CARTA: COMPOSING DEEP-SEARCH TASKS WITH RELATIONAL CONTRACTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Training deep-search agents requires tasks whose web evidence jointly constrains the answer through three relational patterns: *depth*, where clues chain through intermediate entities; *branching*, where multiple conditions converge on one entity; and *coupling*, where distinct relational paths connect the same pair of entities to form a loop. Longer chains alone need not introduce branching or coupling. Moreover, realizing a graph as a question can omit relations or break identity bindings without invalidating the reference answer, making answer correctness insufficient to establish structural fidelity. We introduce **CARTA** (*Compositional Answer-Rooted Task Assembly*), which treats relational structure as a checkable specification. Three operations (EXTEND, BRANCH, COUPLE) compose answer-rooted graphs, which compile into a *Relational Contract* of relation obligations and identity constraints. The contract guides question realization, localizes violations for revision, and scores teacher trajectories for supervision selection. With SFT alone, matched example counts, and a shared teacher for trajectory generation, models trained on CARTA data outperform those trained on recent synthesis corpora across three benchmarks at two backbone scales. Controlled experiments show that mixing composition levels outperforms either level alone; contract-guided realization raises whole-query conformance from 72.9% to 86.3% and improves downstream accuracy by 4.1–6.6 percentage points; and contract-based trajectory selection outperforms random selection among answer-correct rollouts.Together, these results show that explicitly specifying and checking relational structure provides a useful supervision signal for training stronger deep-search agents.

1 INTRODUCTION

Deep-search agents retrieve and integrate evidence scattered across the open web, and synthetic tasks provide a scalable source of training data. A central design choice is how a task's clues jointly constrain the answer: through chains of intermediate entities (*depth*), multiple conditions on one candidate (*branching*), or distinct relational paths forming loops (*coupling*).

Figure 1 illustrates how these patterns combine in a BrowseComp question. The census, sample, target paper, and requested publication form a chain; authorship, method, and sample constraints converge on the target paper; and shared authorship links two papers whose shared-keyword relation closes a loop in the underlying undirected graph. The authorship and keyword clues must refer to the same pair of papers, connecting the constraints across paths. Together, these patterns call for integrating intermediate findings, reconciling multiple conditions, and maintaining entity consistency across paths, providing training opportunities for coordinated evidence integration. Extending a chain alone need not introduce converging constraints or cross-path bindings, making relational composition a design variable beyond chain length.

Recent methods synthesize search tasks through graph-based construction, formal task representations, and difficulty control, with validation of generated tasks (Chu et al., 2026; Du et al., 2026b; Lu et al., 2025; Tao et al., 2026; Deng et al., 2026; Zhao et al., 2026). Building on these approaches, we study how to compose relational structures and carry them through question realization and supervision selection using an explicit contract.

Preserving that structure is a separate challenge from constructing it. Graph-to-question generation can omit or alter a relation, break an entity binding, or introduce an additional constraint, changing

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

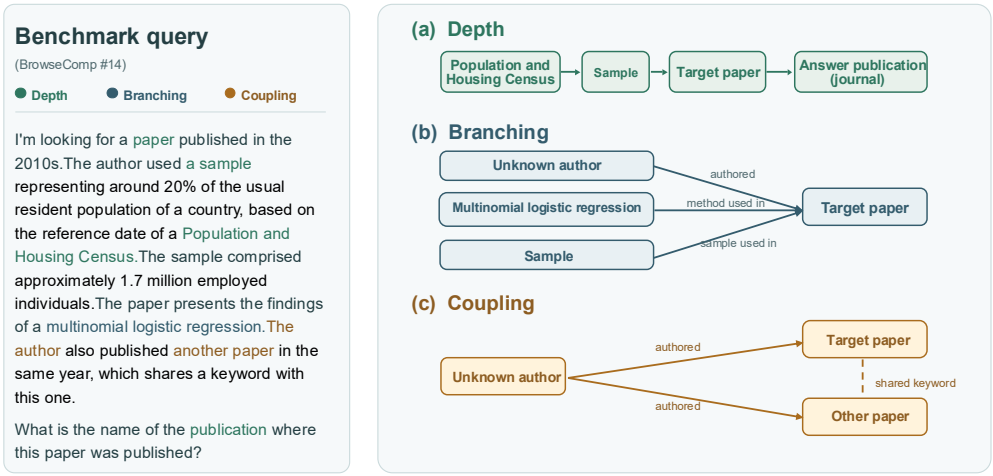


Figure 1: **Relational composition in a deep-search question.** A BrowseComp question (left) and three views highlighting depth, branching, and coupling (right). Repeated labels denote the same entities across panels; the dashed line denotes a symmetric shared-keyword relation. The views show selected question constraints, not a prescribed search trajectory.

what the task asks an agent to resolve. Answer validation alone does not establish structural fidelity: a reference answer may remain compatible with a question after a condition is omitted. In our audit of 240 source graphs, a strong graph-to-query baseline achieves 72.9% whole-query structural conformance despite receiving the full relation inventory, shared entity identifiers, and explicit preservation instructions. This observed gap motivates checking whether the intended relations and bindings survive realization, alongside validating the answer.

We introduce **CARTA** (*Compositional Answer-Rooted Task Assembly*), which treats relational structure as a checkable specification (Figure 2). Three operations—EXTEND, BRANCH, and COUPLE—compose answer-rooted graphs that combine depth, branching, and coupling. The selected structure compiles into a *Relational Contract* of relation obligations and identity constraints. This contract guides question realization and provides the reference for auditing the resulting text, localizing violations for targeted revision and rechecking. The same obligations score teacher trajectories for supervision selection, allowing the constructed structure to inform both task generation and the choice of training demonstrations.

With SFT alone, matched example counts, and trajectories regenerated by a common teacher, CARTA outperforms REDSearcher and OpenSeeker-v1 on BrowseComp, BrowseComp-ZH, and xbench-DeepSearch-2510 at both 4B and 9B scales. Further experimental analysis shows that mixing composition levels outperforms either level alone. On matched source graphs, contract-guided realization raises whole-query conformance from 72.9% to 86.3% and improves downstream accuracy by 4.1–6.6 percentage points. A separate comparison shows that selecting rollouts by contract score outperforms random selection among answer-correct trajectories. Together, these results support relational composition and its preservation as explicit design objectives for deep-search training data.

Our contributions are threefold. **Controllable relational composition:** three operations (EXTEND, BRANCH, COUPLE) compose depth, branching, and coupling within answer-rooted task graphs, with a characterization of their reachable graph family. **Contract-guided realization and supervision selection:** an explicit specification supports relation and identity checks in generated questions, and its obligations guide selection among answer-correct teacher trajectories. **Isolating realization from composition:** paired-source-graph experiments measure downstream effects at fixed structure, while separate audits quantify residual structural and factual errors.

