CONTEXT2ACTION: AN AGENTIC FRAMEWORK FOR REAL-WORLD-AWARE TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Time series are shaped not only by their historical dynamics but also by real-world contexts such as holidays, promotions, weather, news, and unexpected disruptions. Existing methods either focus primarily on numerical patterns or incorporate contextual information through end-to-end prediction, without explicitly determining how such contexts should influence forecasting decisions. We propose **Context2Action**, an agentic framework that translates real-world contexts into explicit and controllable forecasting actions. By analyzing the historical errors of numerical predictors, Context2Action learns how contextual factors are associated with both localized changes and broader temporal deviations. Retrieval-augmented LLM experts then adapt relevant historical experiences to the current forecasting state and generate structured forecast revisions, while a hierarchical decision mechanism filters unsupported actions. The framework is optimized through supervised fine-tuning and forecasting-oriented reinforcement learning, aligning contextual reasoning with numerical forecasting utility. Experiments on four context-rich datasets show that Context2Action outperforms strong numerical, LLM-based, and context-aware forecasting baselines. It also generalizes across heterogeneous numerical predictors without additional expert training, demonstrating its effectiveness and adaptability for real-world-context-aware time series forecasting.

1 INTRODUCTION

Time series forecasting supports decision-making in transportation, energy, finance, and urban management. Recent numerical forecasting models have achieved substantial progress by learning trends, periodicity, long-range dependencies, and cross-variable correlations from historical observations Salinas et al. (2020); Oreshkin et al. (2020); Zhou et al. (2022). Large language models have also been adapted to forecasting by representing numerical sequences as text or reprogramming time-series representations into their embedding spaces Das et al. (2024); Goswami et al. (2024). These advances provide increasingly powerful tools for extrapolating temporal patterns from observed data.

However, real-world trajectories are shaped not only by historical patterns but also by contextual factors in the surrounding world. Holidays, promotions, weather conditions, public events, policy changes, and unexpected disruptions may cause localized or systematic deviations from otherwise regular dynamics. Their effects are heterogeneous and state-dependent: similar contexts can influence different variables or temporal states in different ways. Forecasting under such conditions therefore requires not only recognizing real-world contexts, but also determining how they should alter the current numerical prediction.

This motivates the central research question of this work: *How can real-world contexts be systematically transformed into reliable, controllable, and interpretable revisions of numerical time series forecasts?* Existing methods can be broadly divided into two categories.

First, ***numerical-pattern-driven forecasting models*** learn predictive regularities primarily from time-series observations. Representative deep models improve temporal modeling through linear decomposition, patch-based representations, and variate-centric attention Zeng et al. (2023); Nie et al. (2023); Liu et al. (2024). LLM-based methods further serialize numerical values or align time-series representations with pretrained language models Gruver et al. (2023); Jin et al. (2024).

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.

Despite their strong pattern-modeling capabilities, these methods remain predominantly driven by numerical histories and have limited ability to incorporate heterogeneous, open-world contexts expressed in natural language.

Second, ***context-aware forecasting models*** jointly model time series with external information such as news, textual descriptions, and real-world events. Recent studies employ LLM agents to select relevant news Wang et al. (2024), integrate textual and temporal representations for multimodal prediction Jiang et al. (2025), or directly fuse textual events into generative forecasting architectures Ge et al. (2025). While these methods improve access to external information, context understanding is generally embedded within an end-to-end prediction process. They do not explicitly specify which part of an existing forecast should be changed, how large the change should be, or whether the proposed intervention should be accepted. Directly regenerating future values may also overwrite reliable temporal patterns already captured by numerical predictors.

These limitations reveal a gap between *context understanding* and *forecasting action*. A context-aware forecasting system should determine not only what may influence the future, but also how the current forecast fails under that influence, what concrete revision should be performed, and whether the available evidence is sufficient to justify the revision. This requires grounding contextual reasoning in observed forecasting outcomes rather than relying solely on semantic plausibility.

To address this gap, we propose Context2Action, an agentic framework that translates real-world contexts into structured revisions of numerical forecasts. Instead of replacing specialized predictors, Context2Action uses a pretrained and frozen numerical model to produce base forecasts and analyzes its historical residuals to construct a Residual-Grounded Dual Context Memory. The Local Context Memory captures context-specific effects over localized intervals, while the Global Context Memory characterizes broader level and trend deviations.

Two retrieval-augmented LLM experts then convert relevant historical experiences into complementary forecasting actions. The Local Context Expert specifies the impact start time, duration, and signed adjustment magnitude for each context, while the Global Context Expert calibrates the overall prediction level and temporal trend. The two experts make hierarchical *Accept/No-Action* decisions in a local-to-global manner, retaining the current prediction when a proposed intervention is unsupported. The framework is initialized through supervised fine-tuning and further aligned with forecasting utility through GRPO-based optimization.

Extensive experiments on four context-rich forecasting datasets demonstrate that Context2Action outperforms numerical, LLM-based, and context-aware baselines. Further analyses validate the contributions of its dual memories, retrieval mechanisms, structured actions, and GRPO optimization. Replacing the numerical predictor after expert training also shows that the learned context-to-action capability transfers across heterogeneous forecasting architectures without additional adaptation.

The main contributions are summarized as follows:

- We formulate real-world-context-aware forecasting as a **context-to-action forecast revision problem** and introduce a residual-grounded paradigm that connects contextual information with the actual errors of numerical predictors.
- We develop a **Residual-Grounded Dual Context Memory** and two retrieval-augmented experts that represent localized contextual effects through explicit start-time, duration, and magnitude actions, and systematic deviations through level and trend calibration.
- We propose a **hierarchical and utility-aware decision mechanism** with an explicit *No-Action* option. Through supervised fine-tuning and GRPO-based optimization, it jointly improves experience selection, structured action generation, and forecast revision decisions.

2 METHODOLOGY

2.1 PROBLEM FORMULATION

Let $\mathbf{X}_i = [\mathbf{x}_{t-L+1}, \ldots, \mathbf{x}_t] \in \mathbb{R}^{L \times D}$ denote the historical observations over a look-back window of length L and D variables. The forecasting target over the next H time steps is $\mathbf{Y}_i = [\mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+H}] \in \mathbb{R}^{H \times D}$. Each forecasting instance is associated with real-world contextual in-

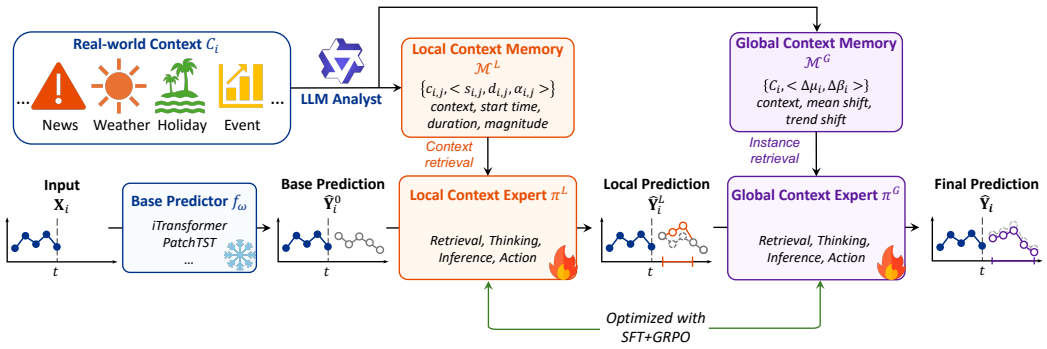


Figure 1: Overview of Context2Action. Real-world contexts are analyzed to construct local and global context memories, which store context-specific actions and window-level corrections, respectively. Given an input instance, a frozen numerical predictor first produces the base forecast. The Local Context Expert retrieves relevant contextual experiences and performs localized revisions, after which the Global Context Expert further calibrates the overall level and trend to obtain the final prediction. Both experts are optimized with SFT and GRPO.

formation. $\mathcal{C}_i = \{c_{i,k}\}_{k=1}^{K_i}$, where each context item contains its temporal information and a description of the corresponding real-world factor, *e.g.,* holiday, promotion, news event, or weather condition. The number of contexts may vary across instances, and $\mathcal{C}_i = \varnothing$ when no contextual information is provided.

Given a dataset $\mathcal{D} = \{(\mathbf{X}_i, \mathcal{C}_i, \mathbf{Y}_i)\}_{i=1}^N$, the objective of real-world-context-aware time-series forecasting is to learn a forecasting function $\widehat{\mathbf{Y}}_i = F_\Theta(\mathbf{X}_i, \mathcal{C}_i)$ by minimizing:

$$\Theta^* = \arg\min_\Theta \frac{1}{N} \sum_{i=1}^{N} \ell\left(\mathbf{Y}_i, F_\Theta(\mathbf{X}_i, \mathcal{C}_i)\right), \tag{1}$$

where $\ell(\cdot, \cdot)$ denotes the forecasting loss.

2.2 Context-to-Action Forecasting Agent Overview

We propose Context2Action, an agentic forecasting framework that translates real-world contexts into explicit forecast revisions, as shown in Figure 1. Given an input sequence $\mathbf{X}_i$, we first introduce a pretrained forecasting model $\mathcal{F}_\omega$ to yield a base forecast $\widehat{\mathbf{Y}}_i^0$. To complement such numerical modeling with understanding of real-world influences, we analyze historical forecasting errors to construct local and global context memories, characterizing contextual effects at different temporal granularities. Based on these memories, the Local Context Expert retrieves semantically relevant context experiences and jointly generates localized actions for all contexts associated with the forecasting instance. The Global Context Expert further retrieves numerically similar historical windows and produces an overall window-level calibration. Each expert also determines whether its proposed action should be applied or rejected through a *No-Action* decision. The resulting local-to-global process establishes a traceable path from contextual evidence retrieval and structured action generation to the final forecast $\widehat{\mathbf{Y}}_i$.

