The Forecast Critic: Leveraging Large Language Models for Poor Forecast Identification

Luke Bhan[1*], **Hanyu Zhang** [2], **Andrew Gordon Wilson** [2], **Michael W. Mahoney** [2], **Chuck Arvin** [2†]

[1]University of California, San Diego [2]Amazon

Abstract

Monitoring forecasting systems is critical for customer satisfaction, profitability, and operational efficiency in large-scale retail businesses. We propose `The Forecast Critic`, a system that leverages Large Language Models (LLMs) for automated forecast monitoring, taking advantage of their broad world knowledge and strong "reasoning" capabilities. As a prerequisite for this, we systematically evaluate the ability of LLMs to assess time series forecast quality, focusing on three key questions. (1) Can LLMs be deployed to perform forecast monitoring and identify obviously unreasonable forecasts? (2) Can LLMs effectively incorporate unstructured exogenous features to assess what a reasonable forecast looks like? (3) How does performance vary across model sizes and reasoning capabilities, measured across state-of-the-art LLMs? We present three experiments, including on both synthetic and real-world forecasting data. Our results show that LLMs can reliably detect and critique poor forecasts, such as those plagued by temporal misalignment, trend inconsistencies, and spike errors. The best-performing model we evaluated achieves an F1 score of 0.88, somewhat below human-level performance (F1 score: 0.97). We also demonstrate that multi-modal LLMs can effectively incorporate unstructured contextual signals to refine their assessment of the forecast. Models correctly identify missing or spurious promotional spikes when provided with historical context about past promotions (F1 score: 0.84). Lastly, we demonstrate that these techniques succeed in identifying inaccurate forecasts on the real-world M5 time series dataset, with unreasonable forecasts having an sCRPS at least 10% higher than that of reasonable forecasts. These findings suggest that LLMs, even without domain-specific fine-tuning, may provide a viable and scalable option for automated forecast monitoring and evaluation.

Introduction

With the rise of large-scale distributed computing, automated purchasing systems that manage inventory decisions at the scale of millions of products are becoming increasingly common. Such purchasing systems require accurate predictions of product demand at all times to make optimal decisions. However, designing accurate forecasters at scale is challenging for a variety of reasons. For example, exogenous shocks like weather events or promotional events can skew observed values, and the inductive biases of time series models may lead to a failure to preserve important time series characteristics, like trends, periodicity, or volatility (Yu et al. 2025b,a). To address these challenges, time series foundation models (Ansari et al. 2024; Goswami et al. 2024; Gruver et al. 2023; Woo et al. 2024) have gained attention, delivering strong cross-timescale performance across domains such as weather prediction and demand forecasting, often without retraining. Even as models produce strong results in aggregate backtests when averaged across a full dataset, these models can sometimes produce poor individual forecasts, with obvious errors like misaligned trends, unrealistic transitions, or missing periodicity. Further, even models which perform well in backtests may produce unrealistic forecasts in the real world, where input data corruption or model deployment errors may trigger unanticipated behavior.

Consequently, to deploy such models in mission-critical real-world applications, we require systems that can reliably identify such inaccurate forecasts without knowing the true future observations. Human experts can often make such judgments by *visually* inspecting historical patterns and assessing whether a forecast is *plausible*, given the observed context. Inspired by this, we design a Large Language Model (LLM) based system that emulates such a qualitative visual human plausibility evaluation, enabling automated assessment of time series forecasts directly from visual plots. We call this system `The Forecast Critic`.

`The Forecast Critic` is designed to interpret a plot of a historical time series, unstructured guidance about what a reasonable forecast looks like, and the corresponding forecast, to determine whether the forecast is plausible or reasonable. Beyond serving as a human-like evaluator, such an approach offers multiple advantages for validating forecasts as they are generated, rather than in hindsight. First, to the extent that LLM behavior parallels human reasoning, the model may automatically detect key features of a time series, including trends, periodic patterns, and volatility levels. This is advantageous over statistical approaches that require domain-based hand-tuning from an engineer and that may incorrectly weigh certain characteristics of the time series over others. Second, LLMs can operate as automated processes, allowing them to process many more time series than would be possible if the process relied on human experts to conduct this forecast

[*]Work completed at Amazon

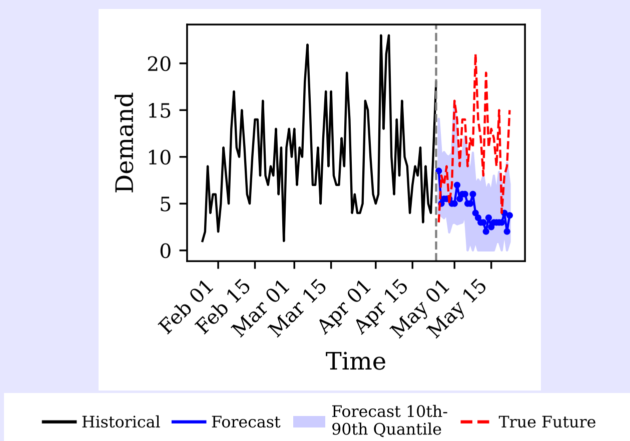
[†]Correspondence to chuarvin@amazon.com

auditing. Lastly, multi-modal LLMs can integrate additional contextual information, including instructions or exogenous features, without retraining. This additional context allows the models to incorporate new signals in an online manner, layering on additional information to foundation models trained on increasingly large datasets (Wolff et al. 2024).

To illustrate the kinds of errors that such an automated LLM agent can identify, we showcase in Figure 1 a real-world forecast from the Chronos model on the M5 dataset. In this example, the LLM correctly identifies that the forecast fails to reflect the high volatility present in the historical time series, and thus it identifies that this forecast is unreasonable. This example highlights the potential of this technology for practical deployment at scale.

Building on this illustrative example, we first evaluate `The Forecast Critic` system on controlled experiments with univariate synthetic data, where we artificially introduce standard but easily-identified perturbations by modifying the forecast trend, translating the forecast up or down, compressing the periodicity of the forecast, and introducing random outlier spikes to the forecasts. Such perturbations are relatively easy for human experts to identify (F1 score: 0.97). We find that LLMs are generally reasonably capable of automatically identifying these unreasonable forecasts, with the best-performing LLM achieving an F1 score of 0.88. We then assess how well these LLM-based forecast critiques can incorporate additional contextual information by providing the LLM with spurious and meaningful information about upcoming promotional events. We demonstrate that LLMs are correctly able to reason about such features, providing a promising avenue for including new features in an online manner through `The Forecast Critic`. Lastly, we conclude by evaluating performance on real-world time series from the M5 dataset (Makridakis, Spiliotis, and Assimakopoulos 2022), using forecasts generated by Chronos at the product granularity. Using this real-world validation, we show that forecasts flagged by `The Forecast Critic` have significantly worse error rates than unflagged forecasts.

Image Input & LLM Response



LLM Response

Looking at this image, I can see historical data showing high volatility with frequent spikes reaching values up to around 20. The forecast appears much less volatile, projecting values mostly between 3-7 with minimal fluctuation compared to the historical pattern.
This forecast fails to capture the significant volatility that is a consistent characteristic of the historical data. The drastic reduction in volatility represents a fundamental mismatch with the historical behavior of this time series, as there's no evidence that volatility was decreasing prior to the forecast point.

Figure 1: Example of a flagged forecast and the corresponding model justification for the M5 dataset using Claude 3.7 Sonnet. The model is fed the image on the left without the future values (shown in red), along with a prompt (given in the M5 section, below). The LLM flags this forecast as unreasonable, as it "fails to capture the significant volatility" in the historical time series.

Related work

LLMs as tools in retail applications. LLMs have been deployed across a variety of retail applications, including marketing (Meguellati et al. 2024), product reviews (Roumeliotis, Tselikas, and Nasiopoulos 2024), product search (Wang and Na 2024), and personalized data visualization (Luera et al. 2025). In particular, they show promise in zero-shot or few-shot contexts, allowing them to reason about novel tasks and situations, without requiring retraining (Zhu et al. 2024), achieving performance levels similar to those of human experts (Chen et al. 2023). LLMs have also been commonly investigated as chat agents, improving sales (Liu et al. 2024) and customer service retention (Pandya and Holia 2023), and they have even been used as customer relationship management (CRM) systems (Huang et al. 2025). Despite initial successes, significant challenges remain in the systematic deployment of LLMs at scale in retail settings (Ren et al. 2024). In this work, we highlight these challenges through a critical examination of time series forecasting.

LLMs for augmenting time series forecasting. Beyond these general retail applications, LLMs have garnered significant attention as powerful tools for enhancing time series forecasting frameworks. These models have shown promising performance across a variety of forecasting tasks (Gruver et al. 2023), though debate remains regarding their effectiveness in zero-shot deployment scenarios (Tan et al. 2024). Importantly, LLMs are not limited to simply generating forecasts. They have also been used to refine baseline forecasts by incorporating key external events like promotions or holiday events that may influence the time series (Ashok et al. 2025; Zhang et al. 2024). LLMs have also proven valuable in detecting anomalies within time series data, often by interpreting visual representations of the time series (Zhou and Yu 2025; Xu et al. 2025). These models are adept at analyzing and interpreting a wide range of time series characteristics, enabling deeper insights into time series behaviors and underlying dynamics (Fons et al. 2024; Cai et al. 2024). Finally, LLMs have been effectively applied to the problem of forecasting model selection. By processing extracted time series features, they can recommend suitable forecasting models and fine-tuned hyperparameters (Wei et al. 2025).