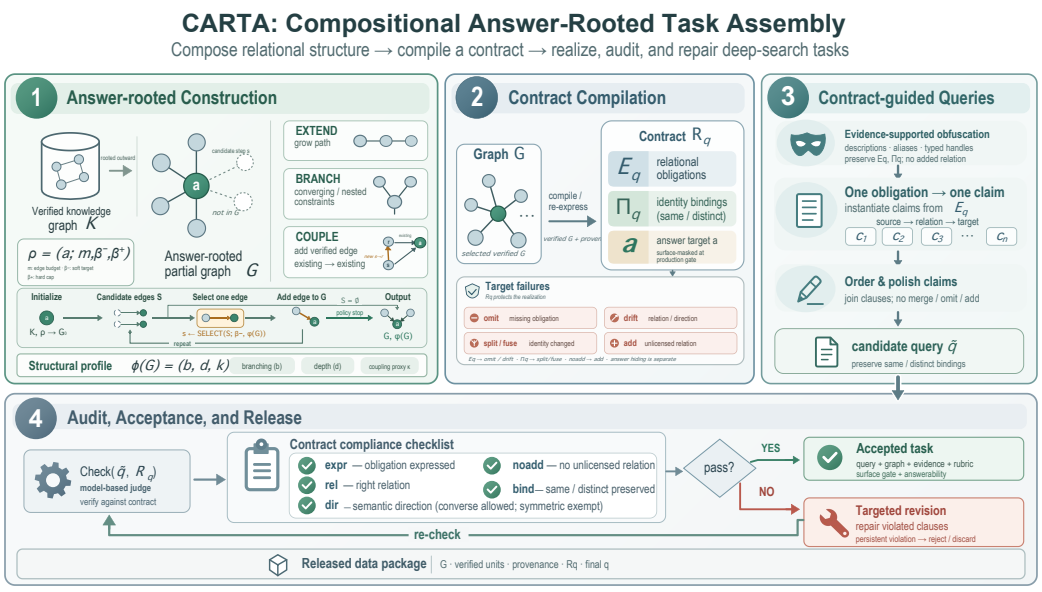


Figure 2: **Overview of CARTA.** Left: construction composes an answer-rooted graph from three primitives (EXTEND, BRANCH, COUPLE). Middle: the selected structure is compiled into a relational contract $R_q = (E_q, \Pi_q, a)$—relation obligations, identity constraints, and the withheld answer. Right: the realized query is audited against the contract, with each violation localized to the obligation that broke.

2 CARTA: Composing and Checking Relational Structure

The intended structure is at risk twice—in how it is composed and in how it is expressed. A realizer mapping a graph straight to a query, $q \sim P_{\mathrm{LLM}}(q \mid G)$, leaves the second loss unobserved: the query can omit an obligation, drift a relation or its direction, split or fuse entities that must stay identical or distinct, or add an unlicensed constraint—all while the reference answer still checks out. CARTA composes the structure deliberately, then audits the realized query for these corruptions, in three parts: a construction language (§2.1), a contract recording what realization can corrupt (§2.2), and an audit against it (§2.3).

Notation. Construction starts from a Wikidata-based knowledge graph (Vrandečić & Krötzsch, 2014). Relations selected for task construction are checked against source pages, and additional relations discovered through web search are incorporated after verification. We denote the resulting graph of verified relations by $\mathcal{K}$. A *construction request* $\rho = (a; m, \beta^-, \beta^+)$ fixes a candidate answer a, an edge budget m, and bounds $\beta^- \preceq \beta^+$ on the profile $\phi(G) = (b, d, \kappa)$: peak in-degree b, longest relational path into the answer d, and treewidth of the undirected projection κ.

2.1 The Structural Object Language

All graphs are answer-rooted: edges point *toward* the answer, so every node has a directed path to a and a itself has no outgoing edge. Construction proceeds *outward* from a, each step adding one edge $(w \to u)$ whose target u already belongs to the partial structure, so the structure stays answer-connected throughout; graphs are simple, and we write V for the current node set and $\mathrm{indeg}(u)$ for the in-degree of u within it. The coupling coordinate κ tracks cross-path coupling (Chu et al., 2026).

Admissible structures. Construction takes the two profile bounds asymmetrically. The hard bound β^+ defines the admissible structure space

$$\begin{aligned} \mathcal{H}_{\mathcal{K}}(\rho) = \{H : H \text{ acyclic, } a\text{-rooted}, \\ H \hookrightarrow \mathcal{K},\ |E(H)| \le m, \\ \phi(H) \preceq \beta^+\} \end{aligned} \tag{1}$$

where $H \hookrightarrow \mathcal{K}$ means every entity and edge of H is an already verified unit of $\mathcal{K}$, so m bounds the evidence atoms a task rests on, not the relations a realizer may propose. All three coordinates are monotone under subgraph inclusion—peak in-degree and longest path cannot decrease, treewidth is minor-monotone—so a coordinate past β^+ is unrecoverable, and the ceiling is enforced step by step through the admissible step set

$$\begin{aligned}\mathcal{A}(G,\rho) = \{\, (w \to u) : u \in V,\ & (w \to u) \notin E(G),\\ & (w \to u) \in E(\mathcal{K}),\\ & G + (w \to u) \text{ acyclic},\\ & |E(G)| < m,\\ & \phi\big(G + (w \to u)\big) \preceq \beta^+ \,\}.\end{aligned} \tag{2}$$

Monotonicity ensures β^+ never blocks a compliant target mid-construction, and gated by an upper-bound estimate of treewidth, $\mathcal{A}$ admits no violating step (§B.4). The soft bound β^- does not restrict $\mathcal{A}$: it is the growth target of the step policy—bounding path length for depth, steering operation choice for branching and coupling—so a step infeasible in $\mathcal{K}$ leaves the realized profile $\widehat{\beta} = \phi(G)$ short of β^-, reported as realized.

Construction primitives. Every step in $\mathcal{A}(G,\rho)$ is exactly one of three primitives (Figure 2, left):

$$\begin{aligned}&\textsc{Extend}: \ w \notin V,\ \ \mathrm{indeg}(u) = 0,\\ &\textsc{Branch}: \ w \notin V,\ \ \mathrm{indeg}(u) \geq 1,\\ &\textsc{Couple}: \ w \in V \setminus \{a\},\ \ w \neq u,\ \ u \not\rightsquigarrow w,\end{aligned} \tag{3}$$

where $u \not\rightsquigarrow w$ means no directed path runs from u to w. The three COUPLE conditions preserve the answer's lack of an outgoing edge, the absence of self-loops, and acyclicity; EXTEND and BRANCH need no guard, since a new source cannot close a cycle. Each expands differently: EXTEND lengthens a dependency path by one mediating relation; BRANCH adds a converging predecessor that pins one entity, and away from the answer nests one subproblem in another; COUPLE links two present entities, closing a cycle that ties two separable paths through a bridge entity (Yang et al., 2018)—a reuse the contract later records as an identity requirement (§2.2). It is also the only primitive that can raise the coupling coordinate.

Lemma 1 (Coupling arises exactly from COUPLE)**.** *Let G be built from $(\{a\}, \emptyset)$ by the primitives of Eq. equation 3, with $|E(G)| \geq 1$. Then $\kappa(G) \geq 2$ if and only if at least one* COUPLE *step was applied; otherwise the undirected projection of G is a tree and $\kappa(G) = 1$.*

Appendix B.3 carries one task through construction, its contract, and realization; the proof is in §B.4.

