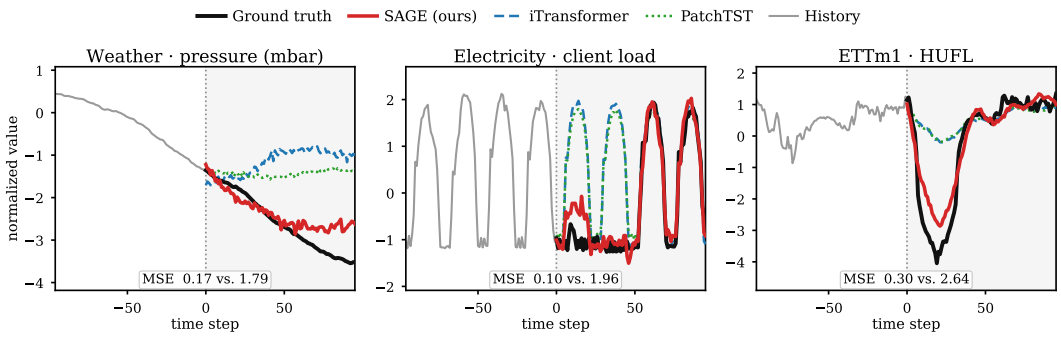


Figure 1: Representative 96-step forecasts on Weather, Electricity, and ETTm1. The selected windows illustrate how SAGE tracks temporal changes when combining numerical history with variable-level descriptors. Each inset reports window-level MSE for SAGE and the strongest displayed baseline.

Our main contribution is **template-based knowledge injection with variable-wise gating**: offline descriptions combine metadata and training-partition statistics, gated cross-attention controls each variable's textual input, and a separately gated bypass preserves numerical cues. Optional LLM assistance is confined to offline template preparation. **Frequency-enhanced encoding and cross-variate context** supply spectral, temporal, and relational representations; **training-only visual alignment** regularizes the full configuration ($N \leq 50$). Evaluation covers eight long-term benchmarks, M4, and text/vision component analyses.

2 Related Work

2.1 Multimodal and Knowledge-Enhanced Time Series Forecasting

GPT4TS (Zhou et al., 2023) fine-tunes only layer normalization and positional embeddings in frozen GPT-2; LLMTime (Gruver et al., 2023) encodes numerical series as strings for zero-shot forecasting. Time-LLM (Jin et al., 2024) reprograms patches into text prototypes with declarative prompts, while TEST (Sun et al., 2024) contrastively aligns time-series embeddings and interpretable text prototypes. Such language components may serve as generic feature extractors rather than sources of temporal knowledge (Tan et al., 2024).

Time-VLM (Zhong et al., 2025) combines retrieval-, vision-, and text-augmented learners with frozen VLMs. Aurora (Wu et al., 2025) uses modality-guided attention and flow matching for zero-shot multimodal forecasting; VLM4TS (He et al., 2026) combines visual screening and multimodal verification for zero-shot anomaly detection. TimesCLIP (Dong et al., 2025) is particularly close to SAGE's backbone, adapting CLIP's text encoder to numerical patch/variate tokens, aligning frozen visual features, and selecting variate representations through cross-attention. TimeCMA (Liu et al., 2024a) aligns time-series and LLM-derived prompt representations, caching prompt embeddings for efficient inference. SAGE emphasizes gated variable-specific fusion of cached descriptions and statistics. The present comparisons do not establish an accuracy or cost advantage over these related multimodal methods.

2.2 Pre-trained Models for Time Series

Supervised forecasters model temporal structure through decomposition and auto-correlation (Autoformer (Wu et al., 2021)), channel-independent patches (PatchTST (Nie et al., 2022)), variate tokens (iTransformer (Liu et al., 2024b)), or 2D transformations (TimesNet (Wu et al., 2022)). Recent baselines address frequency bias (FredFormer (Piao et al., 2024)), joint temporal/channel clustering (DUET (Qiu et al., 2025)), energy-scarce components (Amplifier (Fei et al., 2025)), and selective representation spaces (SRSNet (Wu et al., 2026)). Foundation models include decoder-only TimesFM (Das et al., 2024), trained on 100B real-world time points; Chronos (Ansari et al., 2024), which tokenizes values for T5; MOMENT (Goswami et al., 2024), pretrained by masked recon-