Despite these advancements, limited work has explored the combined challenges of forecast monitoring, which requires an LLM to understand the historical time series and evaluate whether a forecast is a reasonable extrapolation of that history. In this work, we address this gap by focusing on both aspects, with particular attention to developing automated tools to distinguish between accurate and inaccurate (or plausible versus implausible) forecasts.

Can LLMs detect synthetic time series perturbations?

In this section, we address the following questions. (1) Can LLMs detect common forecasting issues from visual plots alone? (2) Are certain issues more difficult for LLMs to detect than others? (3) Can LLMs incorporate exogenous information to flag missing or spurious promotional spikes? We demonstrate that state-of-the-art LLM models can correctly identify when a forecast is translated, contains false spikes, exhibits an unreasonable trend magnitude, or is stretched/-compressed, relative to the true time series. LLM performance is comparable, though somewhat worse, than that of an expert human. Further, we show that performance varies depending on the perturbation introduced. Additionally, we highlight that multi-modal LLMs are successfully able to connect text containing exogenous promotional information with the forecast to improve contextual decision making.

Dataset construction

As demonstrated in (Dooley et al. 2023) and (Taga, Ildiz, and Oymak 2025), synthetic time series are effective for training real-world forecasting models. To begin our systematic study of `The Forecast Critic`, we first apply simple perturbations to synthetically generated time series in a point-wise prediction setting. This approach offers two key advantages over the real-world experiments presented later in this paper. First, it allows us to evaluate the performance of LLMs across a variety of time series errors, helping identify potential situations where LLMs would miss inaccurate forecasts in a real-world deployment. Second, synthetic data provides a clear benchmark for determining when a forecast should be considered inaccurate. In real-world applications, the definition of a “good” forecast may be ambiguous, depending on both the time series and the preferences of forecast users. In the synthetic setting, by contrast, we control the data-generating process, so identifying good versus bad forecasts is more straightforward.

Synthetic time series generation. We construct a synthetic dataset consisting of various basis functions (see the Appendix for dataset details). Let $\mathcal{T} = \{t_0 + k \cdot \Delta t \mid k = 0, 1, \dots, N\}$ be a set of discrete evaluation points, where Δt is the sampling interval, and the series is divided into historical data $t \in \mathcal{T}_{\text{historical}}$ and forecast data $t \in \mathcal{T}_{\text{forecast}}$ such that $\mathcal{T} = \mathcal{T}_{\text{forecast}} \cup \mathcal{T}_{\text{historical}}$. Then, we define the time series $y(t)$ as

$$y(t) = \sum_{i=1}^{n} w_i \cdot b_i(s_i(t + \delta_i)) , \quad t \in \mathcal{T} , \tag{1}$$

where $n \sim \mathcal{U}\{1, 2, 3, 4\}$ is the number of basis functions to include, and for all $i \in [1, n]$, we have $w_i \sim \mathcal{U}(0.5, 2.0)$ as an output scaling weight, $s_i \sim \mathcal{U}(0.5, 2.0)$ as an input scaling weight, $\delta_i \sim \mathcal{U}(0, 4)$ as an input shift, and $b_i \sim \mathcal{U}\{1, 2, ..., 14\}$ as the basis functions defining the time series (given in the Appendix). For all the experiments, we consider a $\Delta t = 0.1$ and a final time horizon of $T = 10$ (yielding 100 discrete points), and we use the time $t = 8$ as the split between the historical window and forecasting period.

User Prompt

You are shown an image of historical data (in black) and a forecast (in blue). Based on the historical trend, assess whether the forecast is reasonable. A reasonable forecast should generally follow the same direction and capture any seasonal trends if there are any. Please provide a brief explanation (1–2 sentences) justifying your decision. Then present your final answer using one of the following options, wrapped in <answer> tags:
<answer> 1 </answer> — The forecast is reasonable.
<answer> 2 </answer> — The forecast is unreasonable.

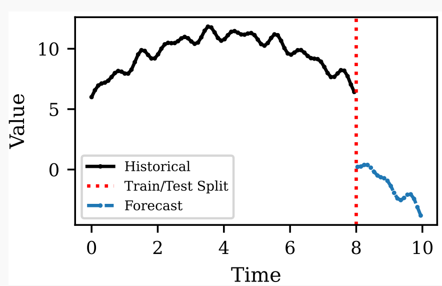


Figure 2: Example prompt asking an LLM whether a forecast (featuring an artificially-induced downward translational shift) is reasonable or unreasonable. In this case, the LLM should respond with unreasonable, along with a justification noting that the forecast is unreasonably low.

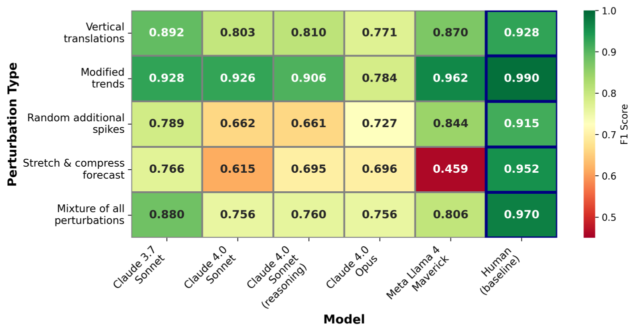


Figure 3: Evaluation of the ability of LLMs to identify reasonable versus unreasonable forecasts based on various synthetic perturbations.

Forecast perturbations. We begin by synthetically injecting perturbations into the forecast data to create clearly inaccurate, or unreasonable, forecasts. As highlighted in previous works (Zhou and Yu 2025), casting time series data as image inputs for multi-modal LLMs is a viable—and sometimes superior—approach for time series tasks. Consequently, we present our inputs to the LLMs in the form of plots, an example of which is shown in Figure 2. We consider five different state-of-the-art LLMs that accept multi-modal text-image inputs and produce text outputs. These models have varying reasoning capabilities and model sizes: Claude 3.7 Sonnet (Anthropic 2025), Claude 4.0 Sonnet (Anthropic 2025), Claude 4.0 Sonnet with reasoning, Claude 4.0 Opus (Anthropic 2025) and Meta Llama 4 Maverick (Meta AI 2025).

To evaluate the LLMs, we consider four distinct types of perturbations (see Figure 7 and the Appendix for examples of each), and we construct case studies where half of the examples contain a perturbation and the other half consist of unaltered, correct forecasts. Each model is prompted, as illustrated in Figure 2, to assess whether a forecast is reasonable. We evaluate performance on 500 unique time series forecasts, equally split between perturbed and unperturbed cases (250 perturbed, 250 without modification). To ensure the perturbed examples are visibly worse than the true time series, we generate 334 problems per perturbation type, and we retain the worst 75% based on symmetric mean absolute percentage error (SMAPE) to obtain our 250 examples of each perturbation. Each LLM is evaluated using its default settings via Amazon Bedrock, and we report both class-specific F1 scores and the overall weighted F1 score (see the Appendix for definitions) in Figure 3. For comparison, we establish a human baseline by having an expert annotator independently label the same 500 plots using the same interface.

Empirical Results

We find that model performance varies substantially depending on the type of perturbation applied. See Figure 3 for a summary, and see Appendix Figures 8, 9, 10, 11 for example LLM anecdotes. We find that, depending on perturbation, the Claude 3.7 Sonnet or Meta Llama 4 Maverick models perform best. This is notable, as both models are older than more recent counterparts like Claude 4.0 Sonnet and Opus. We find that LLMs are quite capable of identifying situations where the forecast has an unrealistic trend (F1 score > 0.9), and they are generally capable of identifying forecasts that have been incorrectly shifted or have spurious forecast spikes. By contrast, the LLMs struggle to identify cases where the forecasts have been unrealistically stretched or compressed (as in Figure 10), even when human experts have little difficulty with those examples. Additionally, we find that the human evaluator outperforms all LLM models in all categories, highlighting the room for improvement. In the final row of Figure 3, we present results from a more difficult "mixture" evaluation, in which any of the four perturbations may be randomly applied. This setup helps to mitigate annotator bias by preventing the human experts from seeing several examples of a single perturbation in sequence, increasing task complexity. Despite this, both the human and Claude 3.7 Sonnet continues to perform well, achieving F1 scores of 0.97 and 0.88, respectively.

These findings suggest that neither model size nor newer model generations reliably improve performance on `The Forecast Critic` task. These results are somewhat surprising. While newer models like Claude 4.0 Opus are generally more capable, we see meaningful degradation on this particular task. Overall, these results show that top performing LLMs are capable of reliably identifying inaccurate forecasts, at least for cases where the inaccuracy is fairly obvious by construction. While the top performing LLMs still underperform human experts, F1 scores on this task often exceed 0.8, indicating a reliable ability to identify inaccurate predictions.

