

TimeClaimBench: A Claim-Level Benchmark for Grounded LLM Explanations of Time Series

Sijia Peng^{1*}, Fan Zhang^{2*}, Yixi Zhou³, Changlun Li⁴, Yun Xiong^{3†}
Yangyong Zhu¹, Xi Chen¹, Yanwei Yu⁵, Nan Tang⁴

¹Shanghai Key Lab of Data Science,

College of Computer Science and Artificial Intelligence, Fudan University

²The University of Tokyo ³Hong Kong Baptist University

⁴HKUST(GZ) ⁵Ocean University of China

pengsijia1216@gmail.com zhang-fan@g.ecc.u-tokyo.ac.jp

Abstract

Large language models (LLMs) are increasingly used to explain numerical time series, but fluent explanations can misstate when a change occurs, overstate a trend, or introduce unsupported patterns. Existing evaluations often score forecasts, answers, or whole explanations, leaving claim-level temporal grounding undermeasured. We introduce TIMECLAIMBENCH, a benchmark that parses explanations into atomic temporal claims and verifies them against a deterministic extracted event graph. Across five LLMs on controlled synthetic series, event-grounded prompting raises the strict graph-supported rate by 18.3 points and reduces graph-unsupported claims by 27.9 points under this fixed operational criterion. Calibration analyses show the boundary of that result. The extractor recovers the designed primary event in 394 of 400 series, but this does not estimate the precision of additional extracted events. A blinded two-annotator comparison on one model and 40 series reverses the automatic ordering: holistic ratings favor Direct and chain-of-thought outputs over event-grounded outputs. TIMECLAIMBENCH therefore distinguishes four axes that should not be conflated: designed-event recovery, the unresolved precision of additional events, claim-to-graph consistency, and whole-explanation human judgment. Code: [GitHub](#).

1 Introduction

Automatically describing time series in natural language is useful, but early time-series captioning work already showed a core risk: fluent descriptions can be factually incorrect with respect to the underlying signal (Jhamtani and Berg-Kirkpatrick, 2021). This risk becomes more consequential as large language models (LLMs) are increasingly used as natural-language interfaces for time-series

analysis. Such interfaces make time-series analysis more accessible, but they also make it easier for users to trust fluent explanations that misstate when a change occurs, overstate the strength of a trend, hallucinate seasonality, or introduce an unsupported causal interpretation. Figure 1 illustrates why these failures need claim-level evaluation rather than only whole-explanation scoring.

Existing work has moved time-series language evaluation closer to natural-language explanations. Truth-conditional captioning and recent time-series captioning work study how to describe salient temporal patterns (Jhamtani and Berg-Kirkpatrick, 2021; Trabelsi et al., 2025); feature-understanding and exam-style benchmarks test whether models recognize trends, noise, anomalies, causality, or other time-series properties (Fons et al., 2024; Cai et al., 2024); BEDTime and CaTS-Bench evaluate time-series description and captioning abilities at broader scale (Sen et al., 2026; Zhou et al., 2026); and LLM-as-a-judge work studies reference-free scoring of candidate time-series explanations (Zheng et al., 2023; Sivalingam et al., 2026). Together, these efforts show an evaluation mismatch: users read multi-claim explanations, while most reported scores collapse each response into a single caption, answer, or judge score.

The gap is therefore not that time-series captioning, understanding, or explanation evaluation is unexplored. The central missing unit is **claim-level temporal grounding**. A single explanation can contain both a correct claim and a wrong one: for example, it may correctly identify a sharp rise, but place the subsequent decline in the wrong interval, overstate a weak trend, or add a seasonal pattern that is absent. Reference-based text metrics and explanation-level judgments can hide these mixed cases. NLP work on faithfulness, attribution, and fact verification provides useful precedents for checking whether generated statements are supported by evidence (Jacovi and Goldberg, 2020;

*Equal contribution

†Corresponding author

Why claim-level time-series faithfulness?

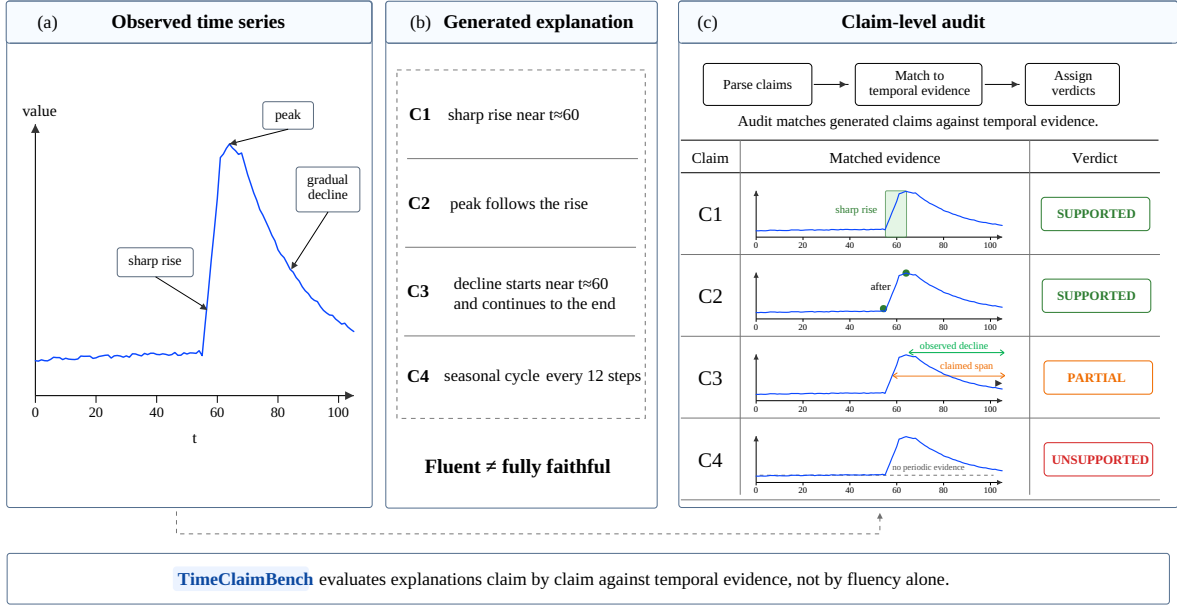


Figure 1: Motivation for claim-level time-series grounding evaluation. A fluent LLM-generated explanation may contain multiple temporal claims, some of which are supported by the observed sequence, while others are only partially supported or unsupported. TIMECLAIMBENCH parses the explanation into atomic claims, matches each claim to temporal evidence, and assigns claim-level verdicts.

Rashkin et al., 2023; Chen et al., 2020; Kamoi et al., 2023; Laban et al., 2022). However, the evidence substrate is different: document and table verification can search for supporting sentences or cells, whereas time-series verification must first induce events from numerical values, intervals, slopes, changes, periodicity, and outliers.

We introduce TIMECLAIMBENCH, a claim-level benchmark for evaluating whether LLM-generated time-series explanations are consistent with numerical temporal evidence. Instead of scoring an explanation only as a whole, TIMECLAIMBENCH decomposes it into temporal claims and checks each claim against a deterministic extracted event graph, exposing mixed cases where a response is partly supported but mislocalized or overstated. TIMECLAIMBENCH combines a controlled synthetic suite covering eight temporal phenomena with a Real-NAB localization diagnostic for real anomaly-window behavior (Lavin and Ahmad, 2015), together with a fixed rule-based verifier and human calibration. Across five LLMs and four prompting conditions, graph-unsupported temporal claims are common; event-grounded prompting raises the strict graph-supported rate by 18.3 points and reduces graph-unsupported claims by 27.9 points. A blinded preference study, however, reverses this au-

tomatic ordering in a one-model, 40-series sample. The combined evidence shows why extracted-event consistency, representation quality, and holistic human judgment must be reported separately. Our contributions are summarized as follows:

- We formulate claim-level temporal grounding for LLM-generated time-series explanations, distinguishing SUPPORTED, PARTIAL, and UNSUPPORTED claims under an explicit extracted-event criterion, with evidence intervals and error types.
- We construct TIMECLAIMBENCH, a benchmark with eight controlled temporal phenomena and a Real-NAB localization diagnostic for real anomaly-window behavior.
- We provide an executable non-LLM verification protocol and a multi-model evaluation showing that graph-unsupported temporal claims are common, chain-of-thought can amplify them, and event-grounded prompting substantially improves claim-to-graph consistency.
- We calibrate the operational metric with extractor-recovery, sparse-reference, claim-audit, and blinded preference analyses, identifying a rank reversal between automatic graph support and holistic human judgment.

2 Related Work

Time-series language benchmarks and explanation evaluation. Recent work adapts language-model architectures and pretrained LLMs to time-series inputs through prompting, reprogramming, embedding alignment, decoder adaptation, or foundation sequence models (Gruver et al., 2023; Jin et al., 2024; Cao et al., 2024; Sun et al., 2024; Pan et al., 2024; Liu et al., 2024; Zhou et al., 2023; Goswami et al., 2024; Das et al., 2024; Ansari et al., 2024). Closer to our setting, time-series captioning, description, QA, and chart-to-text work evaluate whether models describe or reason over temporal and numerical patterns (Jhamtani and Berg-Kirkpatrick, 2021; Kantharaj et al., 2022; Masry et al., 2022; Tang et al., 2023; Fons et al., 2024; Cai et al., 2024; Trabelsi et al., 2025; Sen et al., 2026; Zhou et al., 2026; Chen et al., 2025; Xie et al., 2025; Kong et al., 2025). LLM-as-a-judge methods further provide reference-free or preference-style evaluation for generated responses, including time-series explanations (Zheng et al., 2023; Liu et al., 2023; Kim et al., 2024; Sivalingam et al., 2026). A controlled study of scientific-idea judges further shows that writing style can change scores and rankings under fixed content (Ji et al., 2026a). These lines motivate language-based evaluation for time series, but their scores target forecasts, answers, captions, or holistic rubrics and leave individual temporal claims inside explanations unevaluated.

Contextual time-series systems broaden the evidence available to language models. TimeCAP combines a multimodal encoder with LLM agents for contextualized event prediction (Lee et al., 2025), while TimeXL jointly models time-series and textual context for prediction and explanation (Jiang et al., 2025). These systems motivate a complementary scope distinction. TIMECLAIMBENCH evaluates claims whose evidence appears in the observed signal through an operational event representation. Metadata, documents, and domain knowledge fall outside this evidence scope. Extending the benchmark to that setting would require those sources and a multi-evidence entailment layer in addition to temporal event matching.

Diagnostic benchmarks and deployment-grounded evaluation. Recent benchmarks expose intermediate failure points or tie scores to deployment outcomes. FineState-Bench separates GUI localization, interaction, and exact-state success, while PMD-Bench isolates zero-shot

routing over LoRA-based external memories (Ji et al., 2026d,b). ServImage evaluates generated images against paid commercial tasks and payment decisions, while TradeLens attributes profit and inference cost to trading traces (Ji et al., 2026c; Duan et al., 2026). These benchmarks use domain-specific evidence but share a methodological principle with TIMECLAIMBENCH: the evaluation unit and evidence source should expose the failure under study. TIMECLAIMBENCH applies this principle to time-series explanations by evaluating atomic temporal claims against extracted events and reporting holistic human judgment separately.

Fine-grained factuality and time-series verification. Fine-grained factuality and attribution work provides the evaluation unit most aligned with our goal: these methods decompose generated text into smaller claims and check each claim against evidence (Jacovi and Goldberg, 2020; Atanasova et al., 2023; Rashkin et al., 2023; Laban et al., 2022; Min et al., 2023; Wei et al., 2024). Fact-checking benchmarks and revision or alignment pipelines extend this idea to documents, tables, and mixed evidence (Chen et al., 2020; Aly et al., 2021; Kamoi et al., 2023; Gao et al., 2023; Zha et al., 2023). Most closely, Strong and Vlachos (2025) introduce TSVer for verifying externally sourced factual claims against time-series evidence, whereas TIMECLAIMBENCH audits claims produced inside LLM-generated time-series explanations. The task checks whether temporal events, directions, magnitudes, and intervals induced from the observed sequence support each generated explanatory claim.