2.3 Residual-Grounded Dual Context Memory

Real-world contexts indicate what may influence future observations, but their actual effects on future states are hard to quantify. Similar contexts may lead to different forecasting deviations depending on the affected series, event timing, and temporal state. We therefore ground contextual information in historical forecasting outcomes and construct two complementary memories for local context effects and global temporal deviations.

Given historical observations $\mathbf{X}_i$, a pretrained numerical forecasting model, *e.g.,* iTransformer or PatchTST, produces the base forecast $\widehat{\mathbf{Y}}_i^0 = \mathcal{F}_\omega(\mathbf{X}_i)$, with residual $\boldsymbol{\epsilon}_i = \mathbf{Y}_i - \widehat{\mathbf{Y}}_i^0$. Based on this residual $\boldsymbol{\epsilon}_i$ and context $\mathcal{C}_i$, we incorporate an LLM analyst to jointly examine the context's impact on the prediction.

2.3.1 LOCAL CONTEXT MEMORY

We establish a Local Context Memory $\mathcal{M}^L$ to capture how individual contexts affect localized regions of the forecasting horizon. An LLM analyst is utilized to analyze the impact of all contexts in the instance. Specifically, the LLM analyst, *e.g.,* Qwen3.5-9B, jointly examines each historical context's $c_{i,j} \in \mathcal{C}_i$ semantics and timing, the base forecast, and the aligned forecasting residuals to derive a Local Context Action $\bar{a}^L_{i,j} = \langle s_{i,j}, d_{i,j}, \alpha_{i,j} \rangle$, where $s_{i,j}$ denotes the impact start time, $d_{i,j} \in \{1, \ldots, H\}$ denotes the duration, and $\alpha_{i,j} \in \mathbb{R}$ denotes the signed adjustment magnitude. The residual sequence is aligned with the context timestamp, providing empirical evidence for determining the affected interval and adjustment strength.

The Local Context Memory is constructed as $\mathcal{M}^L = \left\{ \left(c_{i,j}, \bar{a}^L_{i,j} \right) \right\}$, where each entry associates a historical context with its residual-grounded local reference action. Consequently, $\mathcal{M}^L$ preserves grounded event-level experiences of when, where, and to what extent a context influences the numerical forecast. These context–action pairs provide retrieval evidence and action supervision for the Local Context Expert. Ground-truth observations and residuals are used only for training set memory construction to avoid information leakage.

2.3.2 GLOBAL CONTEXT MEMORY

Local context actions may correct context-specific deviations while leaving systematic errors in the overall prediction level or temporal trend. We therefore construct a Global Context Memory $\mathcal{M}^G$ to characterize the remaining window-level discrepancy after local revision.

For each historical instance, the local actions are first applied to the base forecast to obtain a locally revised prediction $\widehat{\mathbf{Y}}^L_i$. The remaining discrepancy with respect to the ground truth $\mathbf{Y}_i$ is summarized as a Global Context Action $\bar{a}^G_i = \langle \Delta\mu_i, \Delta\beta_i \rangle$, where $\Delta\mu_i$ represents the overall shift correction and $\Delta\beta_i$ represents the trend correction.

The Global Context Memory is constructed as $\mathcal{M}^G = \left\{ \left(\mathcal{C}_i, \bar{a}^G_i \right) \right\}$, where each entry associates the historical contexts of a forecasting window with the corresponding global reference action. Unlike the event-centered Local Context Memory, $\mathcal{M}^G$ is organized around complete historical windows and captures systematic forecasting deviations shared by numerically similar temporal states.

2.4 RETRIEVAL-AUGMENTED DUAL CONTEXT EXPERTS

Historical context memories provide grounded evidence of how real-world factors affect numerical forecasts. However, their effects cannot be directly transferred to a new instance, as they may vary with the current temporal state and prediction pattern. We therefore develop two LLM-based experts that retrieve relevant historical experiences and generate structured forecasting actions at complementary temporal scales, *i.e.,* Local Context Expert and Global Context Expert.

The Local Context Expert focuses on context effects within particular forecasting intervals, whereas the Global Context Expert captures broader changes in the overall prediction level and trend. Rather than directly generating future values, both experts operate through constrained action spaces, preserving the temporal prediction of the numerical model.

2.4.1 LOCAL CONTEXT EXPERT

The Local Context Expert π^L identifies the localized effect of each context in $\mathcal{C}_i$. For a context $c_{i,j}$, it constructs a textual query by combining the series identity with the context description and retrieves K semantically relevant experiences from $\mathcal{M}^L$. The retrieval pool is restricted to contexts preceding the current forecasting instance. The expert then compares the candidates and selects the most informative experience as reasoning evidence. The local prompt and interaction examples are provided in Appendix A.1.

The expert follows a rationale-before-action decoding protocol. It first generates an explanation enclosed by the `<think>` and `</think>` tags and subsequently outputs the structured Local Context Action. Conditioned on the current forecasting state and the selected historical experience, the expert first generates a context-impact explanation $z^L_{i,j}$ enclosed by the `<think>` and `</think>`

tags, and then produces a candidate Local Context Action $\hat{a}_{i,j}^L = \left\langle \hat{s}_{i,j}, \hat{d}_{i,j}, \hat{\alpha}_{i,j} \right\rangle$. The explanation relates the current context and retrieved historical evidence to the proposed impact onset, duration, and adjustment magnitude. Its generation process is expressed as:

$$\left(z_{i,j}^L, \widehat{\mathcal{A}}_i^L\right) \sim \pi^L \left(\cdot \mid \mathbf{X}_i, \widehat{\mathbf{Y}}_i^0, c_{i,j}, \bar{a}_{i,j}^L\right), \tag{2}$$

where $\bar{a}_{i,j}^L$ is the candidate memories retrieved from the Local Context Memory $\mathcal{M}^L$. The actions generated for all contexts form $\widehat{\mathcal{A}}_i^L = \left\{\hat{a}_{i,j}^L\right\}_{j=1}^{K_i}$. Each context is assigned an explicit action, while overlapping contexts are considered jointly when determining their timing and magnitude. When $\mathcal{C}_i = \varnothing$, no local action is applied. Applying them to the base forecast produces the locally revised prediction

$$\hat{\mathbf{Y}}_i^L = \mathcal{R}^L \left(\widehat{\mathbf{Y}}_i^0, \widehat{\mathcal{A}}_i^L\right). \tag{3}$$

Here, $\mathcal{R}^L$ applies each accepted local action to its specified interval by adjusting the corresponding base predictions according to the signed magnitude. When multiple actions overlap, their adjustments are jointly applied to the affected positions. If the Local Context Expert selects NO-ACTION, or if no valid context is available, the base prediction is retained, *i.e.,* $\hat{\mathbf{Y}}_i^L = \widehat{\mathbf{Y}}_i^0$.

2.4.2 GLOBAL CONTEXT EXPERT

Local actions capture context-specific changes but may leave systematic deviations in the overall prediction level or trend. The Global Context Expert π^G therefore reasons over the forecasting state after local revision. Unlike the text-based retrieval of the Local Context Expert, it retrieves historical windows from $\mathcal{M}^G$ according to the numerical similarity of their normalized look-back patterns. The expert compares the retrieved candidates and selects the most informative historical window. It then combines the selected experience with the locally revised forecast and a compact summary of $\widehat{\mathcal{A}}_i^L$, such as the number and magnitude of upward and downward adjustments.

The Global Context Expert adopts the same rationale-before-action protocol, first explaining the window-level evidence z_i^G within a `<think>` block and then producing the level and trend corrections $\hat{a}_i^G = \left\langle \Delta\hat{\mu}_i, \Delta\hat{\beta}_i \right\rangle$, which specifies the proposed level and trend corrections. The resulting reasoning trace z_i^G and Global Context Action are generated as:

$$\left(z_i^G, \hat{a}_i^G\right) \sim \pi^G \left(\cdot \mid \mathbf{X}_i, \widehat{\mathbf{Y}}_i^L, \text{Summary}\left(\widehat{\mathcal{A}}_i^L\right), \bar{a}_i^G\right), \tag{4}$$

where $\bar{a}_i^G$ denotes the selected global context memories from $\mathcal{M}^G$ and $\hat{a}_i^G$ specifies the overall level and trend corrections generated by Global Context Expert π^G. The Global Context Expert subsequently proposes $\hat{a}_i^G$ based on $\widehat{\mathbf{Y}}_i^1$, yielding a candidate global revision

$$\widehat{\mathbf{Y}}_i = \mathcal{R}^G \left(\widehat{\mathbf{Y}}_i^L, \hat{a}_i^G\right). \tag{5}$$

Here, $\mathcal{R}^G$ applies the predicted level correction to the complete locally revised forecast and distributes the trend correction linearly across the forecasting horizon. If the Global Context Expert selects NO-ACTION, the local prediction is retained, *i.e.,* $\widehat{\mathbf{Y}}_i = \widehat{\mathbf{Y}}_i^L$. The two experts are trained independently but operate sequentially, allowing the Global Context Expert to calibrate the systematic deviations remaining after localized context revision. The global prompt is provided in Appendix A.2; prompt constraints and information boundaries are detailed in Appendix A.3.

2.5 OPTIMIZATION

2.5.1 SUPERVISED FINE-TUNING

We initialize the Local and Global Context Experts through LoRA-based Hu et al. (2022) supervised fine-tuning of pretrained LLM backbones. The context memories store only residual-grounded context–action pairs and do not retain reasoning traces. During training-data construction, a teacher

LLM generates a rationale for each reference action based on the corresponding context and action label.