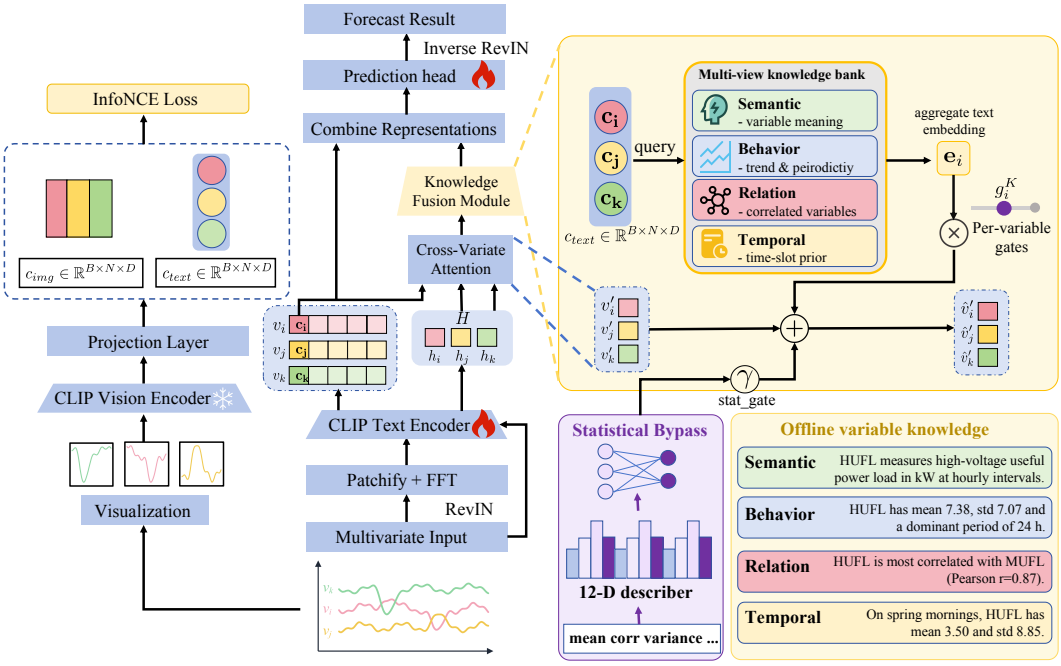


Figure 2: Overview of **SAGE**: shared CLIP text branches encode frequency-enhanced temporal tokens and cross-variable context; gated text attention and a statistical bypass augment them before decoding and inverse normalization. The frozen vision encoder supplies training-only contrastive supervision. The lower callout shows an ETTh1/HUFL knowledge example.

struction on the Time-series Pile; Moirai (Woo et al., 2024), with any-variate attention over 27B observations; and Time-MoE (Shi et al., 2025), a 2.4B-parameter sparse mixture-of-experts. These methods rely on large time-series corpora. OccamVTS (Lyu et al., 2026) studies transfer from pretrained visual representations without billion-scale time-series pretraining, motivating visual priors whose contribution requires controlled analysis.

2.3 Contrastive Learning for Time Series

Time-series contrastive methods align augmented views through temporal and contextual contrasting (TS-TCC (Eldele et al., 2021)), hierarchical instance/temporal objectives (TS2Vec (Yue et al., 2022)), seasonal/trend disentanglement via time- and frequency-domain losses (CoST (Woo et al., 2022)), or time-frequency consistency (TF-C (Zhang et al., 2022)). Soft-CL (Lee et al., 2024) softens assignments to reduce false negatives. Frequency-aware contrastive learning also appears in sequential recommendation (Wang & Zhang, 2026). Cross-modal alignment pairs numerical representations with rendered views to provide complementary supervision.

3 Methodology

Given $\mathbf{X} \in \mathbb{R}^{T \times N}$ with T steps and N variates, SAGE predicts $\hat{\mathbf{Y}} \in \mathbb{R}^{T_{\text{pred}} \times N}$ over T_{pred} steps. Figure 2 shows numerical patch and variate tokens sharing a CLIP text encoder, variable-specific descriptor fusion, and a forecast head with inverse normalization. For $N \leq 50$, frozen visual features provide training-only alignment alongside frequency-enhanced temporal and cross-variate encoding. For $N > 50$, the lightweight configuration retains cross-variate encoding and descriptor fusion while omitting patching and visual alignment. The following subsections detail these components.

3.1 Frequency-Enhanced Language Module

This module treats time series patches as tokens for the pretrained CLIP text Transformer.

We first apply reversible instance normalization to mitigate distribution shift. For variable i, RevIN stores its mean $\mu^{(i)}$ and standard deviation $\sigma^{(i)}$ for output de-normalization. We omit the batch index in per-variable equations for clarity. The normalized sequence $\mathbf{x}^{(i)} \in \mathbb{R}^T$ is divided into L_p overlapping patches of length P and stride S. Let $\mathbf{p}_j^{(i)} \in \mathbb{R}^P$ denote patch j. A learnable tokenizer $\mathbf{W}_{\text{tok}}$ maps it to the shared embedding width D:

$$\mathbf{z}_j^{(i)} = \mathbf{W}_{\text{tok}} \mathbf{p}_j^{(i)}.$$

Time-domain tokens alone may inadequately represent periodic patterns. We apply a Hann-windowed FFT to every patch and project its real and imaginary components into a frequency token $\mathbf{f}_j^{(i)} \in \mathbb{R}^D$. The time token is the query, and the frequency token provides the key and value. A learnable scalar α, initialized at zero, controls this fusion:

$$\mathbf{z}_j^{(i)} \leftarrow \mathbf{z}_j^{(i)} + \alpha \ \mathrm{CrossAttn}\big(\mathbf{z}_j^{(i)}, \mathbf{f}_j^{(i)}, \mathbf{f}_j^{(i)}\big),$$

where $\mathrm{CrossAttn}(Q, K, V)$ denotes cross-attention with query Q, key K, and value V.

A learnable [CLS] token is prepended, and learnable positional embeddings are added to the patch sequence. The sequence then enters the CLIP text Transformer without using its word embedding layer. We replace the original positional embeddings with embeddings matched to the patch sequence. The Transformer blocks and final LayerNorm are fine-tuned with an encoder-specific learning-rate multiplier, while unused projection layers remain frozen. The encoder outputs patch representations $\mathbf{R} \in \mathbb{R}^{B \times N \times (L_p+1) \times D}$ and temporal summaries $\mathbf{c}_{\text{text}} \in \mathbb{R}^{B \times N \times D}$ from the [CLS] positions.

3.2 Cross-Variate Context Modeling Module

The temporal encoder processes each variable independently, but multivariate forecasting also requires dependencies across variables.

