

RSAT: Structured Attribution Makes Small Language Models Faithful Table Reasoners

Jugal Gajjar*, Kamalaskankari Subramaniakuppasamy

Department of Computer Science,
The George Washington University, USA
{jugal.gajjar, kamalaskankaris}@gwu.edu

Abstract

When a language model answers a table question, users have no way to verify which cells informed which reasoning steps. We introduce **RSAT**, a method that trains small language models (SLMs, 1–8B) to produce step-by-step reasoning with cell-level citations grounded in table evidence. Phase 1 (SFT) teaches a structured JSON output format from verified reasoning traces. Phase 2 (GRPO) optimizes a composite reward centered on NLI-based faithfulness, alongside citation validity and parsimony. Across six models from two families—Qwen 2.5 (1.5B/3B/7B) and Llama 3 (1B/3B/8B)—RSAT improves faithfulness $3.7\times$ over SFT alone (0.224→0.826), with near-perfect citation validity (0.992). Post-hoc attribution collapses below 13% format success, confirming that attribution must be integrated into reasoning, not retrofitted. Ablations show the faithfulness reward is essential: removing it drops faithfulness from 0.97 to 0.03.

1 Introduction

When a language model answers a question over a table, users receive an answer with no way to verify *which cells informed which reasoning steps*. This lack of interpretable, cell-level attribution limits trust in table reasoning systems, particularly in high-stakes domains such as financial analysis, journalism, and clinical decision support. While existing methods achieve strong accuracy—TAPAS (Herzig et al., 2020) through cell selection, PASTA (Gu et al., 2022) through operation-aware pre-training, chain-of-thought prompting (Wei et al., 2022) through step-by-step reasoning—none produce structured reasoning traces where every claim is explicitly grounded in specific table cells. The dominant alternative, *post-hoc attribution*, where a model first generates an answer and then retroactively cites evidence, is appealing in principle but,

as we demonstrate, collapses catastrophically on sub-8B models.

We propose **RSAT** (Reinforcement-driven Structured Attribution Training), a method that trains small language models (1–8B parameters) to produce structured JSON output where each reasoning step explicitly cites the table cells it depends on using [row, col] coordinates. RSAT operates in two phases. **Phase 1 (SFT)** fine-tunes on 1,000 verified reasoning traces to teach the structured output format—achieving near-perfect format compliance but only $\sim 22\%$ faithfulness. **Phase 2 (GRPO)** applies group-relative reinforcement learning (Shao et al., 2024) with a composite reward that incentivizes answer correctness, citation validity, *faithfulness*, and parsimony. The faithfulness signal—NLI entailment between cited cell values and reasoning text—is the key innovation: it teaches the model to *ground* its reasoning in actual table evidence rather than producing structurally valid but unsubstantiated citations. GRPO makes this feasible on a single GPU by eliminating the critic model required by PPO, enabling composite reward optimization over non-differentiable signals like NLI scores.

We evaluate RSAT across six models from two architecture families—Qwen 2.5 (1.5B/3B/7B) and Llama 3 (1B/3B/8B)—on three table reasoning benchmarks (WTQ, FeTaQA, TabFact). Our experiments yield four key findings:

1. RSAT improves faithfulness by $3.7\times$ over SFT alone (0.224→0.826 average across all models), while maintaining near-perfect citation validity (0.992) and format compliance (0.993).
2. Post-hoc attribution collapses entirely on some models (0.4% format success for Qwen 3B), averaging under 13% across all six—demonstrating that attribution must be integrated into reasoning, not retrofitted.
3. SFT and GRPO serve distinct, complementary roles: SFT teaches *structure* (+0.61 format,

*Corresponding author.

+0.63 citation validity), while GRPO teaches *quality* (+0.60 faithfulness). Neither alone is sufficient.

4. Ablations reveal faithfulness reward is the sole essential signal—removing it collapses faithfulness from 0.97 to 0.03, while removing other components causes only modest degradation.

The complete code and data for RSAT are available at <https://github.com/JugalGajjar/RSAT>.

2 Related Work

Table Question Answering. Table QA has progressed from semantic parsing (Pasupat and Liang, 2015; Zhong et al., 2017) to specialized pre-trained architectures: TAPAS (Herzig et al., 2020) extends BERT (Devlin et al., 2019) with row and column positional embeddings, TAPEX (Liu et al., 2021) pre-trains by mimicking an SQL executor, and TaBERT (Yin et al., 2020) jointly pre-trains on text-table pairs. More recent work leverages LLM prompting—Binder (Cheng et al., 2022) maps questions to executable SQL or Python, Chain-of-Table (Wang et al., 2024) iteratively transforms tables within the reasoning chain, and TaPERA (Zhao et al., 2024) decomposes questions into sub-questions answered by executable programs. Despite strong accuracy gains, none of these methods produce step-by-step reasoning with explicit cell-level attribution.

Table Fact Verification. TabFact (Chen et al., 2019) established table-based fact verification as a binary classification task. PASTA (Gu et al., 2022) designs operation-aware cloze pre-training to teach table operations, achieving strong accuracy with DeBERTaV3. Other approaches use program-guided reasoning (Yang et al., 2020) or joint verification and retrieval (Schlichtkrull et al., 2021; Eisenschlos et al., 2020). All output binary entailed/refuted labels without explaining *which cells* support the verdict.

Attribution and Faithfulness in LLMs. Rashkin et al. (2023) formalize attributable generation, defining metrics for whether outputs are supported by cited sources. Self-RAG (Asai et al., 2023) trains models to generate self-reflection tokens assessing retrieval relevance and faithfulness during generation. ALCE (Gao et al., 2023b) benchmarks citation generation for long-form text QA. RARR (Gao et al., 2023a) retrofits attributions post-hoc. Bohnet et al. (2022) study attributed

QA with retrieval-augmented approaches. Wallat et al. (2024) argue that correctness alone is insufficient—citations must also be faithful to the model’s actual reasoning, not post-rationalized. All these works operate on unstructured text; RSAT extends attribution to *structured table evidence* with cell-level granularity.