3 The TIMECLAIMBENCH Benchmark

3.1 Overview

Figure 2 gives the end-to-end workflow. The key design choice is to separate *evidence construction* from *language generation*: the same extracted events are used to condition event-grounded prompts and to verify claims after generation. This lets TIMECLAIMBENCH evaluate whether LLMs produce auditable temporal explanations grounded in numerical evidence, rather than merely fluent descriptions. Event-grounded prompting therefore tests evidence use, not a no-evidence fair baseline. Because the same operational extracted event graph is also used by the verifier, Section 5 includes mismatched-event, gold-event, and human-audit checks to characterize this coupling. Accordingly,

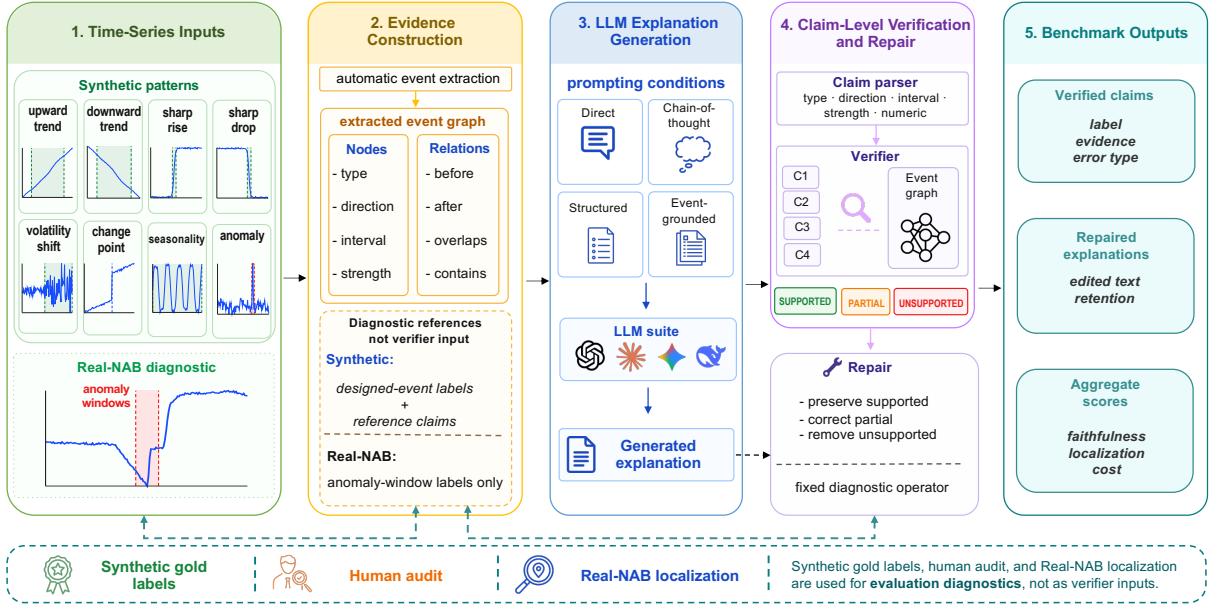


Figure 2: TIMECLAIMBENCH pipeline. Each input series is converted into an extracted event graph; LLM explanations are parsed into atomic claims, verified against the extracted event graph, optionally passed through verifier-guided repair, and summarized as claim-level outputs and aggregate metrics.

a high verifier score means consistency with that representation, not independent evidence that the extractor is precise or that the complete explanation is preferred by humans.

3.2 Design Goals

TIMECLAIMBENCH targets three design goals. First, it evaluates individual temporal claims since a single explanation can mix a correct trend description with a shifted interval, overstated magnitude, or unsupported pattern. Second, it pairs controlled synthetic series with real anomaly windows: synthetic examples provide known event evidence, while the Real-NAB localization diagnostic tests behavior under real noise. Third, it keeps evaluation executable but calibrated. The automatic verifier enables scalable prompt and model comparisons, and verifier ablations and blind human audits expose the boundaries of the automatic scores.

3.3 Benchmark Components

Time-series inputs. The benchmark contains a controlled synthetic split and a Real-NAB localization split. The synthetic split is the main benchmark and covers eight temporal phenomena: upward trends, downward trends, sharp rises, sharp drops, volatility shifts, change points, seasonality, and anomalies. Because these examples are generated with known event types, directions, intervals,

and reference claims, they support auditable claim-level evaluation. The Real-NAB localization split is based on the Numenta Anomaly Benchmark (Lavin and Ahmad, 2015) and tests anomaly-window behavior under real noise. Detailed split sizes and construction choices are reported in Appendix A.1.

Evidence construction. Each time series is converted by a deterministic rule-based event extractor into an extracted event graph. The extractor scans the normalized sequence with fixed temporal operators for trend, sharp change, volatility shift, change point, seasonality, anomaly, peak, and trough evidence; Appendix A.2 gives the exact settings. Each event node stores its type, optional direction, interval, score, and numeric attributes such as slope, magnitude, period, or z-score. Relation edges encode temporal links such as BEFORE, AFTER, OVERLAPS, and CONTAINS. The default verifier uses automatically extracted events; synthetic gold events and NAB anomaly windows are reserved for diagnostics and oracle analyses.

LLM explanation generation. TIMECLAIMBENCH stores model explanations produced under alternative prompting conditions. Direct prompting, zero-shot chain-of-thought (CoT) prompting (Wei et al., 2022; Kojima et al., 2022), and structured prompting expose only the raw series and question, while event-grounded prompting additionally exposes the extracted event list. The down-

stream evaluation protocol is specified in Section 4.

Claim-level verification and verifier-guided repair. The claim parser decomposes each explanation into atomic temporal claims and records attributes such as event type, direction, interval, strength, and numeric values when present. The automatic verifier retrieves compatible extracted events and checks claim–event agreement using type, direction, interval overlap, strength, and numeric consistency. Full support-label boundary cases, parser details, scoring rules, thresholds, and verifier-guided repair procedures are provided in Appendices A.3–A.5. The verifier-guided repair step preserves supported claims, corrects partially supported claims, and removes verifier-unsupported claims.

Outputs. The output layer returns the claim labels, matched evidence, repaired explanations, and token-cost records needed for evaluation. Section 4.2 explains how these outputs are aggregated in the main experiments, while Appendix A.6 provides the formal metric definitions. The first release targets signal-groundable claims. Contextual or multivariate claims would additionally require metadata or documents, channel-indexed events, cross-channel relations, and corresponding parsing and entailment rules.

4 Evaluation Protocol

4.1 Task Settings

We evaluate five LLM backbones under four prompting conditions: direct, chain-of-thought, structured, and event-grounded prompting. Each run uses a model-ready input containing the univariate series and question; the event-grounded prompting condition additionally receives an extracted event list. All runs use the same input order, generation settings, claim parser, automatic verifier, verifier-guided repair module, and cost-accounting pipeline. The main synthetic setting uses the 400-example test split, with 50 examples from each of the eight synthetic temporal phenomena. We also include a Real-NAB localization diagnostic with 100 anomaly-centered windows and 100 normal windows.

4.2 Scoring and Statistical Reporting

For reporting the main synthetic results, we select three headline summaries from the benchmark outputs: strict graph-supported rate, graph-

unsupported rate, and claims per explanation. These metrics separately measure graph-supported claims, graph-unsupported claims, and response verbosity. Verifier-guided repair analyses additionally report retention and supported-claim preservation, while diagnostic analyses report partial rate, interval localization, coverage, cost, and Real-NAB localization when relevant. In result tables, arrows indicate the preferred direction of each metric. For continuity across compact tables, *Faith.* denotes the strict fraction labeled SUPPORTED under the fixed extracted-event-graph criterion; it is an operational graph-support metric, not a human truth score. *Unsup.* denotes the fraction labeled UNSUPPORTED under the same criterion, *Claims/Expl.* claims per explanation, and *Ret.* claim retention after verifier-guided repair. Metrics marked with *+Rep.* are computed after verifier-guided repair; table-specific abbreviations are defined in the corresponding captions.

For the main prompt-level results and uncertainty analyses, we aggregate in two stages: first computing claim-level rates within each model-prompt run, then averaging over the five model runs. Diagnostic tables state any alternative aggregation in their captions. For uncertainty estimates, we use 5,000 bootstrap resamples of the 400 synthetic test series within each fixed model run; paired prompt comparisons preserve shared model-series identifiers. Appendix A.6 gives the formal metric definitions and dataset-specific diagnostics, and Appendix B provides the task notation, prompt templates, and generation settings.

4.3 Human Calibration and Preference Studies

We include blinded human studies to calibrate both claim labels and whole-explanation quality. For the original synthetic audit, two annotators independently label 180 generated claims balanced by automatic verifier label; verifier alignment is computed on the 156 claims where the annotators agree. An independent replication uses 90 new claims balanced over prompt and verifier label; the two annotators agree exactly on 76 claims. For Real-NAB, one blinded annotator audits 90 event-grounded claims to test whether verifier labels remain plausible under real-world noise. Finally, two annotators compare GPT-4.1-mini event-grounded outputs against Direct and CoT outputs on 40 stratified synthetic series. Each annotator completes 80 randomized A/B comparisons and assigns an-

chored 1–5 faithfulness and informativeness ratings plus ternary preferences. Prompt identities, event graphs, verifier labels, and the A/B key are hidden. Detailed protocols, label distributions, and confusion matrices are in Appendices D, F, and G.

4.4 Controls and Diagnostics

All full-scale prompt comparisons use the same fixed rule-based automatic verifier, claim parser, verifier-guided repair module, cost-accounting pipeline, and pre-specified thresholds. These controls make prompt and model comparisons reproducible under a single evaluation operator, but they do not make automatic scores human-level truth estimates. To evaluate the verifier itself, we construct 1,600 controlled claim-level cases from the 400 synthetic test examples and compare majority, type-only, type+interval, full extracted-event-graph, no-interval, and gold-event oracle variants. Auxiliary diagnostics include threshold calibration, coverage, pattern-wise breakdowns, designed-event recovery, sparse-gold re-verification, gold-event versus extracted-event prompting, mismatched event lists, deterministic event verbalization, synthetic–real shape coverage, and Real-NAB localization. The retained 80-example multi-provider pilot is summarized at prompt level in Tables 36 and 37. Appendices A.6, B, and E provide metric definitions, prompt templates and generation settings, and calibration diagnostics, respectively.

5 Experimental Results

We organize the results around four evaluation questions: **RQ1: Graph Support.** How often do generated explanations receive explicit support from the extracted event graph? **RQ2: Evidence Use.** Does explicit temporal evidence improve claim-to-graph consistency? **RQ3: Verifier Calibration.** How reproducible and calibrated is the automatic verifier? **RQ4: Human Judgment.** Does the automatic ordering transfer to blinded whole-explanation ratings?

5.1 Extracted-Graph Support

Main synthetic scores. Free-form prompting often lacks extracted-event support. Table 1 summarizes the 400-example synthetic evaluation. Graph-unsupported claims are common under free-form prompting: Direct yields a 0.493 strict graph-supported rate and a 0.392 graph-unsupported rate, while CoT lowers graph support to 0.360 and

raises graph-unsupported claims to 0.558. Event-grounded prompting is the only condition that consistently improves both operational metrics, reaching 0.676 graph support and 0.113 graph-unsupported rate when averaged over the five LLMs. Appendix Tables 16, 34, and 35 provide per-model, pattern-by-prompt, repair, cost, and token diagnostics. The strongest single configuration is DeepSeek-V4-Flash with event-grounded prompting, which reaches 0.746 graph support and a 0.077 graph-unsupported rate.

Prompt	Faith. $\uparrow$	Δ Faith.	Unsup. $\downarrow$	Δ Unsup.	Claims/Expl.	Ret. $\uparrow$
Direct	0.493	–	0.392	–	10.27	0.530
CoT	0.360	-0.133	0.558	+0.167	26.98	0.298
Structured	0.353	-0.140	0.468	+0.076	6.82	0.477
Event-grounded	0.676	+0.183	0.113	-0.279	8.46	0.850

Table 1: Prompt-level synthetic results on the 400-series test set. *Faith.* and *Unsup.* are first computed within each model–prompt run and then macro-averaged equally over the five LLMs. Δ is relative to Direct; event-grounded differences have $p < 0.001$, with 95% CIs [0.172, 0.195] and [-0.288, -0.270]. Ret. is post-repair claim retention.

Pattern-level diagnostics. The pattern-wise breakdown in Table 2 shows that the gain is not concentrated in one synthetic phenomenon. Event-grounded prompting improves graph support for all eight pattern types, with the largest gains on sharp drops (+0.307), seasonality (+0.273), volatility shifts (+0.227), and anomalies (+0.226). Volatility shifts, seasonality, and anomalies still have the highest event-grounded graph-unsupported rates, indicating that localized or composite temporal behavior remains harder to match under the criterion. Appendix Table 18 shows representative claim-level failures.