Each full normalized sequence $\mathbf{x}^{(i)}$ is mapped to the shared embedding width by a linear tokenizer and augmented with a learnable variable identifier. The N resulting tokens pass through the same CLIP text encoder used above. Its output $\mathbf{H} \in \mathbb{R}^{B \times N \times D}$ contains contextual representations of all variables. Thus, the shared encoder processes patch sequences to capture temporal structure and variable sequences to capture cross-variable structure.

The temporal summaries query $\mathbf{H}$ through cross-attention. Layer normalization LN and a feed-forward network FFN form residual updates:

$$\mathbf{v}' = \mathrm{LN}\big(\mathbf{c}_{\text{text}} + \mathrm{CrossAttn}\big(\mathrm{LN}(\mathbf{c}_{\text{text}}), \mathrm{LN}(\mathbf{H}), \mathrm{LN}(\mathbf{H})\big)\big),$$
$$\mathbf{v}' = \mathrm{LN}(\mathbf{v}' + \mathrm{FFN}(\mathbf{v}')).$$

The result $\mathbf{v}' \in \mathbb{R}^{B \times N \times D}$ combines each variable's temporal summary with information selected from all variables.

3.3 Multi-View Textual Semantic Fusion Module

This module is the central focus of SAGE. It combines three information sources: dataset metadata describing variable meaning and context, numerical descriptors derived from the training series, and representations supplied by the pretrained encoder. The cached descriptions and the direct statistical bypass expose the first two sources to the forecasting model.

For each variable, we construct descriptions from domain metadata and training-split statistics. Table 1 summarizes the knowledge sources, while Figure 2 shows representative HUFL examples. The CLIP text encoder maps the descriptions to the shared embedding space. We also compute a 12-dimensional statistical vector $\mathbf{s}^{(i)} \in \mathbb{R}^{12}$ for variable i. All descriptions, embeddings, and statistical vectors are cached before model training. The construction protocol requires all data-derived quantities in this cache, including correlations, textual descriptors, and bypass statistics, to be computed from the training partition.

Table 1: Offline variable-specific knowledge resources. A real ETTh1/HUFL example is visualized in Figure 2.

Source	Information	Fusion
Semantic	Meaning, unit, sampling context	Attention
Behavioral	Smoothness, volatility, periodicity	Attention
Relational	Cross-variate correlations	Attention
Temporal	Seasonal and time-of-day patterns	Extra view
Statistics	Distribution, trend, spectrum	MLP bypass

Template-based Construction Pipeline. Each human-readable template combines domain metadata with data-driven features. The metadata include the variable name, unit, and domain role from the dataset documentation. Training-set features include distributional statistics, trend slope, dominant FFT period, and Pearson correlations with other variables. The templates may be authored or refined with optional LLM assistance during offline preparation. The relational view names the most correlated variables, restoring context that a channel-independent encoder would otherwise omit. Table 5 evaluates five enhancement modes that control which feature groups are verbalized.

Let $\mathbf{E}_{\text{mv}} \in \mathbb{R}^{N\times K\times D}$ contain K cached views for each variable. Variable-wise cross-attention uses $\mathbf{c}_{\text{text}}^{(b,i)}$ as its query and produces an aggregated embedding $\mathbf{e}^{(b,i)} \in \mathbb{R}^D$. When time-dependent text is available, the input timestamp selects the corresponding slot embedding and appends it as an additional view.

The aggregated embedding is fused with the cross-variate representation through a learnable gate $g^{(i)}$ for variable i:

$$\mathbf{v}'^{(b,i)}_{\text{text}} = \mathbf{v}'^{(b,i)} + g^{(i)}\,\mathrm{CrossAttn}\Big(\mathrm{LN}(\mathbf{v}'^{(b,i)}), \mathrm{LN}(\mathbf{e}^{(b,i)}), \mathbf{e}^{(b,i)}\Big)$$

The gate is initialized to a small positive value, allowing every variable to learn its own reliance on textual priors. Statistical features bypass attention. A two-layer network $\mathrm{MLP}_{\text{stat}} : \mathbb{R}^{12} \to \mathbb{R}^D$ and a separate learnable gate γ_{stat} inject them after text attention:

$$\hat{\mathbf{v}}'^{(b,i)} = \mathbf{v}'^{(b,i)}_{\text{text}} + \gamma_{\text{stat}}\ \mathrm{MLP}_{\text{stat}}\Big(\mathbf{s}^{(i)}\Big)\,.$$

This bypass complements soft textual semantics with hard numerical evidence without consuming attention capacity.

We also regularize temporal summaries toward the aggregated descriptors of their corresponding variates:

$$\mathcal{L}_{\text{text}} = 1 - \frac{1}{BN}\sum_{b=1}^{B}\sum_{i=1}^{N}\cos\Big(\mathbf{c}_{\text{text}}^{(b,i)}, \mathbf{e}^{(b,i)}\Big)\,.$$

This term encourages agreement with the aggregated descriptors of each variate, while the forecasting loss supervises window-specific predictive content. The reported text analysis does not separately quantify the roles of this loss, gated fusion, and the statistical bypass.