RL for LLM Reasoning. RLHF (Ouyang et al., 2022) aligns models with human preferences via learned reward models, while DPO (Rafailov et al., 2023) removes the reward model entirely. DeepSeek-R1 (Guo et al., 2025) demonstrates that GRPO (Shao et al., 2024)—which normalizes rewards within candidate groups, eliminating the critic model—can incentivize chain-of-thought reasoning. Dang and Ngo (2025) show GRPO enhances mathematical reasoning in 1.5B models with modest compute. We design a domain-specific composite reward for table attribution quality—combining NLI-based faithfulness, citation validity, and parsimony—and optimize it with GRPO. To our knowledge, RSAT is the first application of RL-driven attribution training to structured table reasoning.

Positioning. Existing table QA methods optimize accuracy without interpretable attribution. Fact verification outputs binary labels without reasoning traces. Text-domain attribution methods like Self-RAG operate on passage-level evidence, not cell-level structure. RSAT uniquely combines structured reasoning, cell-level citations, NLI-based faithfulness, and RL-driven attribution training in small (1–8B) models.

3 Methodology

3.1 Task Formulation

Given a table T with headers $\{h_1, \dots, h_C\}$ and rows $\{r_1, \dots, r_R\}$, and a natural language question q , the model must produce a structured JSON output containing: (1) a list of reasoning_steps, where each step is a natural language claim paired with cited_cells—a list of $[\text{row}_i, \text{col}_j]$ coordinates referencing specific table cells that support the claim; and (2) a final answer string. Cell coordinates are zero-indexed: $[0, 0]$ denotes the first data cell (row 0, column 0), excluding the header row. For example, a reasoning step might be: {"step": "Team A had 15 wins", "cited_cells": $[[2, 3], [0, 3]]$ }, where cells $[2, 3]$ and $[0, 3]$ contain the relevant values from the wins column. This

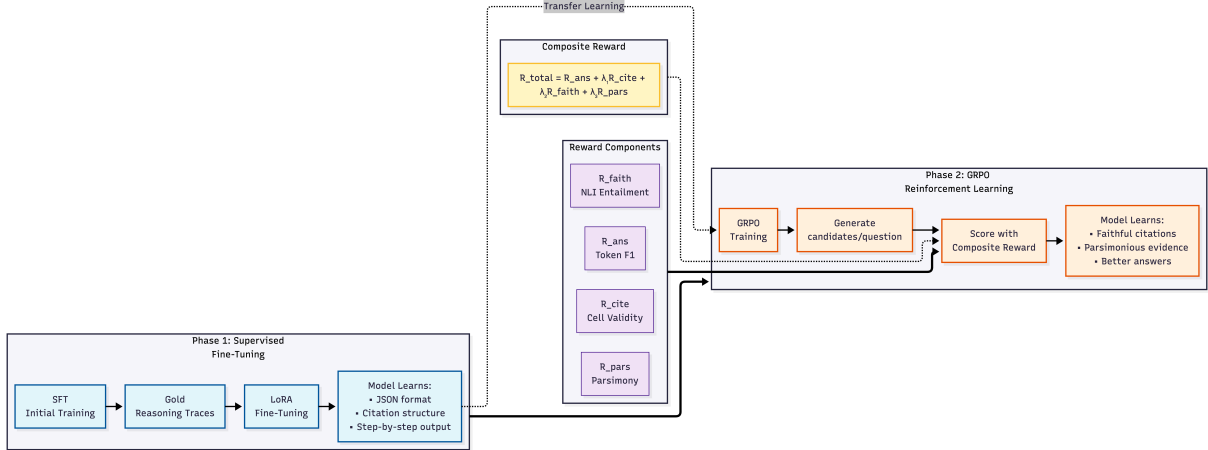


Figure 1: **RSAT overview.** Phase 1 (SFT) teaches the structured output format using verified gold traces. Phase 2 (GRPO) generates multiple candidates per question, scores them with a composite reward, and updates the policy. The faithfulness reward is the critical signal that grounds reasoning in actual table evidence.

Table 1: **Dataset statistics.** SFT examples include teacher-generated reasoning traces; GRPO examples contain only table-question-answer triples.

Split	WTQ	FeTaQA	TabFact	Total
SFT train	630	135	135	900
SFT val	70	15	15	100
GRPO train				38,647
GRPO val	(combined pool)			16,624
GRPO test				19,126
Total corpus				75,397

format enables post-hoc *auditing*: a user can independently verify each claim by checking the cited cells against the original table.

Tables are serialized into a flat text format that preserves structural information: [HEADER] col1 | col2 | ... [ROW 0] val1 | val2 | ... [ROW 1] The row index markers allow the model to learn a direct mapping between cell references in its output and positions in the serialized input.

3.2 Data Construction

We draw from three table reasoning benchmarks: WikiTableQuestions (WTQ) (Pasupat and Liang, 2015) for factoid QA, FeTaQA (Nan et al., 2022) for free-form long-answer QA, and TabFact (Chen et al., 2019) for fact verification. Table 1 summarizes the full corpus.

SFT data (1,000 examples). For each example, we prompt Claude Opus 4.5 to generate a structured reasoning trace given the serialized table and question. Each trace is programmatically verified

for (i) JSON validity, (ii) cell coordinate bounds, and (iii) step count (3–4 required). Failed traces are re-generated in batched repair passes with error-specific prompts. Notably, the verification pipeline enforces structural correctness (valid JSON, valid cell coordinates, and step count) but does not guarantee semantic grounding between cited cells and reasoning steps. This explains why SFT achieves near-perfect format compliance but relatively low faithfulness ($\sim 22\%$): the model learns to imitate structure without necessarily grounding its reasoning in the cited evidence. The average verified trace contains 3.2 steps with 2.4 cited cells per step. The set is split 900/100 (train/val), stratified by source.

GRPO data (74,397 examples). A larger pool of table-question-answer triples (no reasoning traces) is drawn from the same benchmarks. The model generates its own candidates during RL; only gold answers are needed for reward computation. We subsample 500 training and 500 test examples per model to keep compute tractable.

