Atomic and Holistic LLM Judges for Reference-Grounded Support Labels: A Prompt-Controlled Comparison

Anonymous ACL submission

Abstract

Atomic decomposition is valuable when an evaluation must produce claim inventories, claim-level precision/recall, or auditable evidence trails. We ask a narrower question that arises in reference-grounded QA evaluation: if the desired output is only a three-way support label, does a self-decomposing atomic judge outperform a matched holistic rubric? The *atomic* judge decomposes candidate answers into factual claims and verifies each claim. The *holistic* judge sees the same inputs and a similarly detailed rubric, but is not asked to decompose. We call this comparison *prompt-controlled*: the two judges share inputs, the underlying model, and similarly detailed instructions, but their realized generation length is not forced equal — we report token cost as part of the result rather than equalize it away.

We evaluate on 200 source examples per dataset on TruthfulQA, ASQA, and QAMPARI, for 400 pointwise rows per dataset and per judge. Judges are scored against benchmark-defined support labels, with a 60-example single-annotator sanity check clarifying that ASQA uses a stricter completeness standard than relaxed human factual-correctness judgments. Across Opus-4.6, GPT-4.1, and Gemini Flash Lite, we find no evidence that the self-decomposing atomic judge outperforms the matched holistic baseline once prompt richness and token cost are accounted for. TruthfulQA is the only dataset where atomic remains competitive and is sometimes slightly better. On ASQA and QAMPARI, the holistic judge is consistently stronger and cheaper.

Diagnostic schema, budget, grounding, and robustness ablations narrow the explanation space, but we treat them as suggestive rather than confirmatory. Overall, the results support a narrow conclusion: for single-call reference-grounded support labeling, atomic decomposition should earn its extra cost against a matched holistic baseline. The study should not be read as a test of multi-stage retrieval-grounded atomic systems.

1 Introduction

LLM-as-a-judge pipelines are increasingly used to evaluate open-ended model outputs, especially when human annotation is slow or expensive. This trend is appealing, but judge behavior can be unstable under prompt wording, output order, verbosity, and context perturbations—especially concerning for factuality evaluation, where a judge must separate correctness from fluency while resisting the temptation to trust its own parametric knowledge over supplied evidence.

One natural response is to make the judge more structured. Instead of asking for a single holistic verdict, we can require the model to decompose a candidate answer into atomic factual claims and verify them against a reference answer or supporting context. This strategy — popularized by FACTSCORE (Min et al., 2023) and instantiated in many downstream factuality pipelines (Wei et al., 2024; Liu et al., 2025; Wan et al., 2025) — is attractive because it appears to create an evidence bottleneck: the final decision must pass through explicit support judgments rather than a global impression of answer quality.

However, atomic judging comes with an immediate confound. In practice, atomic prompts are often longer, specify richer outputs, and encourage the model to spend more tokens on intermediate reasoning. If atomic judges outperform holistic ones, the gain may come from prompt richness or extra generation rather than from decomposition itself.

This paper studies that confound directly in a setting where practitioners need only an overall support label rather than a claim-level audit. We compare an atomic reference-grounded judge against a prompt-controlled holistic judge that receives the same inputs and a similarly detailed rubric, but is

not asked to decompose the answer into claims. The key contrast is therefore not atomic versus careless holistic judging, and not SAFE-style multi-stage factuality evaluation versus rubric judging, but a common single-call self-decomposing atomic prompt versus an equally detailed holistic rubric. We then use targeted schema, budget, grounding, and robustness ablations to ask why the two designs differ.

Our motivating hypothesis is deliberately strong:

> If a self-decomposing atomic prompt provides a structural advantage for reference-grounded support classification, then it should remain at least competitive when the judge sees the same inputs under similarly rich prompts, and any remaining advantage should survive direct accounting of realized token cost.

The full data do not support this hypothesis. We find no general atomic advantage: on one of three benchmarks, atomic and holistic judging are roughly tied; on the other two, the holistic judge is consistently stronger and cheaper. Among the perturbations we tested, reference-content perturbations move verdicts more than candidate-side perturbations (verbosity padding, distractor injection, pairwise order swap). We treat this as a finding about which perturbations changed verdicts, not as a definitive ranking of mechanisms.

The paper's scoped claim follows: for the support-label task studied here, the right baseline for a single-call atomic judge is not a minimal holistic prompt but a matched holistic rubric. On our completeness-heavy QA benchmarks, the atomic judge does not beat that baseline and usually costs more.

2 Related Work

Our work connects three strands of prior research.

First, claim-level factuality evaluation has been highly influential in generation research. FACTSCORE (Min et al., 2023) showed that long-form factuality can be evaluated more faithfully by decomposing generations into atomic claims and verifying them against external knowledge. SAFE (Wei et al., 2024) extends this with multi-stage decomposition and search-grounded per-claim verification. Subsequent factuality pipelines refine different pieces of that recipe. VeriFact (Liu et al., 2025) improves fact extraction and reference-fact construction for long responses. FaStFACT (Wan et al., 2025) emphasizes efficiency and stronger evidence collection for decomposition-based factuality evaluation. UFO (Huang et al., 2024) studies how evaluator behavior changes with the fact source that is provided. These works strongly motivate the appeal of atomic pipelines, but they target a different setting from ours: they evaluate *factual precision* of long-form generations against external evidence, often through multi-stage extract-then-verify architectures with retrieval. Our setting is single-call *reference-grounded support classification*, where the reference is already supplied and the question is whether the deployed self-decomposing single-prompt atomic design — not multi-stage SAFE-style pipelines, which are out of scope — beats a matched holistic support-label baseline.

Second, several works study reference-aware and rubric-based LLM judges. Reference-Guided Verdict (Badshah and Sajjad, 2025) shows that explicit references can improve free-form QA evaluation. YESciEval (D'Souza et al., 2025) studies robust LLM judging for scientific QA. These efforts motivate our focus on reference-grounded evaluation, but they do not isolate decomposition from prompt richness, output format, and realized compute.