The teacher-generated rationale is enclosed by the <think> and </think> tags and placed before the structured action. The acceptance label is determined by whether applying the reference action improves the corresponding historical forecast. The Local Expert therefore learns to jointly generate a rationale, a reference Local Action Set, and its unified acceptance decision, while the Global Expert learns to generate a rationale, a level-and-trend action, and its acceptance decision. The two experts are trained independently using their respective memories.

2.5.2 GRPO-BASED JOINT OPTIMIZATION

Supervised imitation does not guarantee that the selected experience is informative or that the resulting action improves forecasting accuracy. We therefore further optimize each expert and its decision trajectory using Group Relative Policy Optimization (GRPO) Shao et al. (2024). For each training instance, the policy samples a group of complete trajectories, each consisting of the selected historical experience, a reasoning trace enclosed by the <think> tags, and the resulting action.

Each trajectory is evaluated by:

$$R = \lambda_{\text{fmt}} R_{\text{fmt}} + \lambda_{\text{sel}} R_{\text{sel}} + \lambda_{\text{pred}} R_{\text{pred}}, \tag{6}$$

where R_{fmt} verifies the validity of the reasoning, action, and decision formats. The selection reward encourages the policy to choose historical experiences with similar forecasting effects. For a historical case, we summarize its residual using

$$\mathbf{v} = \left[\text{Mean}(\boldsymbol{\epsilon}), \text{Slope}(\boldsymbol{\epsilon}), \text{Std}(\boldsymbol{\epsilon})\right], \tag{7}$$

and compute R_{sel} from the similarity between the impact vectors of the current instance and the selected experience.

The forecasting reward directly measures the relative error reduction after executing the decision:

$$R_{\text{pred}} = \frac{\text{MAE}\left(\mathbf{Y}_i, \widehat{\mathbf{Y}}_i^{\text{before}}\right) - \text{MAE}\left(\mathbf{Y}_i, \widehat{\mathbf{Y}}_i^{\text{after}}\right)}{\text{MAE}\left(\mathbf{Y}_i, \widehat{\mathbf{Y}}_i^{\text{before}}\right) + \varepsilon}. \tag{8}$$

For the local stage, $\widehat{\mathbf{Y}}_i^{\text{before}} = \widehat{\mathbf{Y}}_i^0$ and $\widehat{\mathbf{Y}}_i^{\text{after}} = \widehat{\mathbf{Y}}_i^L$; for the global stage, they correspond to the predictions before and after global calibration. The residual-derived impact vectors and ground-truth observations are used only for offline optimization and are unavailable during inference.

In this way, GRPO jointly aligns experience selection, contextual reasoning, and structured action generation, while the Local and Global Experts remain independently optimized. GRPO training dynamics are presented in Appendix B. The GRPO hyperparameters are reported in Table 13.

3 EXPERIMENT

3.1 EXPERIMENTAL SETUP

Datasets We evaluate Context2Action on four context-rich forecasting datasets, *i.e.,* M5 Makridakis et al. (2022), Electricity Godahewa et al. (2021), NYC Taxi Rodrigues et al. (2019), and Bitcoin Godahewa et al. (2021), covering retail, energy, transportation, and financial markets. Detailed dataset statistics and context information are provided in Appendix C.1.

Baselines We compare Context2Action with two groups of baselines. *Deep forecasting models* mainly learn temporal patterns from historical observations, including iTransformer Liu et al. (2024), DLinear Zeng et al. (2023), PatchTST Nie et al. (2023), and Autoformer Wu et al. (2021). *LLM-based and context-aware models* incorporate external events or contextual information into forecasting, including Chronos Ansari et al. (2024), AlphaCast Li et al. (2026), Time-LLM Jin et al. (2024), TSS Zhao et al. (2025), EventTSF Ge et al. (2025), IGSTGNN Li et al. (2024), and FNF Wang et al. (2024). Baseline details are provided in Appendix C.2.

Table 1: Overall forecasting performance on four context-rich datasets. The best and second-best results are highlighted in **bold** and underlined, respectively. *Impr.* denotes the relative error reduction over the strongest baseline.

Method	M5				Electricity				NYC Taxi				Bitcoin			
	MAE ($\times 10^3$)	MSE ($\times 10^6$)	RMSE ($\times 10^3$)	WMAPE (%)	MAE ($\times 10^2$)	MSE ($\times 10^4$)	RMSE ($\times 10^2$)	WMAPE (%)	MAE	MSE ($\times 10^2$)	RMSE ($\times 10^1$)	WMAPE (%)	MAE ($\times 10^3$)	MSE ($\times 10^7$)	RMSE ($\times 10^3$)	WMAPE (%)
Deep Forecasting Models																
iTransformer	1.551	6.336	2.517	11.47	1.898	9.538	3.088	4.31	6.202	1.615	1.271	48.30	4.246	2.684	5.182	7.61
DLinear	1.527	6.201	2.490	11.30	2.083	11.023	3.320	4.73	6.260	1.600	1.265	45.63	5.909	6.508	7.354	9.12
PatchTST	1.499	5.826	2.414	11.09	2.426	14.558	3.815	5.51	6.680	1.548	1.244	48.80	3.414	1.858	4.310	6.90
Autoformer	1.685	7.232	2.689	12.46	2.615	15.924	3.990	5.94	7.510	1.918	1.385	54.64	3.008	1.671	4.088	6.11
TimesNet	1.382	4.652	2.156	10.08	2.720	18.901	4.347	6.18	6.050	1.608	1.268	44.07	4.082	2.579	5.078	8.30
LLM-based and Context-Aware Models																
Chronos	1.582	6.414	2.533	11.71	2.897	19.800	4.450	6.58	6.050	1.658	1.288	44.06	3.717	2.661	5.159	7.55
Time-LLM	1.525	6.062	2.462	11.36	2.220	12.372	3.517	5.04	6.498	1.551	1.245	47.31	3.234	1.842	4.280	6.57
TSS	1.426	5.196	2.279	10.55	2.052	10.714	3.273	4.66	5.990	1.540	1.241	43.59	3.089	1.601	4.001	6.28
EventTSF	2.981	19.853	4.456	22.06	3.514	26.533	5.151	7.98	8.120	1.900	1.378	59.12	3.270	1.838	4.288	6.65
IGSTGNN	1.640	7.261	2.694	12.03	3.215	21.433	4.629	7.75	6.16	1.760	1.327	44.87	2.993	1.508	3.883	6.08
FNF	1.768	8.431	2.903	13.37	2.167	13.113	3.621	4.92	10.88	3.425	1.851	79.22	3.052	1.558	3.948	6.21
AlphaCast	1.382	5.325	2.308	10.10	1.957	9.824	3.134	4.44	6.240	1.561	1.249	45.40	2.729	1.242	3.524	5.58
Context2Action	**1.055**	**3.133**	**1.770**	**7.81**	**1.633**	**7.553**	**2.748**	**3.71**	**5.945**	**1.518**	**1.232**	**43.28**	**2.687**	**1.191**	**3.452**	**5.45**
Std.	±0.012	±0.072	±0.020	±0.04	±0.018	±0.170	±0.031	±0.04	±0.063	±0.093	±0.037	±0.21	±0.041	±0.042	±0.060	±0.09
Impr. (%)	↑ 23.66	↑ 32.65	↑ 17.90	↑ 22.52	↑ 13.96	↑ 20.81	↑ 11.01	↑ 13.92	↑ 0.75	↑ 1.43	↑ 0.73	↑ 0.71	↑ 1.54	↑ 4.11	↑ 2.04	↑ 2.33

Implementation Details We report MAE, MSE, RMSE, and WMAPE. The lookback and forecasting horizons for M5, Electricity, NYC Taxi, and Bitcoin are $(56, 28)$, $(336, 48)$, $(168, 24)$, and $(28, 7)$, respectively. All experiments use univariate forecasting. Deep forecasting baselines receive context as discrete features when supported, and we report their better results with or without these features; context-aware baselines follow their original context-encoding designs.

For Context2Action, the numerical backbone is selected independently for each dataset based on validation MAE before memory construction and expert training, using no test information. This selects PatchTST for M5, iTransformer for Electricity and NYC Taxi, and persistence for Bitcoin. The selected backbone remains frozen thereafter. Further details on the evaluation protocol and expert training are provided in Appendices C.3 and C.4, respectively; Table 12 reports the computational and token budgets.

3.2 Overall Experimental Results

Table 1 compares Context2Action with representative numerical and context-aware forecasting methods using MAE, MSE, RMSE, and WMAPE. ***(1)*** *Context can provide information complementary to historical observations.* On M5, AlphaCast ties with TimesNet for the lowest baseline MAE, while TimesNet performs better on the other metrics. iTransformer is the strongest baseline on Electricity, and context-aware methods lead several metrics on Bitcoin. These results suggest that external contexts can provide additional predictive signals beyond numerical histories. ***(2)*** *Effective use of context depends on how it is translated into forecast revisions.* Context-aware baselines vary considerably across datasets, and some underperform numerical predictors despite using additional information. Contextual inputs are therefore useful only when their direction, affected interval, and magnitude are properly identified. ***(3)*** *Context2Action achieves its largest gains on M5 and Electricity.* It obtains the best results across all metrics on both datasets, reducing MAE, MSE, RMSE, and WMAPE by 23.66%, 32.65%, 17.90%, and 22.52% on M5 and by 13.96%, 20.81%, 11.01%, and 13.92% on Electricity. The larger MSE reductions suggest that structured revision is particularly helpful for correcting large deviations. ***(4)*** *The improvement depends on the structure and reliability of the available contexts.* The MAE gains decrease from 23.66% on M5 and 13.96% on Electricity to 0.75% on NYC Taxi and 1.54% on Bitcoin. Discrete and recurrent contexts provide stronger evidence for retrieving relevant experiences and generating revisions, whereas sparse or weakly related contexts often result in smaller adjustments or NO-ACTION. Nevertheless, Context2Action achieves the best results across all four metrics on Bitcoin, with relative gains of 1.54%–4.11%.