Summary: LLMs succeed at identifying clearly inaccurate forecasts, with F1 score approaching 0.9. LLMs also reliably identify modified trends and vertical translations, but they struggle with stretching and compressed periodicity.

Exogenous Feature Injection

A key advantage of using LLMs as forecast auditors lies in their ability to use unstructured context, including plain-language instructions or exogenous features in a zero-shot manner (Ghasemloo and Moradi 2025). In this experiment, we systematically examine how well LLMs can do this in the context of `The Forecast Critic`.

We construct synthetic data reflecting a series of different promotional scenarios, and we evaluate the ability of LLMs to correctly reason about each of the following scenarios.

(a) Historical sales showed no lift during past promotions. There is an upcoming promotion, but there is no spike in the forecast. This is *reasonable*, since the sales do not depend on the promotion.

(b) Historical sales showed no lift during past promotions. There is an upcoming promotion, and there is a spike in the forecast corresponding to the upcoming promotion. This is *unreasonable*, since the forecast expects a lift in sales that is not shown in the historical data.

(c) Historical sales showed a lift during past promotions. There is an upcoming promotion, but there is no spike in the forecast for the upcoming promotion. This is *unreasonable*, since the forecast fails to anticipate a lift in sales from the upcoming promotion.

(d) Historical sales showed a lift during past promotions. There is an upcoming promotion, and there is a spike in the forecast for the upcoming promotion. This is *reasonable*, since the forecast anticipates the upcoming promotional lift.

Examples of each scenario are given in Appendix Figure 12. For each scenario, we evaluate the LLM on 500 examples modifying the synthetic dataset with the corresponding promotional spikes. We then ask the LLM whether the forecast is reasonable or unreasonable, using a prompt that includes the dates of the holidays. An example of the input and output is shown in Figure 4.

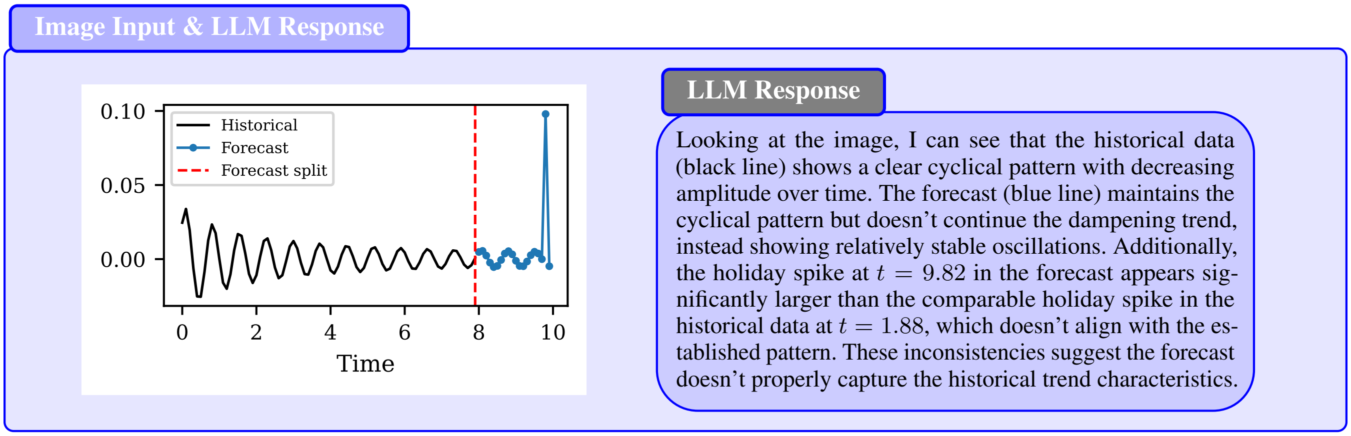


Figure 4: Example of an LLM considering additional promotional context when evaluating the forecast. The LLM reasons that the forecast spike at $t = 9.82$ is much larger than the promotional behavior in the history, so it flags the example as unreasonable.

We present a summary of the results in Figure 5, where we see that across all scenarios, the best performing model (Claude 3.7 Sonnet) achieves F1-scores of around 0.8. Here again, newer versions of the Claude model show degradation in performance on this task. Notably, we also see that scenario (c) is the hardest task across all models tested. In scenario (c), the forecasts are missing a promotional spike, even though such a spike is observed in the historical sales. This result is especially concerning, as detecting cases where a model fails to anticipate an upcoming promotion is a critical use case for such a forecast inspection system. We provide additional illustrative examples of the generated analyses for all scenarios in Appendix Figures 14, 15, 16, and 17.

Summary: LLMs can incorporate additional exogenous information to determine whether a forecast is reasonable, with F1 score up to 0.84. All models struggle most with scenario (c), where the forecast is missing a promotional spike, despite seeing elevated sales in the history.

Can LLMs identify poor real-world forecasts from foundation models?

M5 Dataset construction

While synthetic data is useful for systematic studies of LLMs, they do not fully reflect real-world performance. To bridge this gap, we conduct an additional study to assess whether LLMs can correctly flag inaccurate forecasts from the Chronos model (Ansari et al. 2024) using the M5 time series competition dataset (Makridakis, Spiliotis, and Assimakopoulos 2022). The M5 challenge involves hierarchical forecasting, where participants predict demand for various products across multiple levels, including states, departments, and categories. We consider the product aggregation level, and we randomly sample 1,000 time series and corresponding forecasts.The time series is at the daily granularity, and we present the LLM with 120 days of historical time series and 28 days of forecasts. Because Chronos provides a full distribution forecast, we modify the figures to include the median, 10th and 90th percentiles of the distribution, as shown in Figure 1. We present the LLM only a univariate view of the time series, without additional covariates and contextual data.

Empirical results

Designing a rigorous experiment to evaluate real-world forecasts introduces a core challenge: defining what makes a forecast *good* is inherently ambiguous. To help navigate this, we use the scaled continuously ranked probability score (sCRPS) (see the Appendix for definitions) as our evaluation metric, with higher scores indicating less accurate forecasts. Importantly, sCRPS requires access to ground-truth future values, and it cannot be validated *a priori*, but it provides a viable method for systematically evaluating whether `The Forecast Critic` identified forecasts which were going to be inaccurate. Additionally, in initial experiments we observed that the models considered the vast majority of forecasts unreasonable. To address this, we provided additional instructions to the LLM instructing it to only flag *obvious* forecasting errors, leaving the interpretation of obvious

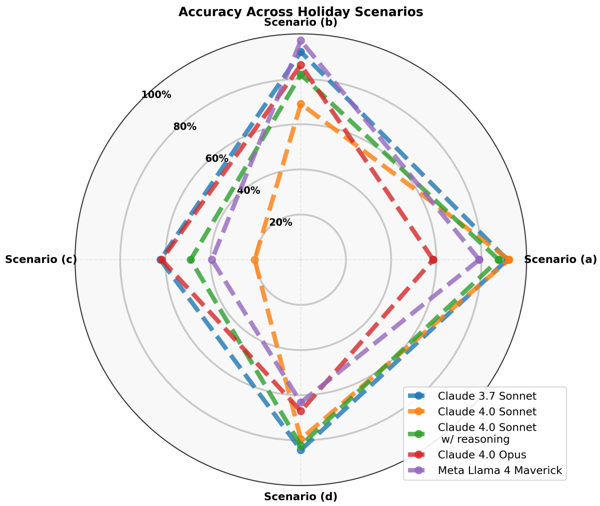


Figure 5: LLM performance on exogenous feature injection across four promotional holiday scenarios. Scenario (a): product sales do not depend on promotions, with no spikes shown in either history or forecast (reasonable forecast). Scenario (b): product sales do not depend on promotions, but forecast incorrectly shows a spike (unreasonable forecast). Scenario (c): product sales depend on promotions with spike in history, but forecast is missing the spike (unreasonable forecast). Scenario (d): product sales depend on promotions, with spikes correctly shown in both history and forecast (reasonable forecast). Accuracy represents the percentage of forecasts correctly classified as reasonable or unreasonable. Claude 3.7 Sonnet achieves the best overall performance with an F1-score of 0.836, followed by Claude 4.0 Sonnet with reasoning (F1: 0.767) and Meta Llama 4 Maverick (F1: 0.706).

largely to the LLM (see Appendix Figure 18).

Under this setup, several key patterns emerge. First, different models are much more likely to flag a forecast as unreasonable, ranging from just 2% for Llama 4 Maverick to nearly 40% for Claude 4.0 Opus. Despite this, forecasts flagged by the LLM as unreasonable consistently have higher sCRPS values than reasonable ones, as shown in Figure 6. Across all models, the distribution of sCRPS for reasonable and unreasonable forecasts are statistically distinct according to a Mann-Whitney U test. See Table 1 for details. Further, the median sCRPS for unreasonable forecasts is at least 10% higher than for reasonable ones. Third, there is clear room to further improve the LLM performance, as Figure 6 shows that all models consider some forecasts with very high sCRPS to be reasonable. Finally, we present anecdotal examples (Appendix Figures 19 and 20), including a failure mode where an LLM incorrectly flags a reasonable forecast due to an outlier spike in the historical data. Despite such individual errors, the aggregate results demonstrate that `The Forecast Critic` is effective at identifying forecasts that have significantly worse accuracy.

