DO TIME-SERIES QA SYSTEMS READ THE TIME SERIES? EVIDENCE USE AND REASONING RELIABILITY

Anonymous authors
Paper under double-blind review

ABSTRACT

In recent years, time-series question answering (QA) systems have made significant progress. However, generating a correct answer does not show whether retaining the supplied numerical series improves task performance, nor whether the prediction is sensitive to changes in that input. While some systems provide rationales, answer accuracy also does not show whether their numerical claims are grounded in the supplied series or whether the stated inference is valid. In this work, we focus on evaluating four time-series QA systems: TimeOmni-1, ChatTS, TimeOmni-VL, and Time-MQA. First, for three systems with released evaluation data, we reproduce their reported results and compare the performance of the systems with their backbones. Then, we introduce a benchmark named COMMON-TSQA, which collects public evaluation datasets from existing time-series benchmarks and unifies their sample representation, task definitions, and answer schemas, while evaluating each system through its own interface under common evaluation criteria. The evaluation uses the original condition and six interventions while keeping the question and target fixed. Our analysis shows that aggregate performance alone can obscure how systems use numerical evidence. Similar task-level scores can arise despite substantial changes in individual predictions. Some interventions induce simple fallback behavior rather than preserved task ability. We also evaluate rationales for factual grounding, inference validity, and consistency with the final answer. We find that rationales often contain time-series claims unsupported by the input. Moreover, the rationale audit shows that agreement between a rationale and its final answer can coexist with incorrect numerical descriptions or invalid intermediate inferences.

1 INTRODUCTION

Recent time-series question-answering (QA) systems extend large language models to tasks such as forecasting, pattern recognition, causal diagnosis, cross-series comparison, and decision-making from numerical time series (Xie et al., 2025; Kong et al., 2025; Guan et al., 2026a;b). Specialized systems report substantial improvements over their underlying language-model backbones, but final-answer accuracy alone does not identify what evidence supported a prediction. As illustrated in Figure 1, the same benchmark answer may be compatible with temporal structure in the series, coarse numerical cues, or information already available from the question, answer choices, and surrounding context. A visible natural-language rationale does not resolve this ambiguity: it may appear coherent while misdescribing the supplied series or drawing an invalid conclusion from its own stated premises.

Prior work has questioned whether strong time-series performance reflects meaningful use of the numerical input, and separate work has examined the correctness of generated time-series explanations (Merrill et al., 2024; Sivalingam et al., 2026). These observations motivate a question: when the numerical evidence for the same item is changed, how does the prediction respond, and does the response remain appropriate for the modified evidence? For systems that expose visible rationales, a question is whether the numerical facts stated in the reasoning are supported by the supplied series, and whether the subsequent conclusion follows from those facts.

We study these questions on four time-series question answering (QA) systems: TimeOmni-1 (Guan et al., 2026a), ChatTS (Xie et al., 2025), TimeOmni-VL (Guan et al., 2026b), and Time-MQA (Kong et al., 2025). For the three systems for which their original evaluation data are available, we first

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

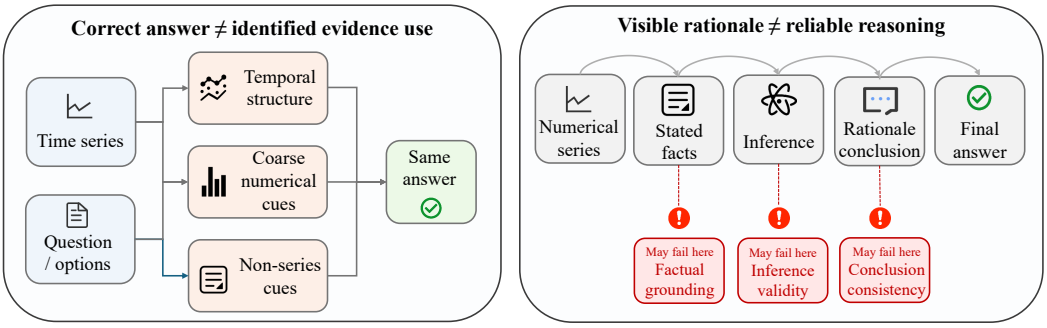


Figure 1: Final-answer correctness and rationales leave distinct ambiguities. (a) The same answer can be compatible with temporal structure, coarse numerical cues, or non-series cues. (b) Even when a rationale is visible, failures can arise in factual grounding, inference validity, or consistency with the final answer.

reproduce the results and compare them to their corresponding backbones. We then construct COMMON-TSQA, a common evaluation benchmark assembled from public time-series evaluation datasets. Rather than introducing new task semantics, COMMON-TSQA retains the source-defined questions, targets, and task objectives while standardizing the representation and scoring needed for cross-system evaluation. Each system is evaluated through its native input and output interface, and outputs are scored using the same task-specific criteria. Each sample is evaluated under original condition and six interventions with fixed question and original target.

Our evaluation separates three questions that are often conflated by final-answer performance. First, we ask whether specialization improves the systems' native task performance (RQ1). Second, we examine how predictions respond when the numerical evidence is changed, including whether apparent performance changes reflect prediction instability, simple fallback behavior, or an answer that should itself change under the intervention (RQ2). Third, for systems with visible reasoning, we audit the path from numerical evidence to stated facts, inference, and final conclusion (RQ3).

Our main contributions are:

- We construct COMMON-TSQA, a common evaluation benchmark across six time-series QA tasks, allowing four specialized systems to be evaluated on the same samples and scoring criteria.
- We develop an intervention-based analysis of numerical-evidence use. The analysis distinguishes prediction turnover, fallback behavior under evidence removal or simplification, and cases in which the intervention changes the answer implied by the numerical input.
- We audit visible rationales for factual grounding, inference validity, and consistency with the final answer, identifying failures at distinct stages of the evidence-to-answer reasoning path.

2 RELATED WORK

Time-series language models. Recent time-series language models extend beyond forecasting to a broad range of understanding and reasoning tasks. CHATTS (Xie et al., 2025) uses a dedicated time-series encoder and aligns its temporal representations with a language model through synthetic alignment data and supervised fine-tuning. CHATTIME (Wang et al., 2025) treats time series as a "foreign language" and models numerical and textual data in a unified framework. TIME-MQA (Kong et al., 2025) builds a large-scale multi-task corpus covering forecasting, imputation, anomaly detection, classification, and open-ended question answering, while TIMEOMNI-1 (Guan et al., 2026a) combines supervised fine-tuning and reinforcement learning for scenario understanding, causality discovery, event-aware forecasting, and decision-making. More recent systems further develop task-specific mechanisms for time-series reasoning. PATRA (Lu et al., 2026) introduces pattern-aware alignment for trend and seasonality together with a task-aware balanced reward for

heterogeneous TSQA tasks. ARTIST (Messica et al., 2026) interleaves reasoning with question-adaptive temporal segment selection through a controller–reasoner architecture. TIMEOMNI-VL (Guan et al., 2026b) instead adopts a vision-centric formulation that maps between time series and images for both understanding and numerical generation.

Evidence dependence and behavioral evaluation. Prior work has questioned whether strong time-series performance necessarily reflects effective use of all available inputs. Tan et al. (2024) ablate the language-model components of OneFitsAll, Time-LLM, and CALF, finding that removing the pretrained language model or replacing it with substantially simpler modules often leaves forecasting performance largely unchanged. CIK (Williams et al., 2025) approaches the question from the data side by constructing forecasting tasks in which textual context is necessary for accurate prediction. In multimodal forecasting, Zhang et al. (2026) find that textual information is most useful when it provides information that cannot be recovered from the time series alone. Merrill et al. (2024) study the role of the numerical input more directly. On an initial multiple-choice benchmark with a single time series, models retain much of their performance when the series is removed. Then they introduce a harder two-series setting, where access to the series provides only limited benefit for most evaluated models. These studies show that strong task performance does not by itself establish substantial dependence on every available input source.

Faithfulness and reliability of generated reasoning. A correct final answer does not guarantee that a rationale provides a reliable account of the evidence or reasoning leading to that answer. Jacovi & Goldberg (2020) distinguish faithfulness from plausibility, with faithfulness concerning whether an explanation reflects factors that actually influence a model's prediction. ERASER (DeYoung et al., 2020) evaluates rationales through human-identified evidence and its effect on model predictions. For chain-of-thought reasoning, Lanham et al. (2023) intervene on intermediate reasoning through perturbation and paraphrasing and measure the resulting changes in the answer. Turpin et al. (2023) show that models can produce convincing rationales that fail to disclose the influence of biasing features. More recent work develops benchmarks for evaluating such failures. FAITHCOT-BENCH (Shen et al., 2026) studies instance-level faithfulness of chain-of-thought reasoning, while RFEVAL (Han et al., 2026) evaluates reasoning faithfulness under counterfactual interventions. In the time-series domain, TSQUERYBENCH (Sivalingam et al., 2026) studies reference-free evaluation of numerical correctness in generated time-series explanations.