3.3 Phase 1: Supervised Fine-Tuning

The SFT phase teaches the model the structured output format. We apply LoRA (Hu et al., 2022) adapters to all linear projection layers (Q, K, V, O, gate, up, down) and fine-tune for 3 epochs with a cosine learning rate schedule.

SFT achieves $\sim 99\%$ format success and $\sim 99\%$ citation validity across all models—but only $\sim 22\%$ faithfulness on average. This gap is the central motivation for Phase 2: the model learns *what* to produce (valid JSON with cell references) but not *how well* to ground its reasoning in the cited evidence.

3.4 Phase 2: Group Relative Policy Optimization

To improve attribution quality beyond supervised learning, we apply GRPO (Shao et al., 2024; Guo et al., 2025). We prefer GRPO over PPO (Schulman et al., 2017) because it eliminates the critic model, reducing memory overhead by roughly half—critical for single-GPU training. We prefer it over DPO (Rafailov et al., 2023) because our multi-component reward requires scoring individual outputs, not ranking pairs.

For each training question, the model generates $G=8$ candidate, where each candidate is scored by the composite reward function (Section 3.5). Advantages are computed as group-relative z-scores—each candidate’s reward minus the group mean, divided by the group standard deviation. The policy is updated via the clipped surrogate objective. We attach a fresh LoRA adapter to the merged SFT checkpoint and train for one epoch over 500 subsampled training examples (250 optimization steps).

3.5 Composite Reward Function

The composite reward balances four objectives plus a hard format gate:

$$R = R_{\text{ans}} + \lambda_1 R_{\text{cite}} + \lambda_2 R_{\text{faith}} + \lambda_3 R_{\text{pars}} + R_{\text{fmt}} \quad (1)$$

Answer reward (R_{ans}): Token-level F1 between the predicted and gold answer. When multiple gold answers exist, we take the maximum.

Citation validity (R_{cite} , $\lambda_1=0.3$): Fraction of cited cell coordinates $[r_i, c_j]$ that fall within the table’s dimensions ($0 \leq r_i < R$, $0 \leq c_j < C$). Measures structural correctness of citations.

Faithfulness (R_{faith} , $\lambda_2=0.5$): For each reasoning step, we concatenate the values of the cited cells into an evidence string and compute the entailment probability against the step’s text using DeBERTa-v3-base NLI (He et al., 2020). The score is averaged across all steps; steps citing no cells receive a faithfulness score of 0. This is the core attribution signal—it measures whether the cited cells actually *support* the claim, not merely whether the citations are structurally valid.

Parsimony (R_{pars} , $\lambda_3=0.2$): Penalizes over-citation. Steps citing ≤ 3 cells receive full score (1.0); the score decays linearly to 0.0 at ≥ 8 cells. This prevents reward hacking through exhaustive citation of all cells.

Format penalty (R_{fmt}): Acts as a hard gate: -1 if the output fails JSON parsing, 0 otherwise. When triggered, this penalty dominates all other components, ensuring the model maintains the structured format learned during SFT.

The weighting hierarchy reflects our design priorities: faithfulness is the primary objective, citation validity is a structural prerequisite, and parsimony is a refinement. We set weights based on preliminary sensitivity analysis on Qwen 7B, tracking faithfulness, answer F1, and training stability across configurations. We found that faithfulness was most sensitive to λ_2 , while λ_1 and λ_3 primarily influenced structural correctness and verbosity. The final configuration ($\lambda_2 > \lambda_1 > \lambda_3$) reflects this hierarchy, prioritizing grounding quality while maintaining valid and concise citations. Results were stable under ± 0.1 perturbations. Full mathematical definitions are in Appendix C.

4 Experimental Setup

Models. We evaluate two model families across three scales: **Qwen 2.5 Instruct** (1.5B, 3B, 7B) (Yang et al., 2025) and **Llama 3 Instruct** (1B, 3B, 8B) (Grattafiori et al., 2024). All models are trained with LoRA (Hu et al., 2022) in bf16 precision on a single NVIDIA H100 80GB GPU. Total compute is approximately 18.4 GPU-hours for the six main models, plus 18.3 GPU-hours for ablation runs—36.8 GPU-hours in total (Appendix H).

Baselines. We compare four methods applied to each model: (1) **Zero-shot**: the base instruction-tuned model prompted with the RSAT system prompt, no fine-tuning; (2) **SFT-only**: Phase 1 only (no GRPO); (3) **Post-hoc**: a two-pass baseline where the model first generates a chain-of-thought answer, then is prompted in a second pass to map its reasoning to cell coordinates in JSON format (using the same system prompt as RSAT to ensure a fair comparison); (4) **RSAT**: the full two-phase pipeline.

Evaluation. We evaluate on 500 held-out examples from the combined WTQ/FeTaQA/TabFact test split using greedy decoding. We report six metrics: **Faithfulness** (NLI entailment between cited cells and reasoning text; *primary metric*), **Answer F1** (token-level overlap with gold answer), **Citation Validity** (fraction of cited cells within table bounds), **Parsimony** (penalizes over-citation; see Section 3.5), **Format Success** (fraction of outputs

parsing as valid JSON), and **Answer EM** (exact string match; reported for completeness but expected to be low due to answer paraphrasing in table QA).

Training Details. **SFT:** LoRA rank 32–64, learning rate 2×10^{-4} , 3 epochs, effective batch size 16. **GRPO:** LoRA rank 16–32, learning rate 5×10^{-5} , 1 epoch over 500 examples, $G=8$, temperature 0.9. Faithfulness scoring uses DeBERTa-v3-base NLI (cross-encoder/nli-deberta-v3-base) during both training and evaluation, ensuring metric consistency. Implementation uses TRL v0.29.0. Full hyperparameters are in Appendix B.

Robustness of Faithfulness Metric. Faithfulness is both the training reward and the primary evaluation metric, computed by the same DeBERTa-v3 NLI model. This introduces a potential train–evaluation circularity. Two observations partially address this concern. First, ablation results show that removing the faithfulness reward causes catastrophic collapse ($0.97 \rightarrow 0.03$) while leaving other metrics largely intact—a model purely gaming the proxy would not require the faithfulness signal so critically. Second, faithfulness improvements are accompanied by consistent answer F1 gains (+0.09 average), indicating that grounding improvements transfer to task performance. A human evaluation study remains the necessary next step to fully validate the NLI proxy.