Third, a growing literature shows that LLM judges are fragile. Frontier judges often generalize better than fine-tuned substitutes (Huang et al., 2025). Judge behavior varies across languages (Fu and Liu, 2025). Repeated runs can be self-inconsistent (Haldar and Hockenmaier, 2025). Statistical testing matters when arguing that judges can replace humans (Calderon et al., 2025). External evidence also does not automatically override model-internal knowledge (Qi et al., 2025). Closest to our robustness analysis, Lee et al. (2026) show that QA judges can fail when references conflict with what the model already "knows."

Our contribution is not a new judge architecture. It is a prompt-controlled empirical study of a popular design choice, together with diagnostic ablations that separate decomposition from schema effects, output budget, and reference quality.

3 Experimental Setup

3.1 Judge designs

We study three pointwise judge variants.

Atomic judge. The judge receives a question, a reference answer, and a candidate answer. It is instructed to break the candidate into atomic factual

claims, identify supported and unsupported claims, and return a structured verdict.

Prompt-controlled holistic judge. The judge sees the same inputs and scores the candidate with a detailed rubric covering correctness, completeness, unsupported detail, and resistance to style bias. It returns structured JSON, but is not asked to decompose. We explicitly tell the holistic judge *not* to decompose. Without that instruction, a holistic prompt sometimes spontaneously enumerates atomic claims, which would re-confound the comparison. We did not change prompts without rerunning.

Schema-matched judge. This ablation keeps atomic-style JSON fields while removing the explicit claim-decomposition instruction. It isolates whether any improvement comes from the decomposition bottleneck itself or from stronger formatting constraints.

For pairwise robustness experiments, we use a matched holistic pairwise judge and test whether verdicts change when candidate order is reversed.

3.2 Prompt control and cost accounting

The main comparison is controlled in four ways:

1. Within each comparison, atomic and holistic prompts use the same underlying judge model.
2. The prompts use the same question, candidate, and reference fields.
3. Both prompts request structured JSON outputs with similarly detailed evaluation instructions.
4. We measure realized token usage directly and report it alongside accuracy.

This setup is *prompt-controlled* rather than literally compute-matched. We do not force identical generation length; instead, we treat realized generation length as part of the empirical outcome. If one judge is only better by spending far more tokens, that should count against it rather than be hidden. This accounting is especially important on ASQA and QAMPARI, where atomic judging often generates substantially longer outputs.

3.3 Datasets and evaluation protocol

Task operationalization. The task is not generic factuality scoring; it is support classification against a supplied reference. Given a question, a reference answer, and a candidate answer, the judge classifies the candidate as `supported`, `partially_supported`, or `unsupported`. Omissions count: a candidate that is locally correct but omits information that the reference includes is `partially_supported`, not `supported`. The task therefore involves both factual accuracy and completeness relative to the reference. We score judges against benchmark-defined labels and cross-check those labels against a 60-example single-annotator sanity pass (Section 5.5); the pass matches the benchmark on TruthfulQA and QAMPARI, and reveals one systematic gap on ASQA, discussed there.

We evaluate on three datasets normalized into a shared JSONL format:

- **TruthfulQA** (Lin et al., 2022), a factuality stress test with supported and unsupported answers.
- **ASQA** (Stelmakh et al., 2022), an ambiguous open-domain QA benchmark where incomplete answers expose coverage and grounding failures.
- **QAMPARI** (Amouyal et al., 2023), a many-answer QA benchmark that stresses coverage rather than single-span correctness.

The primary evaluation uses 200 source examples per dataset. Each source example is converted into two pointwise rows:

- TruthfulQA: one supported answer and one unsupported answer.
- ASQA: one full answer and one benchmark-derived partial answer.
- QAMPARI: one full answer and one benchmark-derived partial answer.

This yields 400 pointwise rows per dataset in the main comparison, with the class composition shown in Table 1.

For ASQA and QAMPARI, the candidates are benchmark-derived partial answers rather than generations from current LLMs. The construction is deterministic. For ASQA, we take the first half of the benchmark's `qa_pairs` list and render each selected pair as "sub-question + first short answer," then concatenate those fragments into a partial long-form candidate. For QAMPARI, we take the first

Dataset	supp.	partial	unsupp.
TruthfulQA	200	—	200
ASQA	200	200	—
QAMPARI	200	200	—

Table 1: Pointwise class composition. Each dataset contributes 400 rows in the main comparison; not every benchmark contains every class. The completeness diagnosis is carried by the `partially_supported` rows in ASQA and QAMPARI.

Dataset	Model	Atomic output tokens	Holistic output tokens
TruthfulQA	Opus-4.6	141 [119, 165]	138 [104, 198]
TruthfulQA	GPT-4.1	106 [99, 116]	104 [89, 127]
ASQA	Opus-4.6	250 [207, 307]	150 [110, 219]
ASQA	GPT-4.1	182 [144, 233]	113 [104, 138]
QAMPARI	Opus-4.6	339 [264, 501]	156 [98, 201]
QAMPARI	GPT-4.1	269 [196, 403]	118 [95, 148]

Table 2: Realized output-token distributions for the expanded main runs, reported as median [Q1, Q3]. Gemini is omitted because the provider interface did not expose per-example output-token counts. Atomic judging becomes much more expensive on ASQA and especially on QAMPARI.

half of the canonical answer list and join the selected answers with commas. In both datasets, the partial candidates therefore preserve benchmark lexical content rather than the free-form style of an LLM generation. This makes the construction transparent and reproducible, but it may also introduce lexical regularity and a strong completeness signal. We therefore interpret the main claims as claims about this benchmark-derived partial-answer setting rather than about arbitrary LLM generations; we return to this in Section 9.

3.4 Models and metrics

Our main model set is Opus-4.6, GPT-4.1, and Gemini Flash Lite. The core claims in this paper are based on 200-source-example runs for all three families.

We report accuracy over all rows, average total tokens, and exact paired sign tests on the 200 source examples per dataset (we average each source's two rows before testing). We also report realized output-token distributions because the main question is partly about whether atomic structure provides benefits beyond the extra generation it tends to induce. Gemini is omitted from the output-token distribution table because the provider interface used in these runs does not expose per-example output-token counts. Unless otherwise noted, the smaller diagnostic ablations in later tables are reported for Opus-4.6 and GPT-4.1 only; Sonnet-4.6 appears only in the prompt-variant stability check (§4.3) as a within-family second model.