3 COMMON-TSQA: A COMMON EVALUATION BENCHMARK

Reported results from current time-series reasoning systems are difficult to compare fairly. Some systems do not release the evaluation data used in their papers, while other systems are evaluated on different tasks, datasets, or splits. Therefore, performance differences may depend on the test set rather than the systems. We need a shared test set where all systems are evaluated on the same canonical samples, task definitions, targets, and evaluation criteria. For this purpose, we build COMMON-TSQA. Instead of creating a new task corpus, we construct it from evaluation sets of existing time-series benchmarks, retaining the original tasks and targets defined by the sources. Using several benchmarks also broadens the coverage of tasks and domains and reduces dependence on the annotation conventions or question formats of any one benchmark. COMMON-TSQA contains samples from six tasks and is constructed to avoid overlap with the training datasets of the evaluated systems. All systems use the same canonical samples and tasks, with the input formatted according to the interface required by each system.

The public benchmark sources include GIFT-Eval, SciTS, TIME, TSB-AD-U, TSAQA, CaTS-Bench, FactoryBench, SenTSR-Bench, TSRBench, and TSR-Suite-OOD (Aksu et al., 2024; Wu et al., 2026; Qiao et al., 2026; Liu & Paparrizos, 2024; Merzouki et al., 2026). Since these samples already have source-defined targets and task semantics, we only standardize what is needed for cross-system evaluation.

For numerical tasks, we keep the source observations and source-defined targets when they can be mapped deterministically to the benchmark schema. For question answering (QA) tasks, we preserve the question meaning, option semantics, and target label. Each record contains the time series, question, options, target, source split, task and domain metadata, and source grouping information. Details on the sources, conversion rules, and task-domain composition can be found in Appendix B.

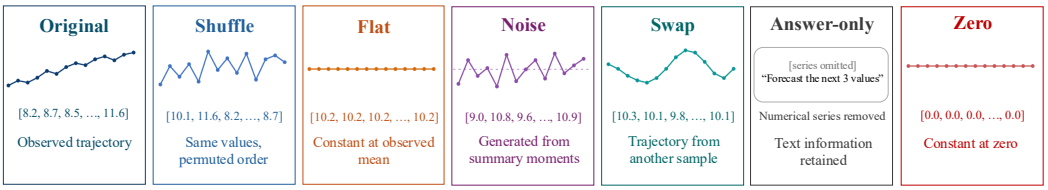


Figure 2: The original input and six interventions to the numerical evidence. Non-series input content is fixed across conditions.

4 Evaluation Settings

4.1 Systems

We evaluate four time-series QA systems: TimeOmni-1 (Guan et al., 2026a), ChatTS (Xie et al., 2025), TimeOmni-VL (Guan et al., 2026b), and Time-MQA (Kong et al., 2025). Each checkpoint is evaluated through its native input interface. Thus, the systems receive the same canonical sample and task content, while the numerical series and accompanying information are rendered in the format required by each system. Details on the backbones, training procedures, and native reproduction are provided in Appendix A.

4.2 Evidence interventions

For each sample, we evaluate the system under the original condition and six interventions to the numerical series. The interventions modify only the numerical evidence; the question, answer choices and their order, task context, ground-truth label, and other non-series information are held fixed. Figure 2 illustrates the seven conditions.

The six interventions are:

- **Shuffle** independently permutes the observed values within each series, preserving its length and value multiset while removing the original temporal order.
- **Flat** replaces the observed values in each series by their mean, approximately preserving the observed level while removing within-series variation.
- **Noise** replaces the observed values with independent Gaussian draws parameterized by the series' observed mean and standard deviation.
- **Swap** replaces the numerical series with a compatible series from another sample.
- **Answer-only** removes the numerical series while retaining the question and all other non-series information.
- **Zero** replaces the numerical values by zero while preserving the series length and input structure.

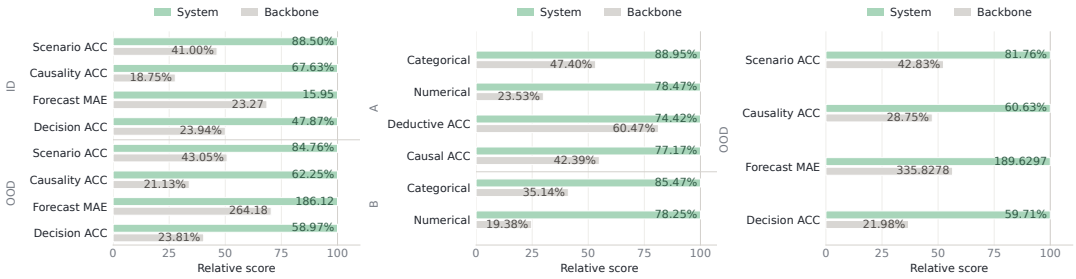
For sample i, we use the notation $z_i = (x_i, q_i, y_i)$, where x_i denotes the numerical series, q_i contains the question and other non-series information, and y_i is the source-defined Original target. Let T_c denote the numerical transformation for condition c, with $T_0(x_i) = x_i$ for original, and let R_s be the input renderer for system s. The prediction under condition c is

$$\hat{y}_{si}^{(c)} = f_s(R_s(T_c(x_i), q_i)) \,. \tag{1}$$

Note that $\hat{y}_{si}^{(c)}$ is evaluated against the Original target y_i. Appendix C specifies the transformation details, masking rules, degenerate cases, numerical rendering, and Swap donor selection procedure.

4.3 Metrics

For forecasting, we report mean absolute scaled error (MASE) (Hyndman & Koehler, 2006). Lower MASE indicates better forecasting performance.



(a) TimeOmni-1 vs. Qwen2.5-7B (b) ChatTS vs. Qwen2.5-14B (c) TimeOmni-VL vs. BAGEL-7B

Figure 3: Systems and matched backbones on native evaluations. Bars are normalized separately within each task so that the better result equals 100; bar heights are intended only for within-task comparison. Labels report the original metric values.

For anomaly detection, we report binary accuracy over the normal and anomalous labels. Pattern recognition, causal diagnosis, and decision intervention are evaluated by accuracy. Cross-series comparison uses relation accuracy, which counts a prediction as correct when its canonical relation matches the target relation.

5 Experiments

5.1 Does specialization improve native task performance? (RQ1)

We consider three systems with released evaluation data: TimeOmni-1, ChatTS, and TimeOmni-VL. We first reproduce their reported evaluations before comparing each specialized system with its matched backbone. The reproduced results are close to the reported values on most native metrics. ChatTS differs by at most 4.0% across its six reported metrics, while six of the eight TimeOmni-1 metrics differ by at most 3.3%. The larger reproduction discrepancies occur in forecasting, particularly for TimeOmni-1 on the out-of-distribution (OOD) split. Full reproduction results are reported in Appendix A.2.

The specialization gains are consistent across the available native comparisons. As shown in Figure 3, TimeOmni-1 outperforms Qwen2.5-7B (Yang et al., 2025) across all of its reported ID and OOD tasks, including both accuracy-based tasks and forecasting. ChatTS likewise improves over Qwen2.5-14B (Yang et al., 2025) on all six reported metrics, and TimeOmni-VL improves over BAGEL-7B (Deng et al., 2025) on all four OOD metrics. These results show that specialization consistently improves native task performance for the systems that can be compared with their matched backbones. However, the comparison does not identify how the numerical series contributes to those predictions. A specialized system may achieve higher native performance without every prediction depending on the sample-specific temporal evidence in the same way.

5.2 How Do Systems Respond to Changes in Numerical Evidence? (RQ2)

Table 1 shows that Original is not uniformly the best-performing condition. Flat achieves the lowest forecasting MASE, whereas the effects on discrete tasks vary across systems and interventions. To interpret these differences, we examine whether similar scores hide different predictions, whether removing or simplifying the series induces a recognizable fallback, and whether the correct answer itself changes with the numerical evidence.

Similar accuracy can hide different predictions. To analyze sample-level sensitivity, we compare the semantic answers produced under Original and Answer-only. Correct-to-wrong (C→W) and wrong-to-correct (W→C) record changes in correctness against the original ground-truth answer. Changed records any difference between the two semantic answers, including a change from one incorrect answer to another. Table 2 shows that ChatTS and TimeOmni-VL retain similar pattern-recognition accuracy after the series is removed, despite changing many individual predictions. Their C→W and W→C transitions largely offset each other on the jointly valid samples.

Table 1: Scores of four time-series QA systems under original and six interventions. All conditions are scored against the original ground-truth answer. Bold indicates the best score. "–" indicates that the system produced no scorable forecasting outputs.