5 Results and Analysis

5.1 Main Results

Table 2 presents results across all six models and four methods. RSAT achieves the best performance on every metric for every model. The most striking result is the faithfulness improvement: averaged across all six models, RSAT achieves 0.826 faithfulness compared to 0.224 for SFT-only—a $3.7\times$ improvement. This confirms that the GRPO composite reward successfully teaches models to ground reasoning in actual table evidence.

Answer F1 also improves consistently across all models (+0.09 average over SFT), confirming that faithfulness training does not sacrifice answer quality. Citation validity and format success remain near-perfect (>0.97), indicating that GRPO preserves the structural gains from SFT.

The more modest gains in answer F1 relative to faithfulness reflect a difference in objectives rather than a limitation. RSAT targets verifiability—

whether reasoning is auditably grounded in table evidence—rather than raw accuracy. In high-stakes domains such as financial analysis or clinical decision support, this distinction matters: a correct answer that cannot be traced to specific evidence offers users no basis for trust or audit.

Phase contribution. Table 3 quantifies what each training phase contributes on average across all six models. SFT produces the structural leap: format success jumps +0.61, citation validity +0.64. However, SFT yields only +0.19 average faithfulness—the model learns to produce well-formed JSON with valid cell references, but cited cells do not reliably support the reasoning claims. GRPO closes this gap, contributing +0.60 faithfulness while preserving structural quality established by SFT.

5.2 Scaling Analysis

Figure 2 presents faithfulness and answer F1 across model sizes for both families. Two patterns emerge.

First, **Qwen consistently outperforms Llama at equivalent scale on faithfulness:** 0.847 vs. 0.480 at $\sim 1.5\text{B}$, 0.946 vs. 0.735 at 3B, converging at 7–8B (0.977 vs. 0.972). This gap may reflect differences in pre-training data composition or Qwen’s larger vocabulary (151K vs. 128K tokens), which provides more expressive capacity for structured JSON output.

Second, **faithfulness scales with model size but with diminishing returns.** Qwen approaches ceiling (> 0.94) at 3B, while Llama requires 8B to reach equivalent performance. This suggests a practical finding: for resource-constrained deployment, Qwen 3B offers the best faithfulness-per-parameter ratio.

One anomaly: Llama 3B shows lower parsimony (0.865) than both Llama 1B (0.967) and 8B (1.000), suggesting that mid-scale models may generate redundant reasoning steps that smaller and larger models avoid.

5.3 Post-hoc Attribution Collapse

The post-hoc baseline performs worse than every other method across all models—frequently worse than zero-shot. Average format success is just 12.7%, compared to 99.3% for RSAT (Figure 3). The failure is particularly dramatic for Qwen 3B (0.4%) and Llama 8B (4.0%). The two-pass approach asks the model to retroactively map free-form reasoning to specific cell coordinates—a task

Table 2: **Main results across six models and four methods.** Best values per model are **bolded**. RSAT achieves the highest scores on every metric for every model. Faithfulness shows the largest improvement: $3.7\times$ over SFT on average. EM is omitted from the main table due to uniformly low values (0.000–0.018) caused by strict string matching; F1 captures partial correctness.

Model	Method	F1 \uparrow	Cite \uparrow	Faith \uparrow	Pars \uparrow	Fmt% \uparrow
Qwen 1.5B	Zero-Shot	0.209	0.391	0.040	0.389	0.436
	SFT	0.371	0.995	0.149	0.918	0.998
	Post-Hoc	0.115	0.182	0.032	0.209	0.244
	RSAT	0.524	0.996	0.847	0.990	0.998
Qwen 3B	Zero-Shot	0.181	0.296	0.040	0.364	0.398
	SFT	0.531	0.996	0.213	0.848	0.998
	Post-Hoc	0.002	0.000	0.000	0.004	0.004
	RSAT	0.592	0.999	0.946	0.996	1.000
Qwen 7B	Zero-Shot	0.277	0.565	0.086	0.501	0.566
	SFT	0.576	1.000	0.234	0.888	1.000
	Post-Hoc	0.088	0.140	0.027	0.117	0.140
	RSAT	0.619	0.992	0.977	0.992	0.992
Llama 1B	Zero-Shot	0.100	0.326	0.008	0.280	0.334
	SFT	0.403	0.947	0.192	0.787	0.952
	Post-Hoc	0.066	0.064	0.004	0.184	0.198
	RSAT	0.537	0.972	0.480	0.967	0.972
Llama 3B	Zero-Shot	0.099	0.368	0.015	0.336	0.368
	SFT	0.546	0.984	0.269	0.822	0.984
	Post-Hoc	0.035	0.056	0.005	0.085	0.090
	RSAT	0.592	0.993	0.735	0.865	0.994
Llama 8B	Zero-Shot	0.054	0.164	0.006	0.142	0.164
	SFT	0.555	0.996	0.288	0.830	0.998
	Post-Hoc	0.013	0.040	0.006	0.038	0.040
	RSAT	0.647	1.000	0.972	1.000	1.000

Table 3: **Phase contribution analysis.** Average per-metric gain from each training stage across all six models. SFT drives structure; GRPO drives faithfulness.

Transition	Δ Fmt	Δ Cite	Δ Faith	Δ F1
Zero-shot \rightarrow SFT	+0.61	+0.64	+0.19	+0.34
SFT \rightarrow RSAT	+0.00	+0.01	+0.60	+0.09

requiring re-reading the table, identifying which cells correspond to which claims, and formatting the result as valid JSON. Small models lack the working memory for this mapping. In contrast, RSAT models produce citations *during* reasoning, making attribution a natural part of the generation process.