3.4 Vision-Language Contrastive Alignment Module

When $N \le 50$, an auxiliary vision-language objective provides additional training supervision. Each normalized variable sequence is rendered as a color-coded line chart, encoded by the frozen CLIP vision encoder, and mapped to the shared width by a learnable projection $\mathbf{W}_{\text{proj}}$:

$$\mathbf{c}_{\text{img}}^{(b,i)} = \mathbf{W}_{\text{proj}}\ \text{CLIP-ViT}\Big(\mathrm{Render}(\mathbf{x}^{(b,i)})\Big)\,, \quad \mathbf{c}_{\text{img}}^{(b,i)} \in \mathbb{R}^D$$

Alignment is performed *independently for each variate*. For variate i, let $\mathbf{S}_{\text{img}}^{(i)}, \mathbf{S}_{\text{text}}^{(i)} \in \mathbb{R}^{B\times D}$ stack the L2-normalized visual and temporal summaries of the B samples in the batch. A bidirectional

InfoNCE loss matches each sample to its own rendering, using the remaining $B - 1$ same-variate samples as in-batch negatives, and the result is averaged over the N variates:

$$\mathcal{L}_{\text{align}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} \Big[\text{CE}\Big(\tau\, \mathbf{S}_{\text{img}}^{(i)} \mathbf{S}_{\text{text}}^{(i)\top}, \mathbf{I}_B\Big) + \text{CE}\Big(\tau\, \mathbf{S}_{\text{text}}^{(i)} \mathbf{S}_{\text{img}}^{(i)\top}, \mathbf{I}_B\Big) \Big],$$

where $\mathbf{I}_B$ denotes the batch-level matching-pair targets (the identity assignment over the B samples) and $\tau = \exp(s)$ is the learnable logit scale.

A curriculum schedule gradually increases the concentration of hard negatives. The rendered image is derived from the same numerical input, so this objective adds an encoder-induced representation prior. The available ablation evaluates that objective within SAGE; it does not separate CLIP pre-training from generic contrastive regularization. This module is active only during training. For $N > 50$, rendering hundreds of variable-wise images is impractical, so the vision loss is disabled and textual knowledge fusion remains active.

3.5 Forecast Generator and Training Objective

The augmented representation $\hat{\mathbf{v}}'$ is added to the final position of $\mathbf{R}$. This operation injects cross-variable and external knowledge without replacing earlier patch representations. The resulting token sequence is flattened and mapped to T_{pred} forecast steps by a two-layer MLP with GELU activation. RevIN then restores the original scale.

For $N > 50$, the full variable-wise patch path is replaced by a lightweight decoder for computational practicality. Cross-variate context modeling and textual semantic fusion remain active, and a linear layer maps each augmented variable representation to T_{pred} outputs. When time-dependent descriptions are available, the selected slot embedding supplies an additional gated correction at each forecast step.

The total objective combines forecast MSE with the two alignment losses:

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda_{\text{align}} \mathcal{L}_{\text{align}} + \lambda_{\text{text}} \mathcal{L}_{\text{text}},$$

where $\mathcal{L}_{\text{MSE}}$ is the error between $\hat{\mathbf{Y}}$ and the ground-truth future sequence.

The coefficients λ_{align} and λ_{text} weight vision-language and text alignment, respectively. The CLIP text encoder uses an encoder-specific learning-rate multiplier to control adaptation of the pretrained parameters.

4 Experiments

4.1 Setup

Datasets. We evaluate SAGE on 8 widely-used long-term forecasting benchmarks and the M4 short-term forecasting competition. The long-term benchmarks include: ETTh1, ETTh2, ETTm1, ETTm2 (electricity transformer temperature, N=7) (Zhou et al., 2020), Electricity (ECL, N=321), Traffic (N=862), Weather (N=21) (Wu et al., 2021), and Exchange (N=8) (Lai et al., 2017). For long-term forecasting, we use a fixed lookback window T=96 and prediction horizons $T_{\text{pred}} \in \{96, 192, 336, 720\}$. For short-term forecasting, we evaluate on the M4 dataset (Makridakis et al., 2020) comprising 100,000 time series across six frequencies (Yearly, Quarterly, Monthly, Weekly, Daily, Hourly).

Table 2 summarizes dataset characteristics and active configurations.

Metrics. Long-term forecasting is evaluated with Mean Squared Error (MSE) and Mean Absolute Error (MAE). Short-term forecasting follows the M4 competition protocol with SMAPE, MASE, and Overall Weighted Average (OWA).

Table 2: Long-term benchmark characteristics and SAGE pathways. All datasets use $T = 96$ and prediction horizons in $\{96, 192, 336, 720\}$.

Dataset	N	Observations	Interval	Pathway
ETTh1	7	17,420	1 hour	Full
ETTh2	7	17,420	1 hour	Full
ETTm1	7	69,680	15 minutes	Full
ETTm2	7	69,680	15 minutes	Full
ECL	321	26,304	1 hour	Lightweight
Traffic	862	17,544	1 hour	Lightweight
Weather	21	52,696	10 minutes	Full
Exchange	8	7,588	1 day	Full

Table 3: Dataset-averaged MSE / MAE comparison across prediction horizons $\{96, 192, 336, 720\}$ with lookback T=96. **Bold**: best; underline: second best among the listed methods. The final two rows report first-place counts for each metric.

