

OmniDA: Grounding Agentic Multimodal Data Analysis with Evidence-Aware Process Rewards

Xuhang Zhu^{1*}, Liyao Li^{1*}, Ningtao Wang², Yanyu Chen²,
Hao Chen¹, Wentao Ye¹, Ziquan Wang¹, Haoxing Zhai¹,
Yu Cheng², Gang Chen¹, Xiu Tang^{1†}, Haobo Wang^{1†}

¹Zhejiang University, ²Ant Group
{zhuxuhang, liliyao, tangxiu, wanghaobo}@zju.edu.cn

Abstract

Real-world data analysis often requires joint reasoning over heterogeneous artifacts such as tables, charts, and textual reports. Training reliable multimodal large language model (MLLM) agents for this setting remains challenging, as large-scale aligned training data are scarce, and outcome-only rewards are too coarse to distinguish grounded cross-source analysis from spurious success. We introduce OmniDA¹, a training recipe that addresses these two challenges with a scalable multimodal data-synthesis pipeline and an Evidence-Aware Process Reward (EAPR) that supervises both evidence recovery and procedural validity. Through this pipeline, we construct OmniDA-11K, a high-quality training corpus for multimodal data analysis. When instantiated with Qwen3-VL-8B, OmniDA improves over the corresponding agentic baseline by 19.2 points on average across five challenging benchmarks and achieves the strongest overall performance among existing open-source data-analysis agents. These results highlight the value of supervising evidence recovery and analytical procedure beyond final-answer correctness.

1 Introduction

Large language models (LLMs), and more recently multimodal large language models (MLLMs), have enabled agents that translate analytical goals into executable programs, interact with external environments, and derive answers from intermediate observations. Recent progress has been encouraging in structured settings such as tables, spreadsheets, and databases (Qiao et al., 2025; Zhang et al., 2025a). Real-world analysis, however, rarely resides in a single source. Analysts often need to

combine tables, charts, and textual reports, where relevant evidence is distributed across modalities and must be aligned through entities, time periods, thresholds, and derived statistics (Foroutan et al., 2025; Qi et al., 2026; Kim et al., 2025). Although recent benchmarks have begun to characterize this setting, training reliable MLLM-based agents for heterogeneous data analysis remains substantially less mature than evaluation.

We argue that the main bottleneck is not only multimodal perception, but also supervision. In multimodal data analysis, a correct final answer does not necessarily imply a grounded analytical trajectory. Agent may misread a chart value, relax a textual constraint, or silently broaden a filtering scope, yet still arrive at the correct answer by chance, as shown in Figure 1. Under standard outcome-based reinforcement learning, such a trajectory still receives positive reinforcement, because the final answer collapses a long reasoning process into a single correctness judgment. Outcome rewards therefore conflate answer correctness with analytical faithfulness, even though these two properties can diverge sharply in multimodal data analysis.

Training a generalist multimodal data-analysis agent faces two intrinsic challenges. **First, large-scale multimodal training data are scarce.** Existing multimodal data-analysis benchmarks are typically small and evaluation-oriented, providing limited support for large-scale agent training. Constructing aligned tables, charts, and textual reports together with non-trivial, verifiable questions requires substantial expert effort (Foroutan et al., 2025; Kim et al., 2025), and there is still no broadly adopted pipeline for synthesizing such training tasks at scale while preserving cross-modal dependencies and controllable reasoning difficulty. **Second, outcome rewards are too coarse to supervise grounded analysis.** Final-answer correctness alone cannot determine whether an agent has re-

*Equal contribution.

†Corresponding author.

¹Code and data are available at <https://github.com/moonwalker-z/OmniDA>.

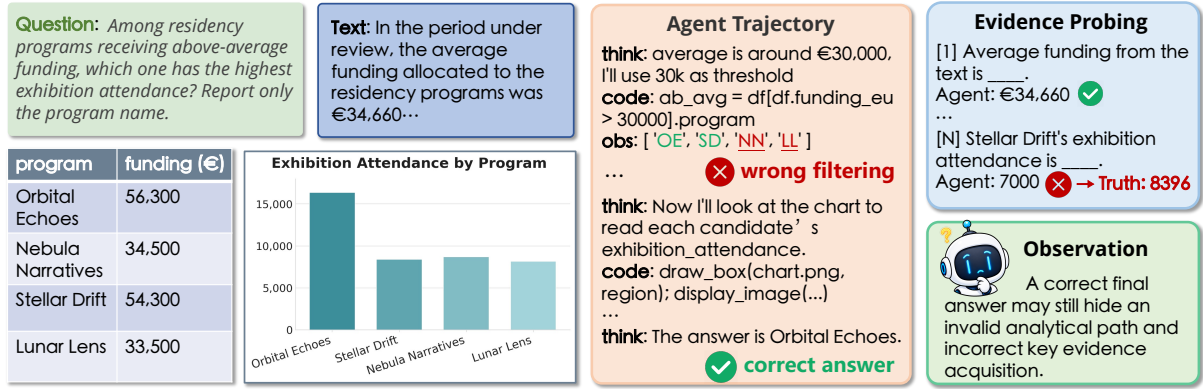


Figure 1: A motivating example of the supervision gap in multimodal data analysis. The agent reaches the correct final answer despite violating a textual constraint and failing to recover a required chart evidence value. Standard outcome rewards would reinforce this trajectory, whereas our Evidence-Aware Process Reward (EAPR) penalizes it by probing intermediate evidence and evaluating procedural validity.

covered the intermediate evidence required by the task, nor whether it has used that evidence through a valid analytical procedure. As a result, shortcut-based or spuriously grounded trajectories may still be reinforced.

To tackle these challenges, we introduce OmniDA, a training framework for grounded multimodal data-analysis agents. We first develop a scalable pipeline for constructing multimodal analytical training tasks from large-scale tabular seeds collected from the web. The pipeline allocates information across tables, charts, and textual reports; composes multimodal analytical primitives to control question structure and difficulty; and refines synthesized instances through code-based verification, majority voting, information-exclusivity checking, and format diversification. Through this pipeline, we construct OmniDA-11K, a training corpus of 11K multimodal data-analysis instances. We then propose Evidence-Aware Process Reward (EAPR), a trajectory-level reward that supervises both *what* evidence the agent has recovered and *how* it uses that evidence. EAPR combines question-specific process rubrics with executable evidence checklists that probe whether the agent has established the essential intermediate evidence before producing its final answer.

Built on this framework, we train an agent initialized from Qwen3-VL-8B using cold-start supervised fine-tuning followed by EAPR-guided reinforcement learning. Across five challenging data-analysis benchmarks, OmniDA improves over the corresponding agentic baseline by 19.2 points on average and achieves the strongest overall performance among existing open-source data-analysis

agents. The learned supervision also transfers beyond fixed-answer QA, improving the DAComp-DA score by 10.5 points on open-ended data-analysis tasks.