3.3 Ablation Study

We evaluate the effects of expert optimization, hierarchical revision, and knowledge retrieval. The **Base Predictor** denotes a numerical deep forecasting model pretrained on each dataset and kept

Table 2: Ablation results in terms of MAE. The best and second-best results are marked in **bold** and underlined.

Variant	M5 ($\times 10^3$)	Electricity ($\times 10^2$)	NYC Taxi	Bitcoin ($\times 10^3$)
Base Predictor	1.499	1.898	6.202	3.028
SFT-only	1.442	2.187	6.157	2.954
Local Expert Only	1.137	1.799	5.968	2.710
Global Expert Only	1.517	1.734	6.196	2.735
Random Retrieval	1.245	1.854	5.997	3.175
Top-1 Retrieval	1.137	1.748	5.958	2.776
Context2Action	**1.055**	**1.633**	**5.945**	**2.687**

Table 3: Transferability across numerical predictors on M5. PatchTST-trained experts are transferred to other pretrained predictors without retraining. *Impr.* denotes relative MAE reduction.

Predictor	Stage	MAE ($\times 10^3$)	MSE ($\times 10^6$)	RMSE ($\times 10^3$)	WMAPE (%)	*Impr.* (%)
iTransformer	Base Predictor	1.551	6.336	2.517	11.47	–
	+ Local Expert	**1.260**	**4.569**	**2.138**	**9.32**	**18.77**
	+ Global Expert	1.277	4.735	2.176	9.45	17.67
DLinear	Base Predictor	1.527	6.201	2.490	11.30	–
	+ Local Expert	1.259	4.414	2.101	9.31	17.57
	+ Global Expert	**1.223**	**4.158**	**2.039**	**9.05**	**19.91**
Autoformer	Base Predictor	1.685	7.232	2.689	12.46	–
	+ Local Expert	1.332	4.883	2.210	9.85	20.94
	+ Global Expert	**1.304**	**4.696**	**2.167**	**9.65**	**22.60**

frozen throughout the context-aware revision process. **SFT-only** removes GRPO while retaining the complete dual-expert framework. **Local Expert Only** removes the Global Context Expert, whereas **Global Expert Only** removes the Local Context Expert. **Random Retrieval** randomly samples candidate cases, and **Top-1 Retrieval** directly uses the nearest case without candidate selection.

(1) Forecasting-oriented GRPO is essential for effective expert optimization. SFT-only is consistently inferior to Context2Action and even degrades the Electricity forecast. This indicates that SFT learns the basic context-to-action mapping, whereas GRPO further aligns knowledge selection and action generation with forecasting utility. ***(2) The two experts provide effective context-specific revisions.*** The Local Expert Only variant substantially improves the base predictor across all four datasets, validating context-specific local revision. The complete model further reduces MAE, showing that the Global Context Expert captures complementary level and trend deviations that remain after local revision. ***(3) Relevant context retrieval provides useful historical knowledge.*** Random Retrieval consistently performs worse than the complete model and even harms performance on Bitcoin. Relevant event- or time-series-based retrieval therefore provides useful historical knowledge rather than merely additional prompt content. ***(4) Multi-candidate retrieval and LLM-based knowledge selection are more effective than directly using the nearest case.*** Top-1 Retrieval is inferior to Context2Action on all datasets. Retrieving multiple candidates allows the expert to compare historical cases, while GRPO-guided knowledge selection helps identify and exploit the most informative evidence for forecast revision.

3.4 Transferability across Numerical Predictors

Although the main experiments use a validation-selected backbone for each dataset, we further examine whether the learned context-to-action capability depends on the numerical predictor. The experts trained with PatchTST are directly applied to iTransformer, DLinear, and Autoformer forecasts, while keeping the context memories, expert backbone, and parameters fixed without additional training or adaptation. Table 3 supports the following observations.

(1) The Local Context Expert transfers consistently across numerical predictors. It reduces MAE by 18.77%, 17.57%, and 20.94% over iTransformer, DLinear, and Autoformer, respectively, while improving all reported metrics. This indicates that the learned context-impact reasoning is not tied to the predictor used during training. ***(2) Context2Action retains substantial gains without retraining.*** After sequential local and global revisions, the final MAE improvements range from 17.67% to 22.60%. Similar reductions are observed in MSE, RMSE, and WMAPE, demonstrating that Context2Action can serve as a transferable revision layer for different forecasting architectures. ***(3) Global calibration is more sensitive to the underlying forecast state.*** It further improves DLinear and Autoformer, but slightly degrades the locally revised iTransformer result. This suggests that local context actions transfer more consistently, whereas global level and trend calibration depends more strongly on the numerical characteristics of the base forecast.

3.5 LLM Backbone Analysis

We examine the sensitivity of Context2Action to the LLM backbone used by both Context Experts.

We replace Qwen3.5-9B with Qwen3-4B and Llama-Nemotron-Nano-8B while keeping the numerical predictor, context memories, training procedure, and all other settings unchanged. PatchTST without the Context Experts is included as the basic predictor.

Table 4: Performance with different LLM backbones on M5. The same backbone is used for both context experts. PatchTST is the numerical predictor without expert reasoning.

Configuration	MAE ($\times 10^3$)	MSE ($\times 10^6$)	RMSE ($\times 10^3$)	WMAPE (%)
PatchTST (Base Only)	1.499	5.826	2.414	11.09
Qwen3-4B	1.236	4.239	2.059	9.14
Llama-Nemotron-Nano-8B	<u>1.216</u>	<u>3.989</u>	<u>1.997</u>	<u>9.00</u>
Qwen3.5-9B (Context2Action)	**1.055**	**3.133**	**1.770**	**7.81**

As shown in Table 4, all three LLM backbones substantially improve the numerical-only prediction, demonstrating that the proposed context-to-action mechanism is not restricted to a specific LLM. Qwen3.5-9B achieves the best performance, reducing MAE by 29.62% over PatchTST. The smaller Qwen3-4B and Llama-Nemotron-Nano-8B also reduce MAE by 17.55% and 18.88%, respectively. These results suggest that the choice of LLM backbone affects contextual action generation, while the smaller backbones still provide clear improvements over the numerical predictor.

4 Related Works

4.1 Time Series Forecasting

Deep time series forecasting methods primarily improve the modeling of temporal patterns from historical observations. Autoformer introduces progressive decomposition and an auto-correlation mechanism for long-term forecasting Wu et al. (2021), while TimesNet captures multi-period temporal variations by transforming one-dimensional sequences into two-dimensional representations Wu et al. (2023). PatchTST improves long-range modeling through channel-independent patch representations Nie et al. (2023), and iTransformer represents entire variables as tokens to capture cross-variable dependencies Liu et al. (2024). Meanwhile, DLinear demonstrates that simple linear models can remain highly competitive for long-term forecasting Zeng et al. (2023). Despite their different architectures, these methods mainly extrapolate numerical patterns from historical observations and predefined covariates. In contrast, Context2Action complements numerical predictors by explicitly reasoning over open-world contexts and translating them into structured forecasting actions.

4.2 LLM-based and Context-aware Time Series Forecasting

Recent studies have introduced LLMs and real-world information into time series forecasting. Chronos models numerical sequences as discrete tokens, while Time-LLM reprograms time-series representations into the embedding space of a frozen LLM Ansari et al. (2024); Jin et al. (2024). Other methods explicitly incorporate external contexts: From News to Forecast analyzes news through LLM-based reflection Wang et al. (2024); IGSTGNN models incident impacts in traffic forecasting Li et al. (2024); EventTSF integrates textual events with temporal observations Ge et al. (2025); AlphaCast enables human–LLM collaborative reasoning Li et al. (2026); and TimeSeriesScientist organizes forecasting through a general-purpose agent workflow Zhao et al. (2025). Unlike these methods, Context2Action grounds contextual effects in historical forecasting errors and translates them into explicit local and global actions, enabling controllable and interpretable context-aware forecasting while preserving the numerical predictor.

5 Conclusion

In this work, we introduced Context2Action, an agentic framework for real-world-context-aware time-series forecasting. The framework grounds contextual understanding in historical forecasting errors, organizes reusable local and global context experiences, and translates them into explicit forecasting actions through retrieval-augmented LLM experts. A hierarchical decision mechanism further evaluates the proposed actions to avoid unsupported revisions. Experiments on four real-world datasets demonstrate improvements over numerical, LLM-based, and context-aware forecasting baselines, while additional analyses verify the effectiveness of the dual memories, structured actions, retrieval strategy, and forecasting-oriented optimization.

AI USE STATEMENT

Generative AI tools assisted with code review, figure preparation, and manuscript editing. They were also used to generate rationale supervision and to implement the LLM-based context experts described in the paper. The authors reviewed all generated content and take full responsibility for the final manuscript, implementation, and results.

REPRODUCIBILITY STATEMENT

We have uploaded the experimental code in the Supplementary Material. Appendices A–D specify the expert interactions, information boundaries, optimization procedure, datasets, chronological splits, baseline implementations, backbone selection, training configurations, and computational budgets. Reproduction requires the benchmark datasets and pretrained LLM, embedding, and numerical forecasting checkpoints identified in the paper and Supplementary Material.