Reachability. Let $\mathcal{C}_{\mathcal{K}}(\rho)$ be the structures reachable from a by the three operations under ρ. Admissibility gives $\mathcal{C}_{\mathcal{K}}(\rho) \subseteq \mathcal{H}_{\mathcal{K}}(\rho)$; the reverse is constructive.

Proposition 1 (Reachability of the construction language)**.** *Let H be answer-rooted and acyclic with $H \hookrightarrow \mathcal{K}$, $|E(H)| \leq m$, and $\phi(H) \preceq \beta^+$. Then H is reachable from a by the three construction primitives.*

The two inclusions give $\mathcal{C}_{\mathcal{K}}(\rho) = \mathcal{H}_{\mathcal{K}}(\rho)$ (proof in §B.4): the operators are constructively complete for the constrained space, though the implemented reachable set may be smaller, since the treewidth gate uses an upper-bound estimate of κ. Reachability is stated over the supply $\mathcal{K} = \mathcal{K}_0 \cup E_{\mathrm{confirmed}}$, the initial graph plus edges added by evidence discovery (§2.3), so a request can name a target composition and record whether a realizing instance exists.

2.2 THE RELATIONAL CONTRACT

The object language says which structures can be built, not whether a finished query still expresses one. CARTA closes that gap by compiling the selected, evidence-supported structure into a checkable specification—the *relational contract*

$$R_q = (E_q, \Pi_q, a), \tag{4}$$

which persists as the reference the query is audited against (Figure 2, middle). $E_q = \{(h, r, t)\}$ collects the entity–relation–entity obligations the query must express, each a verified, individually checkable graph unit; shared-attribute obligations are symmetric and carry no direction. $\Pi_q = (\sim, \neq)$ pairs a coreference relation over mentions with a distinctness relation over entities, fixing which mentions denote one entity and which entities stay apart. The answer a, inherited from G, constrains the target alone. Beyond the individual obligations, E_q also determines how they compose—answer-side and intermediate branching, dependency depth, cross-path reuse. The relational contract thus serves as a *structure-carrying rubric* for realization and auditing.

Conformance. A query *conforms* to R_q when it preserves every recorded requirement:

$$\begin{aligned} q \models R_q \iff & \bigwedge_{e \in E_q} \mathrm{expr}(q, e) \\ & \wedge \bigwedge_{e \in E_q} \big(\mathrm{rel}(q, e) \wedge \mathrm{dir}(q, e)\big) \\ & \wedge\ \mathrm{noadd}(q, E_q)\ \wedge\ \mathrm{bind}(q, \Pi_q). \end{aligned} \tag{5}$$

$\mathrm{expr}(q, e)$ holds when q asserts obligation e between the entities it names (against omission); $\mathrm{rel}(q, e)$ and $\mathrm{dir}(q, e)$ when its relation and semantic direction match e, with licensed converses equivalent and $\mathrm{dir} \equiv \top$ for symmetric obligations (against drift); $\mathrm{noadd}(q, E_q)$ when q adds no inter-entity relation beyond E_q (against addition); $\mathrm{bind}(q, \Pi_q)$ when q realizes every required coreference as one entity and keeps required-distinct entities apart (against split or fuse). Each conjunct rules out one corruption. Conformance covers E_q and Π_q; the answer target a is handled by a separate production gate (§2.3).

2.3 Contract-Guided Synthesis

Construction is governed by the request ρ. The selected structure is then compiled into a contract R_q, which serves as the shared reference for realization, audit, and targeted revision.

Constructing under the request. Construction starts from $G = (\{a\}, \emptyset)$ and repeatedly draws a step from $\mathcal{A}(G, \rho)$ (Eq. equation 2) under a staged SELECT policy, until a stopping condition or exhaustion. The reachable structures follow from the operations and hard constraints, not a template (the full loop is in Appendix B.2). COUPLE is the hard case: it constrains both endpoints (Eq. equation 3), so its admissible candidates are scarce and the verified graph may offer none. CARTA then turns to an agentic procedure that proposes entity pairs and gathers evidence for the missing relation, admitting one only once retrieved evidence confirms it under the graph's own verification standard. Confirmed edges join $E_{\mathrm{confirmed}}$, the supply over which §2.1 states reachability; Appendix B.6 covers tool routing, confirmation, shared attributes, and per-graph limits.

Realizing the contract. Realization is where the corruptions actually occur (Appendix B.3 walks them on the running example). CARTA guards against this by realizing R_q in three linked steps, all sharing it as their reference. First, direct entity cues—which can expose the answer or permit single-name lookup (Deng et al., 2026)—give way to evidence-supported descriptions, aliases, typed referring expressions, or symbolic handles, as in prior deep-search synthesis (Team et al., 2025; Lu et al., 2025); the substitution adds no unsupported fact and no relation beyond E_q, weakening identification without changing which entity participates in a relation. The realizer is given these descriptions in place of the non-answer entity names, so no later step can put a name back into the query. Then each obligation is lowered into its own clause, head and tail in subject and object position for directed ones (§B.7). Finally, a fluency pass folds the planned claims into prose without touching the rubric they must satisfy.

Auditing against the contract. The audit tests q directly against R_q. $\mathrm{Check}(q, R_q)$ operationalizes Eq. equation 5 as a model-based judge returning the violated conjuncts Δ, each naming a corruption and localizing it to the affected obligation or binding, not the whole query. Revision edits the corresponding claims and re-audits; a violation that survives revision rejects the query rather than relaxing R_q. As a learned judge rather than a rule-based test, Check estimates conformance rather

Model	Training	BrowseComp	BC-ZH	xbench-2510
Controlled comparison (matched size and examples): Qwen3.5-9B				
No SFT (base)	-	26.6	35.9	31.7
REDSearcher	SFT	40.9	54.3	43.3
OpenSeeker-v1	SFT	44.3	54.7	50.0
CARTA	SFT	**52.4**	**57.0**	**56.0**
Controlled comparison (matched size and examples): Qwen3.5-4B				
No SFT (base)	-	16.9	33.6	30.3
REDSearcher	SFT	37.7	47.8	44.0
OpenSeeker-v1	SFT	39.7	46.4	48.0
CARTA	SFT	**46.6**	**48.4**	**54.0**

Table 1: **Main results on deep-search benchmarks.** Within each block, rows differ only in the 5k-example corpus naming them; all share the §3.1 protocol (details in §B.1), and scores are three-run means ("-": not applicable).

than deciding it. A deterministic surface check then verifies that no withheld name appears in q, completing the answer-hiding component a that conformance leaves to production.

Both gaps are thereby addressed: the object language composes the intended structure instead of sampling what an unconstrained traversal of $\mathcal{K}$ returns, and the audit checks it survives realization instead of inferring survival from a still-valid answer. What reaches training is a task whose relational composition is both chosen and checked.

3 Experiments

We evaluate the downstream utility of CARTA tasks (§3.2), attribute the gain across three pipeline stages—task selection, realization, and rollout selection (§3.3)—and audit the fidelity, answerability, and factual support of the resulting tasks (§3.4).