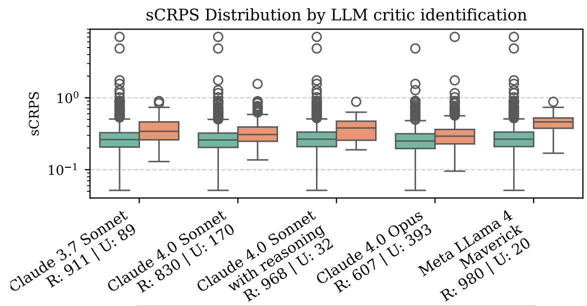


Figure 6: Comparison of sCRPS distributions on the forecasts flagged as reasonable and unreasonable for each critic model. Here, $R : X|U : Y$ stands for the number of reasonable and unreasonable forecasts flagged by the model. Claude 3.7 Sonnet assessed 91.1% of the forecasts as reasonable, while Claude 4.0 Opus assessed only 60.7% of the forecasts as reasonable.

Summary: LLMs can identify real-world forecasts which are likely to be inaccurate. While the number of unreasonable forecasts varies across LLMs (from 2% up to 40%), in all cases the sCRPS of unreasonable forecasts exceeds the sCRPS of reasonable forecasts by a significant margin.

Conclusion

In this work, we have introduced `The Forecast Critic`, a multi-modal system to assess whether a newly generated forecast is reasonable or unreasonable, plausible or implausible, based on a visualization plus plain-language instructions. We have also evaluated how well several modern LLMs can identify good versus obviously-inaccurate forecasts—an essential capability for building scalable and reliable forecasting systems. In our experiments using synthetic time series data, we find that LLMs can reliably detect forecast errors (F1 score > 0.8), albeit slightly below the performance of expert human evaluators. We also demonstrate that our system can effectively use additional text information about promotional covariates, highlighting the promise of LLMs customizing their critiques to incorporate additional context when exogenous shocks arise. These results also highlight shortcomings of the LLMs. For example, LLMs excel at detecting misaligned trends and vertical translations, but they struggle to identify stretched or compressed periodicity. We also conduct experiments using real-world time series from the M5 dataset with probabilistic forecasts from the Chronos model. We show that unreasonable forecasts have higher sCRPS values than reasonable ones, indicating that the system is detecting unusually inaccurate forecasts.

Overall, our findings underscore the potential and shortcomings of LLMs as automated forecast auditors. While their current performance remains slightly below that of expert human evaluators, `The Forecast Critic` framework provides a scalable and reasonable approach to forecast auditing. In future work, we plan to extend this work to include

Model	Claude 3.7 Sonnet	Claude 4.0 Sonnet	Claude 4.0 Sonnet with reasoning	Claude 4.0 Opus	Meta Llama 4 Maverick
Reasonable, Median	0.2624	0.2591	0.2661	0.2505	0.2661
Reasonable, Mean	0.3022	0.3008	0.3063	0.2505	0.2661
Reasonable, Standard Deviation	0.3105	0.3206	0.3044	0.24	0.3022
Unreasonable, Median	0.3417	0.3081	0.3804	0.2924	0.4622
Unreasonable, Mean	0.3752	0.3472	0.3813	0.3403	0.4658
Unreasonable, Standard Deviation	0.1615	0.1704	0.1479	0.3739	0.168
Median, % Diff	30.2	18.9	43	16.7	73.7
Mean, % Diff	24.18	15.45	24.50	18.11	52.49
Mann-Whitney, U stat	25336	50339	9208	90515	3578
p-value	5.04×10^{-9}	3.84×10^{-9}	9.36×10^{-5}	1.14×10^{-10}	1.14×10^{-6}

Table 1: Aggregate sCRPS statistics for the distribution of forecasts classified as reasonable/unreasonable by each model for the M5 product dataset.

a wider variety of modern LLMs, expand our treatment of additional covariate information, and evaluate our performance on additional real-world time series forecast datasets.

References

Ansari, A. F.; Stella, L.; Turkmen, A. C.; Zhang, X.; Mercado, P.; Shen, H.; Shchur, O.; Rangapuram, S. S.; Arango, S. P.; Kapoor, S.; Zschiegner, J.; Maddix, D. C.; Wang, H.; Mahoney, M. W.; Torkkola, K.; Wilson, A. G.; Bohlke-Schneider, M.; and Wang, B. 2024. Chronos: Learning the Language of Time Series. *Transactions on Machine Learning Research*.

Anthropic. 2025. Claude 3.7 Sonnet System Card. Accessed: 2025-09-15.

Anthropic. 2025. System Card: Claude Opus 4 & Claude Sonnet 4. https://www.anthropic.com/claude-4-system-card. Accessed: 2025-09-15.

Ashok, A.; Williams, A. R.; Zheng, V. Z.; Rish, I.; Chapados, N.; Étienne Marcotte; Zantedeschi, V.; and Drouin, A. 2025. Beyond Naïve Prompting: Strategies for Improved Zero-shot Context-aided Forecasting with LLMs. arXiv:2508.09904.

Cai, Y.; Choudhry, A.; Goswami, M.; and Dubrawski, A. 2024. TimeSeriesExam: A time series understanding exam. arXiv:2410.14752.

Chen, J.; Ma, L.; Li, X.; Thakurdesai, N.; Xu, J.; Cho, J. H. D.; Nag, K.; Korpeoglu, E.; Kumar, S.; and Achan, K. 2023. Knowledge Graph Completion Models are Few-shot Learners: An Empirical Study of Relation Labeling in E-commerce with LLMs. arXiv:2305.09858.

Dooley, S.; Khurana, G. S.; Mohapatra, C.; Naidu, S. V.; and White, C. 2023. ForecastPFN: Synthetically-trained zero-shot forecasting. In *Advances in Neural Information Processing Systems*.

Fons, E.; Kaur, R.; Palande, S.; Zeng, Z.; Balch, T.; Veloso, M.; and Vyetrenko, S. 2024. Evaluating Large Language Models on Time Series Feature Understanding: A Comprehensive Taxonomy and Benchmark. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 21598–21634.

Ghasemloo, M.; and Moradi, A. 2025. Informed Forecasting: Leveraging Auxiliary Knowledge to Boost LLM Performance on Time Series Forecasting. arXiv:2505.10213.

Goswami, M.; Szafer, K.; Choudhry, A.; Cai, Y.; Li, S.; and Dubrawski, A. 2024. MOMENT: a family of open time-series foundation models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Gruver, N.; Finzi, M.; Qiu, S.; and Wilson, A. G. 2023. Large language models are zero-shot time series forecasters. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23.

Huang, K.-H.; Prabhakar, A.; Dhawan, S.; Mao, Y.; Wang, H.; Savarese, S.; Xiong, C.; Laban, P.; and Wu, C.-S. 2025. CRMArena: Understanding the Capacity of LLM Agents to Perform Professional CRM Tasks in Realistic Environments. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 3830–3850.

Liu, N.; Chen, L.; Tian, X.; Zou, W.; Chen, K.; and Cui, M. 2024. From LLM to Conversational Agent: A Memory Enhanced Architecture with Fine-Tuning of Large Language Models. arXiv:2401.02777.

Luera, R.; Rossi, R.; Dernoncourt, F.; Siu, A.; Kim, S.; Yu, T.; Zhang, R.; Chen, X.; Lipka, N.; Zhang, Z.; Gyeom Kim, S.; and Yeon Lee, T. 2025. Personalizing Data Delivery: Investigating User Characteristics and Enhancing LLM Predictions. In *Companion Proceedings of the ACM on Web Conference 2025*, WWW '25, 1167–1171.

Makridakis, S.; Spiliotis, E.; and Assimakopoulos, V. 2022. M5 accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting*, 38(4): 1346–1364. Special Issue: M5 competition.

Meguellati, E.; Han, L.; Bernstein, A.; Sadiq, S.; and Demartini, G. 2024. How Good are LLMs in Generating Personalized Advertisements? In *Companion Proceedings of the ACM Web Conference 2024*, WWW '24, 826–829.

Meta AI. 2025. The Llama 4 herd: The beginning of a new era of natively multimodal intelligence. *AI at Meta*. Accessed: 2025-09-15.

Pandya, K.; and Holia, M. 2023. Automating Customer Service using LangChain: Building custom open-source GPT Chatbot for organizations. arXiv:2310.05421.

Ren, Q.; Jiang, Z.; Cao, J.; Li, S.; Li, C.; Liu, Y.; Huo, S.; He, T.; and Chen, Y. 2024. A survey on fairness of large language models in e-commerce: progress, application, and challenge. arXiv:2405.13025.

Roumeliotis, K. I.; Tselikas, N. D.; and Nasiopoulos, D. K. 2024. LLMs in e-commerce: A comparative analysis of GPT and LLaMA models in product review evaluation. *Natural Language Processing Journal*, 6: 100056.

Taga, E. O.; Ildiz, M. E.; and Oymak, S. 2025. TimePFN: Effective multivariate time series forecasting with synthetic data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 20761–20769.