REFERENCES

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

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. *arXiv preprint arXiv:2310.10688*, 2024.

Yunfeng Ge, Ming Jin, Yiji Zhao, Hongyan Li, Bo Du, Chang Xu, and Shirui Pan. EventTSF: Event-aware non-stationary time series forecasting. *arXiv preprint arXiv:2508.13434*, 2025. URL https://arxiv.org/abs/2508.13434.

Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I Webb, Rob J Hyndman, and Pablo Montero-Manso. Monash time series forecasting archive. *arXiv preprint arXiv:2105.06643*, 2021.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shun Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. In *International Conference on Machine Learning*, 2024.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew Gordon Wilson. Large language models are zero-shot time series forecasters. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0861. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/3eb7ca52e8207697361b2c0fb3926511-Abstract-Conference.html.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net, 2022.

Yushan Jiang, Wenchao Yu, Geon Lee, Dongjin Song, Kijung Shin, Wei Cheng, Yanchi Liu, and Haifeng Chen. TimeXL: Explainable multi-modal time series prediction with LLM-in-the-loop. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/c5433ab4056ca58db67be4578c384cba-Abstract-Conference.html.

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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Unb5CVPtae.

Yuxin Li, Xinyu Zhang, Yilun Wang, Chao Chen, et al. Incident-guided spatiotemporal traffic forecasting. *arXiv preprint arXiv:2410.19688*, 2024. URL https://arxiv.org/abs/2410.19688.

Zhe Li et al. Alphacast: Knowledge retrieval-augmented large language models for time series forecasting. In *International Conference on Learning Representations*, 2026.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=JePfAI8fah.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. M5 accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting*, 38(4):1346–1364, 2022.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Jbdc0vTOcol.

Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-beats: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020.

Filipe Rodrigues, Ioulia Markou, and Francisco C Pereira. Combining time-series and textual data for taxi demand prediction in event areas: A deep learning approach. *Information Fusion*, 49: 120–129, 2019.

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

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in LLM-based time series forecasting with reflection. *arXiv preprint arXiv:2409.17515*, 2024. URL https://arxiv.org/abs/2409.17515.

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*, volume 34, pp. 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. In *ICLR*, 2023.

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.

Haokun Zhao, Xiang Zhang, Jiaqi Wei, Yiwei Xu, Yuting He, Siqi Sun, and Chenyu You. Timeseriesscientist: A general-purpose ai agent for time series analysis, 2025. URL https://arxiv.org/abs/2510.01538.

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.

APPENDIX

A PROMPT TEMPLATES AND EXPERT INTERACTION

This section presents the prompt design and structured interaction used by the two context experts. In our implementation, the Local Context Expert and Global Context Expert are referred to as the *Local Expert* and *Global Expert*, respectively. Both experts follow a rationale-before-action protocol: they first generate an evidence-grounded explanation enclosed by the `<think>` and `</think>` tags and then output an executable forecasting action under a fixed schema.

The prompts are designed around three principles. First, historical evidence selection is separated from action generation, allowing the selected supporting cases to be explicitly inspected. Second, the experts reason over structured forecasting states rather than directly regenerating future values. Third, strict output schemas make the resulting actions directly parseable and applicable to the numerical forecast.

A.1 LOCAL CONTEXT EXPERT PROMPT

The Local Context Expert identifies context-specific effects over localized forecasting intervals. Its implementation follows a two-stage AgentLoop consisting of historical evidence selection and joint context-impact reasoning.

Stage 1: Historical Evidence Selection For each visible non-recurring context, the retriever first returns at most five semantically related historical cases. Each candidate contains its context description, semantic similarity, and residual pattern aligned relative to the historical context timestamp. The Local Context Expert then selects one candidate for each current context.

The first-stage Prompt is deliberately constrained to evidence selection and does not request an explanation or forecasting action. This separates the retrieval decision from subsequent contextual reasoning.

Local Context Expert: Stage-1 Evidence Selection

```
You are selecting historical recall evidence for a
time-series context adjustment task.

Choose the one candidate whose context description and
aligned residual pattern best match each target context.

Consider:
1. semantic mechanism and affected entity;
2. expected impact direction;
3. affected temporal interval;
4. residual-pattern magnitude and duration.

Do not explain.
Output only the SELECTED_RECALLS table.

TARGET_CONTEXTS
event_slot | context_timestamp | context_description
E1 | <timestamp> | <target context 1>
E2 | <timestamp> | <target context 2>
...

TOP5_RECALL_CANDIDATES

[E1]
rank | context_description | similarity | aligned_residuals
1 | <historical context> | <score> |
    d-3:<value>, ..., d+7:<value>
...
5 | <historical context> | <score> |
```

Table 5: Inputs provided to the Local Context Expert after evidence selection.

Input component	Description
Current contexts	Context descriptions, timestamps, affected series, and observable event attributes.
Base prediction shape	Relative shape of the frozen numerical forecast over the current horizon.
Selected historical evidence	Residual patterns from the historical cases selected in Stage 1 and aligned to the current context dates.
Recent context pressure	Summary of upward, downward, and context-free days in the look-back window.
Task-specific indicators	Structured context information such as SNAP activity or scheduled event times.

```
    d-3:<value>, ..., d+7:<value>

[E2]
...

OUTPUT FORMAT

SELECTED_RECALLS
event_slot | selected_rank
E1 | <rank>
E2 | <rank>
...
```

For the representative M5 instance, the Local Context Expert selects the first candidate for each of the four non-SNAP contexts:

Representative Stage-1 Output

```
SELECTED_RECALLS
event_slot | selected_rank
E1 | 1
E2 | 1
E3 | 1
E4 | 1
```

Environment-Aligned Evidence Injection After evidence selection, the environment aligns the residual pattern of each selected historical case with the timestamp of the corresponding current context. These aligned values are injected into a forecasting-window table together with the base prediction shape and the visible contexts.

The resulting second-stage input contains the following information:

The aligned historical residuals serve as empirical evidence rather than direct action labels. The expert may reduce or override their influence when the semantics, current prediction shape, or co-occurring contexts suggest that direct transfer is inappropriate.

Stage 2: Joint Context Reasoning and Action Generation All current contexts are analyzed jointly in a single invocation. The Prompt requires the expert to reason about three aspects in order:

1. the semantic mechanism and expected direction of each context;
2. the affected temporal interval based on context meaning and aligned historical evidence;
3. the adjustment magnitude after considering the base prediction shape, retrieved evidence, recent context pressure, and interactions among multiple contexts.

The output must contain one structured action for every visible context.

Local Context Expert: Stage-2 Rationale and Action Prompt

```
You are a time-series local context adjustment expert.

A frozen numerical model has already produced the base
prediction over the current forecasting horizon. Historical
recall values are relative forecasting residuals from previously
observed, semantically similar contexts.

Infer how each visible context should adjust the numerical
forecast using:
- the context semantics and timestamp;
- the selected historical residual evidence;
- the current numerical prediction shape;
- recent context pressure in the look-back window;
- interactions among overlapping contexts.

Historical residuals are evidence rather than fixed templates.
Do not mechanically copy them when the current prediction
state or co-occurring contexts indicate a different adjustment.

Inside <think>...</think>, write three compact paragraphs:

Paragraph 1:
Analyze the semantic mechanism and expected impact
direction of every visible context.

Paragraph 2:
Determine the impact start time and duration using the
context meaning and the aligned historical residual evidence.

Paragraph 3:
Determine the signed adjustment magnitude after jointly
considering the prediction shape, selected evidence, recent
context pressure, and interactions among contexts.

OUTPUT REQUIREMENTS

1. Output one complete <think>...</think> block.
2. Output exactly one EVENT_EFFECT_TABLE after </think>.
3. Use exactly these columns:

event_name | effect_start_date |
estimated_duration_days | average_impact_pct

4. effect_start_date must be a valid timestamp.
5. estimated_duration_days must be a positive integer.
6. average_impact_pct must be a signed percentage.
7. Output one row for every required context.
8. Do not output any other table.

REQUIRED_CONTEXTS
<context names, timestamps, and descriptions>

LOOKBACK_CONTEXT_PRESSURE_SUMMARY
<upward/downward/context-free summary>

TARGET_FORECASTING_INSTANCE
- series: <series identity>
- prediction_window_start: <timestamp>
- forecasting_horizon: <H>

FORECASTING_WINDOW_TABLE
time_offset | timestamp | base_prediction_shape |
visible_context | selected_recall_residuals |
task_specific_context
```

Table 6: Representative Local Context Actions generated for an M5 forecasting window.

Context	Effect Start	Duration	Signed Adjustment
ValentinesDay	2016-02-15	1 day	$+1.1\%$
PresidentsDay	2016-02-16	1 day	$+1.1\%$
Warehouse processing delay	2016-02-23	7 days	-11.7%
Pricing-file outage	2016-03-05	1 day	-11.0%
SNAP count 1	SNAP active day	1 day	-5.0%
SNAP count 2	SNAP active day	1 day	-11.5%
SNAP count 3	SNAP active day	1 day	-10.6%

...

The corresponding machine-readable action schema is:

Local Context Action Schema

```
<think>
<context-impact rationale>
</think>
EVENT_EFFECT_TABLE
event_name | effect_start_date |
estimated_duration_days | average_impact_pct
<context> | <timestamp> | <duration> | <signed percentage>
```

This schema directly instantiates the Local Context Action

$$\hat{a}^L_{i,j} = \left\langle \hat{s}_{i,j}, \hat{d}_{i,j}, \hat{\alpha}_{i,j} \right\rangle, \tag{9}$$

where $\hat{s}_{i,j}$, $\hat{d}_{i,j}$, and $\hat{\alpha}_{i,j}$ denote the predicted impact start time, duration, and signed magnitude, respectively.