4 Main Evaluation

This section asks the central question: with both judges receiving equally rich instructions, does a single-call atomic judge improve support-label accuracy enough to justify its extra tokens? The short answer is dataset-specific. Atomic remains competitive on TruthfulQA. On ASQA and QAMPARI, holistic judging is stronger and cheaper.

4.1 Overall pattern

Table 3 reports the expanded main results across three datasets and three model families. Three patterns are immediately visible.

First, TruthfulQA is the only dataset where atomic judging remains preferred. The gap is tiny for Opus-4.6 and GPT-4.1, but Gemini Flash Lite shows a somewhat larger atomic edge. Even so, the TruthfulQA story is still much weaker than a universal atomic advantage claim, because the gain is dataset-specific and does not transfer to ASQA or QAMPARI.

Second, ASQA shows the opposite pattern, and the gap widens rather than shrinks after expansion. The holistic judge is substantially stronger for the two proprietary frontier models we paired-test (Opus-4.6 and GPT-4.1; Gemini's accuracy is also higher under holistic but its output-token distribution is unavailable, so we exclude it from this paired comparison). On GPT-4.1, holistic improves accuracy by 0.155 while using about 65 fewer total tokens per example. On Opus-4.6, the gain is 0.0725 accuracy with about 77 fewer tokens.

Third, QAMPARI is the clearest case where the self-decomposing atomic judge fails to beat the matched holistic baseline. The holistic judge is both better and dramatically cheaper, and this result becomes easier to defend statistically on the expanded evaluation. The output-token distributions in Table 2 make the cost gap especially clear: atomic judging often expands into very long outputs, while the holistic judge remains compact.

4.2 Paired comparisons

Table 4 reports exact paired sign tests on the expanded runs. These tests sharpen the qualitative reading of Table 3.

Dataset	Method	Opus-4.6		GPT-4.1		Gemini Flash Lite	
		Acc.	Tokens	Acc.	Tokens	Acc.	Tokens
TruthfulQA	Atomic	0.9750	336.40	0.9775	284.52	0.9625	295.27
TruthfulQA	Holistic	0.9700	364.66	0.9675	292.44	0.9300	309.33
ASQA	Atomic	0.7775	569.92	0.5650	475.19	0.5500	504.52
ASQA	Holistic	0.8500	492.81	0.7200	410.12	0.6975	423.74
QAMPARI	Atomic	0.9750	803.47	0.9275	676.45	0.9525	599.20
QAMPARI	Holistic	0.9925	510.68	0.9975	420.06	0.9925	423.29

Table 3: Expanded main pointwise results on 200 source examples per dataset (400 pointwise rows after candidate construction). Accuracy is measured over all rows. Tokens are average total tokens per example.

Dataset	Model	ΔAcc. (H-A)	ΔTokens (H-A)	p
TruthfulQA	Opus-4.6	-0.0050	+28.27	0.6250
TruthfulQA	GPT-4.1	-0.0100	+7.92	0.1250
TruthfulQA	Gemini Flash Lite	-0.0325	+14.06	2.44e-04
ASQA	Opus-4.6	+0.0725	-77.11	2.98e-08
ASQA	GPT-4.1	+0.1550	-65.07	6.39e-15
ASQA	Gemini Flash Lite	+0.1475	-80.78	2.45e-12
QAMPARI	Opus-4.6	+0.0175	-292.79	0.0156
QAMPARI	GPT-4.1	+0.0700	-256.39	7.45e-09
QAMPARI	Gemini Flash Lite	+0.0400	-175.91	4.02e-04

Table 4: Exact paired sign tests between prompt-controlled holistic (H) and atomic (A) judges on the expanded main runs. Positive ΔAcc. means holistic is more accurate; negative ΔTokens means holistic is cheaper.

Dataset	Model	ΔAcc. 95% CI	ΔTok. 95% CI	Stab. 50%
TruthfulQA	Opus-4.6	[-0.015, +0.005]	[+24.7, +31.9]	0.64
TruthfulQA	GPT-4.1	[-0.020, -0.003]	[+5.8, +10.0]	0.87
ASQA	Opus-4.6	[+0.048, +0.100]	[-89.0, -65.8]	1.00
ASQA	GPT-4.1	[+0.118, +0.193]	[-73.5, -57.2]	1.00
QAMPARI	Opus-4.6	[+0.005, +0.033]	[-334, -253]	0.97
QAMPARI	GPT-4.1	[+0.048, +0.095]	[-294, -221]	1.00

Table 5: Bootstrap 95% CIs (10,000 iterations) and subsample stability (fraction of 1,000 draws at 50% sample size that preserve the sign of the accuracy effect).

TruthfulQA gives the only atomic-favoring results in the paper, but even there the story is mixed: Opus-4.6 and GPT-4.1 show only very small, statistically weak atomic edges, while Gemini Flash Lite shows a clearer but still dataset-specific atomic preference. In contrast, ASQA strongly favors the holistic judge for all three model families. QAMPARI also favors the holistic judge across all three families, with the largest practical gains coming from cost rather than raw accuracy.

To strengthen uncertainty quantification beyond the sign test, Table 5 reports bootstrap 95% confidence intervals (10,000 iterations) and 50%-subsample sign stability (1,000 draws at 50% of the data, fraction preserving the sign of the accuracy effect) for Opus-4.6 and GPT-4.1. ASQA and QAMPARI holistic advantages are extremely stable: the 95% CI for the accuracy delta excludes zero, and 100% of 50%-subsamples preserve the sign. TruthfulQA is different: Opus-4.6 is a near-tie with a CI spanning zero, while GPT-4.1 shows a small atomic-favoring effect whose magnitude remains about one percentage point.