In summary, our main contributions are three-fold:

- We identify a central *supervision gap* in multimodal data analysis: final-answer correctness alone cannot distinguish grounded cross-source analysis from spurious reasoning.
- We propose OmniDA, which combines a scalable multimodal data-synthesis pipeline with Evidence-Aware Process Reward (EAPR) to supervise both evidence recovery and procedural validity.
- We construct OmniDA-11K, a training corpus produced by our synthesis pipeline, and show that the resulting agent substantially improves over the corresponding base model and prior open-source data-analysis agents.

2 Method

2.1 Problem Formulation

We consider a multimodal data-analysis task as $s = (q, \mathcal{E}, y)$, where q is a natural-language query, $\mathcal{E}$ is a heterogeneous data environment containing tables, charts, and textual reports, and y denotes the ground-truth answer. Given q and $\mathcal{E}$, an agent produces a ReAct-style trajectory τ consisting of reasoning steps, code actions, environment observations, and a final answer $\hat{y}$.

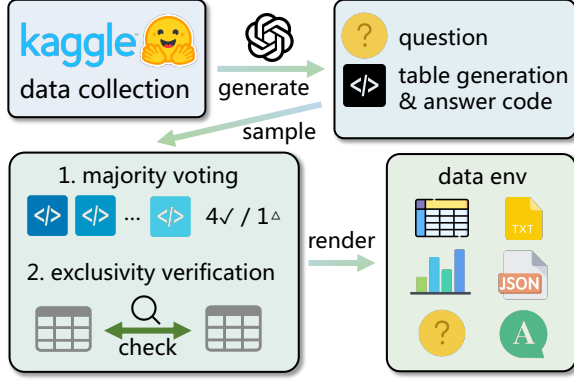


Figure 2: Overview of the multimodal data-analysis task synthesis pipeline, including primitive-based question synthesis, code-based answer verification, and format diversification.

The goal is to learn a policy π_θ that generates trajectories leading to correct and grounded answers:

$$\max_{\theta} \mathbb{E}_{s \sim \mathcal{D}, \tau \sim \pi_\theta(\cdot | q, \mathcal{E})} [R(\tau; s)]. \quad (1)$$

2.2 Scalable Multimodal Data-Analysis Task Synthesis

2.2.1 Data Preparation and Primitive-based Query Synthesis

Developing versatile data-analysis agents requires learning from complex and diverse heterogeneous data environments. However, publicly available data resources rarely provide aligned tables, charts, and text, as real-world analyses often expose only final reports or visualizations while keeping the underlying data sources private. To address this scarcity, our pipeline adopts a simulation-based synthesis strategy that constructs multimodal data environments from publicly available tabular seeds. Specifically, it collects raw tables from Kaggle and Hugging Face across diverse domains, and extracts their schemas and value distributions as priors. The synthesis process then plans modality-specific information allocation, determining which variables, relations, and derived statistics should be preserved in tables, visualized in charts, or verbalized in textual reports. Guided by this plan, an LLM expands and rewrites the original schema, and generates executable table-construction code to instantiate synthetic records. The resulting structured data serve as the shared underlying environment for rendering aligned heterogeneous artifacts.

To enable flexible and diverse question synthesis, we systematically define seven analytical reasoning primitives for multimodal data analysis: extrac-

tion, selection, aggregation, arithmetic, comparison, ranking, and set operation. Detailed definitions and examples are provided in Appendix A. We then synthesize questions by composing these primitives across modalities, so that solving a question requires intermediate results obtained from one source to be combined with, compared against, or further transformed using information from another source. By varying the chain length, primitive diversity, and number of cross-modal transitions, the pipeline produces questions of different difficulty levels, from simple analytical queries to multi-hop problems requiring joint reasoning over tables, charts, and text.

2.2.2 Verification and Format Diversification

Each synthesized question is verified through independent code-based solving. Given a question q and its underlying multimodal data environment $\mathcal{E}$, we sample an expert model N times to generate independent answer programs. Executing these programs produces a set of candidate answers $\{\hat{y}_i\}_{i=1}^N$. Since these answers may vary in surface form and are difficult to compare purely with rules, we use a judge model to determine answer equivalence, apply majority voting to obtain the final reference answer y , and discard instances without a clear majority. In our implementation, we use DeepSeek-V3.2 (DeepSeek-AI, 2025) as the expert code-generation model and GLM-4.7-flash (Zhipu AI, 2026) as the judge model.

Finally, we diversify the presentation formats of synthesized data while preserving the underlying answer-relevant information. Tables are rendered in different textual formats, charts are visualized with varied templates and styles, and textual reports are expressed with diverse narrative styles, information orders, and levels of conciseness while keeping all factual statements unchanged. This reduces format-specific artifacts and encourages agents to reason over the underlying multimodal evidence. Through this pipeline, we construct OmniDA-11K, a large-scale synthetic training corpus for multimodal data-analysis agents.

2.3 Evidence-Aware Process Reward

To move beyond outcome-only supervision, we introduce Evidence-Aware Process Reward (EAPR), a trajectory-level reward mechanism for training multimodal data-analysis agents. As shown in Figure 3, EAPR assigns fine-grained rewards from two complementary aspects: evidence awareness and

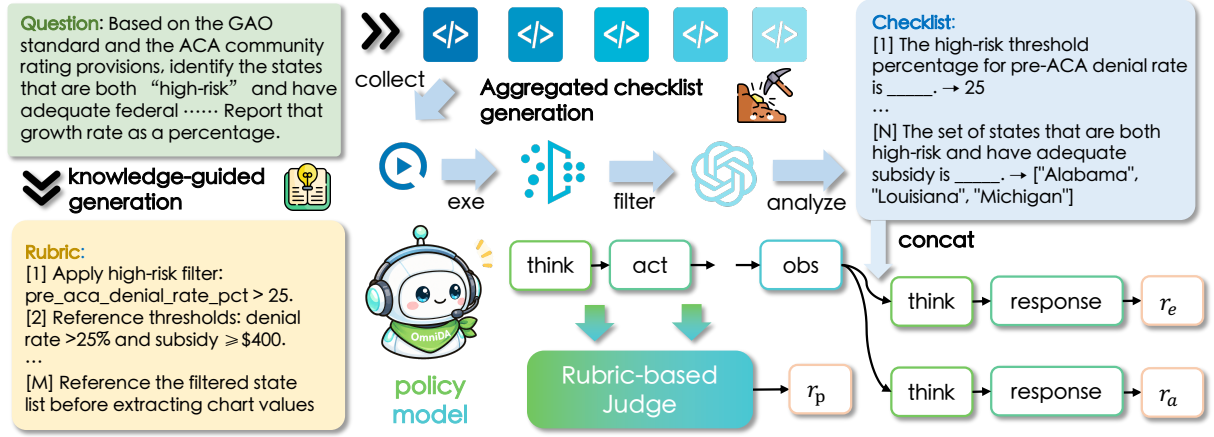


Figure 3: EAPR-guided training pipeline of OmniDA. EAPR combines grounded evidence probing and question-specific process rubrics to provide fine-grained trajectory-level rewards for optimizing multimodal data-analysis agents.

process validity. For evidence awareness, EAPR builds a question-specific evidence checklist and probes the pre-answer agent state to verify whether the agent has recovered essential grounded evidence. For process validity, EAPR constructs a question-specific trajectory rubric and applies it to the completed rollout, checking whether the analysis follows the required procedure.