Task	Metric	Model	Original	Shuffle	Flat	Noise	Swap	Answer-only	Zero
Forecasting	MASE ↓	TimeOmni-1	1.739	1.929	**1.321**	2.060	26.848	17.418	7.235
		ChatTS	–	–	–	–	–	–	–
		Time-MQA	2.258	2.287	**2.106**	2.260	23.380	38.241	12.706
		TimeOmni-VL	2.650	3.001	**2.604**	3.291	36.109	38.546	13.221
Anomaly detection	Accuracy (%) ↑	TimeOmni-1	**60.70**	55.44	45.61	46.32	45.96	45.61	45.61
		ChatTS	62.46	**68.88**	45.61	58.60	54.39	54.39	45.61
		Time-MQA	64.91	**65.46**	45.61	56.14	52.28	45.61	45.61
		TimeOmni-VL	**70.88**	56.49	46.32	46.67	47.02	45.61	45.61
Pattern recognition	Accuracy (%) ↑	TimeOmni-1	**60.51**	47.67	26.00	44.14	52.01	52.81	25.52
		ChatTS	**59.91**	48.15	23.76	45.75	59.07	59.23	23.76
		Time-MQA	**55.06**	51.85	33.07	50.08	52.65	43.98	33.23
		TimeOmni-VL	58.35	50.40	18.62	51.20	56.02	**58.91**	19.42
Cross-series comparison	Accuracy (%) ↑	TimeOmni-1	58.79	57.06	35.78	**59.41**	55.51	50.87	38.49
		ChatTS	**54.93**	49.32	31.91	48.74	46.86	46.23	26.31
		Time-MQA	46.81	44.49	29.01	**47.64**	40.23	34.82	25.53
		TimeOmni-VL	53.00	**53.31**	26.89	44.68	35.20	35.59	19.73
Causal diagnosis	Accuracy (%) ↑	TimeOmni-1	33.67	31.90	30.89	**33.92**	25.32	27.85	21.27
		ChatTS	28.35	28.86	27.09	29.11	30.63	**31.39**	28.86
		Time-MQA	27.19	25.57	24.56	**27.34**	23.29	19.24	16.96
		TimeOmni-VL	**33.42**	29.87	25.82	30.13	26.58	29.62	19.75
Decision intervention	Accuracy (%) ↑	TimeOmni-1	**43.43**	23.43	26.29	21.14	34.29	21.14	29.16
		ChatTS	26.14	21.71	28.57	24.57	24.57	26.29	**29.71**
		Time-MQA	26.17	24.00	**27.43**	20.57	19.43	22.29	20.57
		TimeOmni-VL	**49.71**	23.43	30.29	30.86	42.86	27.43	26.29

Thus, similar aggregate scores can conceal changes in which samples are answered correctly. The predictions are sensitive to the numerical input, but this sensitivity alone does not establish that the input is used appropriately. Appendix D.1 reports the other discrete tasks.

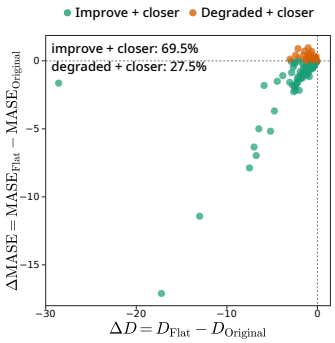
Table 2: Pattern-recognition transitions from Original to Answer-only.

System	C→W (%)	W→C (%)	Changed (%)
TimeOmni-1	24.11	16.34	48.22
ChatTS	14.13	12.84	30.50
Time-MQA	17.62	11.88	34.65
TimeOmni-VL	19.69	20.38	47.95

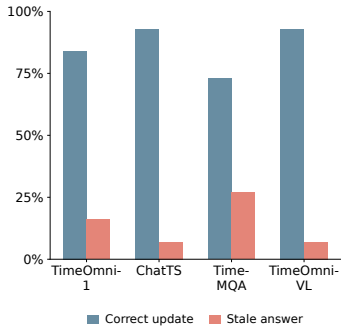
Answer-only induces single-class fallback in anomaly detection. We inspect the predicted label distribution to determine whether the remaining Answer-only accuracy reflects discrimination between samples. All four systems predict a single class: TimeOmni-1, Time-MQA, and TimeOmni-VL label every sample as *normal*, whereas ChatTS labels every sample as *anomalous*. A constant prediction is correct whenever a sample belongs to the selected class, so its accuracy depends only on that class's frequency. The residual Answer-only score reflects a fixed-class fallback rather than retained discrimination between normal and anomalous samples.

Flat shifts forecasts toward a history-mean reference. To understand the forecasting behavior associated with Flat, we compare Original and Flat on samples that are scorable under both conditions. We use a constant forecast equal to the mean of the Original history as a fixed reference. Let D denote the mean absolute distance from a model forecast to this reference, divided by the same Original-history scale used for MASE. Unlike MASE, which measures error against the future observations, D measures similarity to the reference forecast. Figure 4(a) plots the changes in these two quantities: $\Delta D < 0$ indicates movement toward the reference, and $\Delta\mathrm{MASE} < 0$ indicates improved forecasting.

Flat moves nearly all paired forecasts toward the history-mean reference. For TimeOmni-1 and Time-MQA, this movement is more often accompanied by lower error; for TimeOmni-VL, it is more often accompanied by higher error. The common response is therefore a shift toward a simple level



(a) TimeOmni-1 original and flat forecasting results.



(b) Correct updates vs. stale answers.

Figure 4: Paired diagnostics of intervention responses. (**a**) Each point is a forecasting sample scorable under both Original and Flat. The horizontal axis is $\Delta D = D^{\text{Flat}} - D^{\text{Original}}$, the change in scaled distance to the fixed Original-history-mean forecast; the vertical axis is $\Delta\text{MASE} = \text{MASE}^{\text{Flat}} - \text{MASE}^{\text{Original}}$. Improve + closer denotes $\Delta D < 0$ and $\Delta\text{MASE} < 0$; Degraded + closer denotes $\Delta D < 0$ and $\Delta\text{MASE} > 0$. (**b**) On cross-series sample–intervention pairs whose correct answer changes, the bars show correct updates and stale answers. Correct updates match the recomputed target, and stale answers match the original ground-truth answer.

forecast, not a uniform improvement in prediction. This distinction helps explain why removing temporal variation can improve some forecasts without demonstrating more effective modeling of the original temporal structure. Full paired results and an alternative last-value reference are provided in Appendix D.2.

Errors against the original answer can include appropriate updates. An intervention can change the answer implied by the input. For instance, flattening two series makes both ranges zero, so a range-comparison question may no longer have its original answer. We therefore recompute answers for the subset of cross-series questions whose targets can be determined unambiguously from the transformed series. For sample–intervention pairs whose correct answer changes, a *correct update* matches the recomputed answer, whereas a *stale answer* matches the original ground-truth answer. These labels refer to the two targets, not to whether the system repeats its own Original prediction. Figure 4(b) compares their relative shares within these two categories; other errors are reported separately in Appendix D.3.

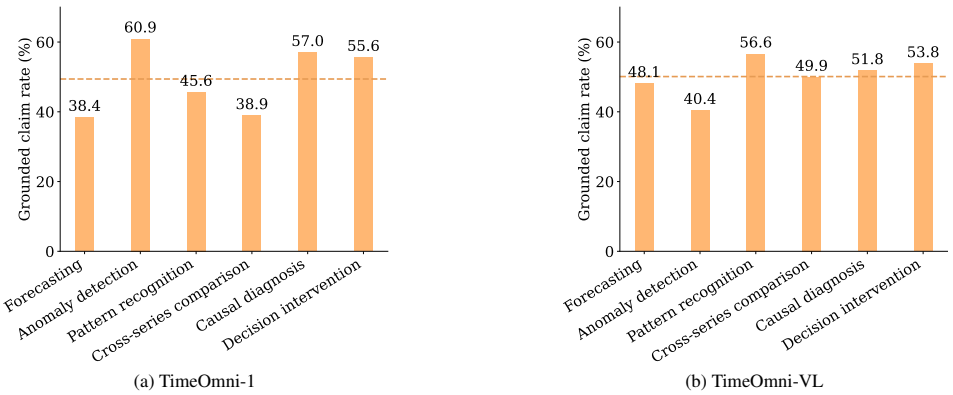
Correct updates are more frequent than stale answers for every system, and scoring against the recomputed answers increases accuracy on the eligible subset. The distinction is particularly relevant to Flat and Zero, which change the target much more often than Shuffle in this subset. Some errors against the original answer therefore reflect appropriate responses to the transformed evidence rather than a loss of task ability. However, stale answers and other errors remain, showing that appropriate updating is not consistent across all samples and interventions.

5.3 Where Does the Evidence-to-Answer Reasoning Path Fail? (RQ3)

In this section, we evaluate the visible rationales of TimeOmni-1 and TimeOmni-VL under Original using GPT-5.6 Sol[1] (OpenAI, 2026) as the LLM judge, without providing the ground-truth answer. We first examine individual numerical claims and then distinguish factual grounding, inference validity, and conclusion consistency at the output level. Appendix F provides the factual-grounding prompt, along with the protocol and results of an independent human validation of the LLM-based adjudication.

Numerical claims are only partially grounded in the input. We extract atomic claims about the observed time series and check them against the supplied numerical evidence. Grounded claims are judged supported by that evidence; the remaining labels distinguish contradicted statements

[1]Accessed via the OpenAI API between July 19 and September 24, 2026.



(a) TimeOmni-1 (b) TimeOmni-VL

Figure 5: Grounded claim rate across six tasks under Original. Dashed lines indicate the overall rates, 49.4% for TimeOmni-1, and 50.1% for TimeOmni-VL.