Tan, M.; Merrill, M. A.; Gupta, V.; Althoff, T.; and Hartvigsen, T. 2024. Are Language Models Actually Useful for Time Series Forecasting? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Wang, H.; and Na, T. 2024. Rethinking E-Commerce Search. *SIGIR Forum*, 57(2).

Wei, W.; Yang, T.; Chen, H.; Rossi, R. A.; Zhao, Y.; Dernoncourt, F.; and Eldardiry, H. 2025. Efficient Model Selection for Time Series Forecasting via LLMs. arXiv:2504.02119.

Wolff, M.; Olivares, K. G.; Oreshkin, B. N.; Ruan, S.; Yang, S.; Katoch, A.; Ramasubramanian, S.; Zhang, Y.; Mahoney, M. W.; Efimov, D.; and Quenneville-Belair, V. 2024. SPADE Split Peak Attention DEcomposition. In *NeurIPS Workshop on Time Series in the Age of Large Models*.

Woo, G.; Liu, C.; Kumar, A.; Xiong, C.; Savarese, S.; and Sahoo, D. 2024. Unified training of universal time series forecasting transformers. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Xu, X.; Wang, H.; Liang, Y.; Yu, P. S.; Zhao, Y.; and Shu, K. 2025. Can Multimodal LLMs Perform Time Series Anomaly Detection? arXiv:2502.17812.

Yu, A.; et al. 2025a. Understanding the Implicit Biases of Design Choices for Time Series Foundation Models. arXiv:2510.19236.

Yu, A.; et al. 2025b. Understanding Transformers for Time Series: Rank Structure, Flow-of-ranks, and Compressibility. arXiv:2510.03358.

Zhang, H.; Arvin, C.; Efimov, D.; Mahoney, M. W.; Perrault-Joncas, D.; Ramasubramanian, S.; Wilson, A. G.; and Wolff, M. 2024. LLMForecaster: Improving Seasonal Event Forecasts with Unstructured Textual Data. arXiv:2412.02525.

Zhou, Z.; and Yu, R. 2025. Can LLMs Understand Time Series Anomalies? In *The Thirteenth International Conference on Learning Representations*.

Zhu, J.; Cai, S.; Deng, F.; Ooi, B. C.; and Wu, J. 2024. Do LLMs Understand Visual Anomalies? Uncovering LLM's Capabilities in Zero-shot Anomaly Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, MM '24, 48–57.

Definition of statistical metrics

Weighted F1-Score

The weighted F1 score is given by

$$\text{Weighted F1 Score} = \sum_{i=1}^{N} \left(\frac{n_i}{\sum_{j=1}^{N} n_j} \cdot F1^{(i)} \right) ,$$

where $F1^{(i)}$ is the $F1$ score for the ith class, n_i is the number of elements in each class, and N is the total number of classes.

Scaled Continuous Ranked Probability Score (sCRPS)

sCRPS is a normalized version of the standard CRPS, where the total CRPS is divided by the sum of the absolute values of the true time series. Formally, for a given true time series y and a set of forecast quantile predictions $\hat{Y}$, it is defined as

$$\text{sCRPS}(y, \hat{Y}) = \frac{\sum_{t \in \mathcal{T}_{\text{forecast}}} \text{CRPS}(y_t, \hat{Y}_t)}{\sum_{t \in \mathcal{T}_{\text{forecast}}} |y_t|} ,$$

where $t \in \mathcal{T}_{\text{forecast}}$ is the number of forecasted points.

To compute the CRPS, we use a Riemann integral approximation technique that uniformly averages the quantile loss over a discrete set of quantiles. The CRPS for a single observation is given by:

$$\text{CRPS}(y, \hat{Y}) = 2 \int_0^1 \text{QL}_q \left(y, F_{\hat{Y}}^{-1}(q) \right) \, dq \, ,$$

where we use $q \in \{0.1, 0.2, ..., 0.9\}$ as the set for the discrete quantiles, $F_{\hat{Y}}^{-1}(q)$ to indicate the predicted quatile at q, and the quantile loss function $\text{QL}_q(\cdot, \cdot)$ is defined as:

$$\text{QL}_q(y, \hat{y}) = q \cdot (y - \hat{y})^+ + (1 - q) \cdot (\hat{y} - y)^+ \, ,$$

where $(x)^+ = \max(x, 0)$.

Additional details on the construction of the synthetic dataset

For our synthetic dataset, we construct unique time series via

$$y(t) = \sum_{i=1}^{n} w_i \cdot b_i(s_i(t + \delta_i)) \, , \quad t \in \mathcal{T} \, ,$$

where $n \sim \mathcal{U}\{1, 2, 3, 4\}$ is the number of base functions to include, and for all $i \in [1, n]$, we have $w_i \sim \mathcal{U}(0.5, 2.0)$ as an output scaling weight, $s_i \sim \mathcal{U}(0.5, 2.0)$ as an input scaling weight, $\delta_i \sim \mathcal{U}(0, 4)$ as an input shift, and $b_i \sim \mathcal{U}\{1, 2, ..., 14\}$ as the base functions defining the time series, given in Table 2 below.

Synthetic experiments under controlled perturbations: Additional results and perturbation construction

Perturbation types and construction

In this study, we consider four types of point-wise perturbations: translations, trend modifications, additional spiking, and stretching of the signal. Each perturbation simulates a different kind of temporal or structural anomaly. We provide visual examples of each type of modification in Figure 7. Here, we define a general framework, and then we describe each perturbation mathematically:

Consider the discrete signal representation as in (1),

$$y(t), \quad t \in \mathcal{T} = \{t_0 + k \cdot \Delta t \mid k = 0, 1, \ldots, N\},$$

where Δt is the sampling interval, and the series is divided into historical data $y(t)$ for $t \in \mathcal{T}_{\text{historical}}$ and forecast data $y(t)$ for $t \in \mathcal{T}_{\text{forecast}}$.

Following this formulation, each perturbation is given as follows.

- **Vertical Shift (Shift scale ω).** Given the original time series $y(t)$, the forecast data is perturbed by adding a vertical shift ω multiplied by the forecast mean:

$$y_{\text{vertical shift}}(t) := y(t) + \omega \times \frac{1}{|\mathcal{T}_{\text{forecast}}|} \sum_{\theta \in \mathcal{T}_{\text{forecast}}} y(\theta), \quad t \in \mathcal{T}_{\text{forecast}}.$$

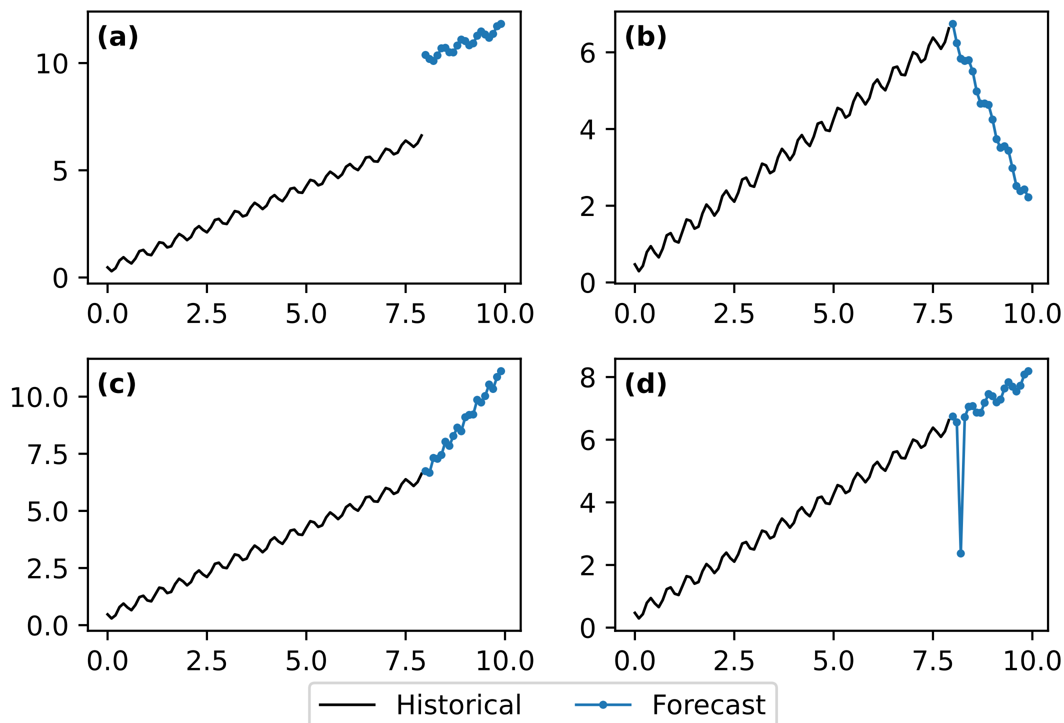


Figure 7: Examples of various perturbations on a synthetic time series dataset. (a) shows the translate up perturbation, (b) shows the trend modification, (c) shows the stretch modification, and (d) has the forecast containing an additional spike. In this case, ($\mathcal{T}_{\text{historical}} = [0, 8]$ and $\mathcal{T}_{\text{forecast}} = (8, 10]$ where $\Delta t = 0.1$. Additionally, the hyperparameters for each perturbation are set as $\omega = 0.5$, $\beta = -3$, $\alpha = 3$, $\gamma = 0.5$, and $N_{\text{spikes max}} = 3$.