5.2 Evidence Use

Statistical and verifier-guided repair evidence. Event-grounded prompting consistently improves graph support and reduces graph-unsupported claims. The gains are stable under paired bootstrap resampling: relative to Direct, event grounding raises graph support by 0.183 and reduces graph-unsupported rate by 0.279 ($p < 0.001$ for both; Appendix Table 14). CoT has low retention after verifier-guided repair (0.298), meaning that many generated claims must be removed, while event-grounded explanations retain 0.850 of claims. The mismatched-event stress test in Table 3 gives a complementary check: replacing the correct event list

Pattern	Direct Faith.	Event Faith.	Δ Faith.	Event Unsup.	Event Part.	Event Claims/Expl.
Upward trend	0.519	0.691	+0.172	0.008	0.301	8.70
Downward trend	0.493	0.685	+0.192	0.023	0.292	8.54
Sharp rise	0.552	0.714	+0.162	0.047	0.240	8.96
Sharp drop	0.452	0.759	+0.307	0.059	0.183	8.74
Volatility shift	0.332	0.558	+0.227	0.254	0.187	8.40
Change point	0.513	0.719	+0.206	0.063	0.218	7.92
Seasonality	0.385	0.658	+0.273	0.214	0.128	10.19
Anomaly	0.410	0.636	+0.226	0.235	0.129	6.26

Table 2: Pattern-wise diagnostics for 400 synthetic series, pooled across five models. Direct Faith. and Event Faith. report strict graph-supported fractions, while the remaining Event columns provide diagnostics on graph-unsupported claims, partially supported claims, and claim volume.

with events from another series drops graph support by 0.481 for DeepSeek-V4-Flash and 0.422 for GPT-4.1-mini. This suggests that event-grounded improvements come from evidence use rather than prompt formatting alone; extractor-sharing effects are examined in the verifier controls below.

Graph support versus informativeness. Appendix Table 17 shows why graph support should not be conflated with informativeness. CoT covers more gold events than direct prompting because it generates many more claims, but it also produces the highest verifier-unsupported rate. Event-grounded prompting is more conservative: it has higher graph support and lower graph-unsupported rate, but lower gold-event coverage than CoT. The deterministic event verbalizer provides an extractor-aligned reference: direct template verbalization reaches 0.936 graph support and a 0.040 graph-unsupported rate, with high event coverage and no API generation cost. This shows that strong automatic scores are possible when language is tightly constrained to the extracted event graph, but it should be read as an extractor-aligned reference rather than an LLM explanation method. It is not an upper bound on explanation quality: it verbalizes the extractor’s own event representation and omits richer but harder-to-verify explanatory language. Its high score shows that verifier-aligned language can be recognized reliably, but TIME-CLAIMBENCH should not be used to optimize automatic verifier scores alone.

5.3 Verifier Calibration

Controlled verifier and event-list checks. Table 4 first checks the verifier on controlled synthetic claim cases. Type-only matching reaches only 0.456 macro-F1, while adding interval and extracted-event-graph checks raises macro-F1 to

0.733; the gold-event oracle reaches 0.884, indicating that automatic event extraction is a major remaining bottleneck.

The generation-side event-list ablation likewise shows that sparse gold lists are not a simple upper bound: they reduce claim volume and graph support for both tested models (Appendix Table 19). The frozen extractor recovers the designed primary event in 394/400 series (98.5%; exact type and IoU ≥ 0.5), but this does not estimate precision for additional events. Re-verifying all 8,000 generations against sparse gold graphs yields model-averaged strict support of 0.074 for event grounding versus 0.130 for Direct; on gold-compatible claim types, strict rates are 0.519 versus 0.748 and supported-or-partial rates are 0.998 versus 0.983 (Appendix Tables 20 and 21). Since gold graphs contain only designed spans, this diagnoses reference sensitivity rather than independent truth.

Human calibration. In the original 180-claim audit (Table 5), three-way annotator agreement is $\kappa = 0.680$, whereas verifier agreement on the 156 consensus cases is 0.410 ($\kappa = 0.090$; macro-F1 0.353). Many claims receiving automatic PARTIAL/UNSUPPORTED labels are human-supported. Thus, graph-unsupported rates should be interpreted as conservative representation-mismatch diagnostics rather than absolute hallucination rates.

The independent 90-claim audit shows the same pattern: annotators agree on 76/90 cases (0.844; $\kappa = 0.572$), whereas verifier agreement on the consensus cases is 26/76 (0.342; $\kappa = 0.003$; macro-F1 0.234). Humans confirm 23/26 verifier-SUPPORTED claims, and in 42/50 mismatches, the verifier assigns a stricter label to a human-SUPPORTED claim. Because the audit is balanced by prompt and verifier label, these results reflect conditional calibration rather than corpus accuracy

Model	Event evidence	Faith. $\uparrow$	Δ Faith.	Unsup. $\downarrow$	Part.	Claims/Expl.	Ret.	Cost/Expl.
DeepSeek-V4-Flash	Matched events	0.734	+0.000	0.078	0.188	8.51	0.881	\$0.00043
DeepSeek-V4-Flash	Mismatched events	0.253	-0.481	0.294	0.453	8.85	0.571	\$0.00047
GPT-4.1-mini	Matched events	0.710	+0.000	0.085	0.205	9.26	0.865	\$0.00137
GPT-4.1-mini	Mismatched events	0.288	-0.422	0.302	0.410	9.30	0.589	\$0.00134

Table 3: Mismatched event-list ablation on 80 stratified synthetic series. Faith./Unsup. are strict supported/unsupported fractions under the correct graph. Mismatched prompts retain the raw series but receive another series’ events; the drop shows sensitivity to supplied evidence beyond prompt formatting.

Verifier	Macro-F1	Sup. F1	Part. F1	Unsup. F1	IoU
Majority unsupported	0.222	0.000	0.000	0.667	0.000
Type-only	0.456	0.552	0.000	0.817	0.531
Type+direction	0.456	0.552	0.000	0.815	0.531
Type+interval	0.733	0.819	0.564	0.817	0.604
Event graph (no interval)	0.458	0.552	0.005	0.817	0.537
Event graph	0.733	0.819	0.564	0.817	0.601
Gold-event oracle	0.884	0.856	0.797	1.000	0.651

Table 4: Verifier baselines and ablations on 1,600 synthetic claims from 400 test examples: one supported, one interval-mislocalized partial, and two unsupported claims per series. Sup., Part., and Unsup. denote class-specific F1; IoU is computed on supported and partial claims with predicted support intervals.

Comparison	N	Exact	Macro-F1	κ	Auto-Sup. Accept	Non-Sup. F1
Annotators (3-way)	180	0.867	–	0.680	–	–
Annotators (binary)	180	0.883	–	0.696	–	–
Verifier vs consensus gold	156	0.410	0.353	0.090	0.807	0.333

Table 5: Original balanced 180-claim audit. Consensus uses 156 annotator-agreement cases. Auto-Sup. Accept is human confirmation of verifier-SUPPORTED; Non-Sup. F1 merges PARTIAL/UNSUPPORTED.

(Appendix Table 25). Threshold tuning does not close this gap (Appendix E).

Real-world stress test. Both representative LLMs identify 0.960 of positive Real-NAB windows with an overlapping anomaly-like claim and have low explicit false-anomaly rates on normal windows (Table 6). Broad precision (0.505), interval IoU (0.080), and verifier–human agreement on a 90-claim audit (0.389; $\kappa = 0.083$) remain low, so the controlled split is the main setting (Appendix G). Across eight z-normalized descriptors, mean synthetic-range coverage is 84.6%, with 79.5% of real windows covered on at least six; this is descriptive shape coverage, not distributional equivalence (Appendix Table 11).

5.4 Operational Consistency and Human Judgment

On the same 40 series, graph-supported fractions are 0.711/0.577/0.355 for event-grounded/Direct/CoT. Blinded ratings reverse this ordering: event-grounded outputs score below

Model	Faith. $\uparrow$	Unsup. $\downarrow$	Pos. Hit $\uparrow$	FA $\downarrow$	Prec. $\uparrow$	IoU $\uparrow$
DeepSeek-V4-Flash	0.671	0.049	0.960	0.020	0.505	0.080
Event verbalizer	0.961	0.015	0.960	0.560	0.505	0.080
GPT-4.1-mini	0.617	0.049	0.960	0.010	0.505	0.080

Table 6: Real-NAB localization. Pos. Hit is positive-window overlap; FA is explicit false-anomaly rate on normal windows; Prec. is broad anomaly-like localization precision; IoU uses positive NAB windows. Faith./Unsup. use the extracted graph.

Metric	EG vs. Direct	EG vs. CoT
Faith. Δ [95% CI]	−1.938 [−2.225, −1.638]	−2.000 [−2.288, −1.688]
Info. Δ [95% CI]	−0.900 [−1.088, −0.688]	−1.525 [−1.738, −1.300]
Overall EG/Base/Tie	2/70/8	3/75/2

Table 7: Blinded preference study on GPT-4.1-mini, 40 series, and two annotators. Deltas are EG minus baseline; CIs use 5,000 series bootstraps. All Holm-adjusted paired tests have $p \leq 1.082 \times 10^{-6}$. Counts pool 80 descriptive judgments per comparator.

both baselines on faithfulness and informativeness (Table 7). This is a same-sample rank reversal, not a faithfulness–informativeness trade-off.

A period-4 event occurs in 25/40 extracted graphs and the same 25 event-grounded texts, but in 0/40 Direct/CoT texts. The human-rating gap is descriptively larger in this post hoc subgroup. Although strata confounding and absent claim-level adjudication preclude an error-rate or causal interpretation, the trace shows how graph-to-text adherence can propagate an additional event accepted by the shared verifier (Appendix F).

5.5 Summary of Findings

Event grounding improves consistency with the supplied representation, but not holistic preference in this sample. The calibration studies distinguish designed-event recall, the unresolved precision of additional events, claim-to-graph consistency, and whole-explanation human judgment.

6 Conclusion

We introduced TIMECLAIMBENCH, an executable claim-level benchmark for measuring consistency between LLM-generated time-series explanations and extracted temporal events. Across five LLMs, event-grounded prompting increases support under the shared graph, but a blinded study on one model and 40 series reverses that ordering under holistic human ratings. TIMECLAIMBENCH therefore reports claim-to-graph consistency as an operational diagnostic alongside designed-event recovery and human evaluation. It leaves the precision of additional extracted events unresolved.

Limitations

TimeClaimBench is limited to univariate series and a fixed rule-based event graph that both conditions and verifies event-grounded outputs; scores therefore measure representation consistency, not independent truth. Designed-event recovery does not establish precision for additional events, gold graphs are non-exhaustive, and negative-polarity parsing is imperfect.

Synthetic series isolate one primary phenomenon rather than realistic multi-pattern mixtures, while Real-NAB lacks complete event-type labels. The preference study covers one model, 40 series, and two annotators; its period-4 analysis is post hoc, stratum-confounded, and lacks claim-level adjudication.

The benchmark does not assess forecasting, causality, or external semantics. Extensions require broader data and models, independent event-precision annotation, and multivariate or contextual evidence protocols.

Ethical Considerations

TimeClaimBench is intended for evaluation and diagnosis, not as a substitute for domain expertise or automated decision-making in high-stakes settings. Experiments use synthetic and public time-series data, and the public artifact contains no private or personally identifiable information.

Human annotations support claim-level calibration and whole-explanation preference evaluation. Only aggregate results and de-identified records released with annotator permission are included; annotator identifiers, private randomization keys, and identifying metadata are excluded. The artifact documents data provenance, evaluation code, and the verifier’s operational limits.

Acknowledgments

We thank Shaobo Wang, Bowen Hao, Xunwen Zheng, and Shiman Yuan for data annotation and quality checks.

References

- Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. [The fact extraction and VERification over unstructured and structured information \(FEVEROUS\) shared task](#). In *Proceedings of the Fourth Workshop on Fact Extraction and VERification*, FEVER ’21, pages 1–13, Dominican Republic. Association for Computational Linguistics.
- Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Yuyang Wang. 2024. [Chronos: Learning the language of time series](#). *Transactions on Machine Learning Research*. Expert Certification.
- Pepa Atanasova, Oana-Maria Camburu, Christina Lioma, Thomas Lukasiewicz, Jakob Grue Simonsen, and Isabelle Augenstein. 2023. [Faithfulness tests for natural language explanations](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL ’23, pages 283–294, Toronto, Canada. Association for Computational Linguistics.
- Yifu Cai, Arjun Choudhry, Mononito Goswami, and Artur Dubrawski. 2024. [TimeSeriesExam: A time series understanding exam](#). In *Proceedings of the NeurIPS 2024 Workshop on Time Series in the Age of Large Models*, NeurIPS Workshops ’24, Vancouver, BC, Canada.
- Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. 2024. [TEMPO: Prompt-based generative pre-trained transformer for time series forecasting](#). In *Proceedings of the Twelfth International Conference on Learning Representations*, ICLR ’24, Vienna, Austria.

- Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyong Zhou, and William Yang Wang. 2020. [TabFact: A large-scale dataset for table-based fact verification](#). In *Proceedings of the International Conference on Learning Representations*, ICLR '20, Online.
- Ziwen Chen, Xiaoyuan Zhang, and Ming Zhu. 2025. [TS-CLIP: Time series understanding by CLIP](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, EMNLP '25, pages 4646–4664, Suzhou, China. Association for Computational Linguistics.
- Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. 2024. [A decoder-only foundation model for time-series forecasting](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *ICML '24*, pages 10148–10167, Vienna, Austria. PMLR.
- Qiqi Duan, Changlun Li, Chen Wang, Fan Zhang, Mengxiang Wang, Dayi Miao, Peixian Ma, Jiangpeng Yan, Liyuan Chen, Shuoling Liu, Preslav Nakov, Yuyu Luo, and Nan Tang. 2026. [Can agentic trading systems pay for their own intelligence?](#) *ArXiv preprint*, arXiv:2607.10286.
- Elizabeth Fons, Rachneet Kaur, Soham Palande, Zhen Zeng, Tucker Balch, Manuela Veloso, and Svitlana Vytrenko. 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*, EMNLP '24, pages 21598–21634, Miami, Florida, USA. Association for Computational Linguistics.
- Luyu Gao, Zhu Yun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023. [RARR: Researching and revising what language models say, using language models](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL '23, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.
- Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. 2024. [MO-MENT: A family of open time-series foundation models](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *ICML '24*, pages 16115–16152, Vienna, Austria. PMLR.
- Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew Gordon Wilson. 2023. [Large language models are zero-shot time series forecasters](#). In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, volume 36 of *NeurIPS '23*, pages 19622–19635, New Orleans, LA, USA. Curran Associates, Inc.
- Alon Jacovi and Yoav Goldberg. 2020. [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*, ACL '20, pages 4198–4205, Online. Association for Computational Linguistics.
- Harsh Jhamtani and Taylor Berg-Kirkpatrick. 2021. [Truth-conditional captions for time series data](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, EMNLP '21, pages 719–733, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Fengxian Ji, Yuke Li, Jingpu Yang, Juanfan Wu, Fan Zhang, Zhexuan Cui, Yu Xie, Min Peng, Qianqian Xie, Xiuying Chen, and Zhuohan Xie. 2026a. [Style wins, substance loses: A diagnosis of LLM-as-judge in idea generation](#). *ArXiv preprint*, arXiv:2608.01666.
- Fengxian Ji, Zhuohan Xie, Jingpu Yang, Fan Zhang, Zirui Song, and Xiuying Chen. 2026b. [Parametric memory decoding for zero-shot routing in LoRA-based external parametric memory](#). *ArXiv preprint*, arXiv:2607.04118.
- Fengxian Ji, Jingpu Yang, Zirui Song, Lang Gao, Junhong Liang, Zhenhao Chen, Jinghui Zhang, and Xiuying Chen. 2026c. [ServImage: An image generation and editing benchmark from real-world commercial imaging services](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL '26, pages 43504–43529, San Diego, California, United States. Association for Computational Linguistics.
- Fengxian Ji, Jingpu Yang, Zirui Song, Yuanxi Wang, Zhexuan Cui, Yuke Li, Qian Jiang, and Xiuying Chen. 2026d. [FineState-Bench: Benchmarking state-conditioned grounding for fine-grained GUI state setting](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, ACL '26, pages 43073–43088, San Diego, California, United States. Association for Computational Linguistics.
- Yushan Jiang, Wenchao Yu, Geon Lee, Dongjin Song, Kijung Shin, Wei Cheng, Yanchi Liu, and Haifeng Chen. 2025. [TimeXL: Explainable multi-modal time series prediction with LLM-in-the-loop](#). In *Proceedings of the 39th International Conference on Neural Information Processing Systems*, volume 38 of *NeurIPS '25*, pages 135431–135467, San Diego, CA, USA.
- Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y. Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, and Qingsong Wen. 2024. [Time-LLM: Time series forecasting by reprogramming large language models](#). In *Proceedings of the Twelfth International Conference on Learning Representations*, ICLR '24, Vienna, Austria.
- Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. [WiCE: Real-world entailment](#)

- for claims in Wikipedia. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, EMNLP '23, pages 7561–7583, Singapore. Association for Computational Linguistics.
- Shankar Kantharaj, Rixie Tiffany Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. 2022. [Chart-to-text: A large-scale benchmark for chart summarization](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL '22, pages 4005–4023, Dublin, Ireland. Association for Computational Linguistics.
- Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024. [Prometheus: Inducing fine-grained evaluation capability in language models](#). In *Proceedings of the Twelfth International Conference on Learning Representations*, ICLR '24, Vienna, Austria.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.
- Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. 2025. [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)*, ACL '25, pages 29736–29753, Vienna, Austria. Association for Computational Linguistics.
- Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. [SummaC: Re-visiting NLI-based models for inconsistency detection in summarization](#). *Transactions of the Association for Computational Linguistics*, 10:163–177.
- Alexander Lavin and Subutai Ahmad. 2015. [Evaluating real-time anomaly detection algorithms – the Numenta Anomaly Benchmark](#). In *Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications*, ICMLA '15, pages 38–44, Miami, FL, USA.
- Geon Lee, Wenchao Yu, Kijung Shin, Wei Cheng, and Haifeng Chen. 2025. [TimeCAP: Learning to contextualize, augment, and predict time series events with large language model agents](#). In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence*, volume 39 of AAAI '25, pages 18082–18090, Philadelphia, PA, USA.
- Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. [G-Eval: NLG evaluation using GPT-4 with better human alignment](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, EMNLP '23, pages 2511–2522, Singapore. Association for Computational Linguistics.
- Yong Liu, Guo Qin, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. 2024. [AutoTimes: Autoregressive time series forecasters via large language models](#). In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, volume 37 of *NeurIPS '24*, pages 122154–122184, Vancouver, BC, Canada. Curran Associates, Inc.
- Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. [ChartQA: A benchmark for question answering about charts with visual and logical reasoning](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, ACL '22, pages 2263–2279, Dublin, Ireland. Association for Computational Linguistics.
- Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. [FActScore: Fine-grained atomic evaluation of factual precision in long form text generation](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, EMNLP '23, pages 12076–12100, Singapore. Association for Computational Linguistics.
- Zijie Pan, Yushan Jiang, Sahil Garg, Anderson Schneider, Yuriy Nevmyvaka, and Dongjin Song. 2024. [S²IP-LLM: Semantic space informed prompt learning with LLM for time series forecasting](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *ICML '24*, pages 39135–39153, Vienna, Austria. PMLR.
- Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2023. [Measuring attribution in natural language generation models](#). *Computational Linguistics*, 49(4):777–840.
- Medhasweta Sen, Zachary Gottesman, Jiaying Qiu, C. Bayan Bruss, Nam Nguyen, and Tom Hartvigsen. 2026. [BEDTime: A unified benchmark for automatically describing time series](#). In *Proceedings of the 43rd International Conference on Machine Learning*, ICML '26, Seoul, South Korea.
- Preetham Sivalingham, Murari Mandal, Dhruv Kumar, and Saurabh Deshpande. 2026. [TSQueryBench: LLM-as-a-judge for time series explanations](#). *ArXiv preprint*, arXiv:2604.02118.
- Marek Strong and Andreas Vlachos. 2025. [TSVer: A benchmark for fact verification against time-series evidence](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, EMNLP '25, pages 29906–29926, Suzhou, China. Association for Computational Linguistics.
- Chenxi Sun, Hongyan Li, Yaliang Li, and Shenda Hong. 2024. [TEST: Text prototype aligned embedding to](#)

- activate LLM’s ability for time series. In *Proceedings of the Twelfth International Conference on Learning Representations*, ICLR ’24, Vienna, Austria.
- Benny Tang, Angie Boggust, and Arvind Satyanarayan. 2023. [VisText: A benchmark for semantically rich chart captioning](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL ’23, pages 7268–7298, Toronto, Canada. Association for Computational Linguistics.
- Mohamed Trabelsi, Aidan Boyd, Jin Cao, and Huseyin Uzunalioglu. 2025. [Time series language model for descriptive caption generation](#). *Engineering Applications of Artificial Intelligence*, 162:112673.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS ’22, Red Hook, NY, USA. Curran Associates Inc.
- Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. 2024. [Long-form factuality in large language models](#). In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, volume 37 of *NeurIPS ’24*, pages 80756–80827, Vancouver, BC, Canada. Curran Associates, Inc.
- Zhe Xie, Zeyan Li, Xiao He, Longlong Xu, Xidao Wen, Tieying Zhang, Jianjun Chen, Rui Shi, and Dan Pei. 2025. [ChatTS: Aligning time series with LLMs via synthetic data for enhanced understanding and reasoning](#). *Proceedings of the VLDB Endowment*, 18(8):2385–2398.
- Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. [AlignScore: Evaluating factual consistency with a unified alignment function](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL ’23, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. 2023. [Judging LLM-as-a-judge with MT-Bench and Chatbot Arena](#). In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, volume 36 of *NeurIPS ’23*, pages 46595–46623, New Orleans, LA, USA. Curran Associates, Inc.
- Luca Zhou, Pratham Yashwante, Marshall Fisher, Alessio Sampieri, Zihao Zhou, Fabio Galasso, and Rose Yu. 2026. [CaTS-Bench: Can language models describe time series?](#) In *Findings of the Association for Computational Linguistics: ACL 2026*, ACL ’26, pages 34479–34519, San Diego, California, United States. Association for Computational Linguistics.
- Tian Zhou, Peisong Niu, Xue Wang, Liang Sun, and Rong Jin. 2023. [One fits all: Power general time series analysis by pretrained LM](#). In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, volume 36 of *NeurIPS ’23*, pages 43322–43355, New Orleans, LA, USA. Curran Associates, Inc.

A Details for TIMECLAIMBENCH

A.1 Benchmark Construction

Synthetic split. The controlled synthetic split makes event types, intervals, directions, and reference claims known by construction, supporting claim-level verification alongside explanation-level preference judgments. It covers eight first-release temporal phenomena: upward and downward trends, sharp rises and drops, volatility shifts, change points, seasonality, and anomalies. Each JSONL record contains a length-80 time series, metadata, gold events, a question, and reference claims; each series has one designed primary phenomenon.

The split contains 1,200 training, 200 development, and 400 test examples. Main experiments use all 400 test examples, with 50 per phenomenon; a stratified 80-example pilot subset supports faster provider checks. Table 8 gives the exact parameter ranges. The generator randomizes locations, spans, directions, and magnitudes where applicable but does not simulate the full complexity or co-occurrence structure of real signals.

Real-NAB localization diagnostic. To test whether the pipeline transfers beyond synthetic control, we add a real-world localization diagnostic based on the Numenta Anomaly Benchmark (NAB) (Lavin and Ahmad, 2015). We use the real NAB subsets *realKnownCause*, *realTraffic*, *realTweets*, *realAWScloudwatch*, and *realAdExchange*. NAB provides timestamped univariate series and labeled anomalous windows. We convert these labels into gold ANOMALY events for localization diagnostics while still using the same automatic event extractor to detect trends, sharp changes, seasonality, volatility shifts, change points, and anomalies.

The Real-NAB localization diagnostic contains 200 fixed-length windows: 100 anomaly-centered windows and 100 normal windows. Values are normalized within each window for compatibility with the event operators. Original values, timestamps,

Phenomenon	Randomized parameters	Gold event span
Upward/downward trend	$ \text{slope} \sim U(0.08, 0.22)$; random start/end	randomized subinterval
Sharp rise/drop	magnitude $\sim U(4, 8)$; end-start 2–5; random location	transition interval
Volatility shift	standard deviation $0.25 \leftrightarrow 1.4$; random direction/split	split to end
Change point	levels $U(96, 99)$ and $U(101, 104)$; random direction/split	split point
Seasonality	period $\in \{6, 8, 10, 12, 16\}$; amplitude $\sim U(1.5, 3.5)$	full series
Anomaly	magnitude $\sim U(5, 9)$; random sign/location	anomaly point

Table 8: Synthetic generator settings. Every series has length 80, Gaussian background noise with $\sigma = 0.5$, and one designed primary phenomenon. Locations, spans, directions, and magnitudes are randomized where applicable.

Split	Size	Use
Synthetic train	1,200	development and prompt checks
Synthetic dev	200	validation and debugging
Synthetic test	400	main evaluation
Synthetic pilot subset	80	fast provider checks
Real-NAB localization diagnostic	200	real-world localization check

Table 9: Benchmark splits. The synthetic test split is the main evaluation set and contains 50 examples for each of the eight synthetic phenomena. The Real-NAB localization diagnostic contains 100 anomaly-centered windows and 100 normal windows.

source file names, and NAB label windows are preserved in metadata.