3.1 Experimental Setup

Data and training. We compare SFT on 5,000 tasks from REDSearcher, OpenSeeker-v1, and CARTA at two backbone sizes, Qwen3.5-9B and Qwen3.5-4B (Qwen Team, 2026). For every condition, trajectories are regenerated with the same DeepSeek-V4-Pro (DeepSeek-AI, 2026) teacher, tools, and decoding, filtered to answer-correct before SFT, so conditions differ only in the task distribution. Within each backbone block, models share an initial checkpoint (itself given no SFT), implementation, and optimization (§B.1); task count and recipe are held fixed, though equal example counts need not imply identical tokens or compute.

CARTA retains each task's constructed graph, contract, and final query (GPT-5.2 for relation verification and realization); difficulty filtering rejects candidates a single checked source page already answers. Search requests exclude benchmark-related keywords or domains to reduce benchmark leakage during synthesis (§B.5).

Evaluation. We evaluate BrowseComp (Wei et al., 2025), BrowseComp-ZH (Zhou et al., 2025), and xbench-DeepSearch-2510 (Xbench-Team, 2025) through a shared deep-search harness with fixed tools, decoding, turn and rollout budgets, judge, and benchmark snapshot. Main scores are three-run means for one checkpoint per condition, measuring decoding and execution variability.

3.2 Downstream Utility of the Synthesized Tasks

Within each controlled block, CARTA is the strongest corpus on all three benchmarks at both scales, reaching 52.4/57.0/56.0 at 9B on BrowseComp/BC-ZH/xbench—over the best matched-data baseline per benchmark by $+8.1/+2.3/+6.0$, with 4B gains of $+6.9/+0.6/+6.0$. Gains are largest on

Ablation	Variant	BrowseComp	BC-ZH	xbench-2510
Task selection (5,000)	low structure	43.0	50.5	48.3
	high structure	46.7	54.6	53.7
	mixed structure	**48.8**	**56.9**	**55.7**
Realization (3,000 pairs)	graph-to-query	40.6	51.4	41.7
	full CARTA	**46.7**	**55.5**	**48.3**
Rollout selection (3,000)	random correct	48.0	55.9	47.3
	Contract-selected	**52.3**	**60.0**	**52.3**

Table 2: One ablation per row group, each at a different pipeline stage (task selection, realization, rollout selection). Accuracy (%), three-run means, Qwen3.5-9B; each group fixes its own cohort, so differences are within-group only, neither comparable across groups nor additive.

BrowseComp at both scales and smallest on BC-ZH (within three points), so the advantage tracks the task distribution rather than model size.

The gain is not longer supervision either: across the three 5k corpora the teacher's search effort is comparable—CARTA marginally lower, if anything—so its advantage is not from longer trajectories or more search per example (per-corpus tool-call counts in Appendix B.8).

3.3 Attributing the Gain

We test these three components separately, each in its own matched cohort, varying one component so accuracies are within-cohort contrasts, not comparable to Table 1.

Structural enrichment at matched supervision volume. We compare two composition levels using disjoint low- and high-structure subsets from the CARTA accepted pool, and an equal mixture of the two (5,000 tasks per arm). The arms share the teacher, correctness filter, and recipe, and are matched on source count, trajectory length, tool calls, and supervised tokens (absolute standardized mean differences ≤ 0.011; per-arm statistics and composition axes in Appendix B.8, Table 5). Depth, branching, and coupling vary jointly—bundled enrichment, not ranked individual effects—though high-structure queries are 7.5% longer in words, so word count is not held exactly fixed. Both subsets with high-structure examples beat low structure on all three benchmarks (high by 3.7–5.4, mixed by 5.8–7.4), and mixing is strongest on every benchmark, modestly above high structure alone ($+2.1/+2.3/+2.0$): with volume matched, the gain is attributable to task structure, not more supervision or longer trajectories.

Realization control transfers to training. We next ask whether the contract's realization-fidelity advantage (§3.4) transfers to training. Per source graph we generate a strong graph-to-query and a full-CARTA query, regenerate teacher trajectories under matched tools, decoding, retry cap, and filter, and train Qwen3.5-9B on 3,000 graph pairs usable in both conditions (common support fixed before evaluation). Full CARTA improves accuracy by $+6.1/+4.1/+6.6$ points at matched supervised tokens (within 1%): the fixed-graph fidelity advantage produces a downstream gain.

Structure-guided trajectory selection. The constructed contract labels not only which tasks realize the intended structure but which *trajectories* do. We fix 3,000 CARTA tasks, sample five teacher rollouts each, and keep those reaching the correct answer as a fixed answer-correct pool. The contract score is the fraction of relation obligations in E_q satisfied by the trajectory's reasoning. Two rules then draw one trajectory per task: *Random* samples uniformly, while *Contract-selected* takes the rollout with the highest contract score. Each yields 3,000 Qwen3.5-9B SFT examples that differ only in the chosen trajectory. Contract-selected outperforms random on all three benchmarks (48.0/55.9/47.3 to 52.3/60.0/52.3, $+4.3/+4.1/+5.0$). Since both draw from the same answer-correct pool, the gain reflects structure alignment rather than answer correctness or sampling budget—the constructed structure carries supervision signal beyond the final answer (structural analysis of successful versus failed trajectories in Appendix B.12).

	Relation level				Identity level		
Condition	Expr. ↑	Drop ↓	Alt. ↓	Add ↓	Split ↓	Merge ↓	Conf. ↑
Source material only	42.8	46.5	10.7	38.6	-	-	18.8
Source material + rubric	79.6	14.2	6.2	12.8	-	-	57.1
Strong graph-to-query	87.2	8.1	4.7	8.4	11.8	7.3	72.9
CARTA without explicit bindings	89.8	6.4	3.8	6.5	8.3	5.2	77.1
Full CARTA	**93.5**	**4.1**	**2.4**	**3.8**	**3.5**	**2.1**	**86.3**

Table 3: Realization on 240 source graphs (realizer GPT-5.2, blind judge GPT-4o), percent. Expressed (query-macro relation recall), Dropped, and Altered (direction-reversed or relation-changed) sum to 100 per row; Added counts triples on entity pairs licensed by no obligation, missing extra relations on already licensed pairs. Split and merge cover the 144 same-entity and 96 distinctness tasks ("-" for source rows). Conformance is estimated under this operational audit over all 240 tasks. Graph IDs are paired; input packages differ between the first two and last three rows.

3.4 Trustworthiness of the Synthesized Tasks

Having shown CARTA data trains a stronger agent, we ask whether the tasks are trustworthy along three axes: whether realization preserves the specification (relation and identity fidelity), the answer is well-posed (answerability), and the facts are supported. The realization comparison scores every condition on the same 240 source graphs before acceptance, so outputs are paired; the fidelity and answerability audits use a fixed 240-task accepted sample; and the factual-support audit draws a separate sample. All use construction-independent judges, checked against human annotation (Appendix B.14); metrics and scoring rules are in Appendix B.10. Reported whole-query conformance is an estimate under this operational audit, not a certification of Eq. equation 5: the addition check detects unlicensed entity pairs but can miss extra relations on already licensed pairs.