Importantly, the dominant failure mode is not incorrect attribution but the failure to produce structured output at all. As shown in Appendix J, 76–100% of post-hoc outputs are empty or non-JSON, indicating that models consistently fail to comply with the required format in the second pass. This suggests a fundamental capacity limitation rather than a weak baseline design: small models lack the working memory to re-read the table and retroac-

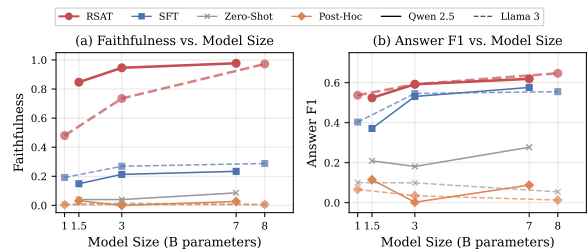


Figure 2: **Scaling curves for faithfulness and answer F1.** RSAT (solid, high) dominates at every scale. Qwen outperforms Llama at small scale; the gap narrows at 7–8B. Post-hoc collapses regardless of scale.

tively map free-form reasoning into structured citations. All methods use the same system prompt and base model, ensuring that the only difference is whether attribution is integrated during generation or applied post-hoc.

5.4 Ablation Study

Table 4 shows the effect of removing individual reward components on Qwen 7B and Llama 8B.

Faithfulness reward is critical. Removing it causes faithfulness to plummet from 0.977 to 0.117 (Qwen 7B) and 0.972 to 0.031 (Llama 8B). No-

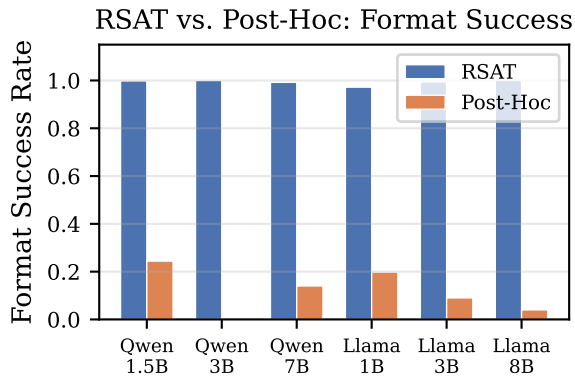


Figure 3: **Post-hoc attribution collapses across all models.** RSAT maintains >97% format success; post-hoc averages 12.7% (dashed line).

Table 4: **Ablation study.** Removing faithfulness reward causes catastrophic degradation (−86% to −97%). Removing parsimony causes significant over-citation. Citation reward has the smallest individual effect, as SFT already handles structural validity.

Model	Variant	F1	Faith	Pars
Qwen 7B	RSAT (full)	0.619	0.977	0.992
	−Faithfulness	0.635	0.117	1.000
	−Parsimony	0.612	0.952	0.604
	−Citation	0.605	0.934	0.993
Llama 8B	RSAT (full)	0.647	0.972	1.000
	−Faithfulness	0.638	0.031	0.996
	−Parsimony	0.626	0.899	0.575
	−Citation	0.617	0.938	1.000

tably, answer F1 remains comparable or slightly improves, while parsimony and citation stay high. Without the faithfulness signal, the model learns to produce perfectly formatted, parsimonious outputs that cite real cells—but the citations are essentially *random*. The model games the remaining rewards without grounding its reasoning. Interestingly, F1 slightly *increases* without the faithfulness signal (0.619→0.635 for Qwen 7B), revealing a mild tension: the model produces marginally better answers when freed from the constraint of justifying them with evidence.

Parsimony reward prevents over-citation. Without it, parsimony drops by ~40% (0.992→0.604 for Qwen, 1.000→0.575 for Llama), and models default to citing 5–6 cells per step rather than 1–2. Faithfulness also degrades as over-citing dilutes evidence quality.

Citation reward provides refinement. Removing it causes only a ~3–4% faithfulness drop, because SFT already teaches valid cell coordinate production. The citation reward serves as a safety net,

Real example: fetaga_test_168 (Qwen 7B)

Q: What span of days was the 2019 Eric Clapton World Tour in Tokyo for?

Gold: Tokyo on 13 April 2019 through 20 April 2019.

SFT (valid JSON, over-citing 15 cells):

```
step: "Rows 1-5, five dates 13-18 April 2019 at Nippon Budokan."
cited_cells: [[1,0],[1,1],[1,3],[2,0],[2,1],[2,3],[3,0],[3,1],[3,3],[4,0],[4,1],[4,3],[5,0],[5,1],[5,3]] # 15 cells
answer: "13-18 April 2019"
```

RSAT (valid JSON, 1 cell per step):

```
step: "Row 1." cited_cells: [[1, 1]]
step: "Row 1." cited_cells: [[1, 0]]
step: "Row 5." cited_cells: [[5, 0]]
... (3 more boundary cells)
answer: "April 13 to April 18"
```

Figure 4: **Qualitative comparison (real test example).** SFT cites 15 cells in one step, bundling five complete table rows. RSAT cites one cell per step, pointing to exactly the evidence needed. (Both models produce slightly imprecise date ranges compared to the gold answer; the comparison illustrates attribution quality, not answer accuracy.)

not a primary learning signal. See Appendix D for visualizations.

5.5 Qualitative Analysis

Figure 4 shows a real test example comparing Qwen 7B SFT-only vs. RSAT on a FeTaQA concert tour question. The contrast reveals GRPO’s learned citation behavior: SFT bundles the entire table section (15 cells across 5 rows) into a single step, while RSAT points to one cell per step—exactly the date cells needed to establish the span. Both outputs are valid JSON with correct answers; the difference is attribution quality and conciseness. RSAT completion length averages 71–211 tokens across models, compared to ~250 tokens for SFT (Appendix I).

6 Discussion and Limitations

The central insight from our experiments is that format compliance and faithfulness are orthogonal capabilities in small language models. Supervised learning solves format almost completely (~99%) but barely touches faithfulness (~22%); only reward-driven optimization bridges the gap. Practically, *high format compliance should not be mistaken for high attribution quality*.