Dataset	Metric	DLinear (2023)	TimesNet (2022)	PatchTST (2022)	Fred Former (2024)	iTrans former (2024b)	Amplifier (2025)	DUET (2025)	SRSNet (2026)	**SAGE** Ours
ETTh1	MSE	0.456	0.458	0.469	0.449	0.454	0.449	0.443	0.442	**0.440**
	MAE	0.449	0.450	0.450	0.442	0.442	0.442	0.438	**0.437**	0.447
ETTh2	MSE	0.559	0.414	0.387	0.378	0.383	0.389	0.376	0.376	**0.373**
	MAE	0.515	0.421	0.404	0.405	0.404	0.405	0.405	0.406	**0.403**
ETTm1	MSE	0.403	0.400	0.387	0.385	0.407	0.385	0.390	0.385	**0.384**
	MAE	0.406	0.407	**0.400**	0.402	0.412	0.402	0.403	0.402	**0.400**
ETTm2	MSE	0.350	0.291	0.281	0.281	0.288	**0.280**	**0.280**	0.284	0.283
	MAE	0.377	0.338	0.332	0.332	0.337	**0.331**	**0.331**	0.334	0.334
ECL	MSE	0.212	0.192	0.205	0.178	0.178	0.173	0.172	0.188	**0.165**
	MAE	0.295	0.289	0.292	0.271	0.274	0.267	0.266	0.281	**0.260**
Traffic	MSE	0.625	0.620	0.481	0.434	0.432	0.485	0.451	0.495	**0.430**
	MAE	0.383	0.395	0.363	0.285	0.282	0.324	0.297	0.327	**0.278**
Weather	MSE	0.265	0.259	0.259	0.256	0.258	0.247	0.251	0.249	**0.243**
	MAE	0.287	0.285	0.285	0.279	0.283	0.276	0.278	0.277	**0.271**
Exchange	MSE	0.353	0.416	0.367	0.374	0.360	0.361	0.341	0.343	**0.332**
	MAE	0.411	0.445	0.420	0.408	0.410	0.402	0.399	0.395	**0.393**
1st Count	MSE	0	0	0	0	0	1	1	0	7
	MAE	0	0	1	0	0	1	1	1	6

Implementation. SAGE fine-tunes the CLIP ViT-B/32 text encoder and keeps its vision encoder frozen. RevIN-normalized series use frequency-enhanced patches of length 16 and stride 8 (Kim et al., 2021); offline descriptions are fused through per-variable gates. For N>50, a lightweight path bypasses patching. Training uses Adam, cosine scheduling, an encoder-specific learning-rate multiplier, and bidirectional InfoNCE alignment in the full configuration. We evaluate 651 configurations across all datasets, horizons, and five text-enhancement modes.

Environment and Computational Cost. Experiments use Ubuntu 22.04, AMD EPYC 9654 processors, 768 GB RAM, four NVIDIA A800 GPUs (one per run), Python 3.11, PyTorch 2.5.1, and CUDA 12.1. Depending on dimensionality and pathway, epochs take roughly 17 s on ETTh1, 21 s on Exchange, 130 s on ECL, 250 s on Weather, and 230–290 s on Traffic. Early stopping typically converges within 10 epochs: under 3 minutes on ETT and under 45 minutes on Traffic. The 651-run sweep used approximately 1,000 single-GPU hours across four GPUs. Offline text construction takes seconds per dataset, amortized across runs.

4.2 Main Results

Among the results listed in Table 3, SAGE has the lowest average MSE on 7 of 8 datasets and the lowest average MAE on 6 of 8. Averaging the displayed dataset-level MSE values gives 0.331 for SAGE and 0.338 for DUET, the strongest listed comparator by this aggregate, corresponding to about a 2.0% reduction. iTransformer (Liu et al., 2024b) has a corresponding average MSE of 0.345.

Amplifier (Fei et al., 2025) and DUET (Qiu et al., 2025) lead on ETTm2 (0.280 MSE, 0.331 MAE); SRSNet (Wu et al., 2026) has the lowest ETTh1 MAE. SAGE's overall MAE is 0.348 at lookback T=96. The datasets span 7–862 variables: ECL and Traffic use the lightweight configuration, and the other six use the full configuration. Offline text preparation avoids LLM calls during training or forecasting. The appendix reports all 32 dataset-horizon combinations.

Within the subset of M4 methods reported in Table 4, SAGE has the lowest values of all three metrics, including an OWA of 0.834. TimesNet and N-HiTS obtain OWA values of 0.851 and 0.861, respectively. The table does not isolate the effect of textual augmentation on M4.

4.3 Text Enhancement Analysis

We conduct a systematic ablation over 5 text enhancement modes across all 8 datasets and 4 prediction horizons. For each dataset-horizon pair, all architectural and text-mode hyperparameters are selected using validation MSE. The selected configuration is then evaluated once on the held-out test split. The 5 modes progressively enrich the text description. **+Stat** adds mean, variance, skewness, and kurtosis. **+Dyn** adds moving averages and trend slopes. **+Stat+Dyn**, abbreviated as +S+D, combines both groups. **+Dyn+Freq** adds FFT-derived spectral features to dynamic descriptions. **+Full** includes statistical, dynamic, spectral, and cross-variable features.

As shown in Table 5, the reported text-augmented configurations improve forecasting in 31 of 32 dataset-horizon combinations, with dataset-level gains shown in the table. Among the modes, +Stat is the most reliable (78.1% win rate across all experiments, selected as the best mode in 45.2% of combinations), followed by +Dyn (22.6%). The combined mode +Stat+Dyn achieves the highest peak improvement (6.3% on Exchange), suggesting that statistical and dynamic features provide complementary information. Interestingly, modes incorporating frequency features (+Dyn+Freq, +Full) are less consistently beneficial, likely because the frequency-enhanced patch embedding already captures spectral structure.

Text gains tend to be larger on lower-dimensional datasets, without a strictly monotonic relationship (Table 5). Exchange (N=8), Weather (N=21), ECL (N=321), and Traffic (N=862) improve by 6.3%, 3.9%, 1.2%, and 0.9%, respectively. Gates may modulate descriptors more effectively with fewer variables, and cross-channel redundancy may affect their incremental value. Gates can attenuate unhelpful descriptors, but this analysis does not isolate these explanations.