Contract-guided realization preserves the specification. We evaluate five generation conditions with GPT-5.2. *Source material only* requests a question from the entity material; *Source material + rubric* adds the selected obligations and bindings but realizes them directly. The final three share the same facts and licensed descriptions: *Strong graph-to-query* receives the full relation inventory, shared entity IDs, answer target, and descriptions with instructions to preserve all relations and identities, under a planning-and-revision budget matched to CARTA's; *CARTA without explicit bindings* drops the Π_q list; *Full CARTA* retains both. Outputs are scored before acceptance with no common-success filtering (§B.10).

Making the selected structure explicit lifts relation recall from 42.8% (source only) to 79.6% once obligations are supplied. The strong graph-to-query baseline already reaches 87.2% recall and 72.9% conformance; over it, full CARTA raises recall to 93.5% and whole-query conformance to 86.3% (+6.3 and +13.3 on unrounded counts), cutting dropped obligations from 8.1% to 4.1% and unlicensed additions from 8.4% to 3.8%. On the same-entity and distinctness subsets, the explicit Π_q list further cuts identity splits from 8.3% to 3.5% and merges from 5.2% to 2.1% (conformance 77.1% to 86.3%, $+9.2$), preserving cross-obligation identity beyond its implicit encoding in shared graph IDs. The baseline reaches only 72.9% under a matched planning-and-revision budget, so the contract's lift to 86.3% is not merely more revision effort; we attribute the residual gain to its clause-localized audit and revision, though fully separating the audit from its organization would need an initial-realization-matched comparison we do not run. Appendix B.15 gives a worked realization with and without the rubric, per-obligation verdicts, and a production training item.

Revision, answerability, and answer exposure. Three further audits bear on task quality. *Revision* lifts fidelity: on the accepted sample, full-CARTA queries satisfy all structural criteria on 90.0% of queries after revision, up from a pre-revision 73.8% on that same sample ($+16.25$ points on unrounded counts; not the 72.9% graph-to-query baseline of Table 3), with final texts preserving 96.3% of the 1,080 relation obligations; conformance trails recall because one broken obligation invalidates an otherwise faithful query. *Answerability*: an independent evidence-backed solver supports the reference answer on 95.0% of tasks and finds a distinct satisfying answer on 7 (2.9%), bounding rather than proving uniqueness. *Answer exposure*: no answer name appears verbatim under a case-insensitive word-boundary check, and a no-tool probe recovers 147 of the 5,000 answers

(2.94%), on which the corpus is not filtered. Reference support and structural conformance stay separate outcomes—a correct answer can survive a structural omission, a faithful rubric can carry an upstream factual error. Per-metric breakdowns are in Appendix B.11 (Tables 6 and 7).

Factual support. An independent cross-family audit re-retrieves evidence for 670 construction units from 150 sampled queries: textual units are unsupported at 2.01% (4.41% if all unresolved cases count as errors, a sensitivity estimate rather than a guaranteed bound). The headline is the more conservative of two judges, robust to deeper retrieval (calibration, per-type breakdown, and sensitivity analysis in Appendix B.13).

4 Related Work

4.1 Search Agents and Training Data

Search agents progressed from browser-assisted models trained by imitation and human feedback (Nakano et al., 2021) and inference-time scaffolds that interleave reasoning and retrieval (Yao et al., 2022; Li et al., 2025b), to agents trained for multi-turn retrieval with reinforcement learning (Jin et al., 2025; Song et al., 2025; Zheng et al., 2025), and then to horizons requiring explicit history management (Gao et al., 2025; Qiao et al., 2025; Wu et al., 2025). To address scarce expert-authored tasks, systems synthesize tasks and trajectories (Wu et al., 2026; Li et al., 2025a; 2026; Liu et al., 2025; Du et al., 2026b;a; Team et al., 2025), while others synthesize trajectories for existing QA questions (Sun et al., 2025).

4.2 Structured Search-Task Synthesis

Multi-hop question construction already filtered candidates answerable without the intended composition (Trivedi et al., 2022); in current search-task synthesis, tasks are constructed by expanding or walking entity-relation graphs and obfuscating entity cues (Li et al., 2025a; 2026; Lu et al., 2025). Existing controls include reasoning depth through random-walk length (Lu et al., 2025), graph topology and evidence dispersion (Chu et al., 2026), topological expansion with target coverage and complexity (Du et al., 2026b;a), shortcut resistance (Deng et al., 2026), or a target difficulty level with solver feedback (Xu et al., 2026). Related work raises target-entity density over curated tables (Tao et al., 2025). For benchmark construction, LoHoSearch controls search-space size and structural complexity, and checks generated questions for subgraph coverage separately from answer satisfaction (Zhao et al., 2026). In deep research, QUEST decomposes each task into verifiable criteria whose score filters SFT trajectories and defines RL rewards (Xie et al., 2026).

WebShaper uses a set-theoretic formalization to guide layer-wise expansion and consistency validation, with layer count controlling coverage and difficulty (Tao et al., 2026). Building on structured construction and validation, CARTA combines three operations for relational composition with a characterization of their reachable graph family. Its contract records relation obligations and identity constraints for clause-localized auditing and revision, and reuses those obligations to select teacher trajectories. Paired-source-graph experiments isolate the downstream effect of this realization workflow from the composition it realizes.

5 Conclusion

We presented CARTA, which treats the relational structure of a synthetic deep-search task as a specification rather than a byproduct of how the task was sampled. Three composition operations build answer-rooted structures, with a characterization of which structures they reach; the selected structure compiles into a *Relational Contract* of relation obligations, cross-obligation identity constraints, and a withheld answer; and the realized question is audited against that contract clause by clause, so structural loss is located rather than inferred. Across matched-example training comparisons, realization fidelity on fixed source graphs, and trajectory selection among equally correct rollouts, the results support treating relational structure as something to be specified and checked rather than assumed.

AI USE STATEMENT

We used large language models for synthetic task generation, teacher trajectory generation, relation verification, and automated evaluation, as described in the main text and appendix. Generative AI tools also assisted with manuscript editing and consistency checks. The authors take responsibility for the final content, methodology, claims, and reported results.

REFERENCES

Zheng Chu, Xiao Wang, Jack Hong, Huiming Fan, Yuqi Huang, Yue Yang, Guohai Xu, Chenxiao Zhao, Cheng Xiang, Shengchao Hu, et al. Redsearcher: A scalable and cost-efficient framework for long-horizon search agents. *arXiv preprint arXiv:2602.14234*, 2026.

Bruno Courcelle. The monadic second-order logic of graphs. i. recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990.

Víctor Dalmau, Phokion G Kolaitis, and Moshe Y Vardi. Constraint satisfaction, bounded treewidth, and finite-variable logics. In *International Conference on Principles and Practice of Constraint Programming*, pp. 310–326. Springer, 2002.

DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

Jia Deng, Yimeng Chen, Xiaoqing Xiang, Ziyang Zeng, Shuo Tang, Wayne Xin Zhao, Feng Chang, Chuan Hao, Yuan Wei, Ran Tao, et al. Fort-searcher: Synthesizing shortcut-resistant search tasks for training deep search agents. *arXiv preprint arXiv:2606.12087*, 2026.