Per-class breakdowns localize the effect to incompleteness detection. On ASQA `partially_supported` rows, holistic improves over atomic by +14.5, +30.5, and +33.0 percentage points for Opus-4.6, GPT-4.1, and Gemini Flash Lite, respectively; on QAMPARI the corresponding gains are +3.5, +13.0, and +9.0 points. By contrast, TruthfulQA's small atomic edge comes from `unsupported` rows (-1.0, -2.0, and -6.0 points for holistic-minus-atomic). The holistic gains are therefore not generic accuracy improvements; they are concentrated in detecting incomplete but locally correct answers.

Taken together, the main evaluation rejects the strongest version of the hypothesis. Atomic decomposition is not a better support-labeling baseline in this setting. TruthfulQA is the only dataset where atomic remains favored, and even there the effect is small for the two strongest families. On ASQA and QAMPARI, the holistic judge is clearly preferable across all three model families, and the bootstrap CIs and subsample stability checks support that conclusion.

4.3 Prompt variant stability

A natural concern with any controlled comparison is that the results might depend on the specific wording of one prompt pair rather than on the structural difference between atomic and holistic judging. To address this, we created two independently written alternative prompt templates for each method family, yielding three atomic and three holistic variants. Each variant preserves the core design principle (claim decomposition for atomic, rubric-based evaluation for holistic) while varying the framing, instruction order, and emphasis.

Dataset	Model	Variant	Accuracy	Tokens
ASQA	Opus-4.6	Atomic v1	0.78	519.9
ASQA	Opus-4.6	Atomic v2	0.78	615.9
ASQA	Opus-4.6	Atomic v3	0.78	643.7
ASQA	Opus-4.6	Holistic v1	0.88	479.5
ASQA	Opus-4.6	Holistic v2	0.94	562.9
ASQA	Opus-4.6	Holistic v3	0.94	568.8
ASQA	Sonnet-4.6	Atomic v2	0.76	627.1
ASQA	Sonnet-4.6	Atomic v3	0.72	646.8
ASQA	Sonnet-4.6	Holistic v2	0.88	574.8
ASQA	Sonnet-4.6	Holistic v3	0.88	583.9
TruthfulQA	Opus-4.6	Atomic v1	0.950	345.2
TruthfulQA	Opus-4.6	Atomic v2	0.975	436.8
TruthfulQA	Opus-4.6	Atomic v3	0.975	463.9
TruthfulQA	Opus-4.6	Holistic v1	0.975	374.8
TruthfulQA	Opus-4.6	Holistic v2	0.975	434.7
TruthfulQA	Opus-4.6	Holistic v3	0.975	452.0

Table 6: Prompt variant stability on diagnostic slices. Three independently worded prompts per method family. On ASQA, all holistic variants outperform all atomic variants for both model families. On TruthfulQA, all variants cluster at 0.95–0.975.

Table 6 reports results for Opus-4.6 and Sonnet-4.6 on ASQA, and Opus-4.6 on TruthfulQA.

The results are encouraging for the paper's scoped claim. On ASQA, every holistic variant outperforms every atomic variant for both Opus-4.6 and Sonnet-4.6, confirming that the holistic advantage is a property of the method family rather than of one specific prompt or one specific model. On TruthfulQA, all six variants cluster tightly at 0.950–0.975, again confirming the near-tie. The main cost penalty for the alternative prompts is higher token usage (the v2 and v3 templates are wordier), but the sign of the accuracy gap is fully stable.

5 Diagnostic Mechanism Analyses

This section asks why the atomic and holistic judges differ. We rule out three obvious explanations — output format, output budget, and reference-side knobs — using small targeted ablations. The conclusion is that none of these alone explains the gap; reference quality is the input the judges are most sensitive to, but we treat that as a perturbation-sensitivity finding rather than a causal claim. These analyses use smaller slices than the main evaluation because each ablation multiplies the number of model calls. They focus on Opus-4.6 and GPT-4.1; Gemini was added to the main evaluation but not to every ablation grid.

5.1 Schema control

The schema ablation asks whether atomic gains, where they exist, come mainly from stricter output formatting. Table 7 suggests that this explanation is too weak. On TruthfulQA, the schema-matched judge can match atomic accuracy, but it is more expensive. On ASQA, it modestly improves over atomic while still falling short of the holistic judge. This weakens the claim that atomic benefits are primarily a JSON-format artifact.

Dataset	Model	Accuracy	Avg. Total Tokens
TruthfulQA	Opus-4.6	0.950	424.85
TruthfulQA	GPT-4.1	0.975	345.78
ASQA	Opus-4.6	0.800	567.48
ASQA	GPT-4.1	0.620	477.68

Table 7: Schema ablation on targeted diagnostic slices: atomic-style JSON without the explicit claim-decomposition instruction. $N = 40$ rows per dataset (TruthfulQA, ASQA), Opus-4.6 and GPT-4.1 only. Accuracy is row-level accuracy against benchmark labels.

Dataset	Method	Budget	Opus-4.6	GPT-4.1
TruthfulQA	Atomic	128	0.975	0.975
TruthfulQA	Atomic	256	0.950	1.000
TruthfulQA	Atomic	512	0.950	0.975
TruthfulQA	Holistic	128	0.975	0.975
TruthfulQA	Holistic	256	0.975	0.975
TruthfulQA	Holistic	512	0.975	0.975
ASQA	Atomic	128	0.760	0.580
ASQA	Atomic	256	0.760	0.600
ASQA	Atomic	512	0.760	0.600
ASQA	Holistic	128	0.880	0.660
ASQA	Holistic	256	0.880	0.680
ASQA	Holistic	512	0.860	0.700

Table 8: Budget ablation on targeted diagnostic slices. $N = 40$ rows per dataset, three budget settings, Opus-4.6 and GPT-4.1. Cells report row-level accuracy. Increasing output budget does not reverse the ASQA ranking.

5.2 Budget control

The budget ablation asks whether atomic judging simply needs more room to write out its reasoning. Table 8 does not support that explanation. On ASQA, increasing the explicit output budget does not rescue atomic judging for either top model family. On TruthfulQA, the ranking stays nearly flat across budget settings. This is consistent with the expanded main results: the ASQA gap is not just a low-budget failure.