4.4 Vision Enhancement Analysis

We isolate the contribution of visual supervision by retraining SAGE with $\lambda_{\text{align}} = 0$, while keeping the model architecture, textual knowledge, optimization settings, and all other loss terms unchanged. Because the visual branch is enabled only when $N \leq 50$, the ablation is conducted on the six eligible datasets. Table 6 reports the test MSE averaged over the four prediction horizons.

Table 4: M4 short-term forecasting. Lower is better; bold and underline mark the best and second-best results.

Method	SMAPE	MASE	OWA
TimesNet	<u>11.800</u>	<u>1.591</u>	<u>0.851</u>
N-HiTS	11.936	1.610	0.861
PatchTST	12.072	1.622	0.869
FEDformer	12.614	1.732	0.918
iTransformer	12.792	1.756	0.931
SAGE	**11.593**	**1.556**	**0.834**

Table 5: Average text gains. Mode: best family; Δ: relative MSE reduction.

Dataset	N	No Text	+Text	Mode	$\Delta(\%)\uparrow$
ETTh1	7	0.450	0.440	+Stat	2.2
ETTh2	7	0.381	0.373	+Dyn	2.1
ETTm1	7	0.395	0.384	+Stat	2.8
ETTm2	7	0.294	0.283	+Stat	3.7
ECL	321	0.167	0.165	+Stat	1.2
Traffic	862	0.434	0.430	+Dyn	0.9
Weather	21	0.253	0.243	+Stat	3.9
Exchange	8	0.354	0.332	+S+D	6.3
Average					**2.8**

Table 6: Average vision-enhancement gains. Δ is the relative MSE reduction.

Dataset	N	No Vision	+Vision	$\Delta(\%)\uparrow$
ETTh1	7	0.448	0.440	1.9
ETTh2	7	0.378	0.373	1.3
ETTm1	7	0.391	0.384	1.8
ETTm2	7	0.290	0.283	2.5
Weather	21	0.244	0.243	0.4
Exchange	8	0.337	0.332	1.5
Average				**1.6**

Alignment reduces MSE on all six eligible datasets, from 0.348 to 0.343 on average (1.6%). Gains are largest for ETTm2 (2.5%), ETTh1 (1.9%), and ETTm1 (1.8%). Visible trends and periodicity in ETT and Exchange may provide complementary shape information; the reductions are consistent with regularization, but attribution to pretrained CLIP requires controls separating the encoder from the contrastive objective.

Weather's smaller 0.4% gain may reflect its 21 heterogeneous, strongly coupled variables: a 96-step history at 10-minute intervals spans only 16 hours, less than a daily cycle. Separate renderings omit interactions among temperature, humidity, pressure, wind, and radiation (Wu et al., 2021). This ablation does not isolate their effect beyond the frequency-enhanced and cross-variate representations.

The visual branch supplies training-only representation regularization: it does not feed image features directly to the forecast head and adds no visual computation at forecasting time.

5 Conclusion

SAGE combines template-based, variable-specific statistical, dynamic, and spectral descriptors through independent gates and a statistical bypass, supported by frequency-enhanced encoding and cross-variable context. Its pretrained CLIP ViT-B/32 backbone fine-tunes the text encoder and freezes the vision encoder for training-only contrastive supervision in the full configuration. Forecasting requires no generative LLM calls.

Among the results listed in our tables, SAGE obtains the lowest average MSE on 7 of 8 long-term datasets, the lowest average MAE on 6 of 8, and an M4 OWA of 0.834. Text enhancement improves 31 of 32 dataset-horizon settings. These results suggest that variable-level descriptors and training-time visual supervision can improve the corresponding SAGE configurations.

Limitations and Future Work. The evaluation reports no estimates of seed variability or matched controls isolating the statistical bypass, text-alignment loss, and pretrained CLIP representations. Small margins therefore remain descriptive of the reported runs. Direct comparisons with closely related multimodal forecasters and measurements of parameter counts, memory, and forecast latency are also needed. Future directions include end-to-end replacements for hand-crafted templates, potentially using small language models distilled from domain experts to improve text quality and domain adaptability; group-wise or hierarchical gates sharing text across correlated channels to address smaller gains in high dimensions; external events or ontologies complementing metadata and training statistics during regime changes; and multivariate-to-multivariate generation and probabilistic forecasting for broader decision support.

AI USE STATEMENT

Generative AI was used to improve the manuscript's clarity and concision, assist with literature checks and the interpretation and presentation of reported results, and format the document. Automated checks verified consistency with earlier versions. The authors are responsible for the final content of this work.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Forty-first international conference on machine learning*, 2024.

Sixun Dong, Wei Fan, Teresa Wu, and Yanjie Fu. Teaching time series to see and speak: Forecasting with aligned visual and textual perspectives, 2025. URL https://arxiv.org/abs/2506.24124.

Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee Keong Kwoh, Xiaoli Li, and Cuntai Guan. Time-series representation learning via temporal and contextual contrasting. *arXiv preprint arXiv:2106.14112*, 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 *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 11645–11653, 2025.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv preprint arXiv:2402.03885*, 2024.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. *Advances in neural information processing systems*, 36:19622–19635, 2023.

Zelin He, Sarah Alnegheimish, and Matthew Reimherr. Harnessing vision-language models for time series anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21690–21698, 2026.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. In *International conference on learning representations*, volume 2024, pp. 23857–23880, 2024.

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*, 2021.

Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long-and short-term temporal patterns with deep neural networks. *arXiv preprint arXiv:1703.07015*, 2017.