Yuwen Du, Rui Ye, Shuo Tang, Keduan Huang, Xinyu Zhu, Yuzhu Cai, and Siheng Chen. Openseeker-v2: Pushing the limits of search agents with informative and high-difficulty trajectories. *arXiv preprint arXiv:2605.04036*, 2026a.

Yuwen Du, Rui Ye, Shuo Tang, Xinyu Zhu, Yijun Lu, Yuzhu Cai, and Siheng Chen. Openseeker: Democratizing frontier search agents by fully open-sourcing training data. *arXiv preprint arXiv:2603.15594*, 2026b.

Jiaxuan Gao, Wei Fu, Minyang Xie, Shusheng Xu, Chuyi He, Zhiyu Mei, Banghua Zhu, and Yi Wu. Beyond ten turns: Unlocking long-horizon agentic search with large-scale asynchronous rl. *arXiv preprint arXiv:2508.07976*, 2025.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.

Jina.ai. Jina, 2026. URL `https://jina.ai/`.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Liwen Zhang, Litu Ou, Jialong Wu, Wenbiao Yin, Baixuan Li, Zhengwei Tao, Xinyu Wang, et al. Websailor: Navigating super-human reasoning for web agent. *arXiv preprint arXiv:2507.02592*, 2025a.

Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Yida Zhao, Liwen Zhang, Litu Ou, Ding-Chu Zhang, Xixi Wu, Xinmiao Yu, et al. Websailor-v2: Bridging the chasm to proprietary agents via synthetic data and scalable reinforcement learning. In *International Conference on Learning Representations*, volume 2026, pp. 93919–93948, 2026.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5420–5438, 2025b.

Junteng Liu, Yunji Li, Chi Zhang, Jingyang Li, Aili Chen, Ke Ji, Weiyu Cheng, Zijia Wu, Chengyu Du, Qidi Xu, et al. Webexplorer: Explore and evolve for training long-horizon web agents. *arXiv preprint arXiv:2509.06501*, 2025.

Rui Lu, Zhenyu Hou, Zihan Wang, Hanchen Zhang, Xiao Liu, Yujiang Li, Shi Feng, Jie Tang, and Yuxiao Dong. Deepdive: Advancing deep search agents with knowledge graphs and multi-turn rl. *arXiv preprint arXiv:2509.10446*, 2025.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Zile Qiao, Guoxin Chen, Xuanzhong Chen, Donglei Yu, Wenbiao Yin, Xinyu Wang, Zhen Zhang, Baixuan Li, Huifeng Yin, Kuan Li, et al. Webresearcher: Unleashing unbounded reasoning capability in long-horizon agents. *arXiv preprint arXiv:2509.13309*, 2025.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Serper. Serper: Google search api, 2026. URL https://serper.dev/.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.

Shuang Sun, Huatong Song, Yuhao Wang, Ruiyang Ren, Jinhao Jiang, Junjie Zhang, Fei Bai, Jia Deng, Wayne Xin Zhao, Zheng Liu, et al. Simpledeepsearcher: Deep information seeking via web-powered reasoning trajectory synthesis. *arXiv preprint arXiv:2505.16834*, 2025.

Zhengwei Tao, Haiyang Shen, Baixuan Li, Wenbiao Yin, Jialong Wu, Kuan Li, Zhongwang Zhang, Huifeng Yin, Rui Ye, Liwen Zhang, et al. Webleaper: Empowering efficiency and efficacy in webagent via enabling info-rich seeking. *arXiv preprint arXiv:2510.24697*, 2025.

Zhengwei Tao, Jialong Wu, Wenbiao Yin, Pu Wu, Junkai Zhang, Baixuan Li, Haiyang Shen, Kuan Li, Liwen Zhang, Xinyu Wang, et al. Webshaper: Agentically data synthesizing via information-seeking formalization. In *International Conference on Learning Representations*, volume 2026, pp. 101872–101889, 2026.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, et al. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*, 2025.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

Denny Vrandečić and Markus Krötzsch. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014. doi: 10.1145/2629489.

Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. Browsecomp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*, 2025.

Jialong Wu, Baixuan Li, Runnan Fang, Wenbiao Yin, Liwen Zhang, Zhenglin Wang, Zhengwei Tao, Ding-Chu Zhang, Zekun Xi, Robert Tang, et al. Webdancer: Towards autonomous information seeking agency. *Advances in Neural Information Processing Systems*, 38:120957–120985, 2026.

Xixi Wu, Kuan Li, Yida Zhao, Liwen Zhang, Litu Ou, Huifeng Yin, Zhongwang Zhang, Xinmiao Yu, Dingchu Zhang, Yong Jiang, et al. Resum: Unlocking long-horizon search intelligence via context summarization. *arXiv preprint arXiv:2509.13313*, 2025.

Xbench-Team. xbench-deepsearch, 2025. URL https://huggingface.co/datasets/xbench/DeepSearch.

Jian Xie, Tianhe Lin, Zilu Wang, Yuting Ning, Yuekun Yao, Tianci Xue, Zhehao Zhang, Zhongyang Li, Kai Zhang, Yufan Wu, et al. Quest: Training frontier deep research agents with fully synthetic tasks. *arXiv preprint arXiv:2605.24218*, 2026.

Fangyuan Xu, Rujun Han, Yanfei Chen, Zifeng Wang, I-Hung Hsu, Jun Yan, Vishy Tirumalashetty, Eunsol Choi, Tomas Pfister, and Chen-Yu Lee. Sage: Steerable agentic data generation for deep search with execution feedback. In *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 3334–3351, 2026.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 2369–2380, 2018.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Jiarui Zhao, Rongzhi Zhang, Lingchuan Liu, Hao Yang, Xunliang Cai, and Xi Su. Lohosearch: Benchmarking long-horizon search agents beyond the human difficulty ceiling. *arXiv preprint arXiv:2606.12837*, 2026.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 414–431, 2025.

Peilin Zhou, Bruce Leon, Xiang Ying, Can Zhang, Yifan Shao, Qichen Ye, Dading Chong, Zhiling Jin, Chenxuan Xie, Meng Cao, et al. Browsecomp-zh: Benchmarking web browsing ability of large language models in chinese. *arXiv preprint arXiv:2504.19314*, 2025.

A Appendix

B Construction and Calibration Details

B.1 Training and Rollout Setup

Supervised fine-tuning. The controlled SFT runs use the VERL training framework with the same Qwen3.5-9B initialization (Qwen3.5-4B for the smaller block), data format, optimizer, and evaluation interface for all training conditions. We match the number of training examples (5k) and fine-tune for 3 epochs with a maximum sequence length of $262{,}144$ (256×1024) tokens, with sequence packing to improve long-context utilization. Training uses bf16 precision and the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, weight decay 0.01, and gradient clipping at 1.0. We use a cosine learning-rate schedule with peak learning rate 2×10^{-5}, minimum learning rate 10^{-7}, and 2 warmup iterations. We mask observation (tool-response) tokens from the SFT loss, so the model is trained only on its own reasoning and actions and not on returned evidence. We report sample count as the primary training unit and token count as a supplementary corpus statistic.