2.3.1 Grounded Evidence Probing

Evidence probing aims to verify whether the agent has recovered the essential grounded facts required by the task before producing the final answer. Rather than relying on surface mentions in the trajectory, we construct a question-specific evidence checklist and use it to probe the pre-answer agent state.

Evidence checklist construction. Given a question, its multimodal data environment, the canonical answer, and multiple independently sampled solution programs verified against the canonical answer, an expert model analyzes the shared solution logic and extracts a set of evidence items that commonly support the correct solution. Each item corresponds to a necessary intermediate fact, such as a numerical statistic, an entity set, a textual condition, or a derived comparison. It is represented by a natural-language claim, a grounded value, and an executable verification snippet that recomputes the grounded value from the underlying data.

Grounding verification. To ensure that the checklist is verifiable, we execute the verification snippet of each evidence item in a sandbox and compare the computed value with the proposed grounded value. Items whose values cannot be verified are dis-

carded, while well-formed mismatches can be automatically corrected using the executed value. We also remove items that directly restate the canonical answer, so that the checklist focuses on intermediate evidence rather than final-answer matching.

Prefix-based evidence probing. We rewrite each verified evidence item into a cloze-style probe with a single blank. During reward computation, we first complete the agent rollout, remove the final-answer turn, and append the evidence checklist to the remaining trajectory prefix. The agent is asked to fill in the checklist using only the information available in this prefix, without calling additional tools or re-solving the task. This allows us to test whether the agent had recovered the required evidence before producing its final answer, without interrupting the original rollout.

Reward computation. Let $\mathcal{P} = \{p_i\}_{i=1}^m$ denote the verified evidence probes for a sample, and let g_i be the grounded answer key of probe p_i . Given the agent’s response $\hat{g}_i$, we compute a binary correctness score with type-aware matching, using relative tolerance for scalar values and exact matching for sets, lists, strings, and booleans. The evidence reward is the average probing accuracy:

$$R_e = \frac{1}{|\mathcal{P}|} \sum_{i=1}^{|\mathcal{P}|} \mathbb{1}[\text{match}(\hat{g}_i, g_i)]. \quad (2)$$

2.3.2 Trajectory Process Rubric

While grounded evidence probing verifies whether the agent has recovered the necessary intermediate facts, it does not fully characterize the quality of the analysis procedure. An agent may identify

several key values but still rely on an incomplete source set, misalign entities across modalities, or apply an incorrect filtering scope. To capture these procedural failures, we further construct a question-specific process rubric for judging the completed ReAct trajectory.

Rubric construction. For each question, we generate a compact set of process criteria conditioned on the query, canonical answer, modality configuration, and data environment. Unlike the evidence checklist, the process rubric does not focus on concrete grounded values. Instead, it specifies the expected procedural behaviors required for solving the task. To make process supervision structured and task-adaptive, we design an expert-informed procedural rubric schema for multimodal data analysis. The schema encodes desired procedural behaviors together with common failure modes, and organizes process validity into five dimensions: source acquisition, cross-source grounding, analytical scope, answer derivation, and observation utilization. For each question, this general schema is instantiated into a compact set of positive, question-specific criteria based on the query and the data environment. A criterion is satisfied only when the trajectory follows the corresponding procedural requirement; otherwise, the unsatisfied criterion implicitly captures the associated failure pattern.

Trajectory-level judgment. Given a completed rollout, we apply the process rubric to the entire ReAct trajectory rather than assigning rewards to individual steps. This avoids brittle step-level attribution while still allowing the judge model to inspect how the agent plans, calls tools, interprets observations, and forms the final answer. Each criterion is evaluated as a binary judgment indicating whether the trajectory satisfies the corresponding procedural requirement.

Reward computation. Let $\mathcal{C} = \{c_i\}_{i=1}^n$ denote the process criteria for a sample, and let $s_i \in \{0, 1\}$ be the judgment for criterion c_i . The process reward is computed as the average satisfaction rate over all applicable criteria:

$$R_p = \frac{1}{|\mathcal{C}|} \sum_{i=1}^{|\mathcal{C}|} s_i. \quad (3)$$

This process reward provides direct supervision on the procedural validity of the completed multimodal analysis trajectory.

2.3.3 EAPR-Guided Policy Optimization

We use EAPR as a trajectory-level shaping signal for reinforcement learning. For each training instance x , the current policy samples a group of G ReAct rollouts $\{\tau_i\}_{i=1}^G$. Each rollout contains the agent’s reasoning, tool calls, observations, and final answer. After the rollout is completed, we compute three rewards: an answer reward R_a , the evidence reward R_e , and the process reward R_p . The probing response used for computing R_e is used only for reward estimation, and is not included in the policy-gradient loss.

We first combine the two EAPR components as:

$$R_{\text{EAPR}}(\tau_i) = \alpha R_e(\tau_i) + (1 - \alpha) R_p(\tau_i), \quad (4)$$

where α controls the relative emphasis on evidence recovery and procedural validity. The overall training reward is then defined as:

$$R(\tau_i) = R_a(\tau_i) + \lambda \mathbb{I}[R_a(\tau_i) = 1] R_{\text{EAPR}}(\tau_i). \quad (5)$$

Here, R_a preserves the primary objective of producing the correct final answer, while R_{EAPR} provides additional supervision on answer-correct rollouts by measuring whether the agent has recovered essential grounded evidence and followed a valid multimodal analysis procedure. For rollouts with incorrect final answers, the EAPR term is masked out and does not contribute to the training reward.