Why SFT fails at faithfulness. We conjecture that SFT teaches surface-level imitation: the model copies the *format* of gold traces—valid JSON, plausible cell references—without internalizing the se-

semantic relationship between cited cells and claims. The 22% faithfulness likely reflects the base rate at which randomly selected valid cells happen to entail the step text. GRPO’s faithfulness reward provides the missing signal.

Practical lessons for GRPO training. For practitioners: *monitor reward curves, not loss*. GRPO loss remained near zero throughout training for all six models, yet reward means consistently increased ($\sim 0.75 \rightarrow \sim 1.5$) and eval metrics confirmed genuine learning. The near-zero loss is expected—GRPO normalizes advantages to zero mean within each group. Training curves are in Appendix F.

Learned conciseness. The parsimony reward drives dramatic output compression: average completion length drops from ~ 250 tokens (SFT) to 71–211 tokens (RSAT), with some models converging to 1.0 reasoning step per output. Whether this represents optimal conciseness or over-compression warrants further investigation.

Limitations. Several limitations should be noted. (1) **Train-eval circularity.** Faithfulness is both the training reward and the primary evaluation metric, both scored by the same DeBERTa NLI model. The model may learn to produce text that scores well on DeBERTa specifically rather than achieving genuine grounding. Additionally, if the scorer is noisy or biased toward certain linguistic patterns, the model may inherit these imperfections. A human evaluation study is needed to validate the NLI proxy. (2) **In-distribution evaluation.** All test examples come from WTQ, FeTaQA, and TabFact; generalization to unseen domains (e.g., financial, medical, or scientific tables) remains untested. These domains often feature more complex schemas and domain-specific semantics that may challenge the model’s ability to ground reasoning accurately. (3) **Low exact match.** EM ranges from 0.000–0.018 across all experiments, reflecting strict string matching on paraphrased answers. F1 captures the partial correctness that EM misses, but this limits direct comparison with systems reporting only EM.

Future work. The most pressing next step is a human evaluation (50–100 examples) to validate NLI-based faithfulness and quantify the train-eval circularity gap. Replacing the NLI scorer with a learned reward model trained on human faithfulness judgments would break this circularity entirely. Cross-domain evaluation on financial or

scientific tables would test generalization. More broadly, RSAT demonstrates that RL with composite, domain-specific rewards can teach small models capabilities that supervised learning alone cannot—a paradigm that may extend beyond tables to attributed reasoning over knowledge graphs, code, or APIs.

7 Conclusion

Faithful attribution in table reasoning is not a post-processing problem — it is a training objective. RSAT demonstrates this through a two-phase approach: SFT teaches structured output format, and GRPO with a composite reward teaches the model to *ground* that structure in actual table evidence. Across six models from two architecture families, RSAT achieves $3.7\times$ faithfulness improvement over SFT alone, with near-perfect citation validity — while post-hoc attribution collapses below 13% format success. Ablations confirm that the NLI-based faithfulness reward is the sole essential signal.

Our results suggest a broader principle: for structured generation tasks where output *format* is easy to learn but output *quality* is not, supervised learning and reinforcement learning play complementary, non-substitutable roles. The complete code and data for RSAT are available at <https://github.com/JugalGajjar/RSAT>.

References

- Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.
- Bernd Bohnet, Vinh Q Tran, Pat Verga, Roei Aharoni, Daniel Andor, Livio Baldini Soares, Massimiliano Ciaramita, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, and 1 others. 2022. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv preprint arXiv:2212.08037*.
- Wenhu Chen, Hongmin Wang, Jianshu Chen, Yunkai Zhang, Hong Wang, Shiyang Li, Xiyu Zhou, and William Yang Wang. 2019. Tabfact: A large-scale dataset for table-based fact verification. *arXiv preprint arXiv:1909.02164*.
- Zhoujun Cheng, Tianbao Xie, Peng Shi, Chengzu Li, Rahul Nadkarni, Yushi Hu, Caiming Xiong, Dragomir Radev, Mari Ostendorf, Luke Zettlemoyer, and 1 others. 2022. Binding language models in symbolic languages. *arXiv preprint arXiv:2210.02875*.

- Quy-Anh Dang and Chris Ngo. 2025. Reinforcement learning for reasoning in small llms: What works and what doesn't. *arXiv preprint arXiv:2503.16219*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.
- Julian Eisenschlos, Syrine Krichene, and Thomas Müller. 2020. Understanding tables with intermediate pre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 281–296.
- Luyu Gao, Zhu Yun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and 1 others. 2023a. 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)*, pages 16477–16508.
- Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Zihui Gu, Ju Fan, Nan Tang, Preslav Nakov, Xiaoman Zhao, and Xiaoyong Du. 2022. Pasta: table-operations aware fact verification via sentence-table cloze pre-training. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 4971–4983.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.
- Jonathan Herzig, Pawel Krzysztof Nowak, Thomas Müller, Francesco Piccinno, and Julian Eisenschlos. 2020. Tapas: Weakly supervised table parsing via pre-training. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4320–4333.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *Iclr*, 1:3.
- Qian Liu, Bei Chen, Jiaqi Guo, Morteza Ziyadi, Zeqi Lin, Weizhu Chen, and Jian-Guang Lou. 2021. Tapex: Table pre-training via learning a neural sql executor. *arXiv preprint arXiv:2107.07653*.
- Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, and 1 others. 2022. Fetaqa: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- 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*, 49:777–840.
- Michael Schlichtkrull, Vladimir Karpukhin, Barlas Oguz, Mike Lewis, Wen-tau Yih, and Sebastian Riedel. 2021. Joint verification and reranking for open fact checking over tables. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6787–6799.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

- Jonas Wallat, Maria Heuss, Maarten de Rijke, and Avishek Anand. 2024. Correctness is not faithfulness in rag attributions. *arXiv preprint arXiv:2412.18004*.
- Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and 1 others. 2024. Chain-of-table: Evolving tables in the reasoning chain for table understanding. *arXiv preprint arXiv:2401.04398*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 23 others. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.
- Xiaoyu Yang, Feng Nie, Yufei Feng, Quan Liu, Zhigang Chen, and Xiaodan Zhu. 2020. Program enhanced fact verification with verbalization and graph attention network. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7810–7825.
- Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. Tabert: Pretraining for joint understanding of textual and tabular data. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 8413–8426.
- Yilun Zhao, Lyuhao Chen, Arman Cohan, and Chen Zhao. 2024. Tapera: Enhancing faithfulness and interpretability in long-form table qa by content planning and execution-based reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12824–12840.
- Victor Zhong, Caiming Xiong, and Richard Socher. 2017. Seq2sql: Generating structured queries from natural language using reinforcement learning. *arXiv preprint arXiv:1709.00103*.