5.3 Grounding sensitivity

The grounding ablation asks whether the judges really rely on the supplied reference. Table 9 shows a sharp split between datasets. TruthfulQA is almost unaffected when the reference is reduced to

Dataset	Method / Model	Identity	Reduced Ref	No Ref
TruthfulQA	Atomic / Opus-4.6	0.950	0.950	0.625
TruthfulQA	Atomic / GPT-4.1	0.975	0.975	0.750
TruthfulQA	Holistic / Opus-4.6	0.975	0.975	0.550
TruthfulQA	Holistic / GPT-4.1	0.975	0.975	0.700
ASQA	Atomic / Opus-4.6	0.740	0.500	0.380
ASQA	Atomic / GPT-4.1	0.540	0.340	0.260
ASQA	Holistic / Opus-4.6	0.860	0.620	0.380
ASQA	Holistic / GPT-4.1	0.700	0.500	0.480

Table 9: Grounding ablation on targeted diagnostic slices. $N = 50$ unique source examples per dataset; perturbations are identity (full reference), reduced reference (first-sentence paraphrase), and removed reference. Cells report row-level accuracy under each perturbation. ASQA is much more sensitive than TruthfulQA to degraded references.

Judge	Full Acc.	Partial Acc.	Overall Acc.
Atomic	0.34	0.54	0.44
Holistic	0.30	0.65	0.48

Table 10: Judge accuracy on LLM-generated ASQA candidates from two answering LLMs (100 questions, 200 rows). Full = complete generation, Partial = truncated to first third.

its first sentence, although performance drops once the reference is removed entirely. ASQA is much more grounding-sensitive: even reduced references cause large accuracy losses for both judge families.

These diagnostics point toward a clearer mechanism story, but they do not prove it conclusively. The datasets appear to differ less in whether decomposition is intrinsically useful and more in how much the task depends on a complete, reliable, and non-conflicting reference.

5.4 External validity: LLM-generated candidates

The main evaluation uses benchmark-derived partial answers rather than realistic LLM generations. To check whether the core pattern transfers, we generated answers to 100 ASQA questions from two answering LLMs (Opus-4.6 and Sonnet-4.6, temperature 0.7). For each generated answer, we created a pointwise pair: the full LLM-generated answer (labeled `supported`) and a truncated version retaining the first third of sentences (labeled `partially_supported`), yielding 200 evaluation rows. Table 10 reports the results.

The holistic judge is overall ahead (0.48 vs. 0.44) and is substantially better on the partial slice (0.65 vs. 0.54). The full-slice accuracy is low for both judges (0.34 / 0.30), with both judges frequently labeling the full LLM-generated answers as `partially_supported` rather than `supported`. Inspection suggests that this is not a judge failure but a label-construction issue: ASQA reference answers often require multi-perspective coverage, so an LLM-generated answer focused on one aspect can be genuinely partial relative to the reference even when our construction labeled it "full." We therefore treat this section as a *noisy external-validity stress check* rather than as direct evidence that the main pattern transfers cleanly. The partial-slice signal is the more interpretable one and is consistent with the main evaluation; the full-slice numbers should be read with the labeling caveat in mind.

Dataset	N	Agreement	Agreement %
TruthfulQA	20	20	100.0
QAMPARI	20	20	100.0
ASQA	20	11	55.0
Overall	60	51	85.0

Table 11: Human–benchmark label agreement on a stratified 60-example, single-annotator sanity check. The annotator's labels match the benchmark on TruthfulQA and QAMPARI; on ASQA, all 9 disagreements follow one direction: the benchmark labels a candidate `partially_supported` while the annotator labels it `supported`.

5.5 Human sanity check

To assess whether the benchmark-defined support labels align with human factuality judgments, an author annotated 60 examples: 20 per dataset, stratified across label categories (supported, partially_supported, unsupported) and including both full and partial candidates. Each example was independently classified as supported, partially_supported, or unsupported before comparing with the benchmark label. Table 11 reports the results.

On TruthfulQA and QAMPARI, the annotator's labels match the benchmark on every example checked. ASQA is the exception. All 9 disagreements follow a single systematic pattern: the benchmark assigns `partially_supported` to candidates that the annotator considers `supported`. These are cases where the candidate provides a correct answer to the question but omits some information present in the multi-perspective reference. For example, a candidate that correctly answers "When was the 13th amendment ratified?" with "December 6, 1865" is labeled `partially_supported` by

the benchmark because the reference also covers the passage date and ratification process, but the annotator considers this a supported answer.

This pattern has two implications. First, the ASQA benchmark applies a stricter completeness standard than a single human annotator, which means the absolute accuracy numbers on ASQA should be interpreted as performance against a demanding completeness criterion rather than against intuitive factual correctness. Second, and more importantly for the paper's scoped claim, the relative ranking between holistic and atomic judges is unaffected: the holistic judge is better at matching these strict completeness labels regardless of whether one considers the benchmark or the annotator as the ground truth. If anything, the disagreement pattern strengthens the holistic advantage interpretation, because the errors the holistic judge avoids (over-rating partial answers as supported) are exactly the cases where the benchmark is stricter than the annotator.

We note the obvious limitation that this is a single-annotator sanity check rather than a multi-annotator study with inter-annotator agreement. It is intended to validate the benchmark labels, not to replace them.

6 Robustness Analyses

Appendix A gives the full robustness tables. The main takeaway is simple: among the input perturbations we test, reference-content perturbations move verdicts substantially, while candidate-side perturbations such as verbosity padding, distractor injection, and pairwise order have only minor effects. We therefore treat reference reliability as the most salient sensitivity of this protocol, not as a proven causal explanation for all atomic–holistic differences.

7 Discussion

The complete results support a much narrower conclusion than the paper's initial motivating intuition. There is no universal atomic advantage in this support-label task: TruthfulQA is the only dataset that still favors atomic judging, and even there the strongest effect comes from Gemini Flash Lite rather than the two stronger model families. ASQA and QAMPARI instead favor prompt-controlled holistic judging, often with lower token cost. The diagnostic analyses do not point to output structure or output budget as the main explanation; among the perturbations we tested, reference-content perturbations produced the largest verdict changes, compatible with UFO (Huang et al., 2024) and the reference-conflict findings of Lee et al. (2026).