We optimize the policy with group-relative policy optimization. Given the rewards of the rollout group, we compute the normalized advantage for each rollout as:

$$\hat{A}_i = \frac{R(\tau_i) - \text{mean}(\{R(\tau_j)\}_{j=1}^G)}{\text{std}(\{R(\tau_j)\}_{j=1}^G) + \epsilon}. \quad (6)$$

The policy is updated by maximizing the clipped objective:

$$\begin{aligned} \mathcal{J}_{\text{EAPR}}(\theta) = \mathbb{E}_{\substack{x \sim \mathcal{D}, \\ \{\tau_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(\cdot|x)}} & \left[\frac{1}{G} \sum_{i=1}^G \frac{1}{|\tau_i|} \sum_{t=1}^{|\tau_i|} \right. \\ & m_{i,t} \min(\rho_{i,t} \hat{A}_i, \text{clip}(\rho_{i,t}, 1 - \epsilon, 1 + \epsilon) \hat{A}_i) \\ & \left. - \beta D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}}) \right]. \quad (7) \end{aligned}$$

where $\rho_{i,t} = \frac{\pi_{\theta}(y_{i,t}|x, y_{i,<t})}{\pi_{\theta_{\text{old}}}(y_{i,t}|x, y_{i,<t})}$ is the token-level importance ratio, and $m_{i,t}$ masks out tokens from external observations so that gradients are applied only to tokens generated by the agent. In this way, EAPR guides the policy toward rollouts that are not only answer-correct, but also evidence-grounded and procedurally reliable.

3 Experiments

3.1 Experimental Setup

Benchmarks and Evaluation Details. We evaluate agent performance on five data-analysis benchmarks that cover diverse source combinations: FCMR (T+C+X; Kim et al., 2025), WikiMixQA (T+C; Foroutan et al., 2025), Chart-MRAG (C+X; Yang et al., 2025b), CharXiv (C; Wang et al., 2024), and DABench (T; Hu et al., 2024). Here, T, C, and X denote table, chart, and textual sources, respectively. For text-only models that do not support chart-image inputs, we follow the plot-to-table translation paradigm (Liu et al., 2023) and replace chart images with linearized textual tables generated by GPT-4o-mini. To evaluate answer correctness, we adopt a model-based evaluation protocol and use DeepSeek-V3.2 as the judge model. Additional details about the benchmarks and evaluation setup are provided in Appendix B.

Baselines. We compare our models against two groups of strong baselines. The first group consists of five frontier general-purpose models, including GPT-5.2 (OpenAI, 2025), Claude-4.6-Sonnet (Anthropic, 2026), Gemini-3-Pro (Google DeepMind, 2026), DeepSeek-V3.2 (DeepSeek-AI, 2025), and Qwen3-VL-235B-A22B (Team, 2025). The second group includes four representative open-source data-analysis agents: DataMind (Qiao et al., 2025), DeepAnalyze (Zhang et al., 2025a), TableGPT2 (Su et al., 2024), and TableGPT-R1 (Yang et al., 2025a).

Implementation Details. The model is initialized from Qwen3-VL-8B (Team, 2025) and optimized through a two-stage pipeline: cold-start supervised fine-tuning (SFT) followed by reinforcement learning (RL). For SFT, GPT-5.2 is used to generate trajectories from OmniDA-11K, from which 3,083 high-quality trajectories are retained after quality checking and filtering. The SFT stage is conducted for 3 epochs with a batch size of 64 and a learning rate of $1e-5$. The RL stage is performed on the remaining portion of OmniDA-11K, using a batch size of 32 and a learning rate of $1e-6$. The evidence-process balancing weight α is set to 0.5, and the EAPR reward coefficient λ is set to 0.4. GLM-4.7-flash (Zhipu AI, 2026) serves as the judge reward model for both final-answer correctness and process-rubric satisfaction.

3.2 Main Results

Table 1 reports the overall results on five data-analysis benchmarks. OmniDA consistently improves over the Qwen3-VL-8B agentic baseline across all benchmarks, demonstrating the effectiveness of our data synthesis and EAPR-guided training pipeline. On average, OmniDA improves the base agent from 38.38 to 57.57, with especially large gains on FCMR, Chart-MRAG, and WikiMixQA. These results suggest that training on multimodal data-analysis environments substantially enhances the agent’s ability to reason over heterogeneous sources rather than only execute generic tool-use behaviors.

Compared with existing data-analysis agents, OmniDA achieves the best performance on all five benchmarks. The improvements are particularly pronounced on multimodal benchmarks that require reasoning over charts and textual evidence. For example, OmniDA outperforms the strongest prior data-analysis agent by 26.61 points on FCMR, 28.90 points on Chart-MRAG, and 17.97 points on WikiMixQA. This indicates that methods primarily designed for tabular or structured data analysis are insufficient for tasks requiring cross-modal evidence integration.

OmniDA also remains competitive with advanced model-based agents despite using a much smaller base model. While it still trails frontier proprietary agents on some benchmarks such as FCMR and CharXiv, it achieves comparable performance on Chart-MRAG and the second-best result on DABench. This suggests that targeted multimodal data-analysis training can substantially narrow the gap between compact open-source models and stronger general-purpose agents.

3.3 Ablation Study

Effect of EAPR components. Table 2 shows that both evidence probing and process rubrics contribute consistently to OmniDA. Removing the evidence component leads to an average drop of 3.02 points, indicating that verifying whether the agent has recovered essential grounded evidence provides useful supervision. Removing the process component yields a comparable average drop of 2.88 points, showing the importance of judging whether the trajectory follows a valid multimodal analysis procedure. The two components are complementary: evidence probing brings clear gains on Chart-MRAG and WikiMixQA, where grounded chart or

Model	FCMR	Chart-MRAG	WikiMixQA	CharXiv	DABench
<i>Advanced Model-based Agents</i>					
GPT-5.2	65.27	45.76	82.03	73.20	78.21
Claude-4.6-Sonnet	62.61	38.79	79.72	71.50	85.99
Gemini-3-Pro	66.94	41.78	80.18	74.10	80.54
DeepSeek-V3.2 [†]	13.31	11.05	70.97	27.70	81.32
Qwen3-VL-235B-A22B	14.14	22.34	53.45	50.30	76.26
<i>Data Analysis Agents</i>					
DataMind [†]	11.34	5.48	39.17	19.80	78.60
DeepAnalyze [†]	7.98	5.23	18.89	11.90	74.71
TableGPT2 [†]	12.32	6.06	40.55	23.50	46.69
TableGPT-R1 [†]	14.85	9.97	49.31	27.00	57.97
<i>Our Agents</i>					
Qwen3-VL-8B (Agentic)	12.46	17.11	49.77	43.70	68.87
OmniDA (Ours)	41.46	38.87	67.28	56.20	84.05

Table 1: Overall performance of different agents on five benchmarks. Models marked with [†] are text-only models; on chart-containing benchmarks, chart images are replaced with linearized textual tables generated by GPT-4o-mini.