Table 3: Output-level rationale audit under original setting. Values are pass rates (%) for the reported factual-grounding, inference-validity, and conclusion-consistency criteria. Conclusion consistency concerns agreement with the system's own final answer.

	Factual grounding	Inference validity	Conclusion consistency
TimeOmni-1	2.1	27.3	64.0
TimeOmni-VL	11.2	34.3	92.2

from statements that cannot be verified. Figure 5 reports the grounded claim rates across tasks. Only about half of the extracted claims are supported for either system, but similar overall rates conceal different task-level weaknesses. TimeOmni-1 has lower grounding rates in forecasting and cross-series comparison than in anomaly detection, whereas TimeOmni-VL has its lowest rate in anomaly detection. Thus, neither system provides uniformly reliable numerical descriptions across tasks.

Conclusion consistency does not guarantee grounded or valid reasoning. We evaluate the rationale beyond its individual claims. Factual grounding checks whether the stated premises are supported by the numerical input under the strict output-level criterion. Inference validity asks whether all assessable key reasoning steps are valid given the stated premises, even when those premises are factually incorrect. Conclusion consistency checks whether the rationale's conclusion agrees with the system's final answer, not whether that answer is correct.

Table 3 shows low pass rates under the strict grounding criterion, and fewer than half of the outputs pass inference validity for either system. Conclusion consistency is substantially higher, especially for TimeOmni-VL. These measures capture different requirements: supported individual statements do not establish a grounded rationale, and agreement with the final answer does not establish that the preceding inference is valid.

Case 1 in Appendix E illustrates the distinction. The rationale correctly reports amplitudes of 14 and 7.29, but concludes that the latter is larger; its final answer follows that mistaken comparison. The relevant numerical values are correct and the conclusion agrees with the answer, but the inference is invalid. The quantitative results and cases show why checking only the final conclusion overlooks failures in describing the evidence and reasoning from it.

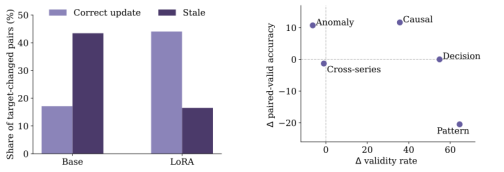
6 Discussion

We relate the observed behavior to each system's public training design and available checkpoints. The analysis identifies limitations that remain after specialization, rather than attributing errors to a particular training choice. Appendix G documents the public evidence and release constraints; a training signal that is not documented is not assumed to be absent.

TimeOmni-1: task-level optimization does not ensure grounded intermediate reasoning. TimeOmni-1 combines human-guided chain-of-thought (CoT) supervision with reinforcement learning on forecasting, causal, and decision tasks. Its native performance gains therefore coexist with the grounding and inference failures identified in RQ3. This contrast shows that task and CoT supervision can improve final-answer performance without ensuring that the numerical statements and subsequent inferences in a rationale are reliable. The public objectives emphasize task-level outcomes, leaving intermediate numerical verification as a direction for further study.

ChatTS: broad task coverage does not ensure reliable evidence-responsive answers. Our review of the ChatTS corpus identifies multi-series, cross-series, causal, and rationale-like supervision (Appendix G.2). The observed errors therefore occur within task families represented in the public training data, rather than only on wholly unrepresented types of questions. RQ2 also shows both appropriate updates and incorrect responses after the evidence changes. These observations motivate examining how training connects numerical evidence to the required answer, rather than treating broad task coverage as sufficient evidence of reliable numerical reasoning.

Time-MQA: adaptation changes output validity and correctness unevenly. The released LoRA adapter allows a comparison between a fixed public base and the same base with specialization applied. Figure 6(a) shows more correct updates and fewer stale answers on target-changed cross-series pairs after adaptation. The gains are not uniform across interventions: the largest increases in correct updates occur under Flat and Zero. Figure 6(b) further separates output validity, the fraction of outputs with recoverable semantic answers, from paired-valid accuracy, which is measured on samples with valid answers from both checkpoints. In pattern recognition, adaptation substantially increases validity but lowers paired-valid accuracy; in causal diagnosis, both improve. Thus, an aggregate accuracy gain can reflect different changes in output validity and correctness across tasks. The comparison uses matched inputs, decoding, and scoring, but is a checkpoint contrast rather than an exact training-stage ablation. Details are provided in Appendix G.3.



(a) Target-aware Base–LoRA responses. (b) Validity and paired-valid accuracy changes

Figure 6: **(a)** Correct updates and stale answers on 327 cross-series pairs whose targets change after intervention; other errors are reported in Appendix G.3. **(b)** Base→LoRA changes in output validity and paired-valid accuracy across discrete tasks.

TimeOmni-VL: answer consistency is narrower than reasoning reliability. Despite its CoT-based training, TimeOmni-VL exhibits a substantial gap between conclusion consistency and the grounding and inference criteria in RQ3. Therefore, aligning the rationale's conclusion with the final answer is a narrower success than providing a numerically supported and valid derivation.

These findings distinguish three requirements that specialization need not improve together: producing valid answers, responding appropriately to changed numerical evidence, and providing grounded, logically valid rationales.

7 Conclusion

We evaluate four time-series QA systems through native reproduction, numerical-evidence interventions, and visible-rationale audits. Specialization improves native task performance, but this does not establish reliable use of numerical evidence or reliable visible reasoning. Similar task-level scores can conceal substantial prediction turnover, while intervention gains can accompany shifts toward simple forecasting references. When the numerical input changes the correct answer, scoring against the original target can also penalize appropriate updates. The rationale audits further distinguish supported numerical claims, valid inference, and consistency with the final answer. These findings show that predictive performance, evidence-responsive behavior, and rationale reliability should be evaluated separately.

AI USE STATEMENT

In this work, we used generative AI tools to assist with code editing and debugging, literature search, and manuscript drafting and editing. We also used generative AI tools to support the rationale-grounding analysis through an LLM-based judge. We did not use generative AI to formulate or prove mathematical claims. All AI-assisted work was reviewed by the authors. Experimental results and quantitative analyses were independently verified, AI-assisted code was inspected and tested, cited literature was checked against the original sources. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

REFERENCES

Taha Aksu, Gerald Woo, Juncheng Liu, Xu Liu, Chenghao Liu, Silvio Savarese, Caiming Xiong, and Doyen Sahoo. GIFT-eval: A benchmark for general time series forecasting model evaluation. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. Eraser: A benchmark to evaluate rationalized nlp models. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4443–4458, 2020.

Tong Guan, Zijie Meng, Dianqi Li, Shiyu Wang, Chao-Han Huck Yang, Qingsong Wen, Zuozhu Liu, Sabato Siniscalchi, Ming Jin, and Shirui Pan. Timeomni-1: Incentivizing complex reasoning with time series in large language models. In *International Conference on Learning Representations*, volume 2026, pp. 152139–152170, 2026a.

Tong Guan, Sheng Pan, Johan Barthelemy, Zhao Li, Yujun Cai, Cesare Alippi, Ming Jin, and Shirui Pan. Timeomni-VL: Unified models for time series understanding and generation. In *Forty-third International Conference on Machine Learning*, 2026b.

Yunseok Han, Yejoon Lee, and Jaeyoung Do. RFEval: Benchmarking reasoning faithfulness under counterfactual reasoning intervention in large reasoning models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Zelin He, Boran Han, Xiyuan Zhang, Shuai Zhang, Haotian Lin, Qi Zhu, Haoyang Fang, Danielle C. Maddix, Abdul Fatir Ansari, Akash Chandrayan, Abhinav Pradhan, Bernie Wang, and Matthew Reimherr. SenTSR-bench: Thinking with injected knowledge for time-series reasoning. In *The 29th International Conference on Artificial Intelligence and Statistics*, 2026.

Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688, 2006.

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4198–4205, 2020.

Baoyu Jing, Sanhorn Chen, Lecheng Zheng, Boyu Liu, Zihao Li, Jiaru Zou, Tianxin Wei, Zhining Liu, Zhichen Zeng, Ruizhong Qiu, et al. Tsaqa: Time series analysis question and answering benchmark. In *Proceedings of the Fifth Workshop on Generation, Evaluation and Metrics (GEM)*, pp. 944–979, 2026.

Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. Time-mqa: Time series multi-task question answering with context enhancement. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29736–29753, 2025.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Qinghua Liu and John Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. *Advances in Neural Information Processing Systems*, 37:108231–108261, 2024.

Junkai Lu, Peng Chen, Xingjian Wu, Yang Shu, Chenjuan Guo, Christian S. Jensen, and Bin Yang. PATRA: Pattern-aware alignment and balanced reasoning for time series question answering. In *Forty-third International Conference on Machine Learning*, 2026.

Mike A Merrill, Mingtian Tan, Vinayak Gupta, Thomas Hartvigsen, and Tim Althoff. Language models still struggle to zero-shot reason about time series. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3512–3533, 2024.