We treat Real-NAB as a localization diagnostic rather than as the primary benchmark. NAB primarily supports anomaly-window and change-like evaluation under real noise. It does not provide complete gold labels for all eight synthetic phenomenon types, and its anomaly windows can be coarser than the temporal spans expressed in natural-language claims. Therefore, Real-NAB results should be read as localization and robustness diagnostics, not as complete human-equivalent claim verification.

Subset	Windows	Annotation used
Anomaly-centered	100	NAB windows converted to gold anomaly events
Normal	100	no NAB anomaly window in selected span

Table 10: Real-NAB localization diagnostic construction.

Synthetic–real shape diagnostic. We compare 400 synthetic series with the 200 Real-NAB windows using eight descriptors after z-normalizing each window. For each descriptor, coverage is the fraction of Real-NAB values inside the synthetic 5th–95th percentile range. Table 11 reports feature-wise coverage of 0.780–0.905, with mean 0.846. In addition, 79.5% of Real-NAB windows are covered on at least six of eight descriptors. These summaries test coarse shape overlap, not distributional equivalence or semantic realism.

Z-normalized descriptor	Real within synthetic Q05–Q95
Absolute trend slope	0.835
Lag-1 autocorrelation	0.820
First-difference std.	0.790
Maximum absolute step	0.780
Spectral entropy	0.895
Peak absolute z-score	0.885
Standardized range	0.905
Log volatility ratio	0.855
Mean	0.846

Table 11: Descriptive shape coverage for 200 Real-NAB windows relative to 400 synthetic series after per-window z-normalization. Overall, 79.5% of real windows fall in the synthetic 5th–95th percentile range for at least six of eight descriptors. This is not a test of distributional equivalence.

A.2 Event Extraction and Extracted Event Graph

The verifier converts each time series into temporal events using deterministic operators with fixed settings across all models and prompts; Table 12 summarizes the default settings. The extractor emits typed events with intervals, scores, and attributes such as direction, magnitude, slope, period, or z-score. Overlapping events of the same type are deduplicated when interval IoU is at least 0.8, keeping the higher-scoring event.

The extracted events form an extracted event graph whose nodes are temporal events. Edges encode temporal relations such as BEFORE, AFTER, OVERLAPS, and CONTAINS. The verifier retrieves candidate events by claim type and uses the graph to check whether the claimed temporal structure is compatible with the extracted evidence.

A.3 Claim-Level Labels and Error Taxonomy

Table 13 expands the three claim-level labels, SUPPORTED, PARTIAL, and UNSUPPORTED, into operational boundary cases for annotation and verifier diagnostics. For claims not fully supported, finer error types are recorded to characterize specific

Operator	Evidence emitted	Key fixed settings
Trend	Upward/downward trend events with slope, interval, and strength.	Least-squares slopes over full, first-half, second-half, and middle-half spans; minimum length 8; slope threshold 0.03; weak < 0.08, moderate < 0.16, strong otherwise.
Sharp rise/drop	Local sharp-change events with direction, magnitude, and interval.	Difference window 5; z-normalized change at least 2.0 relative to first-difference standard deviation.
Volatility shift	Intervals where local variability changes substantially.	Before/after window 10; variance-ratio-style volatility ratio at least 1.8.
Change point	Mean-shift events with split point and affected interval.	All candidate splits scanned; at least four points per side; best mean-shift score > 1.0.
Seasonality	Periodic events with period and strength.	Periods 4–24, capped at half the sequence length; autocorrelation at least 0.45.
Anomaly	Point or short-window anomaly events.	Absolute z-score at least 3.0.
Peak/trough	Local extrema used as auxiliary evidence.	Prominence at least 1.0.
Deduplication	Removal of redundant events.	Same-type events deduplicated when interval IoU is at least 0.8; higher-scoring event is kept.

Table 12: Fixed event-extraction settings used by the default TIMECLAIMBENCH verifier.

failure modes, including interval shifts, overstatement, numeric mismatch, hallucinated events, unsupported causal statements, and chronology violations. These error types support failure analysis but are not additional classes in the main evaluation.

A.4 Claim Parsing and Verification

Claim parsing. The claim parser maps free-form explanations to atomic temporal claims. It detects event types with lexical patterns, including trends, rises, drops, volatility changes, change points, seasonality, anomalies, peaks, troughs, and recoveries. It extracts explicit index spans such as “indices 20–30”, maps coarse phrases such as “early”, “middle”, “late”, “first half”, and “final segment” to approximate spans, and records optional direction, strength, and numeric values.

The parser is conservative. If a sentence contains multiple temporal phenomena, it is split into separate claims when possible. If no temporal phenomenon can be identified, the claim type is marked as unknown and normally requires strong matching evidence to be labeled supported. Unsupported causal language is marked when the explanation introduces an external cause not observable from the univariate sequence.

Claim–event scoring. Let c be a parsed claim and e a candidate event from the extracted event graph. For each claim c , the verifier retrieves candidate events whose types are compatible with the

claim. It computes a weighted agreement score:

$$\begin{aligned}
 S(c, e) = & w_t S_{\text{type}}(c, e) \\
 & + w_d S_{\text{direction}}(c, e) \\
 & + w_i S_{\text{interval}}(c, e) \\
 & + w_s S_{\text{strength}}(c, e) \\
 & + w_n S_{\text{numeric}}(c, e).
 \end{aligned}$$

The fixed weights are $(w_t, w_d, w_i, w_s, w_n) = (.35, .20, .25, .10, .10)$. The type score measures compatibility between the claim type and event type. The direction score checks sign or trend direction when present. The interval score is interval IoU when a claim has an explicit or inferred span. The strength score compares coarse strength or magnitude categories. The numeric score checks extracted values when the claim contains a concrete value, count, index, or period.

Optional attributes absent from the claim receive a neutral component score of 1.0 rather than triggering weight renormalization. If the claim specifies direction, strength, or numeric content but the matched event lacks the corresponding attribute, the verifier assigns a conservative default score of 0.8 for that component. If a claim has no explicit or inferred interval, the interval component is therefore neutral and the interval score is not used as a hard support gate, but the matched event must still be compatible in type and direction.

Label assignment. The best event match determines the automatic label. A claim is labeled SUPPORTED when the best score is at least 0.75 and, for claims with explicit intervals, interval IoU is at least 0.5. After parsing, a direction-incompatible atomic

Label	Error type	Definition	Typical example	Boundary rule
SUPPORTED	–	The main temporal phenomenon, direction, and approximate interval are grounded in the series.	“A sharp rise occurs around indices 20–25.”	Exact wording need not match the event name if the phenomenon and interval are compatible.
PARTIAL	interval shift	The right phenomenon is present but localized to a shifted, too broad, or too narrow interval.	“The increase lasts through the final third” when the event ends near the middle.	Use PARTIAL when the claim would become supported after interval correction.
	overstatement	A grounded pattern is described with excessive strength, magnitude, duration, or frequency.	“A strong trend” for a weak but detectable trend.	Use PARTIAL when the phenomenon exists but a salient attribute is wrong.
	numeric error	The event is present, but the claim gives an inaccurate value, count, period, or index.	“The cycle repeats every 8 steps” when the detected period is closer to 12.	Use PARTIAL when the temporal pattern is grounded but the numeric detail is wrong.
	mixed claim	One part of a claim is grounded, but another part adds unsupported content.	“The series rises near $t = 20$ because of an external shock.”	Use PARTIAL when supported and unsupported parts are inseparable in the same claim.
UNSUPPORTED	pattern hallucination	The explanation introduces a temporal pattern absent from the evidence.	Claiming seasonality or a change point when no compatible event is detected.	Use UNSUPPORTED when no compatible event can ground the main claim.
	wrong direction or timing	The claimed event has the wrong sign or occurs in a region where no compatible event is visible.	Claiming a sharp drop where the evidence shows a sharp rise.	Direction-incompatible atomic claims are gated to UNSUPPORTED; use PARTIAL only for inseparable mixed claims with a grounded subclaim.
	causal claim	The explanation infers an external cause not observable from the univariate series.	“The drop is caused by an external shock.”	Causal language is unsupported unless causal evidence is explicitly provided.
	chronology violation	The claim uses future-window evidence to justify an earlier observation under a decision time.	“At $t = 30$, the series has entered a sustained decline” when the decline is only visible after $t = 50$.	Use this error when the claim may be true with full-sequence access but invalid at the specified decision time.

Table 13: Full claim-level support taxonomy and boundary rules used for annotation and verifier diagnostics.

claim is gated to UNSUPPORTED. Otherwise, a claim is labeled PARTIAL only when a compatible candidate event exists and the best score is at least 0.45, but the claim fails full support because of a non-direction grounding defect. Operationally, this includes an explicit or inferred interval whose IoU with the matched event is below 0.5 but still indicates a shifted or broadened span, compatible type and direction with mismatched strength or magnitude, numeric mismatch for a claimed value, count, index, or period, or an inseparable mixed claim whose grounded subclaim can be matched while the remaining content is unsupported. For mixed sentences, the parser first splits separable temporal claims when possible; remaining inseparable mixed claims are treated as PARTIAL only when a grounded subclaim is matched. Remaining claims are labeled UNSUPPORTED. All thresholds are fixed before model comparison and are used consistently across experiments.

A.5 Verifier-Guided Repair

The verifier-guided repair module uses verifier output to produce a conservative edited explanation. It preserves claims labeled SUPPORTED, cor-

rects claims labeled PARTIAL using the matched grounded event when possible, and removes claims labeled UNSUPPORTED. The repair prompt is instructed not to introduce new patterns, numeric values, or causal explanations that are not present in the evidence.

Because the repaired output is guided and re-evaluated by the same verifier, post-repair graph support is a fixed-criterion measurement rather than independent human evidence. We therefore report claim retention and supported-claim preservation with the post-repair graph-supported rate.

A.6 Metric Definitions

Table abbreviations. Across tables, *Faith.* denotes the strict SUPPORTED fraction under the fixed extracted-event graph, *Unsup.* the UNSUPPORTED fraction under that graph, *Part.* partial rate, *Expl.* explanation, *Ret.* claim retention after verifier-guided repair, *Sem. Pres.* supported-claim preservation, *Chron. Viol.* chronology violations, *Rep.* verifier-guided repair, *Sup./Supp.* supported, *Non-Sup.* non-supported, and *Cov.* coverage. *Claims/Expl.*, *Cost/Expl.*, and *Tokens/Expl.* are per-explanation averages. *Faith.+Rep.* and *Un-*

sup.+Rep. report the corresponding metric after verifier-guided repair. For Real-NAB and calibration tables, *Pos. Hit*, *FA*, *Prec.*, *IoU*, and *Macro-F1* denote positive-window hit rate, false-anomaly rate, precision, interval intersection-over-union, and macro-averaged F1. *Non-Sup. F1* is the binary F1 after collapsing PARTIAL and UNSUPPORTED into a single non-supported class.

Within each model-prompt run, automatic claim-label rates are computed as claim-level micro-averages over all extracted claims. Cross-model aggregation is specified in each table caption. Let *S*, *P*, and *U* denote SUPPORTED, PARTIAL, and UNSUPPORTED. Let $\mathcal{D}$ be a set of generated explanations and

$$N_C = \sum_{r \in \mathcal{D}} |\mathcal{C}(r)|$$

be the total number of extracted claims. If $N_C = 0$, claim-level rates are defined as 0.

Strict graph-supported rate. The main operational metric is the strict supported-claim fraction under the extracted event graph:

$$\text{Faith} = \frac{1}{N_C} \sum_{r \in \mathcal{D}} \sum_{c_i \in \mathcal{C}(r)} \mathbf{1}[\hat{y}_i = S].$$

Partial claims do not receive credit in the main score. We report them separately:

$$\text{PartialRate} = \frac{1}{N_C} \sum_{r \in \mathcal{D}} \sum_{c_i \in \mathcal{C}(r)} \mathbf{1}[\hat{y}_i = P],$$

$$\text{UnsupportedRate} = \frac{1}{N_C} \sum_{r \in \mathcal{D}} \sum_{c_i \in \mathcal{C}(r)} \mathbf{1}[\hat{y}_i = U].$$

Claims per explanation. Claims per explanation measures how many auditable temporal claims a prompt produces:

$$\text{Claims/Expl.} = \frac{1}{|\mathcal{D}|} \sum_{r \in \mathcal{D}} |\mathcal{C}(r)|.$$

This metric is reported because a prompt may generate more coverage by producing many more claims, while also increasing unsupported content.