Model	FCMR	Chart-MRAG	WikiMixQA
OmniDA (Ours)	41.46	38.87	67.28
w/o evidence	38.80	35.22	64.52
w/o process	37.39	36.13	65.44
w/o both	33.75	31.56	64.06

Table 2: Ablation results of the evidence probing and process rubric components.

Model	FCMR	Chart-MRAG	WikiMixQA
OmniDA (Ours)	41.46	38.87	67.28
w/o process	37.39	36.13	65.44
Holistic Judge	36.97	35.80	65.44

Table 3: Process-rubric design ablation. Task-specific rubrics outperform both removing process supervision and replacing them with a generic holistic judge.

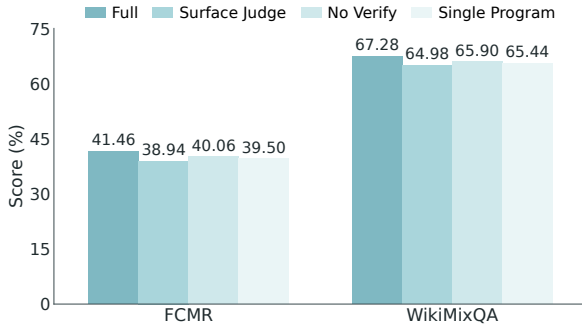


Figure 4: Ablation of evidence probing designs.

textual facts are crucial, while the process rubric is especially helpful on FCMR, which requires more complex cross-modal reasoning. Removing both components results in the largest degradation, with an average drop of 6.08 points, confirming the benefit of jointly supervising evidence recovery and procedural validity.

Effect of evidence probing design. Figure 4 further examines key design choices in the evidence component. Replacing prefix-based cloze probing with a surface trajectory judge leads to the largest

degradation on both FCMR and WikiMixQA, indicating that directly probing whether the agent has recovered the evidence is more effective than judging whether the evidence is merely mentioned in the trajectory. Removing sandbox verification consistently lowers performance, showing that executable grounding helps improve the reliability of the evidence checklist. Extracting evidence from a single solution program causes a larger drop, suggesting that multi-program aggregation produces more stable and essential evidence items.

Effect of task-specific process rubrics. Table 3 examines whether a generic holistic trajectory judge can replace our task-specific process rubrics. Here, the holistic judge refers to a task-agnostic evaluator that assigns an overall judgment to the entire reasoning trajectory, without decomposing the process into task-specific criteria or reasoning axes. The holistic judge fails to improve over removing the process reward, performing worse on FCMR and Chart-MRAG and tying on WikiMixQA. This suggests that coarse trajectory-level judgments pro-

Model	DAComp-DA				
	Acc.	Ins.	Read.	Depth	DA Score
<i>Advanced Model-based Agents</i>					
o3	29.54	23.95	25.24	23.81	28.20
Qwen3-235B-A22B	13.11	21.50	3.64	1.56	13.25
<i>Our Agents</i>					
Qwen3-VL-8B	3.94	6.47	0.48	0.16	5.11
OmniDA (Ours)	12.76	25.18	7.24	4.56	15.57

Table 4: Performance of different agents on DAComp-DA: Open-ended Data Analysis across five dimensions, including accuracy (Acc.), insightfulness (Ins.), readability (Read.), analytical depth (Depth), and DA score.

vide weak or even misaligned supervision, as they do not capture the task-specific reasoning steps required by different multimodal benchmarks. In contrast, our task-specific, axis-structured rubrics consistently outperform the holistic judge by 4.49, 3.07, and 1.84 points on FCMR, Chart-MRAG, and WikiMixQA, respectively. These results highlight the importance of tailoring process supervision to the reasoning structure of each task rather than relying on generic trajectory-level assessment.

3.4 Generalization to open-ended data analysis

To examine whether OmniDA generalizes beyond short-answer data-analysis tasks, we further evaluate it on DAComp-DA, an open-ended data analysis benchmark (Lei et al., 2025) where agents are required to produce analytical reports rather than single final answers. As shown in Table 4, OmniDA substantially improves over the Qwen3-VL-8B baseline across all evaluation dimensions, increasing the DA Score from 5.11 to 15.57. The gains are especially clear in accuracy and insightfulness, suggesting that our training pipeline improves not only answer correctness but also the agent’s ability to derive useful analytical observations from data. OmniDA also outperforms the much larger Qwen3-235B-A22B agent in DA Score and achieves comparable or better insightfulness and readability scores. These results indicate that evidence-aware process supervision learned from synthetic multimodal analysis tasks can transfer to more open-ended data-analysis scenarios.

4 Related Work

4.1 Data Analysis Agents

Data-analysis agents are expanding automated analytics from passive question answering to end-to-

end workflows involving data understanding, planning, tool execution, and grounded conclusion generation (Luo et al., 2026; Wu et al., 2026b; Zhu et al., 2026). Representative efforts include DataMind (Qiao et al., 2025), DeepAnalyze (Zhang et al., 2025a), and TableGPT-R1 (Yang et al., 2025a), which advance agent training for data analysis, autonomous data science, and complex tabular reasoning. Beyond table-centric settings, multimodal data analysis has gained increasing attention, with benchmarks such as WikiMixQA (Foroutan et al., 2025), DataCross (Qi et al., 2026), and FCMR (Kim et al., 2025) evaluating reasoning over tables, charts, and textual sources. Existing resources remain small and mostly evaluation-oriented due to the cost of collecting aligned multimodal data and verifiable questions. We address this with an automated synthesis pipeline that creates large-scale multimodal environments and verifiable training tasks.

4.2 Rubric-Based Process Rewards

Reward design plays a crucial role in post-training LLMs, as reward signals guide which behaviors and reasoning patterns are reinforced during optimization (Chen et al., 2026a,b; Wu et al., 2026a). In particular, process rewards provide denser supervision than outcome-only rewards by evaluating intermediate reasoning steps or actions rather than only final answers (Zheng et al., 2025; Zhang et al., 2025b; Wang et al., 2025). Recent work has instantiated such rewards as rubrics, enabling interpretable and fine-grained assessment (Huang et al., 2025; Sheng et al., 2026). AutoRubric (Jia et al., 2025) extracts reasoning checkpoints from successful multimodal trajectories, and RaR (Gunjal et al., 2025) uses structured rubrics as RL rewards beyond directly verifiable answers. We extend this line to multimodal data-analysis agents by combining grounded evidence probing with trajectory-level process rubrics.