Trajectory rollout. We construct training trajectories in the ReAct format (Yao et al., 2022), in which a shared DeepSeek-V4-Pro teacher solves each synthesized task over multiple turns. At each step the agent emits a free-form *Thought* (unrestricted natural-language reasoning for planning and decomposition), takes an *Action* and, for tool calls, receives the returned *Observation* before choosing the next move. Available capabilities include web search and page retrieval (Serper, 2026; Jina.ai, 2026), Google Maps, Google Scholar, and Python execution; these capabilities are available during rollout as well as during entity-cue obfuscation. ANSWER commits the final response. Rollouts are executed over query arrays in parallel and each is capped at 100 turns. We keep only trajectories whose final answer matches the reference before SFT (§3.1).

B.2 Construction Loop

Algorithm 1 states the loop summarized in §2.3. A step is identified with the single edge it adds, and $\mathcal{S}$ collects the steps that are admissible at the current partial structure: their source and target match one case of Eq. equation 3, their edge is supported by $\mathcal{K}$, and the resulting structure still satisfies

Model / system	Training	BrowseComp	BC-ZH	xbench-2510
Proprietary agents (published reference)				
Claude-4.5-Sonnet	-	24.1	42.4	-
Gemini-3-Pro	-	37.8	66.8	-
OpenAI-o3	-	49.1	68.7	-
GPT-5-High	-	54.9	63.0	-
Large-scale open-source agents (published reference, ≈30B)				
WebSailor-V2-30B-SFT	SFT	24.4	28.3	-
WebSailor-V2-30B-RL	SFT+RL	35.3	44.1	-
WebLeaper-30B-SFT	SFT	27.7	-	-
Tongyi DeepResearch-30B	CPT+SFT+RL	43.4	46.7	55.0
REDSearcher-30B	CPT+SFT+RL	42.1	49.8	-
OpenSeeker-v1-30B-SFT	SFT	29.5	48.4	-
OpenSeeker-v2-30B-SFT	SFT	46.0	58.1	-

Table 4: **Contextual positioning for Table 1.** Published proprietary and large-scale (≈30B) open-source agents, reproduced from the respective technical reports and public benchmark tables (Li et al., 2025a; 2026; Tao et al., 2025; Du et al., 2026b;a; Team et al., 2025; Chu et al., 2026). These tiers are contextual only: they start from different base checkpoints, are not run under our protocol, and where reported their training sets are larger than ours (15k WebLeaper; 11.7k/10.6k OpenSeeker-v1/v2). "-" denotes not reported or not applicable.

Algorithm 1 Answer-outward construction

Require: knowledge graph $\mathcal{K}$; request $\rho = (a; m, \beta^-, \beta^+)$
Ensure: graph G; realized profile $\widehat{\beta}$

$G \leftarrow (\{a\}, \emptyset)$
repeat
  $\mathcal{S} \leftarrow$ admissible steps at G
  if $\mathcal{S} = \emptyset$ **then**
    break
  end if
  $s \leftarrow \textsc{Select}(\mathcal{S}, \beta^-, \phi(G))$
  $G \leftarrow G + s$
until the policy elects to stop
$\widehat{\beta} \leftarrow \phi(G)$
return G, $\widehat{\beta}$

Eq. equation 1. SELECT is the only component the operations and bounds leave free; it weights the candidate steps by the coordinatewise shortfall of $\phi(G)$ against the soft target β^-, so it determines which structures are produced often, not which are reachable (§2.1), and the same holds when the policy additionally targets positional properties such as recurring branching.

B.3 A WORKED EXAMPLE

We thread one small task through the three stages—construction, contract, and realization—to make the definitions of §2 concrete on a single running structure. The structure is deliberately small enough to verify by eye and to exhibit all four realization failures; §B.15 shows items taken from the reported runs instead. The answer is the field-hockey club *Club Egara*; it is withheld in the emitted query and referred to as the target throughout.

Construction. Starting from the answer, construction selects four evidence-supported edges, all pointing toward the answer Ans:

$$\begin{aligned} &R \to \mathrm{Ans}\ (\text{Ramón Alegre played for}),\\ &C \to \mathrm{Ans}\ (\text{2018–19 División de Honor won by}),\\ &D \to R\ (\text{David Alegre is the brother of}),\\ &D \to \mathrm{Ans}\ (\text{David Alegre started playing at}). \end{aligned} \tag{6}$$

Two steps attach Ramón and the 2018–19 División de Honor as converging predecessors of the answer, and $D \to R$ adds David as a predecessor of Ramón. The final edge $D \to \mathrm{Ans}$ is a COUPLE step: both endpoints are already present, so it closes the undirected triangle D–R–Ans and ties the sibling path to David's own path into the club. The resulting profile is $\phi(G) = (3, 2, 2)$—peak in-degree three at the answer, longest path $D \to R \to \mathrm{Ans}$ of length two, and treewidth two from the single COUPLE step (Lemma 1).

The contract. Compiling G gives $R_q = (E_q, \Pi_q, a)$. E_q lists the four obligations of Eq. equation 6, each a separately checkable claim. Π_q records that *David Alegre* denotes one person across the sibling and origin obligations, that the answer denotes one club across the three obligations that reach it, and that the answer's predecessors—Ramón, the competition, and David—remain distinct. The first binding is what a realization severs when it splits David into two unrelated people, so the sibling path and David's own path no longer meet at one person and the coupling loop opens; the distinctness requirement is what it breaks when it collapses the two brothers into one player, so the sibling relation and the branch it anchors disappear. The contract records not a description of the graph but exactly the quantities whose loss would change the task.

Realization and its failures. Realization first obfuscates entity cues: *David Alegre* becomes *a Spanish field-hockey international*, *Ramón Alegre his brother, also a field-hockey player*, and *Club Egara* is withheld, so the emitted query pins the answer as the club a Spanish field-hockey international first played for as a boy, where his brother also played, and that won the 2018–19 División de Honor. Against this task the corruptions are concrete. A realizer may *omit* the *2018–19 División de Honor* obligation, so an answer-side branch disappears; *drift is the brother of* into the weaker *is related to*, losing the sibling tie the coupling depends on; *split* the shared *David Alegre* into a brother of Ramón in one clause and an unrelated youth player in another, so the sibling path and the direct path no longer meet; or *add* a direct tie between Ramón and the 2018–19 División de Honor, a shortcut absent from E_q that reaches the title without passing through the club. Each leaves a fluent, plausible query that specifies a different reasoning problem while the reference answer stays valid, which is what the conformance check of Eq. equation 5 is designed to catch.

B.4 TREEWIDTH AND REACHABILITY

The third profile coordinate κ is the treewidth of the undirected projection of G. We adopt it following Chu et al. (2026), for whom bounded treewidth marks constraint structures that decompose along a tree and are in that sense tractable (Courcelle, 1990; Dalmau et al., 2002). In CARTA the coordinate plays a narrower role: it measures how far the selected obligations depart from a tree, which is exactly the cross-path reuse that COUPLE introduces and nothing else does (Lemma 1). We use it to organize coverage of the composition space and report it as a structural property of the constructed task; the retrieval separation of the accepted tasks is controlled and reported separately (§B.5).