This result is useful precisely because it is conditional. Atomic decomposition remains the natural tool when the desired output is a claim inventory, a precision/recall decomposition, or an auditable chain of per-claim decisions. Our experiment asks a different deployment question: if a practitioner already has a reference answer and only needs a three-way support label, should the default judge spend tokens decomposing the candidate? On completeness-heavy QA benchmarks, our answer is no. The main failure mode is not that the holistic judge reasons more deeply; it is that the self-decomposing atomic judge can verify locally correct subclaims and still miss that the candidate covers only part of the reference.

The practical implication is therefore modest but concrete. Reference-grounded evaluation pipelines should first check whether the reference is complete, non-conflicting, and aligned with the label definition. If the task is support classification rather than claim-level factuality accounting, a matched holistic rubric is a strong baseline that should be reported before adopting a more expensive atomic bottleneck. Atomic structure may still be preferable for long-form factuality systems, but its value should be tied to the artifact it produces—claim-level evidence—rather than assumed from decomposition alone.

8 Conclusion

We revisited a common assumption in LLM judging — that explicit atomic decomposition should improve reference-grounded factuality evaluation — but only for the support-label version of that question. The data do not show that a single-call self-decomposing atomic judge is a better default than a matched holistic rubric. TruthfulQA is the only dataset where atomic remains favored. On ASQA and QAMPARI, the prompt-controlled holistic judge is consistently stronger and cheaper, with gains concentrated in `partially_supported` cases. The resulting recommendation is narrow: for reference-grounded support classification, report a well-matched holistic rubric before paying the cost of decomposition; use decomposition when the desired artifact is claim-level evidence, precision/recall, or an audit trail.

9 Limitations

The limits of the study all point to the same boundary. This paper is a controlled comparison of single-call, reference-grounded support classification, not a general verdict on every atomic factuality pipeline. The 200-source samples and smaller diagnostic slices trade breadth for paired control: they are large enough to expose the within-example atomic–holistic contrast, but they should not be read as a deployment-wide benchmark of all tasks, judges, or model scales.

The benchmark construction also makes incompleteness unusually explicit. ASQA and QAMPARI `partially_supported` candidates are benchmark-derived partial answers, created by truncating answer lists, rather than naturally sampled incomplete generations. This gives transparent labels for the comparison, but it emphasizes a particular form of missing information. The LLM-generated ASQA stress check and the 60-example human sanity check are included to test whether that label definition behaves sensibly: they support the same interpretation, while also showing that ASQA accuracy should be read as accuracy against a strict reference-completeness standard.

The main boundary is therefore about the output a practitioner wants. Atomic decomposition remains valuable when the evaluation product is a claim inventory, an auditable evidence trail, or a precision/recall accounting over claims; multi-stage systems such as SAFE (Wei et al., 2024) and FaStFACT (Wan et al., 2025) pursue that broader goal. Our result addresses a narrower use case: when the desired output is a three-way support label against a supplied reference, decomposition should earn its extra cost against a matched holistic rubric rather than be assumed to be the better baseline.

References

Samuel Amouyal, Tomer Wolfson, Ohad Rubin, Ori Yoran, Jonathan Herzig, and Jonathan Berant. 2023. QAMPARI: A benchmark for open-domain questions with many answers. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 97–110, Singapore. Association for Computational Linguistics.

Sher Badshah and Hassan Sajjad. 2025. Reference-guided verdict: LLMs-as-judges in automatic evaluation of free-form QA. In *Proceedings of the 9th Widening NLP Workshop*, pages 251–267, Suzhou, China. Association for Computational Linguistics.

Nitay Calderon, Roi Reichart, and Rotem Dror. 2025. The alternative annotator test for llm-as-a-judge: How to statistically justify replacing human annotators with llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16051–16081, Vienna, Austria. Association for Computational Linguistics.

Jennifer D'Souza, Hamed Babaei Giglou, and Quentin Munch. 2025. Yescieval: Robust llm-as-a-judge for scientific question answering. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13749–13783, Vienna, Austria. Association for Computational Linguistics.

Xiyan Fu and Wei Liu. 2025. How reliable is multilingual llm-as-a-judge? In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11040–11053, Suzhou, China. Association for Computational Linguistics.

Rajarshi Haldar and Julia Hockenmaier. 2025. Rating roulette: Self-inconsistency in llm-as-a-judge frameworks. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, Suzhou, China. Association for Computational Linguistics.

Hui Huang, Xingyuan Bu, Hongli Zhou, Yingqi Qu, Jing Liu, Muyun Yang, Bing Xu, and Tiejun Zhao. 2025. An empirical study of llm-as-a-judge for llm evaluation: Fine-tuned judge model is not a general substitute for gpt-4. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5880–5895, Vienna, Austria. Association for Computational Linguistics.

Zhaoheng Huang, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2024. Ufo: a unified and flexible framework for evaluating factuality of large language models. *arXiv preprint arXiv:2402.14690*.

Dongryeol Lee, Yerin Hwang, Taegwan Kang, Minwoo Lee, Younhyung Chae, and Kyomin Jung. 2026. Judging against the reference: Uncovering knowledge-driven failures in llm-judges on qa evaluation. *arXiv preprint arXiv:2601.07506*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Xin Liu, Lechen Zhang, Sheza Munir, Yiyang Gu, and Lu Wang. 2025. Verifact: Enhancing long-form factuality evaluation with refined fact extraction and reference facts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17908–17925, Suzhou, China. 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*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Siya Qi, Rui Cao, Yulan He, and Zheng Yuan. 2025. Evaluating llms' assessment of mixed-context hallucination through the lens of summarization. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 16480–16503, Vienna, Austria. Association for Computational Linguistics.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. ASQA: Factoid questions meet long-form answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8273–8288, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yingjia Wan, Haochen Tan, Xiao Zhu, Xinyu Zhou, Zhiwei Li, Qingsong Lv, Changxuan Sun, Jiaqi Zeng, Yi Xu, Jianqiao Lu, Yinhong Liu, and Zhijiang Guo. 2025. Fastfact: Faster, stronger long-form factuality evaluations in llms. *arXiv preprint arXiv:2510.12839*.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V Le. 2024. Long-form factuality in large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 80756–80827.