A Data Construction Details

Gold trace generation. Each training example is generated by prompting Claude Opus 4.5 with a system prompt specifying the JSON format, followed by the linearized table and question. The system prompt instructs the model to produce 3–4 reasoning steps, each citing 1–4 specific cells using $[\text{row}_i, \text{col}_j]$ coordinates.

Verification pipeline. Generated traces undergo three automated checks: (1) JSON parse validity; (2) all cited cell coordinates are within table bounds; (3) step count is between 3–4. Failed examples are batched and re-prompted with error-specific instructions. Final repair rate: WTQ required 18 repair batches; FeTaQA and TabFact were 100% valid on first generation.

B Training Hyperparameters

Table 5: Hyperparameters for SFT and GRPO training across all model scales.

Parameter	SFT		GRPO	
	1-3B	7-8B	1-3B	7-8B
LoRA rank r	32	64	16	32
LoRA α	64	128	32	64
LoRA dropout	0.05	0.05	0.05	0.05
Learning rate	2e-4	2e-4	5e-5	5e-5
Batch size	4	8	2	4
Grad. accum.	4	2	8	4
Eff. batch size	16	16	16	16
Epochs	3	3	1	1
Scheduler	cosine	cosine		
Precision	bf16	bf16		
Max seq. length	2048	1024		
Group size G	—	8		
Temperature	—	0.9		

C Reward Component Definitions

Full mathematical definitions for the composite reward (Section 3.5).

Answer F1. Given predicted answer tokens P and gold answer tokens G :

$$\text{Prec} = \frac{|P \cap G|}{|P|}, \quad \text{Rec} = \frac{|P \cap G|}{|G|} \quad (2)$$

$$R_{\text{ans}} = \frac{2 \cdot \text{Prec} \cdot \text{Rec}}{\text{Prec} + \text{Rec}} \quad (3)$$

When multiple gold answers exist, we take the maximum F1.

Citation validity. For output with N total cited cells across all steps:

$$R_{\text{cite}} = \frac{1}{N} \sum_{i=1}^N \mathbf{1}[0 \leq r_i < R \wedge 0 \leq c_i < C] \quad (4)$$

Faithfulness. For each reasoning step s_k with cited cells $\{(r_i, c_j)\}$, we concatenate the cell values into evidence string e_k , then compute:

$$R_{\text{faith}} = \frac{1}{K} \sum_{k=1}^K \text{NLI}_{\text{entail}}(e_k, s_k) \quad (5)$$

where $\text{NLI}_{\text{entail}}$ is the entailment probability from DeBERTa-v3-base (cross-encoder/nli-deberta-v3-base). Steps with no cited cells receive score 0.

Parsimony. For each step with n_k cited cells:

$$R_{\text{pars}} = \frac{1}{K} \sum_{k=1}^K \begin{cases} 1.0 & n_k \leq 3 \\ \frac{8-n_k}{5} & 3 < n_k < 8 \\ 0.0 & n_k \geq 8 \end{cases} \quad (6)$$

D Ablation Visualization

Figure 5 visualizes the ablation study results from Section 5, showing the effect of removing each reward component on faithfulness and parsimony.

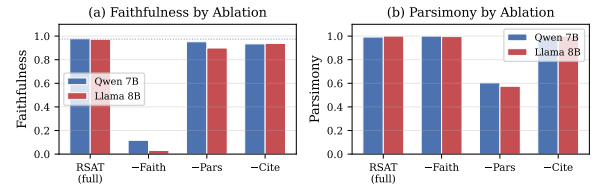


Figure 5: **Ablation impact on faithfulness and parsimony.** Removing faithfulness reward causes near-total collapse; removing parsimony causes significant over-citation.

E Output Format Specification

The RSAT output format is a JSON object with the following schema:

```
{
  "reasoning_steps": [
    {
      "step": "<natural language claim>",
      "cited_cells": [
        [<row_index>, <col_index>],
        ...
      ]
    },
    ...
  ],
  "answer": "<final answer string>"
}
```

F Training Curves

SFT convergence. Figure 6 plots training and validation loss for all six models during SFT. All models converge smoothly with no significant overfitting: validation loss plateaus after epoch 1 while training loss continues to decrease slightly. The Llama models start at higher loss (~ 0.70) compared to Qwen (~ 0.56 – 0.70) but converge to comparable final values.

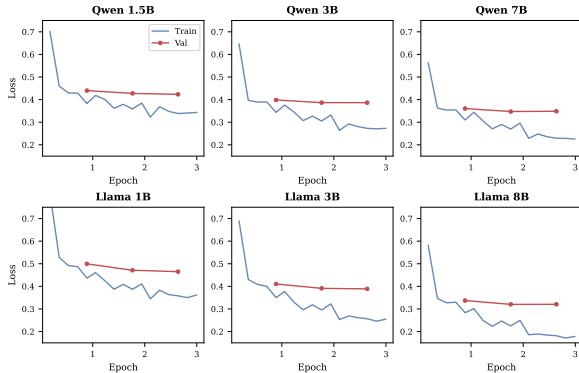


Figure 6: **SFT training curves.** Train and validation loss over 3 epochs for all 6 models. All models converge without overfitting.

GRPO reward dynamics. Figure 7 shows the composite reward mean during GRPO training (250 steps). All models show a clear upward trend from ~ 0.75 to $\sim 1.5+$, confirming that GRPO successfully optimizes the reward signal. The near-zero GRPO loss is expected and not indicative of a training failure.