Seunghan Lee, Taeyoung Park, and Kibok Lee. Soft contrastive learning for time series. In *International Conference on Learning Representations*, volume 2024, pp. 46815–46839, 2024.

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, 2024a. URL https://arxiv.org/abs/2406.01638. Accepted at AAAI 2025.

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*, volume 2024, pp. 11116–11140, 2024b.

Sisuo Lyu, Siru Zhong, Weilin Ruan, Qingxiang Liu, Qingsong Wen, Hui Xiong, and Yuxuan Liang. Occamvts: Distilling vision models to 1% parameters for time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24216–24225, 2026.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The m4 competition: 100,000 time series and 61 forecasting methods. *International Journal of Forecasting*, 36(1): 54–74, 2020.

Jingchao Ni, Ziming Zhao, ChengAo Shen, Hanghang Tong, Dongjin Song, Wei Cheng, Dongsheng Luo, and Haifeng Chen. Harnessing vision models for time series analysis: A survey. *arXiv preprint arXiv:2502.08869*, 2025.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

Zijie Pan, Yushan Jiang, Sahil Garg, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. s^2 ip-llm: Semantic space informed prompt learning with llm for time series forecasting. In *Forty-first International Conference on Machine Learning*, 2024.

Xihao Piao, Zheng Chen, Taichi Murayama, Yasuko Matsubara, and Yasushi Sakurai. Fredformer: Frequency debiased transformer for time series forecasting. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 2400–2410, 2024.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. Duet: Dual clustering enhanced multivariate time series forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 1185–1196, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin. Time-moe: Billion-scale time series foundation models with mixture of experts. In *International conference on learning representations*, volume 2025, pp. 34635–34667, 2025.

Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. Test: Text prototype aligned embedding to activate llm's ability for time series. In *International Conference on Learning Representations*, volume 2024, pp. 37854–37881, 2024.

Mingtian Tan, Mike A Merrill, Vinayak Gupta, Tim Althoff, and Tom Hartvigsen. Are language models actually useful for time series forecasting? *NeurIPS*, 37:60162–60191, 2024.

Zhikai Wang and Weihua Zhang. Frequency-aware adaptive contrastive learning for sequential recommendation. *arXiv preprint arXiv:2601.17057*, 2026.

Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, et al. Context is key: A benchmark for forecasting with essential textual information. *arXiv preprint arXiv:2410.18959*, 2024.

Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi. Cost: Contrastive learning of disentangled seasonal-trend representations for time series forecasting. *arXiv preprint arXiv:2202.01575*, 2022.

Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *Forty-first International Conference on Machine Learning*, 2024.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

Xingjian Wu, Jianxin Jin, Wanghui Qiu, Peng Chen, Yang Shu, Bin Yang, and Chenjuan Guo. Aurora: Towards universal generative multimodal time series forecasting. *arXiv preprint arXiv:2509.22295*, 2025.

Xingjian Wu, Xiangfei Qiu, Hanyin Cheng, Zhengyu Li, Jilin Hu, Chenjuan Guo, and Bin Yang. Enhancing time series forecasting through selective representation spaces: A patch perspective. *Advances in Neural Information Processing Systems*, 38:23328–23354, 2026.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8980–8987, 2022.

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.

Xiang Zhang, Ziyuan Zhao, Theodoros Tsiligkaridis, and Marinka Zitnik. Self-supervised contrastive pre-training for time series via time-frequency consistency. *Advances in neural information processing systems*, 35:3988–4003, 2022.

Siru Zhong, Weilin Ruan, Ming Jin, Huan Li, Qingsong Wen, and Yuxuan Liang. Time-vlm: Exploring multimodal vision-language models for augmented time series forecasting. *arXiv preprint arXiv:2502.04395*, 2025.

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. *arXiv preprint arXiv:2012.07436*, 2020.

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*, pp. 27268–27286. PMLR, 2022.

Tian Zhou, Peisong Niu, Liang Sun, Rong Jin, et al. One fits all: Power general time series analysis by pretrained lm. *Advances in neural information processing systems*, 36:43322–43355, 2023.

A Full Per-Horizon Forecasting Results

Table 7: Full per-horizon long-term forecasting results (MSE↓ / MAE↓). We compare with representative methods spanning 2022–2026 under lookback T=96. **Bold**: best; underline: second best among all nine methods. Each method retains adjacent MSE and MAE columns.