A Robustness Details

We test three pointwise perturbations on targeted slices: verbosity padding, reference distractors, and swapped references. Table 12 shows that, among the tested perturbations, swapped references move verdicts the most. TruthfulQA is relatively stable. ASQA is not. Both atomic and holistic judges become highly unstable when the reference is replaced with one from another example, suggesting that within this perturbation set the reference content carries most of the verdict signal.

Dataset	Method / Model	Consistency	Swap Ref	Distractor	Verbosity
TruthfulQA	Atomic / Opus-4.6	0.950	0.050	0.025	0.025
TruthfulQA	Atomic / GPT-4.1	0.975	0.000	0.025	0.025
TruthfulQA	Holistic / Opus-4.6	0.975	0.025	0.000	0.000
TruthfulQA	Holistic / GPT-4.1	0.925	0.050	0.025	0.000
ASQA	Atomic / Opus-4.6	0.500	0.500	0.040	0.020
ASQA	Atomic / GPT-4.1	0.520	0.480	0.040	0.060
ASQA	Holistic / Opus-4.6	0.500	0.500	0.080	0.040
ASQA	Holistic / GPT-4.1	0.620	0.340	0.060	0.100

Table 12: Pointwise robustness on targeted diagnostic slices ($N = 50$ source examples per dataset, four perturbations). *Consistency* = accuracy on the unperturbed (identity) condition. *Swap Ref / Distractor / Verbosity* = perturbation-induced verdict flip rate relative to the identity condition. Reference swap is by far the most damaging perturbation on ASQA.

Pairwise order bias is much less severe than reference perturbation in the current study. TruthfulQA and QAMPARI are perfectly stable under candidate swap for the evaluated strong models. ASQA is nearly as stable: GPT-4.1 has zero flips, while Opus-4.6 shows a small but non-zero flip rate.

Dataset	Model	Position Flip Rate	Accuracy All Rows
TruthfulQA	Opus-4.6	0.00	1.00
TruthfulQA	GPT-4.1	0.00	1.00
ASQA	Opus-4.6	0.04	0.94
ASQA	GPT-4.1	0.00	1.00
QAMPARI	Opus-4.6	0.00	1.00
QAMPARI	GPT-4.1	0.00	1.00

Table 13: Pairwise order robustness on targeted diagnostic slices ($N = 25$ source examples per dataset, candidate-order swap). *Position Flip* = fraction of source examples whose verdict changes when candidate order is reversed.

B Qualitative Examples

Table 14 gives representative paired examples from the expanded runs. The most common ASQA and QAMPARI failure is that atomic judging over-rewards local correctness and misses incompleteness, whereas the holistic judge more often maps incompleteness to the intended `partially_supported` label. TruthfulQA provides a contrasting case where the atomic judge can be more conservative when the reference explicitly marks the question as ambiguous.

C Judge Prompt Templates

This appendix provides the exact system prompts and user templates for all judge variants used in the paper.

C.1 Atomic judge (v1, main)

System prompt.

> You are a careful factuality judge. Break the candidate answer into atomic factual claims, ground each claim in the reference answer or context, and then produce a structured verdict. Ignore verbosity, tone, or writing style unless they change factual content.

User template.

> `Question:` {question}
>
> `Reference answer:` {reference_answer}
>
> `Candidate answer:` {candidate_answer}
>
> Return valid JSON with keys: `verdict` (one of `supported`, `partially_supported`, or `unsupported`), `score` (integer 1–5), `atomic_claims` (list of strings), `supported_claims` (list of strings), `unsupported_claims` (list of strings), `confidence` (number between 0 and 1), `rationale` (short string).

C.2 Holistic judge (v1, main)

System prompt.

> You are a careful factuality judge. Evaluate the candidate answer against the reference answer or context with a detailed rubric, but do not explicitly decompose the answer into atomic claims. Ignore style or verbosity unless they change factual content.

User template.

> `Question:` {question}
>
> `Reference answer:` {reference_answer}
>
> `Candidate answer:` {candidate_answer}
>
> Judge holistically on: correctness, completeness, unsupported detail, robustness to style bias.
>
> Return valid JSON with keys: `verdict` (one of `supported`, `partially_supported`, or `unsupported`), `score` (integer 1–5), `evidence_summary` (short string), `unsupported_or_missing_points` (list of strings), `confidence` (number between 0 and 1), `rationale` (short string).

Dataset	Question	Candidate excerpt	Gold	Atomic	Holistic	Note
ASQA	Who played Zordon in the original Power Rangers?	"Earlier episodes … David Fielding"	partial	supp.	partial	Holistic preserves the incompleteness judgment; atomic treats the locally correct span as fully supported.
ASQA	When did the US enter the Second World War in Europe?	"start funding/supplying … 11 March 1941"	partial	supp.	partial	Atomic accepts the sub-fact; holistic notices the main question is not fully answered.
QAMPARI	Who won a Coke Zero Sugar 400?	Partial list of winners from the reference set	partial	supp.	partial	Atomic verifies each item independently and misses missing winners; holistic tracks coverage.
QAMPARI	Which team played in a competition won by Collingwood FC?	Partial list of eligible teams	partial	supp.	partial	Atomic rewards subset correctness; holistic better tracks coverage.
TruthfulQA	Is it legal to possess cannabis?	"Yes, it is legal to possess cannabis"	unsupp.	unsupp.	partial	Atomic is more conservative and better matches the intended label.

Table 14: Representative paired examples from the expanded runs. The ASQA and QAMPARI examples illustrate the dominant error pattern: atomic judging upgrades incomplete but locally correct answers to `supported`. Supp. = supported; unsupp. = unsupported.

C.3 Atomic judge v2

System prompt.

You are a meticulous fact-checking assistant. Your task is to identify every key factual assertion in the candidate answer, verify each assertion against the reference answer, and then aggregate your findings into a structured JSON verdict. Focus exclusively on factual accuracy and completeness; disregard differences in phrasing, tone, or length.