5 Conclusion

In this work, we propose OmniDA, a training recipe for grounded multimodal data-analysis agents. It combines scalable multimodal data synthesis with Evidence-Aware Process Reward (EAPR), which supervises both evidence recovery and procedural validity beyond outcome-only feedback. Experiments across diverse data-analysis benchmarks demonstrate the effectiveness of our framework.

Limitations

This work has two limitations that suggest directions for future work. First, our training and evaluation are currently limited to data-analysis tasks with verifiable outcomes, where answers can be checked by rule-based or executable verification. Extending the framework to more open-ended data-analysis tasks, such as long-form analytical reporting, remains an important future direction. Second, due to computational constraints, we have not trained or validated OmniDA on larger open-source base models. Future work can further examine how the proposed data synthesis and reward framework scale with stronger model backbones.

Ethical Considerations

All data used in this work are derived from publicly available sources. No proprietary or confidential information is used in our data construction, training, or evaluation.

Acknowledgments

This work was supported by the “Pioneer” R&D Program of Zhejiang (No. 2024C01021), the National Regional Innovation and Development Joint Fund (No. U24A20254), the National Natural Science Foundation of China (NSFC) (No. 62502439), the Ningbo Key Research and Development Program (No. 2025Z190), and the Fundamental Research Funds for the Central Universities (No. 226-2025-00085). This work was also supported by Ant Group. Generative AI tools were used to assist with code debugging and language polishing, including grammar correction and style refinement.

References

- Anthropic. 2026. Claude Sonnet 4.6 System Card. <https://www.anthropic.com/claude-sonnet-4-6-system-card>.
- Hao Chen, Zhanming Shen, Liyao Li, Yanyu Chen, Xuhang Zhu, Xiaomeng Hu, Qi Zhang, Ru Peng, Xiaoyu Shen, Haobo Wang, and Junbo Zhao. 2026a. Momentum for reasoning: Dense intrinsic signals in policy optimization. *CoRR*, abs/2606.08815.
- Yanyu Chen, Jiyue Jiang, Dianshi Yu, Zheng Wu, Jiahong Liu, Jiaming Han, Xiao Guo, Jinhu Qi, Yu Li, Yifei Zhang, and Irwin King. 2026b. LC-ERD: mining latent logic for self-evolving reasoning via consistency-regulated reward decomposition. *CoRR*, abs/2605.24005.
- DeepSeek-AI. 2025. Deepseek-v3.2: Pushing the frontier of open large language models. *CoRR*, abs/2512.02556.
- Negar Foroutan, Angelika Romanou, Matin Ansari-pour, Julian Martin Eisenschlos, Karl Aberer, and Rémi Le-bret. 2025. Wikimixqa: A multimodal benchmark for question answering over tables and charts. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Findings of ACL, pages 24941–24958. Association for Computational Linguistics.
- Google DeepMind. 2026. Gemini 3 pro model card. Last updated: May 2026.
- Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Bing Liu, and Sean Hendryx. 2025. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *CoRR*, abs/2507.17746.
- Xueyu Hu, Ziyu Zhao, Shuang Wei, Ziwei Chai, Qianli Ma, Guoyin Wang, Xuwu Wang, Jing Su, Jingjing Xu, Ming Zhu, Yao Cheng, Jianbo Yuan, Jiwei Li, Kun Kuang, Yang Yang, Hongxia Yang, and Fei Wu. 2024. Infiagent-dabench: Evaluating agents on data analysis tasks. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, Proceedings of Machine Learning Research, pages 19544–19572. PMLR.
- Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, Xijun Gu, Peiyi Tu, Jiaxin Liu, Wenyu Chen, Yuzhuo Fu, Zhiting Fan, Yanmei Gu, Yuanyuan Wang, Zhengkai Yang, and 2 others. 2025. Reinforcement learning with rubric anchors. *CoRR*, abs/2508.12790.
- Mengzhao Jia, Zhihan Zhang, Ignacio Cases, Zheyuan Liu, Meng Jiang, and Peng Qi. 2025. Autorubric-r1v: Rubric-based generative rewards for faithful multimodal reasoning. *CoRR*, abs/2510.14738.
- Seunghye Kim, Changhyeon Kim, and Taeuk Kim. 2025. FCMR: robust evaluation of financial cross-modal multi-hop reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 23352–23380. Association for Computational Linguistics.
- Fangyu Lei, Jinxian Meng, Yiming Huang, Junjie Zhao, Yitong Zhang, Jianwen Luo, Xin Zou, Ruiyi Yang, Wenbo Shi, Yan Gao, Shizhu He, Zuo Wang, Qian Liu, Yang Wang, Ke Wang, Jun Zhao, and Kang Liu. 2025. Dacomp: Benchmarking data agents across the full data intelligence lifecycle. *CoRR*, abs/2512.04324.
- Fangyu Liu, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhui Chen, Nigel Collier, and Yasemin Altun. 2023. Deplot: One-shot visual language reasoning by plot-to-table translation. In *Findings of the Association for Computational Linguistics*:

- ACL 2023, Toronto, Canada, July 9-14, 2023, Findings of ACL, pages 10381–10399. Association for Computational Linguistics.
- Yuyu Luo, Guoliang Li, Ju Fan, and Nan Tang. 2026. [Data agents: Levels, state of the art, and open problems](#). In *Companion of the International Conference on Management of Data, SIGMOD Companion 2026, Bengaluru, India, 31 May 2026 - 5 June 2026*, pages 571–579. ACM.
- OpenAI. 2025. Gpt-5.2 system card. <https://openai.com/index/gpt-5-system-card-update-gpt-5-2/>.
- Ruyi Qi, Zhou Liu, and Wentao Zhang. 2026. [Dat-across: A unified benchmark and agent framework for cross-modal heterogeneous data analysis](#). *CoRR*, abs/2601.21403.
- Shuofei Qiao, Yanqiu Zhao, Zhisong Qiu, Xiaobin Wang, Jintian Zhang, Zhao Bin, Ningyu Zhang, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2025. [Scaling generalist data-analytic agents](#). *CoRR*, abs/2509.25084.
- Leheng Sheng, Wenchang Ma, Ruixin Hong, Xiang Wang, An Zhang, and Tat-Seng Chua. 2026. [Reinforcing chain-of-thought reasoning with self-evolving rubrics](#). *CoRR*, abs/2602.10885.
- Aofeng Su, Aowen Wang, Chao Ye, Chen Zhou, Ga Zhang, Gang Chen, Guangcheng Zhu, Haobo Wang, Haokai Xu, Hao Chen, Haoze Li, Haoxuan Lan, Jiaming Tian, Jing Yuan, Junbo Zhao, Junlin Zhou, Kaizhe Shou, Liangyu Zha, Lin Long, and 14 others. 2024. [Tablegpt2: A large multimodal model with tabular data integration](#). *CoRR*, abs/2411.02059.
- Qwen Team. 2025. [Qwen3-v1 technical report](#). *CoRR*, abs/2511.21631.
- Weiyun Wang, Zhangwei Gao, Lianjie Chen, Zhe Chen, Jinguo Zhu, Xiangyu Zhao, Yangzhou Liu, Yue Cao, Shenglong Ye, Xizhou Zhu, Lewei Lu, Haodong Duan, Yu Qiao, Jifeng Dai, and Wenhai Wang. 2025. [Visualprm: An effective process reward model for multimodal reasoning](#). *CoRR*, abs/2503.10291.
- Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, Alexis Chevalier, Sanjeev Arora, and Danqi Chen. 2024. [Charxiv: Charting gaps in realistic chart understanding in multimodal llms](#). In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.
- Xingjian Wu, Junlin Liu, Xingchen Liu, Xuhang Zhu, Jianing Wang, Linsen Guo, Xiaoyu Li, Xuezhi Cao, and Xunliang Cai. 2026a. [Contrastive reinforced policy optimization via privileged self-distillation](#). *CoRR*, abs/2607.28026.
- Xingjian Wu, Xuhang Zhu, Xingchen Liu, Junlin Liu, Jianing Wang, Linsen Guo, Xiaoyu Li, Xuezhi Cao, and Xunliang Cai. 2026b. [Clawtrack: Towards trace-level evaluation and improvement of real-world autonomous agents](#). *CoRR*, abs/2607.28037.
- Saisai Yang, Qingyi Huang, Jing Yuan, Liangyu Zha, Kai Tang, Yuhang Yang, Ning Wang, Yucheng Wei, Liyao Li, Wentao Ye, Hao Chen, Tao Zhang, Junlin Zhou, Haobo Wang, Gang Chen, and Junbo Zhao. 2025a. [Tablegpt-r1: Advancing tabular reasoning through reinforcement learning](#). *CoRR*, abs/2512.20312.
- Yuming Yang, Jiang Zhong, Li Jin, Jingwang Huang, Jingpeng Gao, Qing Liu, Yang Bai, Jingyuan Zhang, Rui Jiang, and Kaiwen Wei. 2025b. [Benchmarking multimodal RAG through a chart-based document question-answering generation framework](#). *CoRR*, abs/2502.14864.
- Shaolei Zhang, Ju Fan, Meihao Fan, Guoliang Li, and Xiaoyong Du. 2025a. [Deepanalyze: Agentic large language models for autonomous data science](#). *CoRR*, abs/2510.16872.
- Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2025b. [The lessons of developing process reward models in mathematical reasoning](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10495–10516, Vienna, Austria. Association for Computational Linguistics.
- Congming Zheng, Jiachen Zhu, Zhuoying Ou, Yuxiang Chen, Kangning Zhang, Rong Shan, Zeyu Zheng, Mengyue Yang, Jianghao Lin, Yong Yu, and Weinan Zhang. 2025. [A survey of process reward models: From outcome signals to process supervisions for large language models](#). *CoRR*, abs/2510.08049.
- Zhipu AI. 2026. Glm-4.7-flash. <https://docs.bigmodel.cn/cn/guide/models/free/glm-4.7-flash>.
- Xuhang Zhu, Xiu Tang, Sai Wu, Yixuan Tang, Haobo Wang, Chang Yao, Ruichen Xia, and Gang Chen. 2026. [Bridge-sql: Bridging single- and multi-turn text-to-sql via preference-aligned question rewriting](#). In *Database Systems for Advanced Applications - 31st International Conference, DASFAA 2026, Jeju, South Korea, April 27-30, 2026, Proceedings, Part IV*, volume 16538 of *Lecture Notes in Computer Science*, pages 120–135. Springer.

A Analytical Reasoning Primitives

To support flexible question synthesis, we define seven analytical reasoning primitives for multimodal data analysis. These primitives are grounded in core relational data-analysis operations, but are extended beyond structured tables to heterogeneous sources including tables, charts, and textual reports.

Primitive	Definition	Representative Operations	Multimodal Examples
Extraction	Locate and read specific information from a source without performing computation.	Cell lookup, row/column reading, chart value reading, axis/legend identification, text fact or rule extraction.	Read a product’s revenue from a table; read a bar value from a chart; extract an industry benchmark or policy rule from a report.
Selection	Identify a subset of entities or records satisfying one or more conditions.	Numeric filtering, categorical filtering, range filtering, temporal filtering, membership filtering, deduplication.	Select products whose satisfaction exceeds a reported benchmark; select regions appearing in a text-defined priority list.
Aggregation	Apply statistical functions to a set of values, optionally grouped by an entity or category.	Sum, mean, count, max, min, median, standard deviation, percentile, correlation, group-by aggregation.	Compute average sales from a table; count chart categories above a threshold; aggregate values for entities selected by a textual rule.
Arithmetic	Derive new values through numerical computation or formula application.	Addition, subtraction, multiplication, division, percentage, ratio, growth rate, difference, rounding, formula application.	Compute profit margin from table values; calculate deviation from a chart trend; apply a formula defined in text to values from other modalities.
Comparison	Evaluate relative magnitude, equality, or truth conditions between values or statements.	Greater/less-than checks, threshold comparison, range check, pairwise comparison, claim verification.	Compare table revenue with a textual target; check whether a chart value exceeds the platform average stated in a report.
Ranking	Order entities according to a metric or identify top/bottom performers.	Ascending/descending sorting, Top- k , Bottom- k , rank position, argmax/argmin, percentile rank.	Rank products by table revenue; identify the top regions by charted growth; find the lowest-risk item among text-filtered candidates.
Set Operation	Combine or compare entity sets using set-theoretic operations.	Intersection, union, difference, membership check, subset check.	Intersect products selected from a chart with those satisfying a table condition; remove entities excluded by a textual policy.

Table 5: Definitions and examples of the seven analytical reasoning primitives used for multimodal question synthesis.

Table 5 provides their definitions, representative operations, and multimodal examples.

B Benchmark Details and Evaluation Setup

We evaluate on five data-analysis benchmarks covering complementary modality settings. Here, T , C , and X denote table, chart, and textual sources, respectively.

FCMR. We use the hard split of this statement-verification benchmark (Kim et al., 2025), consisting of 714 questions over textual reports, tables, and charts ($T+C+X$). Predictions are scored by rule-based exact match over the canonicalized set of true statements.

WikiMixQA. We use the oracle table-chart subset of this Wikipedia-grounded QA benchmark (Foroutan et al., 2025), containing 217 multiple-choice questions over tables and charts ($T+C$).