Dataset	T_{pred}	DLinear (2023)		TimesNet (2022)		PatchTST (2022)		FredFormer (2024)		iTransformer (2024b)		Amplifier (2025)		DUET (2025)		SRSNet (2026)		SAGE Ours	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.386	0.400	0.384	0.400	0.414	0.423	0.378	0.400	0.386	0.407	**0.376**	**0.399**	0.377	**0.399**	0.383	0.402	0.383	0.406
	192	0.437	0.432	0.436	**0.429**	0.460	0.444	0.435	0.433	0.441	0.432	0.442	0.437	0.429	**0.429**	0.433	0.432	**0.428**	0.437
	336	0.481	0.459	0.491	0.469	0.501	0.469	0.485	0.464	0.487	0.461	0.478	0.459	0.471	**0.454**	0.476	0.459	**0.467**	0.461
	720	0.519	0.506	0.521	0.500	0.500	0.464	0.496	0.469	0.503	0.468	0.501	0.471	0.496	0.468	**0.474**	**0.455**	0.483	0.484
ETTh2	96	0.333	0.387	0.340	0.382	0.302	0.348	0.291	0.347	0.297	0.349	0.298	0.346	0.300	0.354	0.296	0.350	**0.289**	**0.340**
	192	0.477	0.476	0.402	0.414	0.388	0.404	0.372	0.401	0.380	0.402	0.378	0.399	0.372	0.402	0.370	0.402	**0.367**	**0.397**
	336	0.594	0.541	0.452	0.440	0.426	**0.431**	0.419	0.434	0.428	0.432	0.428	0.432	0.415	0.432	**0.413**	0.433	0.414	**0.431**
	720	0.831	0.657	0.462	0.447	0.431	0.432	0.431	0.438	0.427	**0.431**	0.452	0.444	**0.416**	0.432	0.425	0.438	0.423	0.445
ETTm1	96	0.345	0.372	0.338	0.375	0.329	0.367	0.326	0.365	0.334	0.366	0.320	0.363	0.324	0.363	0.319	0.362	**0.318**	**0.359**
	192	0.380	0.389	0.374	0.387	0.367	0.385	0.363	0.386	0.377	0.392	0.364	0.387	0.369	0.389	**0.359**	0.385	0.362	**0.382**
	336	0.413	0.413	0.410	0.411	0.399	0.410	0.395	0.410	0.426	0.426	0.395	0.409	0.404	0.413	**0.391**	**0.406**	0.399	0.412
	720	0.474	0.449	0.478	0.453	**0.454**	**0.439**	0.456	0.445	0.491	0.463	0.462	0.450	0.463	0.448	0.470	0.455	0.458	0.448
ETTm2	96	0.193	0.274	0.187	0.267	0.175	0.259	0.177	0.262	0.180	0.264	0.178	0.262	**0.174**	**0.258**	0.181	0.264	0.176	0.268
	192	0.284	0.339	0.249	0.311	**0.241**	**0.305**	0.243	0.306	0.250	0.311	0.243	0.306	0.243	**0.305**	0.243	0.307	0.243	**0.305**
	336	0.369	0.393	0.321	0.360	0.305	0.349	**0.302**	**0.347**	0.311	0.352	0.305	0.348	0.304	0.348	0.306	0.350	0.308	**0.347**
	720	0.554	0.500	0.408	0.414	0.402	0.416	0.404	0.414	0.412	0.420	**0.393**	**0.409**	0.399	0.413	0.407	0.416	0.407	0.416
ECL	96	0.197	0.282	0.168	0.272	0.181	0.269	0.148	0.243	0.148	0.247	0.149	0.244	0.145	0.240	0.161	0.257	**0.136**	**0.231**
	192	0.196	0.285	0.184	0.278	0.199	0.289	0.165	0.260	0.162	0.257	0.165	0.259	0.163	0.257	0.172	0.265	**0.157**	**0.249**
	336	0.209	0.296	0.198	0.296	0.212	0.300	0.180	0.274	0.178	0.275	0.176	0.270	0.175	0.270	0.190	0.285	**0.171**	**0.267**
	720	0.245	0.319	0.218	0.310	0.228	0.311	0.218	0.307	0.225	0.317	0.204	0.296	0.204	0.296	0.231	0.318	**0.195**	**0.293**
Traffic	96	0.650	0.396	0.593	0.388	0.469	0.363	0.403	0.273	0.399	0.268	0.450	0.307	0.407	0.272	0.471	0.314	**0.393**	**0.259**
	192	0.598	0.370	0.617	0.396	0.467	0.355	0.429	0.282	**0.421**	0.276	0.489	0.325	0.431	0.284	0.480	0.316	0.422	**0.274**
	336	0.605	0.373	0.629	0.399	0.475	0.358	0.441	0.287	**0.437**	0.284	0.484	0.319	0.456	0.299	0.496	0.325	0.439	**0.281**
	720	0.645	0.394	0.640	0.396	0.514	0.375	**0.463**	**0.297**	0.471	0.300	0.517	0.345	0.509	0.334	0.531	0.353	0.468	0.298
Weather	96	0.196	0.247	0.172	0.214	0.177	0.228	0.163	0.207	0.174	0.217	0.165	0.210	0.163	0.207	0.167	0.211	**0.157**	**0.201**
	192	0.237	0.267	0.219	0.260	0.225	0.265	0.224	0.260	0.221	0.261	0.212	0.256	0.218	0.260	0.215	0.258	**0.206**	**0.248**
	336	0.283	0.298	0.280	0.306	0.278	0.300	0.278	0.301	0.278	0.302	0.267	0.296	0.274	0.299	0.270	0.298	**0.266**	**0.292**
	720	0.345	**0.335**	0.365	0.359	0.354	0.348	0.357	0.348	0.358	0.352	0.344	0.343	0.349	0.346	0.346	0.343	**0.343**	0.341
Exchange	96	0.088	0.218	0.107	0.234	0.086	0.215	0.084	0.202	0.086	0.218	**0.083**	0.202	0.085	0.202	0.084	**0.201**	0.084	0.205
	192	0.176	0.315	0.226	0.344	0.205	0.335	0.183	0.302	0.177	0.306	0.175	0.297	0.173	0.299	0.171	0.303	**0.170**	**0.296**
	336	**0.308**	0.415	0.431	0.487	0.385	0.459	0.335	0.418	0.331	0.422	0.328	0.414	0.322	**0.408**	0.324	0.410	0.315	0.410
	720	0.839	0.695	0.901	0.714	0.793	0.672	0.893	0.711	0.847	0.695	0.858	0.696	0.785	0.688	0.792	0.667	**0.758**	**0.659**
1st Count		1	1	0	1	2	3	2	2	2	1	3	2	2	6	4	3	16	19