Representative Local Context Output Table 6 shows the structured Local Context Actions generated for an event-dense M5 forecasting window. The instance contains cultural holidays, a warehouse processing disruption, a pricing-file outage, and multiple SNAP conditions.

A representative rationale generated before these actions can be summarized as follows:

Representative Local Rationale The expert first identifies the holidays as weak positive factors whose effects may shift to the following day. It interprets the warehouse interruption and pricing-file outage as negative operational constraints. It then aligns the selected historical residuals to the current event dates, using the sustained negative residuals to assign a multi-day warehouse effect and a single-day pricing-system effect. Finally, it calibrates the magnitudes jointly with the base forecast shape, recent context pressure, SNAP indicators, and overlaps among the visible contexts.

A.2 Global Context Expert Prompt

The Global Context Expert operates after the local actions have been applied or rejected. Its purpose is not to reinterpret individual contexts, but to determine whether the resulting forecast still exhibits a systematic level or trend mismatch.

Unlike the semantic retrieval used for the Local Context Expert, the Global Context Expert retrieves complete historical forecasting windows according to the numerical similarity of their normalized look-back patterns. Every retrieved candidate remains an independent real historical window; candidates are not averaged, blended, or converted into a synthetic trajectory.

Global Prompt Inputs The Global Context Expert receives the following information:

Table 7: Inputs provided to the Global Context Expert.

Input component	Description
Locally revised forecast	Forecasting state after applying or rejecting the Local Context Actions.
Recent rolling windows	Level and trend statistics of recent historical windows relative to the current forecast.
Retrieved historical windows	Five real historical windows with similarity scores and observed level/trend corrections.
Local-action summary	Number and magnitude of upward, downward, and unchanged forecasting steps.
Look-back context pressure	Summary of historical upward and downward contextual effects preceding the current window.

All level and trend quantities are represented as percentages relative to the mean of the locally revised forecast. The level action specifies an overall shift, while the trend action specifies the total linear change from the beginning to the end of the forecasting horizon.

Global Rationale and Action Prompt

Global Context Expert: Level-and-Trend Calibration Prompt

```
You are a time-series global calibration expert.

The Local Context Expert has already adjusted the forecast
using visible contexts and historical context evidence.

Your task is to determine whether the overall level and
linear trend of the current forecasting horizon require a
second calibration.

All percentages are represented relative to the mean of the
current locally revised forecast:
- the current forecast mean is defined as 0%;
- mean_shift changes the overall prediction level;
- trend_slope represents the total linear percentage change
  from the first forecasting step to the last.

Analyze:
1. the look-back temporal pattern;
2. recent rolling-window level and trend statistics;
3. the five retrieved real historical windows;
4. the observed level and trend corrections in those windows;
5. the summary of actions generated by the Local Expert;
6. recent contextual pressure in the look-back window.

Every recalled window is a real historical instance.
Do not average or blend the recalled windows into a
synthetic trajectory. Compare them, select the most
informative evidence, or ignore unreliable candidates.

Inside <think>...</think>, write one compact paragraph that:
- describes the current level and trend state;
- identifies the most relevant recalled windows;
- explains how the Local Expert actions affect the remaining
  level and trend decision;
- justifies the final global calibration.

OUTPUT REQUIREMENTS

1. Output one complete <think>...</think> block.
```

Table 8: Representative Global Context Action.

Adjustment	Type	Magnitude
Mean shift	Level multiplier	-1.5%
Trend slope	Linear ramp	-12.0%

```
2. Output only TEMPORAL_ADJUSTMENT_TABLE after </think>.
3. Use exactly these columns:

adjustment_name | adjustment_type | adjustment_pct

4. Output exactly two rows:
   mean_shift | level_multiplier | <percentage>
   trend_slope | linear_ramp | <percentage>

TARGET_WINDOW
- series: <series identity>
- prediction_window_start: <timestamp>
- forecasting_horizon: <H>

RECENT_ROLLING_WINDOWS
<window-level mean and trend statistics>

TOP5_REAL_RECALL_WINDOWS
<similarity, historical mean shift, historical trend shift,
and context-action summaries>

LOCAL_CONTEXT_ACTION_SUMMARY
<upward/downward/unchanged forecasting steps>

LOOKBACK_CONTEXT_PRESSURE_SUMMARY
<historical contextual pressure>
```

The required output schema is:

Global Context Action Schema

```
<think>
<window-level rationale>
</think>
TEMPORAL_ADJUSTMENT_TABLE
adjustment_name | adjustment_type | adjustment_pct
mean_shift | level_multiplier | <signed percentage>
trend_slope | linear_ramp | <signed percentage>
```

This output corresponds to the Global Context Action

$$\hat{a}_i^G = \left\langle \Delta\hat{\mu}_i, \Delta\hat{\beta}_i \right\rangle, \tag{10}$$

where $\Delta\hat{\mu}_i$ controls the overall level shift and $\Delta\hat{\beta}_i$ controls the linear trend shift.

Representative Global Context Output For the same M5 instance, the Global Context Expert observes that most locally adjusted days receive negative revisions and that several similar historical windows exhibit negative trend errors. It therefore generates the following calibration:

Representative Global Rationale The recent rolling windows provide mixed level evidence but increasingly negative trend signals. Among the recalled historical windows, the most relevant cases exhibit negative level or trend corrections that are consistent with the current context-dense forecasting state. Because the Local Context Expert has already applied downward actions to most of the horizon, the Global Context Expert selects only a modest additional level reduction while retaining a clearer negative trend calibration.

A.3 Prompt Constraints and Information Boundaries

The prompt design explicitly separates information available during offline learning from information available during inference.

Historical evidence. Retrieved historical cases may contain forecasting residuals because their outcomes have already been observed. These residuals are used only as evidence describing how a previous numerical predictor behaved under a historical context.

Current forecasting instance. The experts receive the current historical observations, the frozen base or intermediate forecast, visible real-world contexts, and temporally valid historical retrieval results. They never receive the current ground-truth future sequence or current forecasting residual.

Temporal filtering. At test time, every retrieved context or historical window must precede the current forecasting origin. This prevents future forecasting outcomes from entering retrieval.

Rationale supervision. The `<think>` rationales are generated as intermediate supervision during expert training. They are not stored as entries in the Local or Global Context Memories.

Structured outputs. Both experts are required to produce fixed tables with predefined row and column schemas. The Local Context Expert outputs explicit start-time, duration, and signed-magnitude actions, while the Global Context Expert outputs exactly one level shift and one trend shift.

Invalid outputs. An output is considered invalid when the rationale block is incomplete, required contexts are missing, the structured table cannot be parsed, or action values violate the predefined schema. Invalid outputs are not applied to the numerical forecast and receive a negative format reward during GRPO optimization.

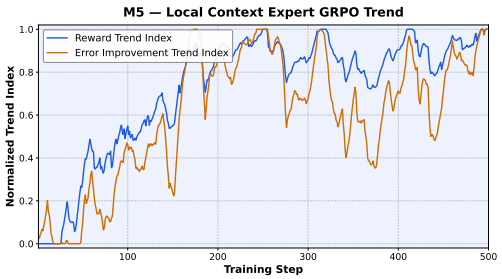
Together, these constraints ensure that the expert interaction remains evidence-grounded, temporally valid, auditable, and directly executable without allowing the LLM to overwrite the complete numerical forecasting sequence.

B GRPO Training Dynamics

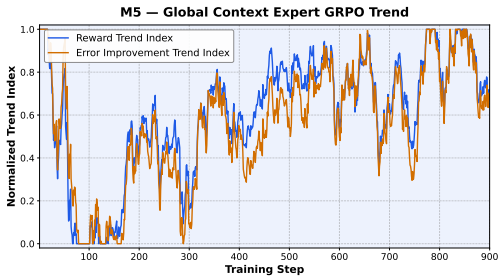
Figure 2 presents the GRPO training dynamics of the Local and Global Context Experts on the four datasets. For each expert, we report the normalized reward trend and the normalized forecasting-error-improvement trend. Each trajectory is independently normalized within its own training run; therefore, the curves reflect relative optimization progress rather than directly comparable reward magnitudes across datasets or experts.

Overall, both experts exhibit clear upward trends during training, although the trajectories are non-monotonic due to the stochastic sampling of retrieval, reasoning, and action trajectories. On M5, the Local Context Expert improves rapidly and remains at a relatively high reward level, while the Global Context Expert shows larger fluctuations but gradually achieves stronger forecasting utility. On Electricity, both experts display stable late-stage error improvements, indicating that GRPO effectively learns both context-specific and window-level forecasting actions. The NYC Taxi experts show comparatively smooth optimization, with reward and error-improvement trends approaching their upper ranges in the later stages. Bitcoin presents more pronounced intermediate fluctuations, but both experts recover and reach high normalized levels toward the end of training.

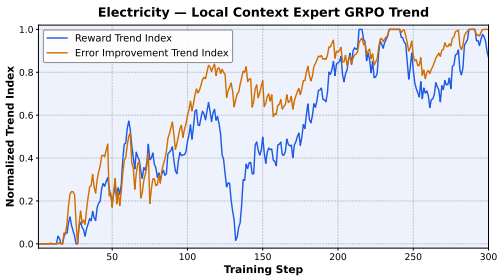
Across most runs, the reward and forecasting-error-improvement curves evolve consistently. This agreement indicates that the forecasting-oriented reward does not merely encourage valid output formats, but also guides the experts toward actions that improve downstream forecasting performance.