Chart-MRAG. We use all 1,204 converted samples from this text-chart reasoning benchmark (Yang et al., 2025b), requiring paired tex-

tual and chart evidence ($C+X$). Final answers are evaluated by a model-based judge.

CharXiv. We evaluate on the reasoning validation split of this scientific chart reasoning benchmark (Wang et al., 2024), which contains 1,000 chart-only questions (C). We use the official judge-based grading protocol.

DABench. We use all 257 available questions from this table-centric data-analysis benchmark (Hu et al., 2024), which are grounded in CSV files (T). A model-based judge is applied, with tolerance-based matching for numeric answers and exact matching for multiple-choice answers.

Evaluation Protocol. For closed-answer benchmarks, we use rule-based scoring: exact option matching for WikiMixQA and exact set matching for FCMR. For DABench, Chart-MRAG, and CharXiv, we use a model-based judge. For text-only baselines, charts are converted into question-blind textual descriptions before inference.

C Prompts

The prompts are presented in Figure 5, 6 and 7.

Prompt for Question-Blind Chart Linearization

Task Description

You convert chart images into textual representations for a text-only QA model. You are **question-blind**: you will only see the chart image, not the downstream question or answer. Do not infer or solve any hidden question. Faithfully describe the information presented in the chart so that the resulting text can serve as a substitute for the original visual input.

Input

Chart image: {chart_image}

Your Task

Produce a faithful Markdown representation of the chart. Capture the chart structure, labels, series, and data values as accurately as possible from the visible content. Preserve information that may be useful for downstream reasoning, while avoiding unsupported inference. If any content is visually ambiguous or unreadable, explicitly indicate the uncertainty rather than guessing.

Output Format

Return Markdown only, following the structure below:

Linearized Chart Representation

Chart type:

Title:

Layout / subplots:

- description

Axes and labels:

- x-axis:

- y-axis:

Legend / series:

- description

Data / visual observations:

- values, relationships, and patterns

Uncertainties:

- mention any ambiguous or unreadable content

Important Rules

- Be **question-blind**: do not assume any downstream question or answer.
- Describe the chart as generally useful evidence, not as a solution to a hidden task.
- Preserve the information presented in the chart as faithfully as possible.
- Include labels, series, and numerical values when they can be reliably read from the chart.
- Do not infer values, relations, or facts that are not supported by the visible chart content.
- Explicitly indicate uncertainty when chart content cannot be reliably determined.
- Return Markdown only and follow the specified structure.

Figure 5: Prompt for question-blind chart linearization used to construct textual chart representations for text-only baselines.

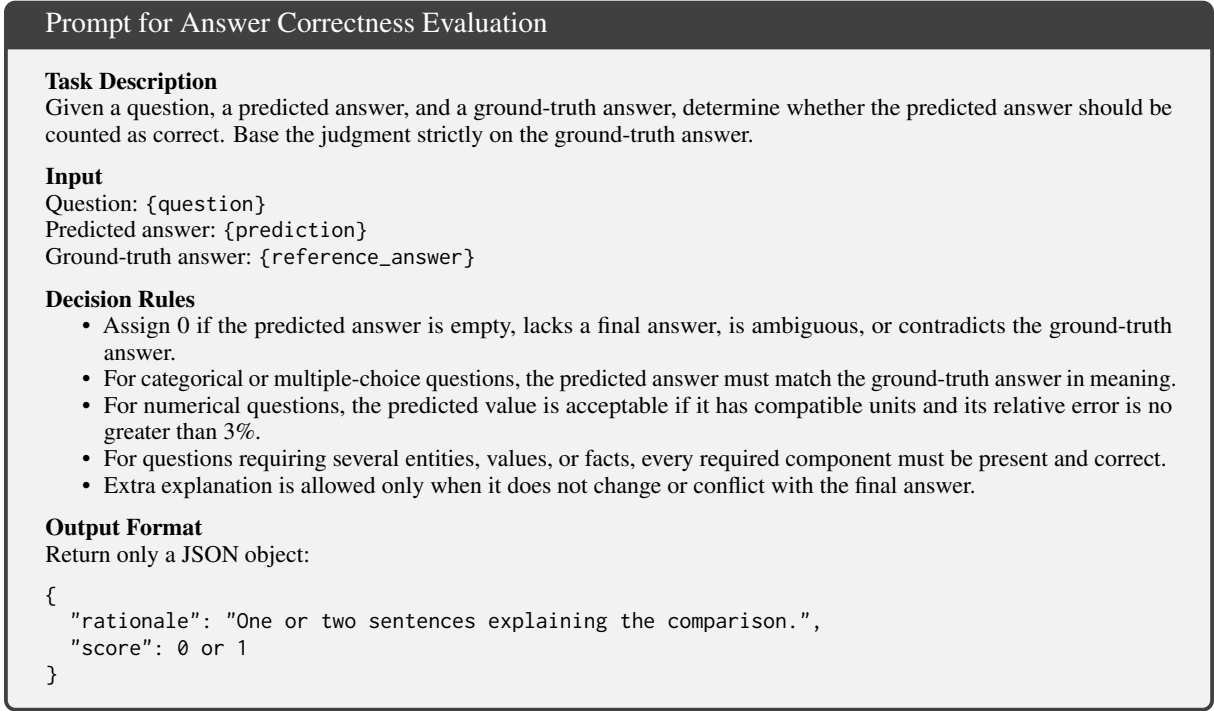


Figure 6: Prompt for answer correctness evaluation used by the model-based judge.

Prompt for Multimodal Data-Analysis ReAct

Task Description

You are given multiple sources of evidence, including a text document, a table, and an image. Answer the question using only the provided evidence.

Input

Question: {question}

Text evidence: {text_evidence}

Table evidence: The table is available as a CSV file at {table_path}. You may inspect it with Python.

Image evidence: The image is available in the execution environment as {image_reference}. You may inspect or process it with Python when visual evidence is needed.

Instructions

- Reason carefully about the relevant information from all provided sources.
- Use Python when useful to inspect the table, compute values, or analyze the image.
- Do not use external knowledge.
- If the provided evidence is insufficient, do not infer unsupported information.
- Provide a concise final answer that directly answers the question.

Response Format

<think>

Describe what evidence is needed and how you will verify the answer.

</think>

<code>

Optional Python code for inspecting the CSV table, performing calculations, or processing the image.

</code>

<observation>

Execution results or visual observations returned by the environment.

</observation>

[Repeat the think-code-observation cycle as needed.]

<think>

Use the observations and provided evidence to derive the final answer.

</think>

<answer>

The final answer.

</answer>

Figure 7: Prompt for multimodal data-analysis ReAct reasoning over textual, tabular, and visual evidence.