Macro-F1. For a label set $\mathcal{Y}$, class-wise precision, recall, and F1 are

$$P_y = \frac{\text{TP}_y}{\text{TP}_y + \text{FP}_y}, \quad R_y = \frac{\text{TP}_y}{\text{TP}_y + \text{FN}_y},$$

$$F1_y = \frac{2P_y R_y}{P_y + R_y}.$$

Macro-F1 is the unweighted average over labels:

$$\text{MacroF1} = \frac{1}{|\mathcal{Y}|} \sum_{y \in \mathcal{Y}} F1_y.$$

For three-way support-label evaluation, $\mathcal{Y} = \{S, P, U\}$. For binary non-supported F1, PARTIAL and UNSUPPORTED are collapsed into the positive non-supported class. If a precision, recall, or F1 denominator is empty, the corresponding value is defined as 0.

Cohen’s κ . We report unweighted Cohen’s κ for two categorical labelers:

$$\kappa = \frac{p_o - p_e}{1 - p_e}, \quad p_e = \sum_{y \in \mathcal{Y}} p_1(y)p_2(y),$$

where p_o is exact agreement and $p_j(y)$ is labeler j ’s marginal proportion for label y . For three-way claim audits, $\mathcal{Y} = \{S, P, U\}$; binary audits merge *P* and *U*. For preference agreement, randomized A/B choices are first decoded to {event-grounded, comparator, tie}; Direct and CoT are mapped to a common comparator label only when the two comparison blocks are pooled.

Interval localization. Intervals are zero-based closed index spans. For $I_1 = [s_1, t_1]$ and $I_2 = [s_2, t_2]$,

$$a = \max\{0, \min(t_1, t_2) - \max(s_1, s_2) + 1\},$$

$$b = \max(t_1, t_2) - \min(s_1, s_2) + 1,$$

$$\text{IoU}(I_1, I_2) = a/b.$$

If a claim has no explicit or inferred interval, the interval-IoU gate is not applied, but the verifier must still find compatible evidence for the claim type and direction.

Coverage. Coverage measures whether faithful claims mention important events. Let $E_{\leq K}(X)$ denote the available highest-scoring extracted events for a series X , up to K events. Top- K supported coverage is

$$\text{TopKCov}_S = \frac{1}{|\mathcal{D}|} \sum_{r \in \mathcal{D}} q_r,$$

$$q_r = \frac{\sum_{e \in E_{\leq K}(X_r)} \mathbf{1}[A(e, r)]}{\max(1, |E_{\leq K}(X_r)|)}.$$

Here, $A(e, r)$ holds if there exists a claim $c_i \in \mathcal{C}(r)$ such that $\hat{y}_i = S$ and c_i matches event e .

Top- K grounded coverage replaces $\hat{y}_i = S$ with $\hat{y}_i \in \{S, P\}$. If fewer than K events are extracted, the denominator uses the number of available events; if no events are extracted, that example contributes 0.

Gold coverage is computed only for synthetic data and measures whether each gold event is matched by a compatible SUPPORTED or PARTIAL claim with interval IoU of at least 0.3. If a series has no gold events for a gold-coverage diagnostic, its contribution is defined as 0.

Repair diagnostics. Let r' be the repaired explanation. Claim retention is

$$\text{Retention} = \frac{|\mathcal{C}(r')|}{|\mathcal{C}(r)|}.$$

If the original explanation contains no extracted claims, retention is defined as 0. In the full synthetic evaluation, this case does not occur (0/8,000 model-prompt outputs), so the convention does not affect aggregate retention trends. Supported-claim preservation measures whether originally supported claims remain present after verifier-guided repair:

$$\text{SupportedPreservation} = \frac{N_{\text{pres}}}{N_S}.$$

Here N_S is the number of originally supported claims, and N_{pres} counts those claims when r' contains a supported claim grounded by an event overlapping the original evidence event. If $N_S = 0$, supported-claim preservation is defined as 0. This is an operational semantic-preservation diagnostic, not an independent human evaluation of repair quality.

Chronology violations. When a decision time t_0 is specified, a claim violates chronology if its supporting evidence interval starts after t_0 , or if the claim requires future-window evidence to justify an earlier observation. Chronology violations are reported as an error-type diagnostic.

Cost. Cost per explanation is computed from provider token usage and the pricing configuration:

$$\text{Cost/Expl.} = \frac{\text{TotalProviderCost}}{|\mathcal{D}|}.$$

Real-NAB localization diagnostics. For Real-NAB, let W^+ be anomaly-centered windows and W^- be normal windows. Let $\mathcal{A}$ denote anomaly-like claim types, including ANOMALY,

SHARPRISE, SHARPDROP, and CHANGEPOINT. A claim is considered grounded for these diagnostics if it is labeled SUPPORTED or PARTIAL.

Positive hit rate measures whether positive windows contain at least one grounded anomaly-like claim overlapping the NAB anomaly window:

$$\text{PosHit} = \frac{1}{|W^+|} \sum_{w \in W^+} \mathbf{1}[P(w)].$$

Here, $P(w)$ holds if there exists an anomaly-like claim $c_i \in \mathcal{A}$ such that $\hat{y}_i \in \{S, P\}$ and $I(c_i) \cap I_{\text{NAB}}(w) \neq \emptyset$.

Explicit false-anomaly rate measures whether normal windows contain an explicit grounded anomaly claim:

$$\text{ExplicitFA} = \frac{1}{|W^-|} \sum_{w \in W^-} \mathbf{1}[F(w)].$$

Here, $F(w)$ holds if there exists a claim c_i such that $\tau(c_i) = \text{ANOMALY}$ and $\hat{y}_i \in \{S, P\}$.

Anomaly-like precision is a broad window-level localization diagnostic:

$$\text{AnomLikePrec} = \frac{\sum_{w \in W^+ \cup W^-} \mathbf{1}[M(w)]}{\sum_{w \in W^+ \cup W^-} \mathbf{1}[G(w)]}.$$

Here, $G(w)$ holds if the explanation contains at least one grounded anomaly-like claim in the selected Real-NAB window. $M(w)$ is defined only as a true match for anomaly-centered windows: it holds when $w \in W^+$, $G(w)$ is true, and at least one grounded anomaly-like claim overlaps the NAB anomaly interval $I_{\text{NAB}}(w)$. For normal windows $w \in W^-$, there is no NAB anomaly interval in the selected span, so $M(w) = 0$ by definition. Thus normal windows with grounded anomaly-like claims contribute to the denominator but not the numerator. If no window contains a grounded anomaly-like claim, anomaly-like precision is defined as 0. For positive windows, we also report mean interval IoU with the NAB anomaly window and normalized center distance:

$$\text{CenterDist}(I_1, I_2) = \frac{|m(I_1) - m(I_2)|}{T},$$

where $m(I)$ is the interval midpoint and T is the window length.

Aggregation and uncertainty. For prompt-level results, we first compute claim-level rates for each model-prompt run and then average over model

runs. Bootstrap confidence intervals resample the 400 synthetic test series within each fixed model run. Paired prompt comparisons preserve shared model–series identifiers.

B Evaluation Protocol

B.1 Task and Generation Settings

The full synthetic evaluation uses the 400-example synthetic test split, with 50 examples for each of the eight temporal phenomena. Each example is converted into a model-ready input consisting of the raw univariate series and a question asking for a concise time-series explanation. For the Real-NAB localization diagnostic, we use 200 fixed-length windows: 100 anomaly-centered windows and 100 normal windows.

We evaluate five LLM backbones from four providers under four prompting conditions. Direct, chain-of-thought, and structured prompts receive only the raw time series and the question. The event-grounded prompt additionally receives the automatically extracted event list and instructs the model to cite listed evidence. All provider calls use temperature 0, the same input order, cached retry logic, and the same downstream claim parser, verifier, verifier-guided repair module, and cost-accounting pipeline. Thus, prompt comparisons differ in prompting conditions, while verification is performed by the same fixed automatic verifier.

B.2 Formal Evaluation Task

Using the metric notation from Appendix A.6, each instance contains a zero-indexed univariate time series $X = (x_0, \dots, x_{T-1})$ and a generated explanation r . The claim parser maps r to a set of atomic temporal claims, $\mathcal{C}(r) = \{c_i\}_{i=1}^{n_r}$. The event extractor maps X to an extracted event graph, $G_X = (\mathcal{V}_X, \mathcal{R}_X)$, where each event $e \in \mathcal{V}_X$ has an event type $\tau(e)$, optional direction $d(e)$, interval $I(e) = [s_e, t_e]$, score, and numeric attributes such as slope, magnitude, period, or z-score. Relations in $\mathcal{R}_X$ encode temporal links such as BEFORE, AFTER, OVERLAPS, and CONTAINS.

For claim c_i , the verifier assigns a support label $\hat{y}_i \in \{\text{SUPPORTED}, \text{PARTIAL}, \text{UNSUPPORTED}\}$ and, when applicable, an evidence event $\hat{e}_i$ and support interval $\hat{I}_i$. The verifier also records error types for non-supported claims when an error category can be identified.

B.3 Model and Prompt Details

The full synthetic evaluation uses five LLM backbones: GPT-4.1-mini, GPT-5.4-mini, DeepSeek-V4-Flash, Gemini-3-Flash, and Claude Sonnet 4.5. We compare four prompting conditions. Direct, chain-of-thought, and structured prompts receive only the raw time series and the question. The event-grounded prompt additionally receives the automatically extracted event list. Generation settings are summarized in Appendix B.1 and prompt templates are presented below.

B.4 Prompt Templates

Direct prompt.

You are a time-series analyst.
Given the following time series, describe the major temporal patterns.
Focus on trends, peaks, drops, volatility, anomalies, seasonality, and change points.
Time series: {series}
Question: {question}
Answer with a concise explanation.

Chain-of-thought prompt.

You are a time-series analyst.
Analyze the time series step by step.
First inspect the overall trend.
Then inspect local peaks and drops.
Then inspect volatility and anomalies.
Finally provide a concise explanation.
Time series: {series}
Question: {question}

Structured prompt.

You are a time-series analyst.
Return a concise bullet list of grounded temporal patterns.
Only include claims that can be checked against the series.
Time series: {series}
Question: {question}

Event-grounded prompt.

You are a time-series analyst.
You are given a time series and automatically extracted events.
Write an explanation using only the listed events.
Do not mention patterns not supported by the event list.
Cite event IDs in brackets.
Time series: {series}
Extracted events: {events}
Question: {question}
Output format:
Explanation:
- [E?] ...
- [E?] ...

C Additional Empirical-Finding Tables

Prompt	Faith. 95% CI $\uparrow$	Unsup. 95% CI $\downarrow$	Δ Faith. (EG–Prompt)	Δ Unsup. (EG–Prompt)
Direct	0.493 [0.485, 0.501]	0.392 [0.383, 0.400]	0.183 [0.172, 0.195]	-0.279 [-0.288, -0.270]
CoT	0.360 [0.355, 0.365]	0.558 [0.552, 0.564]	0.316 [0.306, 0.325]	-0.446 [-0.454, -0.438]
Structured	0.353 [0.344, 0.361]	0.468 [0.458, 0.477]	0.323 [0.311, 0.335]	-0.355 [-0.366, -0.345]
Event-grounded	0.676 [0.667, 0.684]	0.113 [0.105, 0.120]	–	–

Table 14: Bootstrap uncertainty for full synthetic results. Series are resampled within model run before averaging five models. Differences are paired event-grounded-minus-prompt comparisons; all have $p < 0.001$ for strict graph support and graph-unsupported rate.

Verifier label	N	Human sup.	Human part.	Human unsup.	Gate-consistent
Supported	57	0.807	0.035	0.158	0.807
Partial	59	0.797	0.153	0.051	0.203
Unsupported	40	0.750	0.025	0.225	0.250

Table 15: Verifier calibration on the 156 consensus-gold human-audit claims. Human columns show the consensus label distribution within each automatic verifier label. Gate-consistent counts verifier-supported claims as accepted only when humans also mark them supported, and counts verifier-partial/unsupported claims as accepted when humans mark them non-supported.