ID	base function name	Expression $b_i(t)$
1	Gaussian wave	$5e^{-0.00005(t-6)^2}\sin(0.5t)$
2	Linear cos	$0.3+0.5t+0.2\cos(10t)$
3	Linear	$0.3+0.5t$
4	Sin	$\sin(4t)$
5	Sinc	$10(t+\epsilon_{\text{threshold}})^{-1}\sin(5t)$
6	Beat	$\sin(t)\sin(5t)$
7	Sigmoid	$(1+e^{-4t})^{-1}$
8	Log	$\log(1+t)$
9	Sin scaled	$4(t+1)\sin(5(t+1)+4)$
10	Square	$3t^2$
11	Step	$\mathscr{H}(t-3,1)$
12	Multistep	$0.2\mathscr{H}(t-1,1)+0.3\mathscr{H}(t-2.5,1)$ $-0.1\mathscr{H}(t-4,1)+0.4\mathscr{H}(t-5.5,1)$ $-0.3\mathscr{H}(t-7,1)+0.2\mathscr{H}(t-8.5,1)+0.1\mathscr{H}(t-9.5,1)$
13	Chirp	$\sin(10t^2)$
14	Sawtooth	$2\cdot(t\cdot\pi^{-1}-\lceil 0.5+t\cdot\pi^{-1}\rceil)$

Table 2: Various base functions used to construct the synthetic time series dataset. $\mathscr{H}$ corresponds to the Heaviside function and $\epsilon_{\text{threshold}}$ is a numerical stability threshold set to 10^{-10}.

- **Linear Trend Modification (Slope Scaling β).** Assume the forecast data can be decomposed into a linear trend plus residuals:

  $$y(t) = mt + b + r(t), \quad t \in \mathcal{T}_{\text{forecast}},$$

  where m and b denote the slope and intercept estimated from the forecast data, and $r(t)$ represents residuals.
  The perturbation rescales the slope by β:

  $$m' = \beta m.$$

  To maintain continuity at $t_c = t^{(0)}_{\text{forecast}}$, the intercept is adjusted to satisfy:

  $$y(t_c) = m't_c + b' + r(t_c).$$

  Solving for the new intercept gives:

  $$b' = y(t_c) - m't_c - r(t_c).$$

  The resulting perturbed series is:

  $$y_{\text{trend modification}}(t) = m't + b' + r(t).$$

- **Time Stretching (Multiplicative Factor α).** The time axis is resampled with a new step size $\alpha \cdot \Delta t$, stretching ($\alpha > 1$) or compressing ($\alpha < 1$) the signal:

  $$\mathcal{T}' = \{t_0 + k\cdot(\alpha\cdot\Delta t)\}, \quad y'(t') = f(t'), \quad t' \in \mathcal{T}',$$

  where $f(t)$ is the generating signal for the time series (i.e., the right-hand side of (1)).
  Further, to ensure continuity at the forecast boundary:

  $$y_{\text{time stretch}}(t') = y'(t') + \delta,$$
  $$\delta := y(t^{(0)}_{\text{forecast}}) - y'(t^{(0)}_{\text{forecast}}).$$

- **Random Spikes (Modifier γ, Max Count $N_{\textbf{spikes max}}$).** Randomly inject $n \sim \text{Uniform}\{1,\dots,N_{\text{max}}\}$ spikes into the forecast data at positions $\mathcal{T}_{\text{spike}} \subset \mathcal{T}_{\text{forecast}}$, where $|\mathcal{T}_{\text{spike}}| = n$:

  $$y_{\text{spikes}}(t) = \begin{cases} y(t)+\epsilon, & t \in \mathcal{T}_{\text{spike}}, \\ y(t), & \text{otherwise}, \end{cases}$$

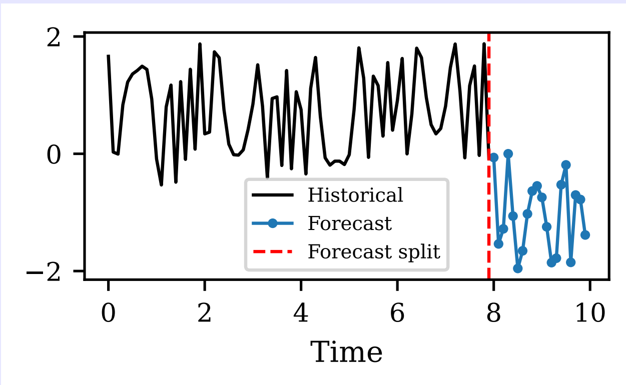
  where the spike magnitude ϵ is defined as

  $$\epsilon = \pm\gamma \cdot \max_{t\in\mathcal{T}_{\text{forecast}}} y(t),$$

  with the sign chosen independently and uniformly at random for each spike.

Example of LLM responses for various synthetic perturbations

Image Input & LLM Response



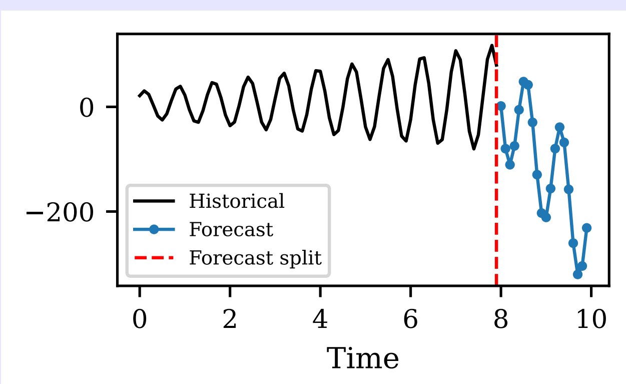
LLM Response

Looking at the image, I can see that the historical data (black line) shows a highly oscillatory pattern with values typically ranging between -1 and 2, with most values staying in the positive range. However, the forecast (blue line) shows oscillations primarily in the negative range (between approximately -2 and 0), representing a significant downward level shift from the historical pattern. While the forecast does capture the oscillatory nature of the data, the substantial shift downward makes it inconsistent with the historical behavior.

The forecast is unreasonable. The forecast has captured the oscillating pattern but incorrectly shifted the entire prediction downward into a predominantly negative range when the historical data mostly fluctuated in the positive range.

Figure 8: Example of an LLM response corresponding to a shift misalignment, as shown in Figure 7(a). The LLM correctly recognizes that the transition point is poor, and hence it flags the forecast as unreasonable.

Image Input & LLM Response



LLM Response

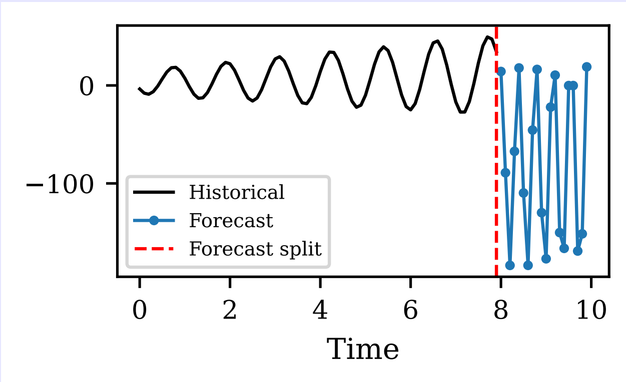
Looking at this image, I can see that the historical data (black) shows a clear oscillating pattern with gradually increasing amplitude over time. The forecast (blue) continues the oscillation pattern but with a dramatically larger amplitude than what would be expected from the historical trend.

The forecast is unreasonable.

The forecast is unreasonable because while it maintains the cyclical nature of the data, it shows a sudden and extreme jump in amplitude that doesn't align with the more gradual amplitude increase shown in the historical pattern.

Figure 9: Example of an LLM response corresponding to a trend misalignment, as shown in Figure 7(b). The LLM correctly recognizes that the amplitude differs from the historical trend, and hence it flags the forecast as unreasonable.

Image Input & LLM Response



LLM Response

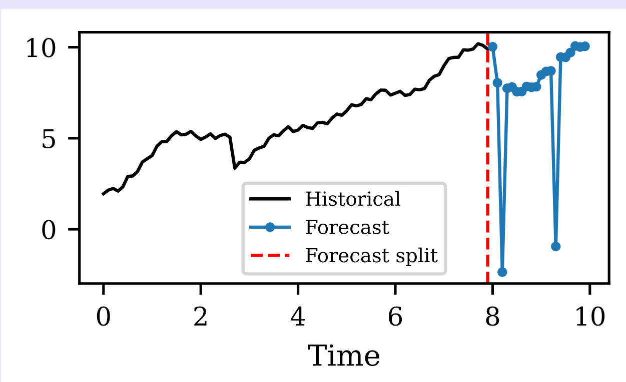
Looking at the image, I can see that the historical data (black line) shows a consistent sinusoidal pattern with regular oscillations between approximately $+25$ and -25. However, the forecast (blue line) dramatically changes in amplitude, showing extreme negative values down to around -150 while maintaining similar frequency. This represents a significant and unexplained departure from the established pattern.