Yanis Merzouki, Coral Izquierdo, Matei Ignuta-Ciuncanu, Marcos Gomez-Bracamonte, Riccardo Maggioni, Alessandro Lombardi, Camilla Mazzoleni, Federico Martelli, Balazs Gunther, Jonas Petersen, et al. Factorybench: Evaluating industrial machine understanding. *arXiv preprint arXiv:2605.07675*, 2026.

Shvat Messica, Jiawen Zhang, Kevin Li, Theodoros Tsiligkaridis, and Marinka Zitnik. Adaptive time series reasoning via segment selection. In *Forty-third International Conference on Machine Learning*, 2026.

OpenAI. GPT-5.6 System Card, 2026. URL `https://deploymentsafety.openai.com/gpt-5-6`.

Zhongzheng Qiao, Sheng Pan, Anni Wang, Viktoriya Zhukova, Yong Liu, Xudong Jiang, Qingsong Wen, Mingsheng Long, Ming Jin, and Chenghao Liu. It's TIME: Towards the next generation of time series forecasting benchmarks. In *Forty-third International Conference on Machine Learning*, 2026.

Xu Shen, Song Wang, Zhen Tan, Laura Yao, Xinyu Zhao, Kaidi Xu, Xin Wang, and Tianlong Chen. Faithcot-bench: Benchmarking instance-level faithfulness of chain-of-thought reasoning. In *International Conference on Learning Representations*, volume 2026, pp. 65918–65948, 2026.

Preetham Sivalingam, Murari Mandal, Dhruv Kumar, and Saurabh Deshpande. TSQuerybench: LLM-as-a-judge for time-series explanations. In *2nd ICML Workshop on Foundation Models for Structured Data*, 2026.

Mingtian Tan, Mike A Merrill, Vinayak Gupta, Tim Althoff, and Thomas Hartvigsen. Are language models actually useful for time series forecasting? *Advances in Neural Information Processing Systems*, 37:60162–60191, 2024.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965, 2023.

Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12694–12702, 2025.

Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, and Alexandre Drouin. Context is key: A benchmark for forecasting with essential textual information. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, 2025.

Wen Wu, Ziyang Zhang, Liwei Liu, Xuenan Xu, Jimin Zhuang, Ke Fan, Qitan Lv, Junlin Liu, Chen Zhang, Zheqi Yuan, et al. Scits: Scientific time series understanding and generation with llms. In *International Conference on Learning Representations*, volume 2026, pp. 114390–114431, 2026.

Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. Chatts: Aligning time series with llms via synthetic data for enhanced understanding and reasoning. *Proceedings of the VLDB Endowment*, 18(8):2385–2398, 2025.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

Fangxu Yu, Xingang Guo, Lingzhi Yuan, Haoqiang Kang, Hongyu Zhao, Lianhui Qin, Furong Huang, Bin Hu, and Tianyi Zhou. TSRBench: A comprehensive multi-task multi-modal time series reasoning benchmark for generalist models. In *Forty-third International Conference on Machine Learning*, 2026.

Xiyuan Zhang, Boran Han, Haoyang Fang, Abdul Fatir Ansari, Shuai Zhang, Danielle C. Maddix, Cuixiong Hu, Andrew Gordon Wilson, Michael W. Mahoney, Hao Wang, Yan Liu, Michael Bohlke-Schneider, Huzefa Rangwala, George Karypis, and Bernie Wang. When does multimodality lead to better time series forecasting? *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856.

Luca Zhou, Pratham Yashwante, Marshall Fisher, Alessio Sampieri, Zihao Zhou, Fabio Galasso, and Rose Yu. Cats-bench: Can language models describe time series? In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 34479–34519, 2026.

A Native Evaluation and Reproduction

A.1 Systems

We evaluate four time-series language models. Table 4 summarizes their backbones, time-series inputs, training procedures, and native evaluation sets.

TimeOmni-1 (Guan et al., 2026a). It is built on Qwen2.5-7B-Instruct (Yang et al., 2025). It is first supervised fine-tuned on chain-of-thought samples from TSR-Suite and then optimized with Group Relative Policy Optimization (GRPO) using task-specific rewards.

ChatTS (Xie et al., 2025). ChatTS uses Qwen2.5-14B-Instruct (Yang et al., 2025) as its language backbone. Training uses synthetic time-series–text data: alignment pre-training first connects the time-series encoder to the language model, followed by supervised fine-tuning on time-series understanding and reasoning tasks.

TimeOmni-VL (Guan et al., 2026b). TimeOmni-VL is built on BAGEL-7B-MoT (Deng et al., 2025). The pretrained BAGEL checkpoint is post-trained on TSUMM-Suite with chain-of-thought supervision for time-series understanding and generation.

Time-MQA (Kong et al., 2025). Time-MQA uses Qwen2.5-7B (Yang et al., 2025) as its backbone. The released Qwen variant is trained on the TSQA corpus with LoRA, covering forecasting, imputation, anomaly detection, classification, and reasoning tasks.

Table 4: Systems and original evaluation settings.

System	Backbone	TS input	Training	Evaluation data
TimeOmni-1	Qwen2.5-7B	Text	CoT-SFT → GRPO	TSR-Suite
ChatTS	Qwen2.5-14B	TS encoder	Alignment → SFT	Eval A/B
TimeOmni-VL	BAGEL-7B	TS image	CoT post-training	TSR-Suite (OOD)
Time-MQA	Qwen2.5-7B	Text + context	Continual PT (LoRA)	TSQA

Table 5: Reproduction results. For MAE, lower is better; all other metrics are higher-is-better.

System	Split	Task/metric	Reproduced	Paper	Difference
TimeOmni-1	ID	Scenario ACC	88.50	90.70	−2.20
	ID	Causality ACC	67.60	69.30	−1.70
	ID	Forecasting MAE	15.95	14.30	+1.65
	ID	Decision ACC	47.90	47.90	0.00
	OOD	Scenario ACC	84.80	87.70	−2.90
	OOD	Causality ACC	62.30	64.00	−1.70
	OOD	Forecasting MAE	186.12	145.53	+40.59
	OOD	Decision ACC	59.00	58.90	+0.10
ChatTS	A	Overall categorical	0.890	0.889	+0.001
	A	Overall numerical	0.785	0.788	−0.003
	A	Deductive reasoning	0.744	0.744	0.000
	A	Causal reasoning	0.772	0.804	−0.032
	B	Overall categorical	0.855	0.862	−0.007
	B	Overall numerical	0.783	0.787	−0.004
TimeOmni-VL	OOD	Scenario ACC	81.80	84.00	−2.20
	OOD	Causality ACC	60.60	61.30	−0.70
	OOD	Forecasting MAE	189.63	163.79	+25.84
	OOD	Decision ACC	59.70	61.40	−1.70

Table 6: Performance of time-series QA systems and their matched backbones. Accuracy and score metrics are reported as percentages, forecasting MAE retains its original scale.

System / backbone	Split	Task / metric	System	Backbone	Improvement
TimeOmni-1 / Qwen2.5-7B	ID	Scenario ACC	88.50	41.00	+47.50
		Causality ACC	67.63	18.75	+48.88
		Forecasting MAE	15.95	23.27	+7.32
		Decision ACC	47.87	23.94	+23.94
	OOD	Scenario ACC	84.76	43.05	+41.71
		Causality ACC	62.25	21.13	+41.12
		Forecasting MAE	186.12	264.18	+78.06
		Decision ACC	58.97	23.81	+35.16
ChatTS / Qwen2.5-14B	A	Categorical score	88.95	47.40	+41.55
		Numerical score	78.47	23.53	+54.94
		Deductive ACC	74.42	60.47	+13.95
		Causal ACC	77.17	42.39	+34.78
	B	Categorical score	85.47	35.14	+50.33
		Numerical score	78.25	19.38	+58.87
TimeOmni-VL / BAGEL-7B	OOD	Scenario ACC	81.76	42.83	+38.93
		Causality ACC	60.63	28.75	+31.88
		Forecasting MAE	189.63	335.83	+146.20
		Decision ACC	59.71	21.98	+37.73

A.2 NATIVE COMPARISON

Reproduction. In Table 5, we compare our results with the reported results. We evaluate TimeOmni-1 on the ID and OOD splits of TSR-Suite. We report the objective metrics from Evaluation Sets A/B for ChatTS. For TimeOmni-VL, we follow the reasoning route on TSR-Suite-OOD.

Comparison with matched backbone. We evaluate each matched backbone on the same tasks in Table 6. These comparisons establish the native specialization gains but do not determine how predictions depend on the numerical series of individual samples.

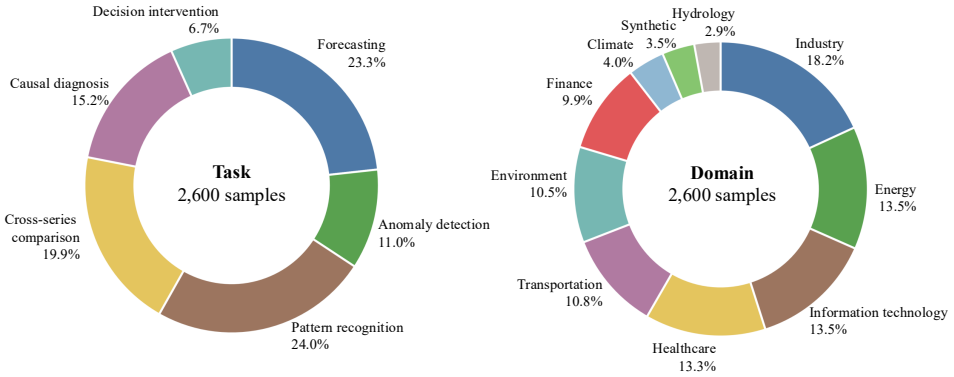


Figure 7: Tasks and domains represented in COMMON-TSQA.