Model	Prompt	Faith. $\uparrow$	Unsup. $\downarrow$	Claims/Expl.	Faith.+Rep. $\uparrow$	Unsup.+Rep. $\downarrow$	Ret. $\uparrow$	Cost/Expl. $\downarrow$
Claude Sonnet 4.5	Direct	0.401	0.542	20.18	0.898	0.041	0.191	\$0.00674
	CoT	0.345	0.590	34.09	0.870	0.039	0.134	\$0.00991
	Structured	0.314	0.514	9.61	0.787	0.063	0.249	\$0.00584
	Event-grounded	0.697	0.140	7.89	0.986	0.004	0.797	\$0.01104
DeepSeek-V4-Flash	Direct	0.544	0.290	5.44	0.975	0.018	0.705	\$0.00051
	CoT	0.429	0.460	16.52	0.974	0.015	0.450	\$0.00088
	Structured	0.407	0.424	5.16	0.987	0.011	0.571	\$0.00064
	Event-grounded	0.746	0.077	8.88	0.992	0.006	0.899	\$0.00044
Gemini-3-Flash	Direct	0.450	0.399	9.96	0.954	0.027	0.536	\$0.00379
	CoT	0.319	0.610	25.30	0.966	0.022	0.303	\$0.00476
	Structured	0.368	0.448	6.90	0.960	0.028	0.542	\$0.00441
	Event-grounded	0.636	0.098	7.84	0.997	0.002	0.871	\$0.00554
GPT-4.1-mini	Direct	0.625	0.297	5.55	0.981	0.011	0.700	\$0.00044
	CoT	0.351	0.552	30.39	0.967	0.018	0.333	\$0.00105
	Structured	0.349	0.474	5.99	0.968	0.021	0.518	\$0.00050
	Event-grounded	0.706	0.082	9.71	0.992	0.001	0.881	\$0.00140
GPT-5.4-mini	Direct	0.445	0.432	10.20	0.949	0.032	0.515	\$0.00150
	CoT	0.357	0.580	28.59	0.952	0.024	0.270	\$0.00267
	Structured	0.326	0.479	6.41	0.970	0.029	0.505	\$0.00151
	Event-grounded	0.595	0.166	7.99	1.000	0.000	0.804	\$0.00260

Table 16: Full synthetic results on 400 series across five LLMs. Faith. is the strict graph-supported fraction; Ret. is post-repair claim retention. Costs use provider token records and the pricing configuration.

Prompt	Expl.	Faith. $\uparrow$	Unsup. $\downarrow$	Claims/Expl.	Top-5 Sup. Cov. $\uparrow$	Top-5 Grounded Cov. $\uparrow$	Gold Sup. Cov. $\uparrow$	Gold Grounded Cov. $\uparrow$
Direct	2000	0.459	0.439	10.27	0.243	0.300	0.721	0.797
CoT	2000	0.354	0.567	26.98	0.337	0.391	0.811	0.868
Structured	2000	0.347	0.474	6.82	0.176	0.250	0.532	0.654
Event-grounded	2000	0.679	0.111	8.46	0.368	0.421	0.611	0.675
Event verbalizer	400	0.936	0.040	11.35	0.832	0.832	0.765	0.767

Table 17: Coverage diagnostics on 400 synthetic series across five LLMs. Top-5 coverage uses the highest-scoring extracted events; gold coverage uses designed events and $\text{IoU} \geq 0.3$. Unlike Table 1, which first computes claim-level rates within each model–prompt run and then averages them equally over the five models, *Faith.* and *Unsup.* here are pooled claim-level micro-averages over all 2,000 explanations for each LLM prompt. The event verbalizer is an extractor-aligned no-API reference.

Failure mode	Run	Series	Problematic claim	Verifier diagnosis
Unsupported pattern	GPT-4.1-mini <i>Direct</i>	test_000003 <i>upward trend</i>	<i>“Around the 50th point, there is a sharp and significant drop... indicating a major change point or anomaly.”</i>	Unsupported: no anomaly or change-point event supports the claim.
Interval mislocalization	DeepSeek-V4-Flash <i>Event-grounded</i>	test_000002 <i>upward trend</i>	<i>“A moderate downward trend follows from index 20 to 60...”</i>	Partial: the phenomenon is plausible, but the span is shifted or too broad.
Causal hallucination	GPT-4.1-mini <i>CoT</i>	test_000032 <i>upward trend</i>	<i>“...abruptly interrupted by a significant drop, suggesting a possible external shock or regime change.”</i>	Unsupported: the explanation adds an external cause not present in the event evidence.

Table 18: Representative claim-level failure cases selected from the full synthetic verification outputs. The cases illustrate unsupported pattern introduction, interval mislocalization, and causal over-interpretation.

Model	Event list	Faith. ↑	Unsup. ↓	Part.	Claims/Expl.	Faith.+Rep. ↑	Ret. ↑	Sem. Pres. ↑	Cost/Expl. ↓
GPT-4.1-mini	Extracted events	0.706	0.082	0.212	9.71	0.992	0.881	1.000	\$0.00140
	Gold events	0.503	0.444	0.053	1.23	0.882	0.859	1.000	\$0.00033
DeepSeek-V4-Flash	Extracted events	0.746	0.077	0.176	8.88	0.992	0.899	1.000	\$0.00044
	Gold events	0.556	0.421	0.023	1.09	0.886	0.924	1.000	\$0.00015

Table 19: Generation-side event-list ablation on 400 synthetic series. Extracted-event rows reuse the submitted event-grounded outputs; gold-event rows rerun generation with sparse designed events. Faith. and Unsup. use the corresponding graph.

Metric	Anomaly	Change point	Downward trend	Seasonality	Sharp drop	Sharp rise	Upward trend	Volatility shift	All
N	50	50	50	50	50	50	50	50	400
Recovered	50 (1.00)	50 (1.00)	47 (0.94)	50 (1.00)	50 (1.00)	50 (1.00)	48 (0.96)	49 (0.98)	394 (0.985)
Mean best IoU	1.000	1.000	0.731	1.000	0.743	0.743	0.720	0.935	0.859

Table 20: Recovery of the single designed primary event by the frozen extractor. Recovery requires exact event type and interval IoU at least 0.5. Mean best IoU is reported for every designed event. This diagnostic estimates primary-event recall, not precision for additional extracted events.

Prompt	Claims	Gold-compatible	Ext. S	Gold S	Compat. S	Compat. S+P
Direct	20,532	3,270	0.493	0.130	0.748	0.983
CoT	53,955	5,783	0.360	0.083	0.764	0.987
Structured	13,633	2,220	0.353	0.106	0.640	0.984
Event-grounded	16,925	2,381	0.676	0.074	0.519	0.998

Table 21: Reference sensitivity for the fixed 8,000 generations. Counts pool five models; rates macro-average model-level proportions. Ext. S uses the extracted graph; Gold S checks all claims against sparse designed events. Compatible columns retain represented claim types. The sparse graph is not exhaustive truth.

This section provides supplementary empirical results, including bootstrap uncertainty estimates (Table 14), verifier calibration against human annotations (Table 15), full model-level results and coverage diagnostics (Tables 16 and 17), representative failure cases (Table 18), and analyses of event-list ablation, event-extractor recovery, and reference sensitivity (Tables 19–21).

D Human Audit Details

Annotation protocol. We conduct the human audit as a calibration study for claim-level time-series grounding. We first sample 180 claims from the full synthetic verification outputs, balanced by automatic verifier label: 60 SUPPORTED, 60 PARTIAL, and 60 UNSUPPORTED. Annotators receive a written guideline defining the three support labels, the error-type taxonomy, and practical boundary rules for interval shifts, magnitude overclaims, mixed claims, and causal statements. During annotation, they see the claim text, the plotted time series, extracted events, gold/reference events, and the full explanation for context, but they do not see the automatic verifier label or each other’s labels. After guideline calibration, two annotators independently label all 180 claims and agree on 156; Table 22 reports the annotator confusion matrix. We use these 156 agreement cases as consensus gold for verifier-alignment reporting; the 24 disagreements are excluded from verifier-alignment metrics and summarized separately. Table 23 reports the resulting label distributions. Of the 24 unresolved disagreements, 13 are SUPPORTED/PARTIAL, 8 are SUPPORTED/UNSUPPORTED, and 3 are PARTIAL/UNSUPPORTED. The automatic verifier labels 20 of these cases UNSUPPORTED, 3 SUPPORTED, and 1 PARTIAL, so unresolved annotation cases are concentrated in verifier-unsupported predictions.

Annotator A label	B supported	B partial	B unsupported
supported	123	6	2
partial	7	12	1
unsupported	6	2	21

Table 22: Human-human confusion matrix for the two calibrated annotators. Rows are Annotator A labels and columns are Annotator B labels.

Independent 90-claim audit. We sample 90 new claims with 10 claims in each prompt-by-verifier-label cell and hide both attributes during annotation. The same two annotators independently apply the

Source	N	Supported	Partial	Unsupported
Annotator A	180	131	20	29
Annotator B	180	136	20	24
Consensus gold	156	123	12	21
Verifier sample	180	60	60	60

Table 23: Label distributions in the calibrated 180-claim human audit. Consensus gold contains only the 156 claims on which the two annotators agree.

Reference	True label	Verifier supported	Verifier partial	Verifier unsupported
Consensus gold	supported	46	47	30
	partial	2	9	1
	unsupported	9	3	9
Annotator A	supported	48	48	35
	partial	3	9	8
	unsupported	9	3	17
Annotator B	supported	47	47	42
	partial	4	10	6
	unsupported	9	3	12

Table 24: Verifier-human confusion matrices grouped by reference label source. Rows are human or consensus labels, and columns are verifier labels. Consensus gold uses the 156 human-agreement cases; Annotator A and Annotator B use all 180 claims.

three-way taxonomy. They agree exactly on 76/90 claims (0.844; unweighted $\kappa = 0.572$); after merging PARTIAL and UNSUPPORTED, agreement is 0.878 ($\kappa = 0.642$). Against the 76 exact-consensus cases, verifier exact agreement is 26/76 (0.342), macro-F1 is 0.234, and $\kappa = 0.003$. Table 25 gives both confusion matrices. Humans confirm 23/26 verifier-SUPPORTED claims, while 42/50 verifier-consensus mismatches give a stricter verifier label to a human-SUPPORTED claim. Because sampling is balanced by prompt and verifier label, all per-label quantities are conditional calibration rather than corpus prevalence or accuracy.

Row label	Column: S	Column: P	Column: U
Annotator 1 vs. annotator 2			
S	65	6	2
P	2	6	2
U	1	1	5
Human consensus vs. verifier			
S	23	23	19
P	1	1	4
U	2	1	2

Table 25: Confusion matrices for the independent 90-claim audit. S, P, and U denote SUPPORTED, PARTIAL, and UNSUPPORTED. Human consensus is exact three-way agreement (76 cases). The sample contains 10 claims per prompt-by-verifier-label cell and therefore supports conditional calibration, not prevalence estimation.

E Verifier Calibration Details

Table 15 groups human consensus labels by the verifier’s three-way label. For finer-grained diagnostics, Table 26 groups non-supported verifier predictions by automatic error type, and Table 27 groups the same consensus-gold claims by verifier score. Because the parent 180-claim audit was intentionally balanced by verifier label, these tables should be read as conditional calibration diagnostics rather than estimates of natural corpus prevalence.

Verifier error type	N	Human sup.	Human part.	Human unsup.	Non-sup. accept
interval_mislocalization	56	0.786	0.161	0.054	0.214
pattern_hallucination	26	0.654	0.000	0.346	0.346
direction_error	10	1.000	0.000	0.000	0.000
pattern_confusion	7	0.857	0.143	0.000	0.143

Table 26: Human calibration of verifier non-supported error types on the consensus-gold subset.

Score bin	N	Mean score	Human sup.	Human part.	Human unsup.
[0.00, 0.25)	40	0.000	0.750	0.025	0.225
[0.25, 0.50)	1	0.480	1.000	0.000	0.000
[0.50, 0.75)	49	0.699	0.796	0.163	0.041
[0.75, 0.90)	16	0.806	0.812	0.125	0.062
[0.90, 1.00]	50	0.961	0.800	0.020	0.180

Table 27: Verifier score calibration on consensus-gold human-audit claims. Scores are produced by the rule verifier before human labels are used.

We also tested whether simple threshold tuning explains the human-verifier gap. With the fixed thresholds used in all main experiments, the verifier achieves 0.410 exact agreement, 0.353 macro-F1, $\kappa = 0.090$, and 0.333 non-supported F1 on the 156 consensus-gold claims. Five-fold cross-validation, selecting thresholds on four folds and evaluating on the held-out fold, yields 0.391 exact agreement, 0.335 macro-F1, $\kappa = 0.066$, and 0.316 non-supported F1. Even the best in-sample thresholds yield only 0.417 exact agreement and 0.357 macro-F1. Thus, the gap is not primarily a threshold-selection artifact.

For the independent two-annotator 90-claim synthetic audit, Table 28 summarizes the verifier-stored error codes for the 42 exact-consensus cases in which human consensus is SUPPORTED but the verifier assigns PARTIAL or UNSUPPORTED.