The forecast is unreasonable.

The forecast is unreasonable because it dramatically increases the amplitude of oscillations and introduces extreme negative values that aren't consistent with the historical pattern, while the frequency remains similar.

Figure 10: Example of an LLM response corresponding to a compression perturbation, as shown in Figure 7(c). The LLM correctly recognizes that the forecast oscillation frequency increase beyond reason, and hence it flags the forecast as unreasonable.

Image Input & LLM Response



LLM Response

Looking at the image, I can see that while the forecast (blue line) continues the overall upward trend from the historical data (black line), it introduces extreme volatility that isn't present in the historical pattern. The historical data shows a relatively smooth upward progression, but the forecast contains dramatic drops (particularly one that goes negative around time 9.5) and sharp fluctuations that aren't justified by any previous patterns in the data.

The forecast is unreasonable. The forecast introduces extreme volatility and dramatic drops (including one below zero) that aren't consistent with the relatively smooth upward pattern shown in the historical data.

Figure 11: Example of an LLM response corresponding to the presence of additional spikes, as shown in Figure 7(d). The LLM correctly recognizes that there are unnecessary spikes in the forecast, and hence it flags the forecast as unreasonable.

Exogenous Feature Injection: Additional results and experimental details

Example of the input images for the promotional input scenarios, considered an example of the prompt with Exogenous Feature Injection as plain text.

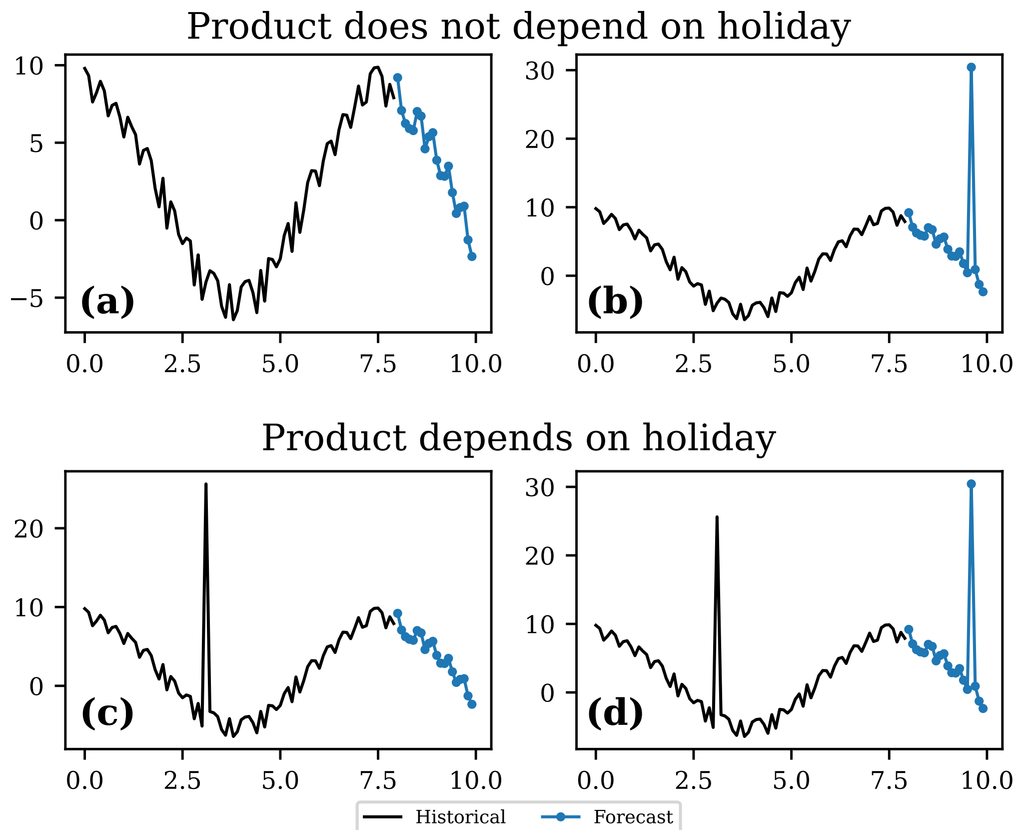


Figure 12: Examples of the four different promotional scenarios considered in the Exogenous Feature Injection section of the main text. The top row considers the case where the product is not dependent on the holiday, but the user provides holiday information. Hence, the true label for (a) is reasonable and for (b) is unreasonable. The bottom row considers the case where the product depends on the holiday. Hence, the true label for (c) is unreasonable, since the forecast is missing the demand spike, and for (d) is reasonable.

User Prompt

You are shown an image of historical data (in black) and a forecast (in blue). Based on the historical trend, assess whether the forecast is reasonable.
A reasonable forecast should generally follow the same direction and capture any seasonal trends if there are any. Note, there is a holiday at $t = 0.321$ in the historical and a second holiday at time $t = 8.47$ in the forecast that may affect the demand.
Please provide a brief explanation (1–2 sentences) justifying your decision. Then present your final answer using one of the following options, wrapped in <answer> tags:
<answer> 1 </answer> — The forecast is reasonable. <answer> 2 </answer> — The forecast is unreasonable.

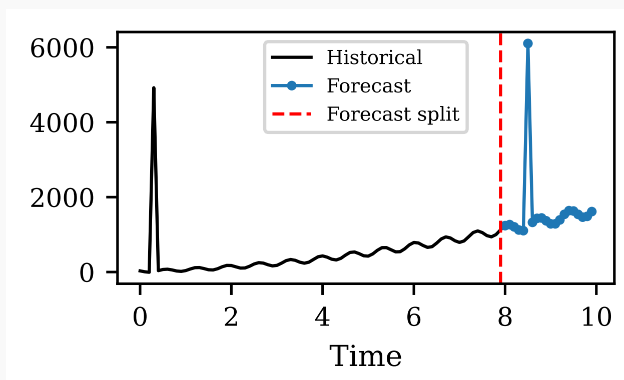
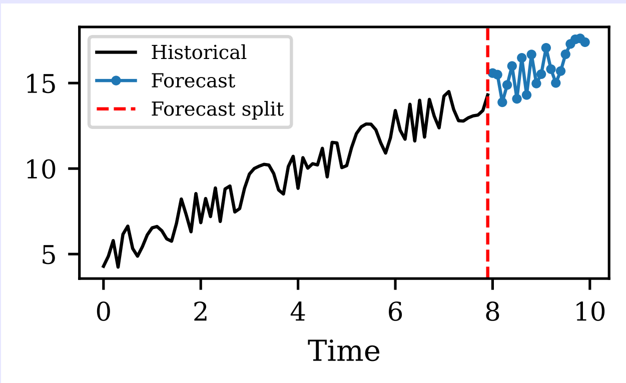


Figure 13: An example prompt for the exogenous signal scenarios in the Exogenous Feature Injection section of the main text. In this scenario, the forecast correct accounts for holiday spikes.

Examples of LLM responses for all Exogenous Feature Injection experiment scenarios

Image Input & LLM Response

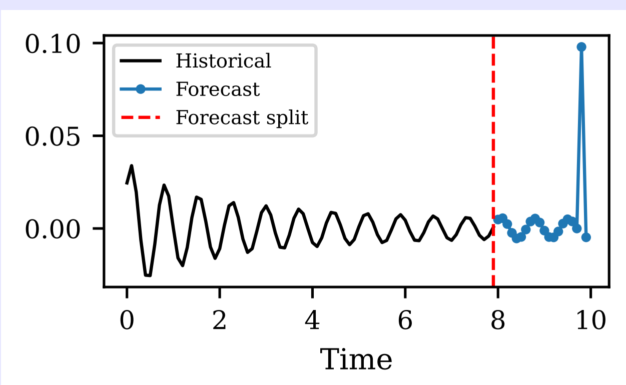


LLM Response

Looking at the image, I can see that the forecast (blue line) continues the same upward trend established by the historical data (black line), and importantly, it maintains the regular oscillation patterns present in the historical data. The forecast also appears to accommodate the expected holiday impact at $t = 8.24$ similarly to how the historical data showed variations around the holiday at $t = 4.29$.

Figure 14: Example of an LLM holiday response for scenario (a) in the Exogenous Feature Injection section of the main text. The LLM correctly recognizes that since the the product does not have a significant spike during the holiday in the forecast history, there should not be a spike in the forecast.

Image Input & LLM Response

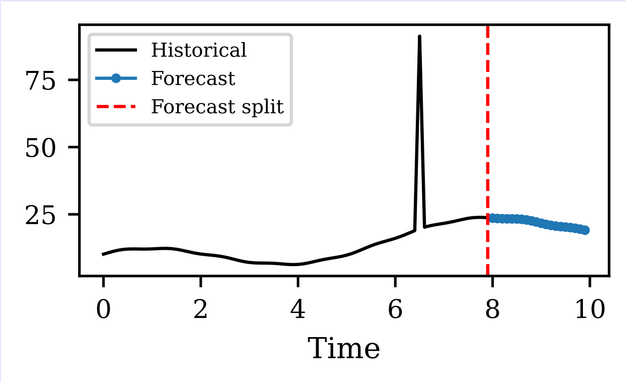


LLM Response

Looking at the image, I can see that the historical data (black line) shows a clear cyclical pattern with decreasing amplitude over time. The forecast (blue line) maintains the cyclical pattern but doesn't continue the dampening trend, instead showing relatively stable oscillations. Additionally, the holiday spike at $t = 9.82$ in the forecast appears significantly larger than the comparable holiday spike in the historical data at $t = 1.88$, which doesn't align with the established pattern. These inconsistencies suggest the forecast doesn't properly capture the historical trend characteristics.