User template.

[Same input fields.] Perform the following analysis: (1) Identify every key factual assertion in the candidate answer. (2) For each assertion, determine whether it is supported by the reference answer. (3) Aggregate your findings into an overall verdict. [Same JSON output schema as v1.]

C.4 Atomic judge v3

System prompt.

You are a rigorous claim-verification judge. Follow a strict step-by-step protocol: first list all discrete claims made in the candidate answer, then check each claim for support in the reference answer, and finally render an overall verdict with a confidence score. Ignore stylistic differences and evaluate only factual substance.

User template.

[Same input fields.] Follow these steps strictly: Step 1: List every discrete claim made in the candidate answer. Step 2: For each claim, check whether the reference answer supports, contradicts, or is silent on it. Step 3: Based on your verification, decide on an overall verdict and score. [Same JSON output schema as v1.]

C.5 Holistic judge v2

System prompt.

You are an expert answer-quality evaluator. Assess the candidate answer by first analyzing how well it covers the information in the reference answer, then checking for factual accuracy, and finally assigning an overall quality score. Do not decompose the answer into atomic claims. Disregard style and verbosity unless they introduce factual errors.

User template.

[Same input fields.] Evaluate the candidate answer holistically in three stages: (1) Coverage: How much of the reference information does the candidate include? (2) Accuracy: Are the included facts correct relative to the reference? (3) Final score: Combine coverage and accuracy into an overall assessment. [Same JSON output schema as v1 holistic.]

C.6 Holistic judge v3

System prompt.

You are a comparative factuality judge. Your job is to compare the candidate answer directly against the reference answer, evaluating factual overlap, missing information, and any unsupported additions. Produce a single holistic judgement without breaking the answer into atomic claims. Ignore differences in writing style or length.

User template.

[Same input fields.] Compare the candidate answer against the reference answer on these dimensions: factual overlap (what correct information do both share?), missing information (what does the reference contain that the candidate omits?), unsupported additions (what does the candidate assert that the reference does not support?). Synthesize your comparison into an overall verdict. [Same JSON output schema as v1 holistic.]

D Human Annotation Details

Table 15 lists representative examples from the 60-example human sanity check, focusing on the 9 ASQA disagreement cases where the benchmark label is `partially_supported` but the annotator labeled `supported`.

E Dataset Construction Details

All three datasets are preprocessed into a shared pointwise JSONL format. Each row contains:

- `id`: Unique identifier encoding dataset, source example, and variant (e.g., `truthfulqa::8::pos`).
- `question`: The input question string.
- `task_type`: Always `free_form_qa` in this study.
- `reference_answer`: The benchmark-provided reference text.
- `candidate_answer`: The answer to be evaluated.
- `gold_label`: One of `supported`, `partially_supported`, or `unsupported`.
- `metadata`: Dataset-specific fields (e.g., `num_qa_pairs` for ASQA, `num_answers` for QAMPARI, `category` for TruthfulQA).

TruthfulQA. Each source example produces two rows: one with a correct answer (`supported`, variant `pos`) and one with a common misconception answer (`unsupported`, variant `neg`). The reference answer is the same for both rows.

ASQA. Each source example produces two rows: a `full` variant containing the complete reference text as the candidate (labeled `supported`), and a `partial` variant constructed by taking the first half of the benchmark's `qa_pairs` list and rendering each pair as "sub-question + short answer" (labeled `partially_supported`).

QAMPARI. Each source example produces two rows: a `full` variant with the complete answer list as the candidate (labeled `supported`), and a `partial` variant containing the first half of the canonical answer list (labeled `partially_supported`).

F Reproducibility Details

Decoding. Main judge runs use greedy decoding (temperature 0). The main experiments use a maximum output budget of 256 tokens; the budget ablation varies this cap over 128, 256, and 512 tokens.

Model versions. The main experiments use `claude-opus-4-6-20260301`, `gpt-4.1-2026-02-27`, and `gemini-3.1-flash-lite-001`. API calls for the main result set and diagnostic ablations were made in March 2026. The paper's claim is about judge design rather than model ranking.

Cost. Total API cost for the broader experiment pool, including main runs, ablations, robustness checks, prompt variants, external-validity checks, and runs not tabulated in the current main results, was approximately $350. QAMPARI dominated cost because atomic prompts often produced long outputs.

Question	Candidate excerpt	Gold	Human	Pattern
When was the 13th amendment ratified by the states?	"December 6, 1865"	partial	supp.	Correct date; ref. also covers passage date and ratification process
Who played Zordon in the original Power Rangers?	"David Fielding"	partial	supp.	Correct for early episodes; ref. also covers later voice actor and 1995 film
When is episode 113 of Dragon Ball Super coming out?	"Oct 29, 2017 (original); Jun 1, 2019 (US)"	partial	supp.	Both key dates in Q&A form; benchmark expects full prose
Which political party is currently in power in Ireland?	"Fine Gael–Labour (2011–16); Fine Gael (2016–20)"	partial	supp.	Key parties named; ref. expects fuller coalition history
Japanese hotel run by same family for 1300 years?	"Hōshi Ryokan"	partial	supp.	Correct hotel; ref. also covers Nishiyama Onsen Keiunkan
Who killed the man in Thelma and Louise?	"Louise Elizabeth Sawyer / Susan Sarandon"	partial	supp.	Correct character and actor; benchmark expects full narrative
The creation of the senate was based on the?	"Roman Senate"	partial	supp.	Correct basis; ref. also covers House of Lords and apportionment
When is Fortnite Battle Royale being released on Android?	"Aug 9, 2018 (Samsung exclusive)"	partial	supp.	Correct initial date; ref. covers full rollout timeline
Present chairman of PAC of Lok Sabha?	"Mallikarjun Kharge"	partial	supp.	Correct for 2017; ref. covers multiple years

Table 15: All 9 ASQA disagreement cases from the human sanity check. In every case, the candidate provides a factually correct but incomplete answer. The benchmark labels it `partially_supported` (partial) due to omitted reference information, while the annotator considers it `supported` (supp.). Ref. = reference.