Two diagnostic cases. In CRV091, both annotators support the claim “No clear seasonality or repeating cyclical patterns are evident,” but the verifier labels it UNSUPPORTED. The parser records a seasonality claim without negative polarity, so evidence for absence is treated as a failed positive-event match. This is a directly traced parser lim-

Stored verifier code	Count
Interval mislocalization	23
Pattern confusion	11
Pattern hallucination	4
Direction error	4

Table 28: Verifier-stored error codes for the 42 exact-consensus mismatches in the independent two-annotator 90-claim synthetic audit where human consensus is SUPPORTED and the verifier assigns PARTIAL or UNSUPPORTED. These automatic codes are hypotheses about mechanism, not independently adjudicated causes.

itation and motivates polarity-aware verification rather than threshold tuning alone.

HP025 (test_000137, sharp-drop stratum) traces a different pipeline behavior. The extractor recovers the designed sharp drop at [43, 46] and also emits a full-range downward trend and a strong period-4 seasonality event. The event-grounded output verbalizes the additional events, and the shared verifier gives the period-4 claim score 1.0. Direct and CoT instead describe stable regimes separated by the abrupt drop, and the blinded comparison favors those baselines on this item. Because the annotators rate complete explanations, this is a graph-to-text propagation illustration, not claim-level adjudication of the additional events.

F Blinded Human Preference Study

Protocol. We sample 40 synthetic series, five from each of the eight phenomena, and generate all compared outputs with GPT-4.1-mini. Two annotators independently complete 80 randomized comparisons each: event-grounded versus Direct and event-grounded versus CoT for every series. Raters see the plotted series, original question, and two explanations; assign anchored 1–5 faithfulness and informativeness scores to each; and choose A, B, or Tie for more faithful, more informative, and overall preference. Prompt identities, extracted and gold graphs, verifier outputs, and the randomization key are hidden. Before randomization, we remove bracketed event IDs from event-grounded outputs and the fixed condition-identifying opener from CoT outputs; the remaining explanation text is unchanged.

For each comparator and rating dimension, we average the two annotators’ event-grounded-minus-baseline difference within each series. Confidence intervals use 5,000 nonparametric bootstrap resamples of the 40 series, and two-sided paired

Wilcoxon tests use those same 40 values with Holm correction over four rating tests. Overall-preference counts pool 80 judgments per comparator and are descriptive; the two blocks reuse the same 40 series. Table 7 reports the aggregate ratings in the main text, and Table 29 reports agreement.

Decoded preference	Exact agreement	κ
More faithful	0.888	0.310
More informative	0.850	0.477
Overall	0.900	0.427

Table 29: Inter-annotator agreement over 80 comparisons after decoding randomized A/B positions to {event-grounded, comparator, tie}. Direct and CoT are mapped to a common comparator label only for this pooled summary.

Same-sample rank reversal. On these 40 series, strict graph-supported fractions are 263/370 (0.711) for event-grounded, 127/220 (0.577) for Direct, and 426/1,199 (0.355) for CoT. The blinded ratings favor both baselines on both dimensions. This is not evidence of a faithfulness–informativeness trade-off; it shows that the operational ordering does not transfer to holistic human quality in this sample.

Period-4 propagation diagnostic. A period-4 event appears in 25/40 extracted graphs and in exactly the same 25 event-grounded explanations, compared with 0/40 Direct and 0/40 CoT explanations. All 25 occur in the change-point, upward/downward-trend, and sharp-rise/drop strata (5/5 each); the sampled seasonality stratum has 0/5. Table 30 shows that the human-rating gap is descriptively larger in the period-4 subgroup for both comparators and both dimensions. This split is post hoc, coincides with five sampling strata, and lacks claim-level adjudication; the aggregate gap in the 15-series no-P4 subgroup remains baseline-favoring. We therefore treat it as a candidate representation-linked propagation signal, not an event-precision estimate or causal explanation of the preference result.

G Real-NAB Human Audit Details

We additionally audit 90 Real-NAB event-grounded claims sampled evenly from the verifier’s SUPPORTED, PARTIAL, and UNSUPPORTED predictions. This is a single-annotator diagnostic of real-world label plausibility rather than a replacement for the two-annotator synthetic audit.

Comparison	Faith. Δ (P4)	Faith. Δ (no P4)
EG vs. Direct	−2.520	−0.967
EG vs. CoT	−2.520	−1.133
Comparison	Info. Δ (P4)	Info. Δ (no P4)
EG vs. Direct	−1.180	−0.433
EG vs. CoT	−1.740	−1.167

Table 30: Post hoc period-4 subgroup diagnostics. Δ denotes event-grounded minus baseline after averaging ratings across annotators. The P4 and no-P4 subgroups contain 25 and 15 series, respectively.

Table 31 summarizes the human-label distribution and verifier macro-F1 overall, by model, and by window type. Across all 90 claims, the annotator labels 36.7% as SUPPORTED, 27.8% as PARTIAL, and 35.6% as UNSUPPORTED, with verifier macro-F1 of 0.390.

Verifier calibration. The verifier exactly agrees with the human label on 35/90 claims (0.389; $\kappa = 0.083$). Because the audit contains 30 claims from each automatic verifier label, the per-label results are conditional calibration diagnostics rather than estimates of corpus prevalence. Table 32 reports the conditional human-label distributions, exact agreement, and binary supported-versus-non-supported gate consistency. Table 33 gives the corresponding full three-way confusion matrix.

Slice	N	Supp. (%)	Part. (%)	Unsup. (%)	Macro-F1
All audited claims	90	36.7	27.8	35.6	0.390
DeepSeek-V4-Flash	45	42.2	28.9	28.9	0.403
GPT-4.1-mini	45	31.1	26.7	42.2	0.367
Anomaly windows	45	31.1	31.1	37.8	0.355
Normal windows	45	42.2	24.4	33.3	0.418

Table 31: Single-annotator Real-NAB audit for 90 event-grounded claims. Supp., Part., and Unsup. are human-label percentages; Macro-F1 is reported on a 0–1 scale.

Metric	Supported	Partial	Unsupported
N	30	30	30
Human supp. (%)	33.3	36.7	40.0
Human part. (%)	33.3	36.7	13.3
Human unresp. (%)	33.3	26.7	46.7
Exact	0.333	0.367	0.467
Gate-consistent	0.333	0.633	0.600

Table 32: Real-NAB human-audit calibration by automatic verifier label. Columns correspond to verifier labels. The sample is intentionally balanced by verifier label, so these rates diagnose label reliability rather than corpus prevalence.

Human label	Verifier supported	Verifier partial	Verifier unsupported
Supported	10	11	12
Partial	10	11	4
Unsupported	10	8	14

Table 33: Three-way confusion matrix for the Real-NAB human audit. Rows are human labels and columns are automatic verifier labels.

H Additional Synthetic Diagnostics

This section contains two supplementary synthetic diagnostics tables. Table 34 reports pattern-by-prompt behavior before verifier-guided repair. Table 35 reports additional full synthetic metrics, including partial rates, post-repair claim counts, supported-claim preservation, total cost, and token use.

Pattern	Prompt	Faith.	Unsup.	Part.	Claims/Expl.
Upward trend	Direct	0.519	0.338	0.143	10.07
	CoT	0.374	0.528	0.098	28.11
	Structured	0.417	0.334	0.250	6.80
	Event-grounded	0.691	0.008	0.301	8.70
Downward trend	Direct	0.493	0.308	0.199	10.69
	CoT	0.382	0.475	0.143	26.84
	Structured	0.392	0.303	0.305	6.80
	Event-grounded	0.685	0.023	0.292	8.54
Sharp rise	Direct	0.552	0.364	0.085	10.38
	CoT	0.435	0.477	0.088	26.40
	Structured	0.381	0.393	0.226	6.34
	Event-grounded	0.714	0.047	0.240	8.96
Sharp drop	Direct	0.452	0.447	0.101	10.58
	CoT	0.381	0.537	0.082	25.30
	Structured	0.304	0.508	0.188	6.34
	Event-grounded	0.759	0.059	0.183	8.74
Volatility shift	Direct	0.332	0.529	0.139	9.80
	CoT	0.285	0.616	0.099	27.11
	Structured	0.215	0.612	0.173	7.38
	Event-grounded	0.558	0.254	0.187	8.40
Change point	Direct	0.513	0.417	0.070	10.94
	CoT	0.383	0.552	0.065	26.52
	Structured	0.392	0.431	0.177	6.80
	Event-grounded	0.719	0.063	0.218	7.92
Seasonality	Direct	0.385	0.590	0.025	10.53
	CoT	0.254	0.721	0.025	29.32
	Structured	0.343	0.616	0.041	7.35
	Event-grounded	0.658	0.214	0.128	10.19
Anomaly	Direct	0.410	0.535	0.054	9.14
	CoT	0.351	0.615	0.034	26.22
	Structured	0.345	0.567	0.088	6.72
	Event-grounded	0.636	0.235	0.129	6.26

Table 34: Full pattern-by-prompt diagnostics on the synthetic test split, pooled over the five full-test models. Metrics are computed before repair from claim-level verifier labels.

Model	Prompt	Part.	Claims/Expl.	+Rep.	Sem. Pres.	↑	Total Cost	↓	Tokens/Expl.
Claude Sonnet 4.5	Direct	0.057	3.86	0.557	\$2.6975	895.9			
	CoT	0.066	4.58	0.569	\$3.9652	1106.4			
	Structured	0.172	2.39	0.422	\$2.3380	820.8			
	Event-grounded	0.163	6.29	0.959	\$4.4150	2280.5			
DeepSeek-V4-Flash	Direct	0.166	3.83	0.999	\$0.2051	2102.4			
	CoT	0.111	7.44	0.974	\$0.3525	3416.7			
	Structured	0.169	2.95	1.000	\$0.2578	2564.7			
	Event-grounded	0.176	7.98	1.000	\$0.1757	2533.0			
Gemini-3-Flash	Direct	0.151	5.34	0.941	\$1.5148	2075.5			
	CoT	0.070	7.67	0.975	\$1.9057	2400.4			
	Structured	0.184	3.74	0.994	\$1.7646	2270.4			
	Event-grounded	0.266	6.83	1.000	\$2.2153	4152.8			
GPT-4.1-mini	Direct	0.078	3.89	1.000	\$0.1754	682.8			
	CoT	0.097	10.11	0.965	\$0.4206	1062.9			
	Structured	0.176	3.10	0.999	\$0.1990	707.6			
	Event-grounded	0.212	8.55	1.000	\$0.5582	2298.8			
GPT-5.4-mini	Direct	0.123	5.25	0.952	\$0.6001	786.6			
	CoT	0.063	7.73	0.869	\$1.0674	1043.0			
	Structured	0.195	3.24	1.000	\$0.6049	776.0			
	Event-grounded	0.240	6.42	1.000	\$1.0419	2163.1			

Table 35: Additional full synthetic diagnostics on 400 examples. Sem. Pres. denotes supported-claim preservation: an originally supported claim is counted as preserved when the repaired explanation contains a supported claim grounded by an overlapping support event. Chronology violations were negligible in this setting: all model-prompt rates were at most 0.002.

I Additional Multi-Provider Pilot Diagnostics

The retained 80-example pilot uses the same prompt-level aggregation as the main synthetic setting.

Prompt	Faith. ↑	Unsup. ↓	Claims/Expl.	Faith.+Rep. ↑	Unsup.+Rep. ↓	Ret. ↑	Cost/Expl. ↓
Direct	0.490	0.404	10.54	0.956	0.025	0.519	\$0.00260
CoT	0.367	0.551	26.69	0.948	0.022	0.306	\$0.00387
Structured	0.339	0.480	6.89	0.932	0.036	0.462	\$0.00259
Event-grounded	0.663	0.118	8.20	0.989	0.004	0.852	\$0.00417

Table 36: Prompt-level pilot results averaged over five LLMs on 80 stratified synthetic series. Faith. is the strict graph-supported fraction; Ret. is post-repair claim retention; Cost/Expl. is estimated API cost.

Table 36 reports the main prompt-level metrics, and Table 37 reports additional prompt-level diagnostics for partial support, post-repair claim volume, supported-claim preservation, cost, and token usage.

Prompt	Part.	Claims/Expl.	+Rep.	Sem.	Pres.	↑	Avg. Total Cost	↓	Tokens/Expl.
Direct	0.106			4.41		0.890		\$0.2081	1313.7
CoT	0.082			7.51		0.877		\$0.3093	1818.2
Structured	0.180			2.99		0.880		\$0.2074	1429.8
Event-grounded	0.219			6.99		0.998		\$0.3337	2646.8

Table 37: Additional prompt-level pilot diagnostics averaged over five LLMs. Sem. Pres. denotes supported-claim preservation: an originally supported claim is counted as preserved when the repaired explanation contains a supported claim grounded by an overlapping support event. Chronology violations were negligible: all pilot rates were at most 0.002. Avg. Total Cost is the mean cost of an 80-example model-prompt run.