Table 7: Tasks, source datasets, and domains represented in COMMON-TSQA after removing all Open-Meteo weather, CAMS air-quality, and ECB exchange-rate samples.

Task	Data sources	Samples n	Domains
Forecasting	GIFT-Eval; SciTS; TSR-Suite (OOD); TIME	605	Energy; environment; finance; healthcare; information technology; transportation; climate
Anomaly detection	TSB-AD-U	285	Environment; healthcare; industry; information technology; synthetic
Pattern recognition	TSAQA; TSRBench	623	Energy; environment; finance; healthcare; information technology; synthetic; transportation
Cross-series comparison	TSAQA; CaTS-Bench	517	Energy; environment; finance; healthcare; information technology; transportation; climate
Causal diagnosis	FactoryBench; SenTSR-Bench; TSR-Suite (OOD)	395	Industry; hydrology
Decision intervention	SenTSR-Bench; TSR-Suite (OOD)	175	Industry; energy
Total		2,600	

B COMMON-TSQA COMPOSITION, SOURCES, AND TASKS

B.1 COMPOSITION AND SOURCES

We retain the source-defined tasks and targets, only standardize the fields needed for a common benchmark representation. The released sources include GIFT-Eval (Aksu et al., 2024), SciTS (Wu et al., 2026), TIME (Qiao et al., 2026), TSR-Suite OOD (Guan et al., 2026a), TSB-AD-U (Liu & Paparrizos, 2024), TSAQA (Jing et al., 2026), TSRBench (Yu et al., 2026), CaTSBench (Zhou et al., 2026), FactoryBench (Merzouki et al., 2026), and SenTSR-Bench (He et al., 2026). Each sample is adapted to the interface of the evaluated system, and its output is transformed to the benchmark format and evaluated using the same specific criteria as the tasks. Table 7 contains the information about the benchmark, Figure 7 summarizes the benchmark composition, and Table 8 lists the publicly available links.

B.2 TASKS

Forecasting. Given an observed time-series history and context provided by the source task, the goal is to predict the future sequence within the specified horizon.

Anomaly detection. For the given time series, it is required to determine whether there is any anomaly in it.

Pattern recognition. This task is to find out the temporal pattern specified in the source sample, which may be trend, periodicity, volatility, or anomaly.

Table 8: Data sources and access links.

Source	Access link
GIFT-Eval	https://huggingface.co/datasets/Salesforce/GiftEval
TIME	https://huggingface.co/datasets/Real-TSF/TIME
SciTS	https://huggingface.co/datasets/OpenTSLab/SciTS
TSB-AD-U	https://www.thedatum.org/datasets/TSB-AD-U.zip
TSAQA	https://huggingface.co/datasets/TSAQA/TSAQA-Benchmark
TSRBench	https://huggingface.co/datasets/umd-zhou-lab/TSRBench
CaTS-Bench	https://huggingface.co/datasets/mhfisher/CaTSBench
TSR-Suite	https://huggingface.co/datasets/anton-hugging/timeomni-1-testbed
FactoryBench	https://huggingface.co/datasets/FactoryBench/FactoryBench
SenTSR-Bench	https://huggingface.co/datasets/ZLHe0/SenTSR-Bench

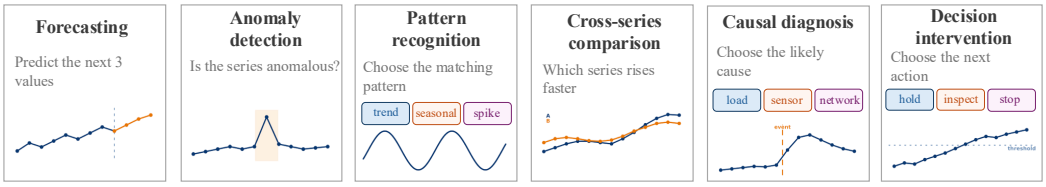


Figure 8: The six tasks in COMMON-TSQA.

Cross-series comparison. Based on the two or more time series, the task is to identify the relationship specified in the source sample between them.

Causal diagnosis. This task is to determine the causal relationship or mechanism between the observed series and context provided by the source task.

Decision intervention. Given the observed time series and context provided by the source task, the task is to select the prescribed action or intervention.

Figure 8 visualizes these six tasks.

B.3 SOURCE SELECTION AND CONVERSION

A source is included only if it contains samples that map to at least one of the six tasks without changing the target of the source. In order to be eligible, a source has to supply information about provenance and split in addition to either a known or independent verification of the target. An output format and scorer have to be defined for each task. All samples must satisfy all seven evaluation criteria and be rendered for all four systems without any loss of task-relevant information.

We check candidate samples from each official source split against the training data available for the evaluated systems. Overlaps are identified and removed using source IDs, fingerprints of normalized records, and grouping information. The grouping checks account for cases where the same series, episode, device, or location appears in a different form.

Before rendering inputs for specific systems, we standardize input representations, option encodings, and output schemas while preserving the source-defined task semantics and ground-truth answers.

C INTERVENTION SEMANTICS

Shuffle. Shuffle independently permutes the mutable observed values within each series. It preserves the series length and value multiset, but destroys the original temporal order.

Flat. Flat replaces all mutable observed values in a series with their mean. The replacement follows the same numerical formatting used by the benchmark, so the displayed constant may be a rounded version of the underlying mean. Flat removes within-series variation while approximately preserving the observed level.

Table 9: Original–Answer-only transitions across the five discrete tasks. C→W, W→C, and Changed use the joint-valid subset as denominator; all entries are percentages.

Task	System	O Acc.	AO Acc.	C→W	W→C	Changed
Anomaly detection	TimeOmni-1	60.70	45.61	20.64	4.98	25.62
	ChatTS	62.46	54.39	8.42	0.35	8.77
	Time-MQA	64.91	45.61	20.73	0.36	21.09
	TimeOmni-VL	70.88	45.61	38.64	9.85	48.48
Pattern recognition	TimeOmni-1	60.51	52.81	24.11	16.34	48.22
	ChatTS	59.91	59.23	14.13	12.84	30.50
	Time-MQA	55.06	43.98	17.62	11.88	34.65
	TimeOmni-VL	58.35	58.91	19.69	20.38	47.95
Cross-series comparison	TimeOmni-1	58.79	50.87	20.39	11.37	33.33
	ChatTS	54.93	46.23	15.95	7.85	25.57
	Time-MQA	46.81	34.82	23.51	13.81	43.66
	TimeOmni-VL	53.00	35.59	41.81	9.70	55.85
Causal diagnosis	TimeOmni-1	33.67	27.85	20.59	14.44	60.43
	ChatTS	28.35	31.39	7.85	10.89	29.62
	Time-MQA	27.19	19.24	11.78	11.78	42.42
	TimeOmni-VL	33.42	29.62	22.06	16.33	62.18
Decision intervention	TimeOmni-1	43.43	21.14	32.94	10.59	62.94
	ChatTS	26.14	26.29	5.14	6.29	25.71
	Time-MQA	26.17	22.29	10.00	9.38	43.12
	TimeOmni-VL	49.71	27.43	33.73	11.45	66.27

Noise. Noise replaces the mutable observed values with independent Gaussian draws using the observed mean and population standard deviation of the series. The realized samples are not recentered or rescaled, and therefore need not reproduce the original sample moments exactly. When the observed standard deviation is zero, we use a standard deviation of 1.0. Generated values are not clipped and are rendered using the recipient's numerical precision.

Swap. Swap replaces the numerical series with a compatible series from another benchmark record. Donors are selected without using the recipient target or model output and are matched by source dataset, split, task contract, domain, compatibility stratum, and number of series. A donor must come from a different record and evidence group.

Answer-only. Answer-only removes the numerical series while retaining the question and other non-series information. Text interfaces replace each series with `[series omitted]`, whereas native numerical interfaces receive no series input. Question semantics, answer choices, units, length descriptions, and mask/index definitions are left unchanged.

Zero. Zero replaces every mutable numerical value with 0 while preserving the series length, channel structure, and non-series information. Unlike Flat, it preserves neither the original level nor within-series variation.

D Additional analysis for RQ2

D.1 Original–Answer-only prediction transitions

For each task and system, we compare the semantic answers under Original and Answer-only. Correct-to-wrong and wrong-to-correct transitions are determined against the original ground-truth answer. Changed counts any difference between the two answers, including transitions between distinct incorrect answers. All three rates use the jointly valid samples as their denominator. Table 9 extends the main-text analysis to all five discrete tasks. Substantial answer changes occur even in some settings with similar aggregate accuracy. These transitions identify sensitivity to evidence removal, not whether the changed answers are appropriate.