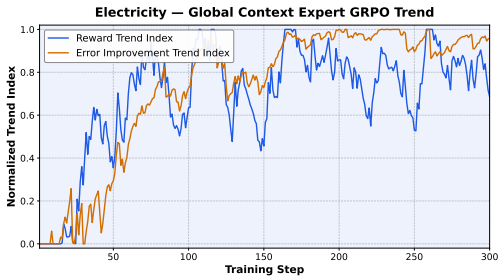
(a) M5: Local Context Expert.



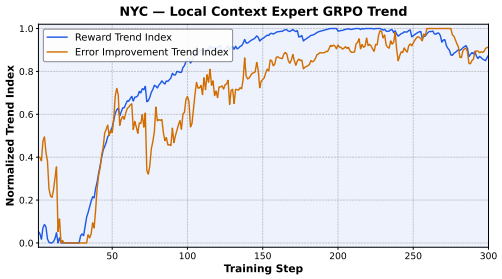
(b) M5: Global Context Expert.



(c) Electricity: Local Context Expert.



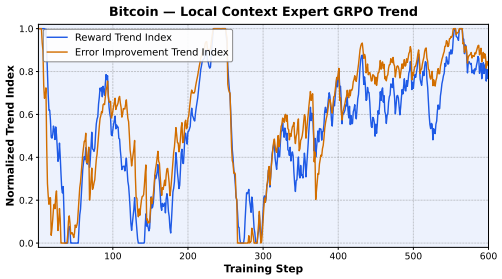
(d) Electricity: Global Context Expert.



(e) NYC Taxi: Local Context Expert.



(f) NYC Taxi: Global Context Expert.



(g) Bitcoin: Local Context Expert.



(h) Bitcoin: Global Context Expert.

Figure 2: Normalized GRPO training dynamics of the Local and Global Context Experts on four datasets. The reward and forecasting-error-improvement trajectories are independently normalized within each training run and therefore reflect relative optimization progress rather than directly comparable absolute values across datasets.

C Experimental Setups

C.1 Datasets and Context Information

We evaluate Context2Action on four context-rich forecasting datasets covering retail, energy, transportation, and financial markets. Table 9 summarizes the forecasting configurations and contextual

information. All datasets are chronologically split, and test-time retrieval is restricted to historical cases preceding the current forecasting origin.

Table 9: Dataset statistics and forecasting configurations.

Dataset	Frequency	Series	Lookback	Horizon	Real-World Contexts
M5	Daily	3 categories	56	28	Promotions, recalls, supply disruptions, weather, and holidays
Electricity	30 minutes	5 states	336	48	Weather, workday/holiday indicators, and public news
NYC Taxi	Hourly	1	168	24	Venue events, start times, event types, and cancellations
Bitcoin	Daily	1	28	7	Timestamped financial and cryptocurrency news

M5. M5 Makridakis et al. (2022) contains daily Walmart sales from stores in California, Texas, and Wisconsin. We aggregate the original item-level records into three category-level series, namely `FOODS`, `HOBBIES`, and `HOUSEHOLD`, covering 1,969 days. The first 1,729 days are used for model development, while the subsequent 240 days form 96 test instances.

To evaluate open-world contextual reasoning, we construct a controlled event-aware extension containing 25 retail event mechanisms and 250 independently described occurrences. The contexts cover promotions, product recalls, supply interruptions, payment failures, severe weather, and other realistic retail operations. Each mechanism appears nine times in training and once in testing with different textual descriptions and independently sampled effects. The injected events preserve the original sales structure: the global mean changes by only 0.05%, and the original and event-enhanced targets retain a Pearson correlation of 0.986. Retrieval queries contain only the product category and event description, excluding event identifiers, impact labels, targets, and residuals.

Electricity. The Electricity dataset Godahewa et al. (2021) contains half-hourly electricity demand from New South Wales, Queensland, South Australia, Tasmania, and Victoria. We use seven days of historical demand to forecast the subsequent day. Data from 2019–2020 are used for model development, and 100 target days from 2021 are used for testing. Contextual information includes weather conditions, workday and public-holiday indicators, and timestamped public news that may affect either the overall demand level or the intraday load pattern.

NYC Taxi. The NYC Taxi dataset Rodrigues et al. (2019) records hourly taxi demand together with events held at major venues. Each instance uses the previous seven days to predict demand over the following 24 hours, with the final 18 daily windows used for testing. Contexts include the event name, scheduled start time, event type, and cancellation status. These contexts mainly induce localized arrival and departure patterns around event hours.

Bitcoin. The Bitcoin dataset Godahewa et al. (2021) contains daily prices and cryptocurrency-related news from 2009 to 2021. We construct 309 training, 50 validation, and 128 test windows, using 28 historical days to predict the following seven days. Each context consists of a news title and a short cleaned excerpt covering factors such as institutional adoption, regulation, payment services, security incidents, and market infrastructure. Retrieval uses only the textual content and does not expose future prices, directions, impact intervals, or residuals.

C.2 Baseline Implementations

Numerical forecasting models. We include DLinear Zeng et al. (2023), PatchTST Nie et al. (2023), iTransformer Liu et al. (2024), Autoformer Wu et al. (2021), and TimesNet Wu et al. (2023). DLinear, PatchTST, iTransformer, and Autoformer are implemented under a common training protocol with two model layers, a hidden dimension of 256, a feed-forward dimension of 512, a batch size of 64, and early stopping. For Electricity, the architecture size and training epochs are reduced to accommodate the longer input sequence. On Bitcoin, numerical models predict log returns, which are accumulated and transformed back to the original price scale for evaluation.

Chronos. Chronos Ansari et al. (2024) is evaluated using the `chronos-t5-small` checkpoint in a zero-shot setting. Since the original model performs univariate forecasting, we additionally apply task-specific conditional scaling based on available structured contexts, such as calendar effects, weather conditions, event intensity, or recent volatility.

Table 10: Numerical predictors used by Context2Action.

Dataset	Predictor	Lookback	Horizon
M5	PatchTST	56	28
Electricity	iTransformer	336	48
NYC Taxi	iTransformer	168	24
Bitcoin	Persistence	28	7

Time-LLM. Time-LLM Jin et al. (2024) reprograms time-series representations into the embedding space of GPT-2. We use six GPT-2 layers with a hidden dimension of 128 and a feed-forward dimension of 512. The model is trained for up to 50 epochs using a learning rate of 10^{-4} and early stopping.

TimeSeriesScientist. TimeSeriesScientist Zhao et al. (2025) is implemented as a LangGraph-based multi-agent workflow that performs data analysis, model selection, numerical forecasting, and ensemble construction. Its final prediction aggregates multiple deep forecasting models using validation-guided ensemble weights.

EventTSF. EventTSF Ge et al. (2025) integrates temporal observations with contextual representations through an event-conditioned flow-matching architecture. We convert task-specific contexts into natural-language descriptions, encode them using Qwen3-4B, and project the pooled embeddings to a 128-dimensional conditional representation. The resulting model is trained for 200 epochs on each dataset.

IGSTGNN. IGSTGNN Li et al. (2024) models the temporal propagation and decay of incident effects through an incident-aware spatio-temporal architecture. We adapt its sequence length, forecasting horizon, and graph construction to each task while preserving its incident-conditioned modeling mechanism.

From News to Forecast. From News to Forecast (FNF) Wang et al. (2024) uses an LLM to interpret external information and generate forecasts through reflection. We implement it with Qwen3-4B and LoRA fine-tuning using rank 8 and scaling factor 16. The model is trained on instruction–input–output examples containing historical observations and corresponding contextual descriptions.

AlphaCast. AlphaCast Li et al. (2026) employs an Investigator–Generator–Reflector workflow for interactive forecasting. We use Qwen3-4B as its LLM backend and provide dataset-specific numerical and contextual covariates. Candidate numerical forecasts are combined using validation-error-based ensemble weights.

C.3 Evaluation Protocol

We report mean absolute error (MAE), mean squared error (MSE), root mean squared error (RMSE), and weighted mean absolute percentage error (WMAPE). All metrics are calculated after reversing normalization or target transformations and are reported in the original target scale. For Context2Action, the reported mean and standard deviation are calculated over 5 repeated runs.

For numerical forecasting baselines, structured contextual variables are included when supported by the corresponding architecture, and we report the better result obtained with or without these variables. LLM-based and context-aware baselines receive contextual information following their original input and prompting designs.

C.4 Implementation Details

Numerical predictors. We use pretrained PatchTST on M5, iTransformer on Electricity and NYC Taxi, and a persistence predictor on Bitcoin. These predictors generate the base forecasts and remain frozen throughout context-expert training and inference.

Table 11: Dataset-specific expert-training configurations. GRPO steps report the total number of optimization steps.

Dataset	SFT Samples	Local GRPO	Global GRPO
M5	400	500	500
Electricity	800	300	300
NYC Taxi	56	300	500
Bitcoin	351	300	300

Table 12: Dataset- and expert-specific runtime budgets and computational configurations. *Steps* denotes the number of GRPO optimization steps; the remaining columns specify the limits used when running each expert and thereby characterize its inference-time resource budget.

Dataset	Expert	Steps	Max prompt	Max response	Max context	Thinking budget	GPU memory utilization
M5	Local	500	6,400	7,000	10,500	900	0.45
M5	Global	500	6,400	7,000	10,500	900	0.45
Electricity	Local	300	4,096	1,500	5,852	1,200	0.55
Electricity	Global	300	4,096	1,500	6,144	1,200	0.55
NYC Taxi	Local	300	4,096	5,000	8,500	900	0.45
NYC Taxi	Global	500	4,096	1,600	6,500	900	0.45
Bitcoin	Local	300	6,400	9,000	16,000	1,500	0.35
Bitcoin	Global	300	6,400	7,000	10,500	900	0.45

LLM and retrieval modules. The Local and Global Context Experts use Qwen3.5-9B, while Qwen3.6-35B-A3B serves as the teacher model for constructing rationale–action supervision. Qwen3-Embedding-4B is used for semantic context retrieval. The Local Expert retrieves the top five textually related historical contexts, whereas the Global Expert retrieves five historical windows according to normalized temporal-pattern similarity. Retrieved cases are real historical instances and are not averaged or synthesized.