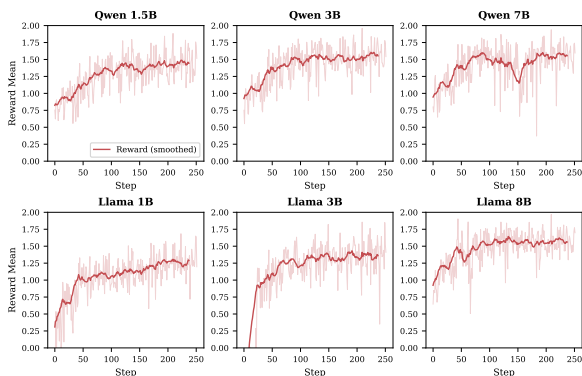


Figure 7: **GRPO reward curves.** Composite reward mean increases for all models despite near-zero loss. Smoothed with a 15-step moving average; raw values shown in light shading.

G GRPO Training Statistics

Table 6 reports per-model GRPO training statistics.

Table 6: GRPO training statistics per model (250 steps each).

Model	Time (min)	Final Reward	Final Entropy
Qwen 1.5B	120	1.560	0.118
Qwen 3B	116	1.623	0.057
Qwen 7B	265	1.590	0.038
Llama 1B	105	1.459	0.197
Llama 3B	143	1.488	0.056
Llama 8B	123	1.660	0.033

The final entropy values (0.03–0.20) confirm that GRPO sharpens the policy distribution: smaller entropy indicates more confident, focused generations. Llama 1B retains the highest entropy (0.197), consistent with its lower faithfulness (0.480). Llama 8B achieves the highest final reward (1.660), aligning with its top faithfulness score (0.972).

H Compute Budget

Table 7: Wall-clock time on a single NVIDIA H100 80GB GPU.

Model	SFT	GRPO	Eval	Total
Qwen 1.5B	7 min	120 min	26 min	153 min
Qwen 3B	11 min	116 min	20 min	147 min
Qwen 7B	16 min	265 min	51 min	333 min
Llama 1B	5 min	105 min	27 min	137 min
Llama 3B	11 min	143 min	25 min	179 min
Llama 8B	18 min	123 min	16 min	158 min
6-model subtotal	68 min	874 min	165 min	18.4 hrs
<i>Ablation runs (6 configs \times train + eval)</i>				18.3 hrs
Grand Total				36.8 hrs

GRPO dominates training time due to generating $G=8$ candidates per question and computing per-step NLI faithfulness scores. Qwen 7B is the most expensive model (265 min GRPO) because its longer completions require more generation and scoring time.

I Completion Length Analysis

Figure 8 compares average completion lengths at the start and end of GRPO training. All models show substantial compression: the average reduction is 40–55% across models. This confirms that the parsimony reward teaches models to produce concise, focused reasoning rather than verbose multi-step chains. The effect is especially pronounced for the 3B and 7–8B models, which begin with longer completions due to greater generation capacity.

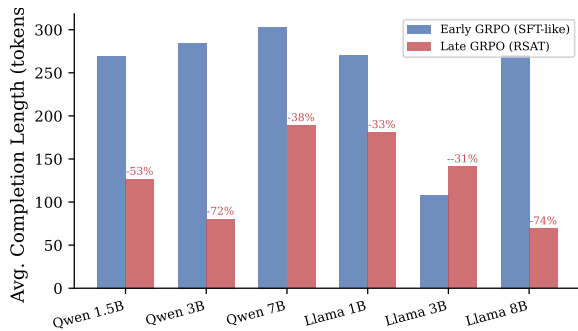


Figure 8: **Completion length reduction during GRPO.** Early GRPO completions (SFT-like behavior) vs. late GRPO completions (RSAT behavior). All models learn to produce shorter, more focused outputs.

J Post-hoc Failure Analysis

Table 8 categorizes the failure modes of post-hoc attribution. The two-pass approach asks the model to first generate a chain-of-thought answer, then retroactively map its reasoning to cell coordinates in JSON format. The dominant failure mode is *empty or non-JSON output* (76–100% of examples), not JSON parse errors. The models simply do not produce the required format in the second pass—they either repeat the question, generate a natural language response, or produce truncated output. This suggests the failure is fundamental: small models lack the working memory to re-read a table and retro-fit structured citations.

Table 8: Post-hoc failure modes (500 test examples per model).

Model	Fmt%	Empty/No-JSON	Valid
Qwen 1.5B	24.4%	75.6%	24.4%
Qwen 3B	0.4%	99.6%	0.4%
Qwen 7B	14.0%	86.0%	14.0%
Llama 1B	19.8%	80.2%	19.8%
Llama 3B	9.0%	91.0%	9.0%
Llama 8B	4.0%	96.0%	4.0%
Average	12.0%	88.1%	12.0%

Counterintuitively, larger models perform *worse*: Llama 8B (4.0%) is below Llama 1B (19.8%). Larger models produce longer free-text responses in the second pass, making JSON compliance harder.

K Faithfulness Improvement Visualization

Figure 9 visualizes the faithfulness improvement from SFT to RSAT for each model. Qwen 1.5B achieves the largest relative gain

(5.7×), while Llama 1B shows the smallest absolute improvement—still a 2.5× gain. Both architecture families benefit substantially from GRPO, confirming that the faithfulness reward transfers across model families.

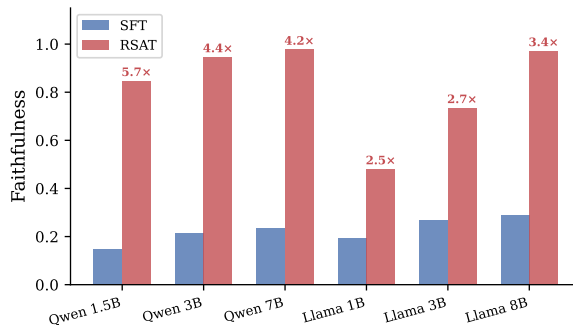


Figure 9: **Faithfulness: SFT vs. RSAT per model.** All models show substantial gains; the 3.7× average is driven by near-ceiling Qwen performance.