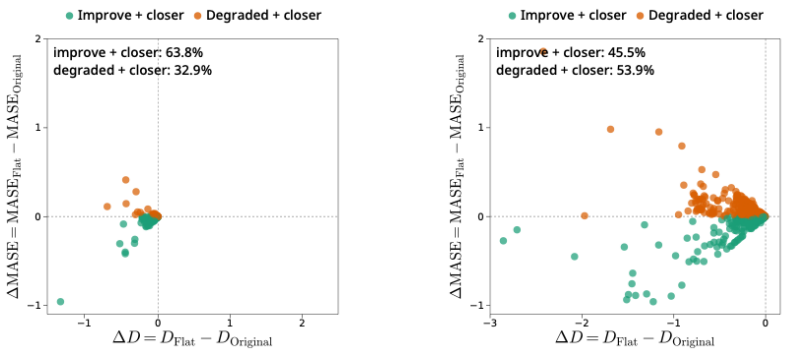


Figure 9: Original–Flat item-level forecasting diagnostics for Time-MQA and TimeOmni-VL with available forecasting results. Panels are enlarged relative to the main-text diagnostic for readability. Negative horizontal values indicate movement toward the fixed history-mean forecast; negative vertical values indicate lower MASE under Flat.

D.2 Flat and simple forecasting references

We compare Original and Flat on samples that are scorable under both conditions. Let $\bar{x}_i$ be the mean of sample i's Original observed history, H_i its forecast horizon, and s_i the Original-history scaling term in MASE. The distance to the fixed history-mean forecast is

$$D_i^{(c)} = \frac{H_i^{-1} \sum_{h=1}^{H_i} \left| \hat{y}_{ih}^{(c)} - \bar{x}_i \right|}{s_i}.$$

Both the reference and the scale remain fixed across conditions. We define $\Delta D_i = D_i^{\text{Flat}} - D_i^{\text{Original}}$ and $\Delta\text{MASE}_i = \text{MASE}_i^{\text{Flat}} - \text{MASE}_i^{\text{Original}}$. Improve + closer requires both differences to be negative; Degraded + closer requires $\Delta D_i < 0$ and $\Delta\text{MASE}_i > 0$.

Figure 9 shows that movement toward the history-mean forecast is common across systems, while its effect on error differs.

D.3 Target-aware evaluation of cross-series interventions

We recompute answers after Shuffle, Flat, Noise, Swap, and Zero for cross-series questions whose targets can be determined exactly from the transformed input.

We distinguish pairs whose correct target changes from those whose target remains unchanged. For target-changed pairs, a correct update matches the recomputed target, a stale answer matches the original ground-truth target, Other contains all remaining outputs. These categories describe the intervention output relative to the two targets; they do not require the Original prediction to have been correct.

Table 10 reports accuracies over all eligible pairs and outcome shares over all target-changed pairs. Figure 4(b) uses a narrower display, normalizing correct updates and stale answers by their combined count. Table 11 separates interventions. This decomposition distinguishes incorrect responses to the transformed series from correct responses that no longer match the original answer.

Table 10: Pooled target-aware results on the exact-oracle subset. Original-target and recomputed accuracies use all eligible pairs; the remaining columns use target-changed pairs as denominator. All entries are percentages.

System	Original-target Acc.	Recomputed Acc.	Correct update	Stale	Other
TimeOmni-1	41.55	72.80	78.90	14.98	6.11
ChatTS	30.49	68.01	82.87	6.12	11.01
Time-MQA	32.88	48.88	50.76	18.04	31.19
TimeOmni-VL	39.01	74.44	77.98	5.50	16.51

Table 11: Target-aware decomposition by system and intervention. Target changed is the share of eligible pairs whose recomputed target changes. Correct-update, stale-answer, and other-error percentages use all target-changed outputs.

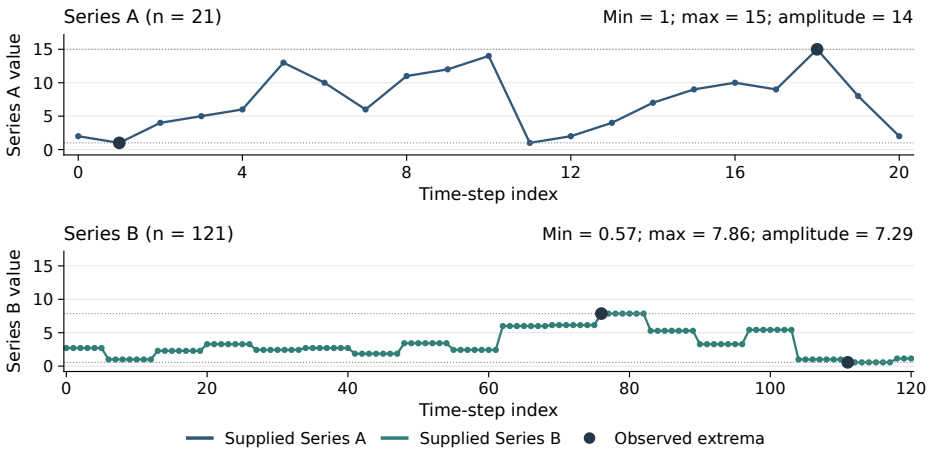
System	Intervention	Target changed	Unchanged-target Acc.	Correct update	Stale	Other
TimeOmni-1	Shuffle	6.7	62.40	88.89	0.00	11.11
	Flat	76.1	90.62	94.12	4.90	0.98
	Noise	13.4	73.28	61.11	27.78	11.11
	Swap	50.4	53.03	62.69	23.88	13.43
	Zero	97.8	66.67	77.10	17.56	5.34
ChatTS	Shuffle	6.7	51.20	44.44	11.11	44.45
	Flat	76.1	84.38	91.18	1.96	6.86
	Noise	13.4	50.86	33.33	27.78	38.89
	Swap	50.4	50.00	67.16	16.42	16.42
	Zero	97.8	33.33	93.89	0.76	5.35
Time-MQA	Shuffle	6.7	47.20	22.22	55.56	22.22
	Flat	76.1	50.00	43.14	15.69	41.17
	Noise	13.4	50.86	44.44	27.78	27.78
	Swap	50.4	39.39	44.78	26.87	28.35
	Zero	97.8	33.33	62.60	11.45	25.95
TimeOmni-VL	Shuffle	6.7	71.20	44.44	11.11	44.45
	Flat	76.1	71.88	85.29	0.00	14.71
	Noise	13.4	75.86	33.33	22.22	44.45
	Swap	50.4	62.12	62.69	17.91	19.40
	Zero	97.8	66.67	88.55	0.76	10.69

E CASES

Case 1. Cross-series comparison (TimeOmni-1)

Question. Given two time series A and B, detect which one has a higher amplitude defined as maximum - minimum.

Response labels. *a_gt_b*: A has the larger amplitude; *a_lt_b*: B has the larger amplitude; *tie*: equal amplitudes.



Target: *a_gt_b*

Answer: *a_lt_b*

Reasoning:

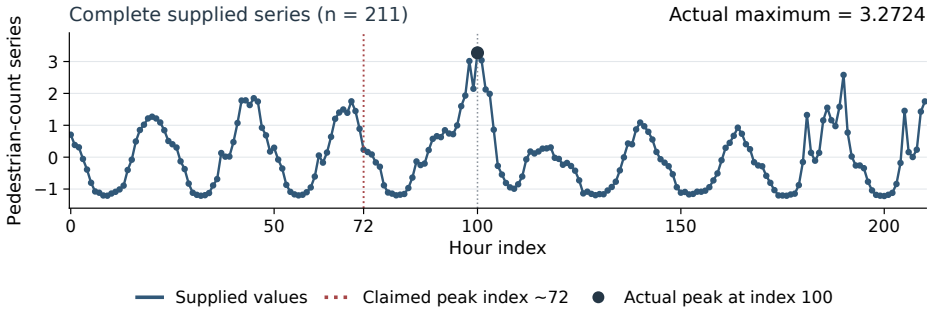
"Series A: Minimum = 1, Maximum = 15 → Amplitude = 14."
"Series B: Minimum = 0.57 (last few points), Maximum = 7.86 (middle section). But the main plateau is 6.00–7.86, so the amplitude is $7.86 - 0.57 \approx 7.29$."
"Therefore, Series B has a higher amplitude than Series A."

Conflict: Series A has amplitude 14, whereas Series B has amplitude approximately 7.29, so Series A has the larger amplitude. The rationale reverses this comparison and concludes that Series B has the higher amplitude.

Case 2. Pattern recognition (TimeOmni-1)

Question. Considering the time series‘ cyclicity, what is the approximate amplitude of its recurring peaks?

Options. A: About 3.0. B: Less than 1.0. C: About 1.5. D: More than 4.0.



Target: A

Answer: B

Reasoning:

“The peak is at 3.2724 (index 72).”
“Option D is possible but less likely, as the highest value is 3.2724, not exceeding 4.0.”
“Therefore, the best estimate is Option A.”