Figure 15: Example of an LLM holiday response for scenario (b) in the Exogenous Feature Injection section of the main text. The LLM correctly recognizes that the spikes at $t = 9.82$ is much larger then the comparable spike in the history, and hence it flags the example.

Image Input & LLM Response

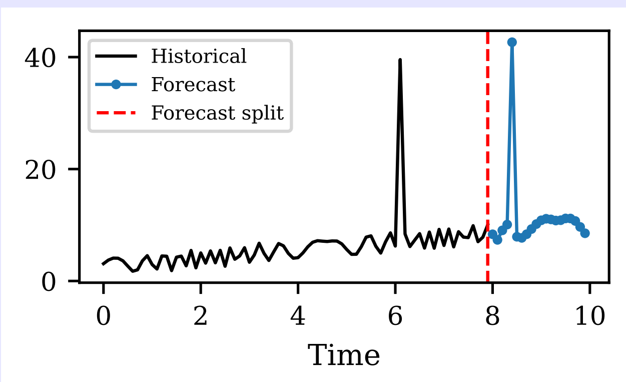


LLM Response

Looking at the image, I can see the historical data shows a relatively flat pattern with a massive spike at $t = 6.53$ (the first holiday), reaching a value around 80. However, the forecast doesn't show any comparable spike at $t = 8.87$ where the second holiday occurs, instead maintaining a smooth, gradually declining trend around the $20 - 25$ value range. Given that holidays typically create similar patterns in time series data, the forecast fails to capture this critical seasonal effect from the first holiday.

Figure 16: Example of an LLM holiday response for scenario (c) in the Exogenous Feature Injection section of the main text. The LLM correctly recognizes that the spike is not comparable at $t = 8.87$, and hence it correctly flags the forecast.

Image Input & LLM Response



LLM Response

Looking at the image, I can see that the historical data shows a relatively stable baseline with a significant spike at $t = 6.09$ (the first holiday). The forecast appropriately captures a similar spike pattern at the second holiday ($t = 8.43$) and maintains a reasonable baseline level consistent with the recent historical trend. The magnitude of the holiday spike in the forecast is proportionate to the historical holiday spike, and the forecast continues the general pattern observed in the historical data.

Figure 17: Example of an LLM holiday response for scenario (d) in the Exogenous Feature Injection section of the main text. The LLM correctly recognizes that the forecast maintains the trend of the historical data and accuractely captures the spikes given at the holiday promotional dates.

Real-world forecasts from foundation models: Additional experimental details and results

Example of probabilistic forecasting prompt input

User Prompt

You are shown an image of historical data (in black) and a forecast (in blue). Your task is to assess whether the forecast appears visually reasonable.
A reasonable forecast should generally follow the same direction as the historical trend and reflect any clear seasonal patterns, if present.
IMPORTANT: Only label a forecast as unreasonable if there is an obvious and significant mismatch — for example, if the forecast goes in the opposite direction of the trend, ignores strong seasonal patterns, or shows extreme jumps that are not supported by the historical data.
Minor deviations or slight over/underestimates are acceptable and should still be considered reasonable.
Please provide a brief explanation (1–2 sentences) justifying your decision. Then present your final answer using one of the following options, wrapped in <answer> tags:
<answer> 1 </answer> — The forecast is reasonable. <answer> 2 </answer> — The forecast is unreasonable.
If you find the forecast unreasonable, clearly explain what makes it obviously inconsistent.

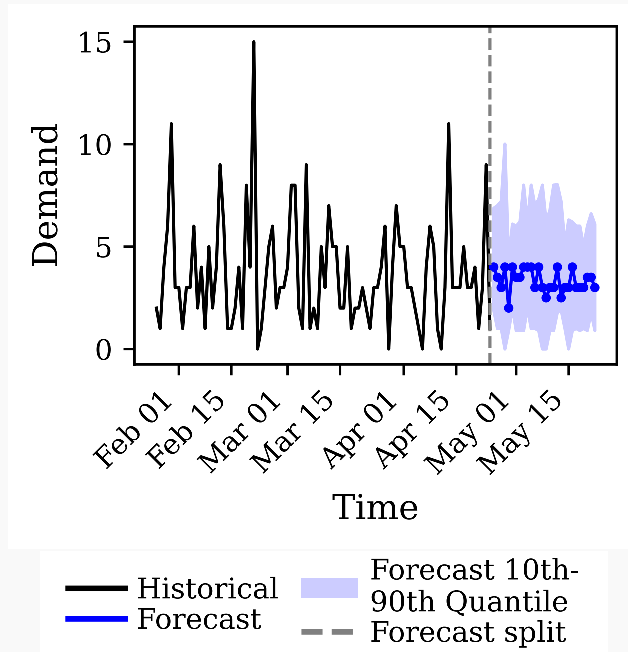
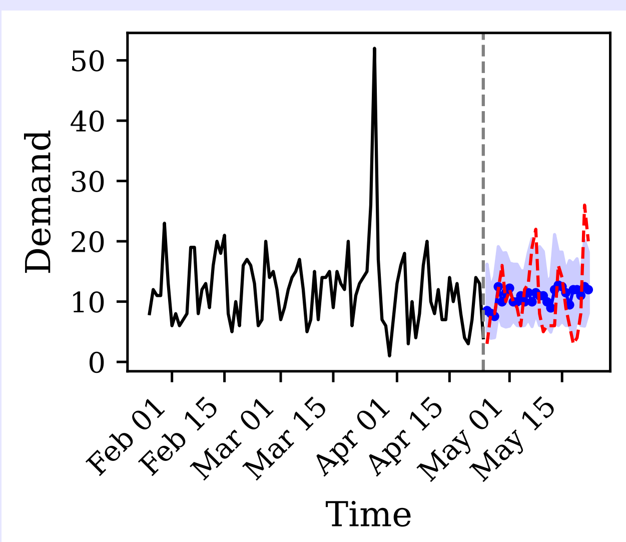


Figure 18: An example prompt for the M5 dataset problems.

Additional anecdotes of flagged forecasts on the M5 dataset with Chronos

Image Input & LLM Response



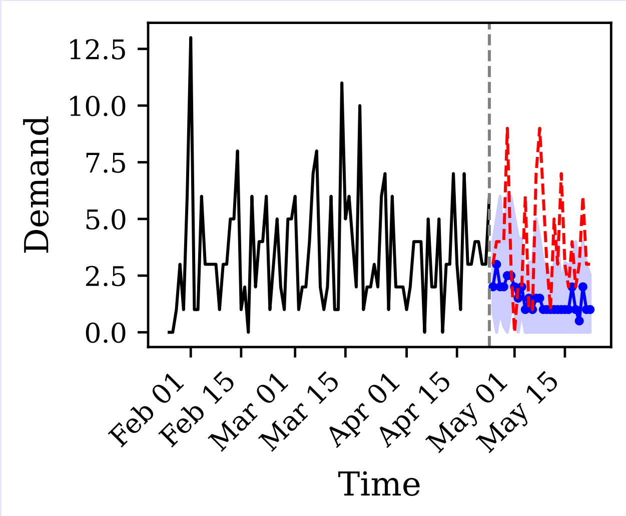
LLM Response

Looking at this image, I can see historical data (black line) that shows considerable volatility with frequent oscillations between approximately 5 and 20, plus one major spike reaching around 50. The blue forecast line appears much smoother and predicts values around 10-15 with minimal fluctuation.

The forecast is unreasonable because it fails to capture the significant volatility pattern that's clearly present in the historical data. The historical series shows regular ups and downs with occasional large spikes, but the forecast appears much smoother without accounting for this established pattern of fluctuation.

Figure 19: Example of a *incorrectly* flagged forecast by the Claude 3.7 Sonnet. This error is due to the fact that the single spike in the historical data caused the LLM to overbias the expected volatility, despite the majority of the data following the forecasted pattern.

Image Input & LLM Response



LLM Response

Looking at the image, I can see historical data (black line) that shows significant volatility with frequent spikes between 0 and approximately 12.5 units. The blue forecast line that follows, however, appears much flatter and lacks the characteristic volatility pattern shown in the historical data.

While the general level of the forecast might be within a reasonable range, it fails to reflect the significant up and down movements that are consistently present throughout the historical period. The forecast is unreasonably smooth compared to the highly volatile historical data. The historical series shows frequent spikes and significant amplitude in its movements, but the forecast shows minimal volatility, suggesting a fundamental mismatch in capturing the time series behavior.

Figure 20: Example of another correctly flagged forecast by LLM forecast critic.