At test time, all retrieved cases must precede the current prediction origin. Retrieval inputs exclude ground-truth targets, forecasting residuals, impact magnitudes, and reference actions.

Expert training. Both experts are independently initialized through LoRA-based supervised fine-tuning and subsequently optimized with GRPO. We use LoRA rank 16 and scaling factor 32. The SFT learning rate is 1×10^{-5} for M5, NYC Taxi, and Bitcoin, and 2×10^{-5} for Electricity. GRPO uses eight rollout trajectories per instance, a learning rate of 5×10^{-6}, and a KL coefficient of 0.001. The complete set of shared GRPO hyperparameters is reported in Table 13.

The reward combines output-format validity, historical-case selection, and numerical forecasting improvement. Invalid structured outputs receive a negative reward, while the principal utility signal is determined by the MAE change after applying the generated action. Teacher-generated rationales are enclosed by the `<think>` and `</think>` tags and are used only for expert supervision rather than being stored in the context memories.

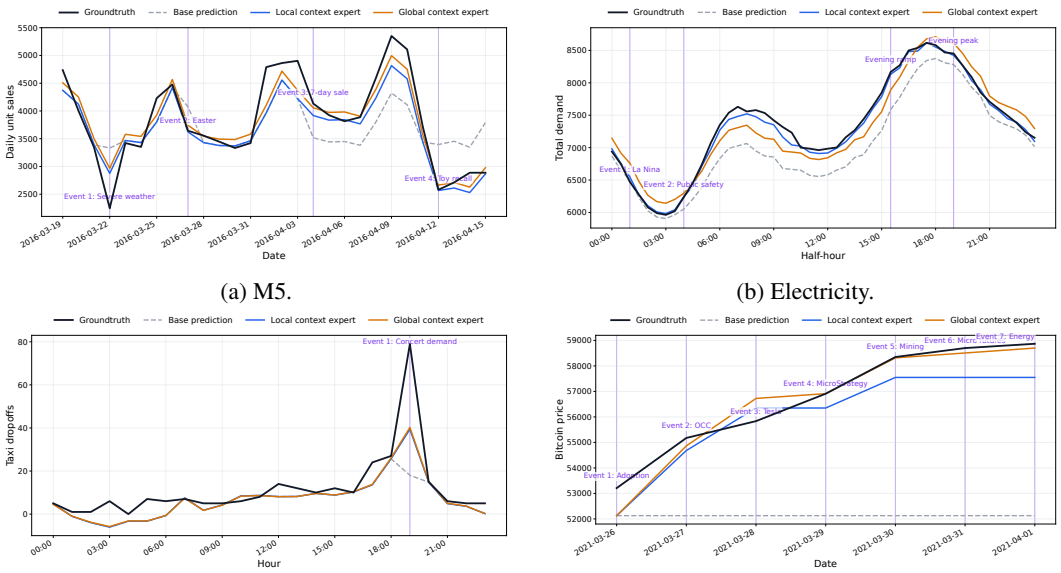
Inference. The Local Expert generates context-specific actions that specify the affected interval and signed adjustment magnitude. The Global Expert subsequently predicts an overall level shift and a linear trend correction. All forecasting actions are applied to the frozen numerical forecast in the original target scale. Ground-truth observations and residual-derived quantities are used only during offline memory construction and optimization and remain unavailable during inference. Table 12 reports the maximum prompt, response, context, and thinking-token budgets together with GPU memory utilization for each expert and dataset, providing a reproducible characterization of the configured inference resource cost.

D Case Studies

To better understand the behavior of Context2Action, we visualize its forecast correction process on a representative M5 sequence in Figure 4. The dashed gray, blue, orange, and black curves denote

Table 13: GRPO optimization hyperparameters shared across datasets and experts.

Hyperparameter	Value
Optimizer learning rate	5×10^{-6}
Rollout samples per prompt	8
Training batch size	1
PPO mini-batch size	1
PPO micro-batch size	1
KL coefficient	0.001
KL estimator	Low-variance KL
Entropy coefficient	0
Sampling temperature	1.0
Top-p	1.0
LoRA rank	16
LoRA alpha	32
LoRA target modules	Q/K/V/O and MLP projections
Precision	BF16
Gradient checkpointing	Enabled
Random seed	42
Tensor parallel size	1
Maximum concurrent sequences	8
Checkpoint strategy	Save at final step
Validation during training	Disabled
Number of GPUs	1



(a) M5.

(b) Electricity.

(c) NYC Taxi.

(d) Bitcoin.

Figure 3: Additional case studies on four datasets. The curves visualize the frozen numerical prediction and the sequential forecasts produced after local and global context actions. The examples illustrate how Context2Action performs localized event-aware corrections and broader level or trend calibration across different application domains.

the Base Predictor, Local Expert output, Global Expert output, and ground truth, respectively, while four contexts are marked in purple.

The Base Predictor captures the overall temporal pattern but misses several context-related deviations. The Local Context Expert performs targeted corrections around the affected intervals, including the sales fluctuations associated with severe weather, Easter, the 7-day sale, and the toy recall. The Global Context Expert further adjusts the overall level and trend, bringing the final forecast closer to the ground truth. These observations illustrate the complementary roles of localized context actions and window-level calibration. To further illustrate how Context2Action translates real-world contexts into forecasting actions, we visualize representative forecasting cases from M5, Electricity, NYC Taxi, and Bitcoin in Figure 3. In each case, the dashed gray curve denotes the

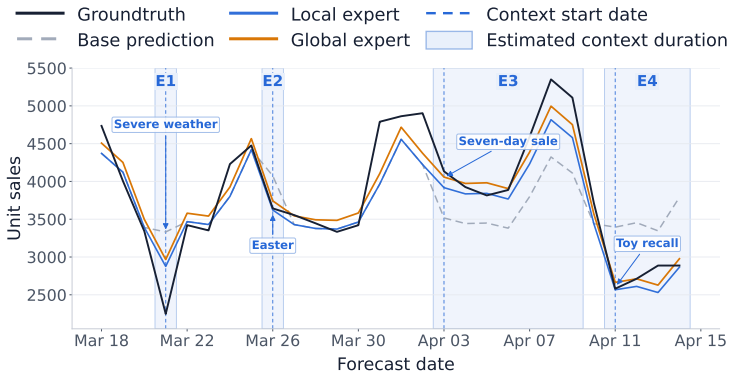


Figure 4: Forecast correction on an M5 dataset. The Local Expert performs context-specific adjustments, and the Global Expert further calibrates the overall level and trend.

frozen numerical predictor, while the colored curves show the sequential corrections produced by the two context experts. These examples highlight how the proposed framework adapts its revision behavior to different temporal structures and contextual patterns.

M5. Figure 3(a) presents one representative `HOBBIES` sequence with four contexts, including severe weather, Easter, a 7-day sale, and a toy recall. The base predictor captures the coarse temporal pattern but misses several context-related deviations, especially the sharp drop under severe weather, the uplift during the sale period, and the decline associated with the recall. After contextual reasoning, the first-stage expert performs targeted revisions on the affected intervals, while the second-stage expert further calibrates the overall level and trend. As a result, the final forecast is consistently closer to the observed trajectory, showing the complementary effects of localized revision and global calibration.

Electricity. Figure 3(b) shows a daily load forecasting case from New South Wales. The numerical predictor systematically underestimates the demand level throughout the day and also fails to capture the intraday shape around the evening ramp and peak. The context-aware corrections improve the forecast from two aspects: one mainly compensates for the overall level bias, while the other further refines the local load profile, especially around the evening ramp and peak period. This case suggests that contextual effects in electricity forecasting are not limited to isolated timestamps, but may jointly influence both the global demand level and fine-grained intraday dynamics.

NYC Taxi. Figure 3(c) shows a concert day in the NYC Taxi dataset. The base predictor misses the strong and highly concentrated demand surge near the concert time. After incorporating the event context, Context2Action substantially increases the forecast around the relevant hours, indicating that the framework can correctly identify the localized impact window of the event. In this example, the second-stage correction is relatively small, which is also reasonable: the main forecasting error is concentrated in a narrow interval rather than in the overall level or trend. This case demonstrates that Context2Action can improve event-sensitive peaks without introducing unnecessary revisions to the rest of the sequence.

Bitcoin. Figure 3(d) illustrates a Bitcoin forecasting case containing seven visible news events, including adoption, OCC, Tesla, MicroStrategy, mining, micro futures, and energy-related news. The base predictor remains nearly flat and fails to reflect the strong upward movement of the market. After incorporating the news contexts, the first-stage expert progressively lifts the forecast according to the event sequence, while the second-stage expert further refines the global trajectory, yielding a smoother and more accurate upward trend. This example highlights the usefulness of the proposed framework in financial forecasting, where multiple news events may jointly drive a sustained directional movement rather than a single short-lived fluctuation.

Overall, these case studies show that the two-stage design of Context2Action is flexible across domains. The first-stage expert mainly captures event-specific or locally concentrated deviations, while the second-stage expert addresses broader level or trend mismatches when necessary. Their relative

contributions vary with the task, but together they provide a traceable and interpretable path from contextual evidence to final forecast revision.