Conflict: The rationale does not actually establish the amplitude of the recurring peaks. It identifies a single maximum value, 3.2724, and treats that value as the answer without examining whether multiple recurring peaks have a similar magnitude. It also places this maximum at index 72, whereas the value at index 72 is 0.2344 and the actual maximum occurs at index 100. Finally, the rationale concludes that Option A is the best estimate, while the reported final answer is B. Thus, the rationale contains both an incorrect localization claim and an unsupported shortcut from a single maximum to the recurring-peak amplitude, and its stated conclusion is inconsistent with the final answer.

Table 12: Blinded human validation of the LLM judge. Agreement compares the LLM-generated labels with independent human annotations on a frozen validation subset covering both systems and all six tasks.

Level	Criterion	N	Agreement (%)	Cohen's κ
Claim	Factual grounding	240	95.8	0.920
Output	Factual grounding	100	98.0	0.969
Output	Inference validity	100	89.0	0.837
Output	Conclusion consistency	100	94.0	0.895

F LLM-AS-JUDGE PROMPT

Human validation of the LLM-based adjudication. We validate the LLM-based adjudication on a frozen blinded subset containing 240 numerical claims and 100 rationale outputs, covering both systems and all six tasks. The human annotator does not observe the LLM-generated labels, target answer, answer correctness, or system identity. As shown in Table 12, the human annotations show strong agreement with the LLM-based adjudication.

Prompt for factual-grounding adjudication

You are evaluating the factual grounding of a model-generated rationale for a time-series task.
Extract atomic claims about the observed time series that can be checked against the supplied numerical input. Split sentences containing multiple independently verifiable claims. Merge semantic duplicates and record their occurrence count. Do not extract final answers, option restatements, predicted or imputed values, recommended actions, generic method descriptions, or external claims that cannot be verified from the supplied series. Do not repeat claims that have already been adjudicated separately.
For each extracted claim, assign one label:

- `supported`: clearly supported by the supplied numerical evidence;
- `unsupported`: clearly contradicted by the supplied numerical evidence;
- `not verifiable`: cannot be evaluated from the supplied numerical evidence.

Use only the numerical evidence provided for the current example. Do not use the system identity, target answer, answer correctness, or answer choices as evidence. Derived diagnostics such as trend, periodicity, autocorrelation, anomaly, or lag may support a judgment, but should not be treated as categorical ground truth when the claim is ambiguous.
For supported and unsupported claims, provide concise numerical evidence. Return each unique claim once with its evidence scope, grounding label, evidence, and occurrence count. Also indicate whether the rationale is complete.
Return JSON matching the provided schema exactly.

Table 13: Public training signals and checkpoint constraints relevant to the discussion. Missing public artifacts limit the comparisons supported by the released systems.

System	Public training signals	Checkpoint constraints
TimeOmni-1	Human-guided CoT-SFT and GRPO, with answer/format rewards and a forecasting-error reward.	Final model available; no released SFT-only or RL-intermediate checkpoint.
ChatTS	Alignment and supervised fine-tuning; the training data released with the paper include multi-series, relational, causal, and rationale-like content.	Historical training data and final model available; no released alignment-only checkpoint.
Time-MQA	Reported training uses 7,000 TSQA and 3,000 OpenOrca samples; final specialization is a LoRA adapter.	Adapter permits a fixed-base contrast; the original base revision is not pinned and no intermediate adapter is released.
TimeOmni-VL	Time-series understanding/generation and CoT-based post-training.	Final model available; no intermediate checkpoints or complete original training-row manifest.

Table 14: Selected candidate tags in the ChatTS corpus. Samples gives the stage size; all other entries are percentages within that stage. Rule-based tags overlap and are not gold annotations.

Stage	Samples	Multi-series	Cross-series	Causal	Rationale
Stage 1	105,085	66.69	6.89	2.27	51.15
Stage 2	30,650	29.34	19.61	10.11	42.39

G Training-Design Diagnostics

G.1 Public training signals and release scope

We examine official papers, supplements, repositories, model cards, and model/data releases. Table 13 summarizes the training signals and checkpoint constraints relevant to the discussion. These observations support descriptive comparisons with the behavioral results.

None of the public releases provides the complete sequence of intermediate training checkpoints. Time-MQA permits a fixed-base contrast because its specialization is released as a separate LoRA adapter, although the original base revision is not pinned. We distinguish this comparison from an exact training-stage ablation.

G.2 ChatTS training corpus

We scan the historical Stage 1 and Stage 2 files, containing 105,085 and 30,650 samples, respectively. Overlapping rule-based tags identify candidate multi-series, cross-series, causal, and rationale-like content. We manually check 57 stage-by-positive-category samples to assess the semantics of the tagging rules. Table 14 reports the categories used in the discussion; the tags are not gold annotations.

The public generation pipeline includes numerical statements and source-series attributes in training text, and TSEvol applies holistic LLM filtering to complete QA pairs. This does not establish that every intermediate numerical statement is checked: our audit identifies no general deterministic per-claim numerical checker applying throughout the pipeline described in the paper.

We also inspect 50 candidate groups, split evenly across stages, for matched questions with changed numerical evidence and recomputed targets. None is verified as a same-parent evidence-change/recomputed-target training pair. This finding is limited to the audited candidates and does not establish that such supervision was absent from private training procedures.

Table 15: Pooled Time-MQA target-aware Base–LoRA contrast. Original-target and recomputed-target accuracies use all 669 eligible sample–intervention pairs. Correct update, stale, and other use the 327 pairs whose correct target changes. All entries are percentages.

Checkpoint	Original-target Acc.	Recomputed Acc.	Correct update	Stale	Other
Base	39.91	27.06	17.13	43.43	39.45
LoRA	24.81	38.27	44.04	16.51	39.45

Table 16: Intervention-level Time-MQA target-aware behavior. "Changed" is the number of eligible pairs for which the intervention changes the recomputed target. Correct-update, stale, and other percentages use the changed-target subset.

Checkpoint	Intervention	Eligible	Changed	Original-target Acc.	Recomputed Acc.	Correct update	Stale	Other
Base	Shuffle	134	9	38.06	35.07	11.11	55.56	33.33
	Flat	134	102	36.57	8.96	1.96	38.24	59.80
	Noise	134	18	37.31	40.30	33.33	11.11	55.56
	Swap	133	67	40.60	30.08	28.36	49.25	22.39
	Zero	134	131	47.01	20.90	21.37	48.09	30.54
LoRA	Shuffle	134	9	32.09	29.85	22.22	55.56	22.22
	Flat	134	102	19.40	41.04	41.18	12.75	46.07
	Noise	134	18	35.07	34.33	16.67	22.22	61.11
	Swap	133	67	26.32	28.57	29.85	25.37	44.78
	Zero	134	131	11.19	57.46	58.78	11.45	29.77

G.3 Time-MQA Base–LoRA checkpoint contrast

Setup. We freeze one public revision of the base declared by the released adapter and compare it with the same base plus the LoRA weights. The tokenizer, model-visible input, prompt renderer, greedy decoding, parser, and scorer are held identical. Invalid outputs remain in the overall accuracy denominator. Because the adapter configuration does not identify the exact original base revision, this is a reproducible checkpoint contrast rather than a reconstruction of the authors' training stages. Forecasting is excluded from the performance interpretation because only two samples are scorable under both checkpoints.

Target-aware responses. Both checkpoints are evaluated on the same 669 eligible cross-series sample–intervention pairs, of which 327 have a changed correct target. Table 15 reports original-target and recomputed-target accuracies over all 669 pairs, and the three outcome shares over the 327 target-changed pairs. Table 16 provides the intervention-level results. Adaptation increases correct updates and reduces stale answers in the pooled comparison, with the largest correct-update increases under Flat and Zero rather than a uniform improvement across interventions.

Validity and paired-valid accuracy. Output validity is the proportion of task samples yielding recoverable semantic answers. Paired-valid accuracy is evaluated on the common subset for which both checkpoints produce valid answers. Table 17 reports these measures alongside overall accuracy. Pattern recognition shows higher validity but lower paired-valid accuracy after adaptation, whereas causal diagnosis improves on both measures. Decision intervention increases validity without changing paired-valid accuracy, and cross-series comparison changes little. The task-specific patterns show why aggregate accuracy gains cannot be interpreted as a uniform change in semantic correctness.

Table 17: Time-MQA validity and paired-valid accuracy. Paired-valid accuracy uses only samples for which both checkpoints produce valid semantic outputs.

	Valid (%)		Paired-valid acc. (%)		Overall acc. (%)	
Task	Base	LoRA	Base	LoRA	Base	LoRA
Anomaly detection	98.95	92.63	55.17	65.90	56.84	61.05
Pattern recognition	26.97	91.49	73.91	53.42	20.06	53.45
Cross-series comparison	26.69	25.73	50.00	48.68	13.73	13.35
Causal diagnosis	50.38	86.08	23.93	35.58	11.65	25.06
Decision intervention	36.00	90.86	24.56	24.56	9.71	21.14