Measuring the structural profile and conformance. We report branching as maximum in-degree and depth as the longest directed path into the answer, and distinguish exact treewidth from its min-fill upper-bound estimate, with non-tree structure measured on the simple undirected projection. Query conformance is assessed independently of the answer: a whole query conforms only if all selected relations, applicable identity bindings, and distinctions are preserved without adding unlicensed relational constraints. Two later experiments each fix a separate set of 3,000 tasks: the *realization-transfer cohort* (§3.3) pairs a strong graph-to-query query against a full-CARTA query per source graph, and the *trajectory-selection cohort* (§3.3) fixes CARTA tasks and varies only which teacher rollout is used for SFT.

Proof of Lemma 1. EXTEND and BRANCH require $w \notin V$, so each step adds one node and one edge and the undirected projection stays acyclic; a tree with at least one edge has treewidth one. Conversely, every partial structure is answer-connected, so the endpoints of a COUPLE edge are already joined by an undirected path; Eq. equation 3 excludes an existing $w \to u$ and $u \not\rightsquigarrow w$ excludes $u \to w$, so that path has length at least two and the step closes an undirected cycle of length at least three, giving $\kappa(G) \geq 2$.

Proof of Proposition 1. Order the edges of H by the shortest directed distance of their targets to a and add them in that order. Every edge is added once, with its target already present and its source either new or already present, so it falls into exactly one case of Eq. equation 3; every partial structure is a subgraph of the acyclic H, which makes the COUPLE conditions hold and, by monotonicity, keeps every partial structure in $\mathcal{H}_{\mathcal{K}}(\rho)$ throughout.

B.5 CONSTRUCTION CONTROLS

An accepted query must first satisfy its selected relational rubric: every selected obligation is evidence-supported, answer-connected, and consistent with the required entity bindings. We retain page-level provenance for each obligation and require the selected obligations collectively to draw on multiple source pages. For each query, we report the number of distinct supporting pages and the minimum number of pages whose union covers all selected obligations, following the motivation of Minimum Source Dispersion (MSD) (Chu et al., 2026). We use these as retrieval-separation statistics rather than claims of statistical independence: topical or hyperlink overlap between pages is a limitation and may be reported separately. A query is rejected when one page independently covers the answer and all required relations, or when an identifying surface reveals the answer without the intended relational composition. These are controls against incidental retrieval shortcuts and invalid evidence; they are not difficulty labels.

Search-stage benchmark leakage prevention. During synthesis, search requests include exclusions for benchmark-related keywords or domains to reduce retrieval of benchmark questions, answers, and dataset pages as source evidence. This safeguard operates at the search request stage; it is not a post-generation overlap audit over entities, near-duplicate questions, or evidence pages, and does not guarantee that all benchmark-related content is excluded.

A construction request instantiates the soft target β^- either on a single coordinate—a floor of $b \geq 4$, $d \geq 4$, one nested subproblem, or $\kappa \geq 2$—or on none, leaving the mix to the SELECT policy. These are soft construction targets, not acceptance requirements; the constructed graph may fall short when the verified relation supply is insufficient.

For construction controls, nesting is an answer-rooted positional indicator: it is positive when convergence occurs on two consecutive layers into the answer—the answer's converging constraints are themselves fed by at least one further convergence one relational step upstream. An isolated fork deeper upstream, or one reached through a single relation into the answer, leaves the indicator at zero. It is therefore a stricter special case of branching away from the answer, and statistics computed with it under-count nested tasks rather than over-count them. It is recorded as a composition indicator computed on the constructed task.

We record the following task-side variables: realized branching, depth, and coupling; selected-edge count; shared-attribute and shared-middle roles; number of source pages; source dispersion; and single-page shortcut indicators. Because exact treewidth is NP-hard, corpus-level coupling is reported as a greedy (min-fill) treewidth estimate, which is an upper bound in general and is also what the admissible upper bound β^+ is checked against during construction; the small running-example projection admits an exact value ($\kappa = 2$). All three coordinates are computed on the constructed graph.

These variables describe the constructed task, not its difficulty. We do not claim that any coordinate determines how hard a task is for a given solver. Establishing that would require a behavioral study under a fixed solver and tool environment, with topic, entity type, query length, and cost band matched across structural regimes, and with consistent trends across more than one solver strength; our construction records the variables such a study would need, and we leave the study itself to

future work. The trajectory-level evidence we do report (Table 8) is conditioned on outcome and is solver-relative. Trajectory length in particular is an observed execution cost, not a difficulty label.

B.6 Agentic Edge Discovery for Coupling

COUPLE steps require an attested edge between two entities the structure has already selected, so their candidate set is constrained at both endpoints, unlike the single-endpoint in-neighborhoods that EXTEND and BRANCH consume. When the current verified graph supplies no such edge, an agent searches for one (§2.3). Two candidate types are admitted: a *directed relation* between two entities already present, and a *shared-attribute edge* asserting a common value—such as a founding year, location, or venue—which is symmetric rather than directed.

Proposal and tool routing. Candidate pairs are drawn from entities already in the structure, given the partial graph and the profile coordinate still short of β^{-}. The agent then selects a tool route from the types of the two endpoints and the relation it is testing: place lookup for locations, scholarly search for publications, programmatic filtering or intersection over retrieved attribute lists when the candidate is a shared value that can be compared exactly, and web search otherwise. The web route first queries the co-occurrence of both names and falls back to a looser query, since the evidence for the two endpoints is by construction documented separately rather than on a shared page.

Confirmation. A proposal is admitted only if it survives three checks, regardless of which route produced it. First, the supporting pages for the two endpoints must be at different URLs. Second, each supporting page is re-read as raw page text rather than a server-side summary, and an entailment judge must confirm the claim against that text; reading the raw page avoids compounding a summarizer's paraphrase with the judge's own. Third, social media hosts and automatically generated or mirrored encyclopedia sites are excluded as cross-page evidence, because they tend to echo the source knowledge graph and therefore cannot serve as independent confirmation. A programmatic route reports the comparison it performed and the pages the compared values were read from, and is subject to the same three checks rather than being trusted for being deterministic. At most six candidate pairs are attempted per graph and at most two such edges are admitted; unconfirmed candidates are discarded and the structure keeps whatever coupling it already has.

How admitted edges enter the contract. A confirmed directed relation enters E_q as an ordinary directed obligation. A shared-attribute edge is symmetric—swapping its endpoints expresses the same fact—so it enters E_q as a symmetric obligation and is exempt from the direction check described in §B.10, in the same way as invertible relations. Coupling is recomputed with admitted edges included. Because the two endpoints of an admitted edge are documented on separate pages, each such edge adds a constraint that no single page satisfies, which is the same retrieval-separation property that the MSD controls target. Discovery is currently attempted only for graphs whose sampled skeleton is block-shaped, so it is a supplementary supply of coupling rather than its primary mechanism.

B.7 Realization Constraints in the Pipeline

The realization stage of the production pipeline (§2.3) writes one clause per obligation before the clauses are joined into prose. Figure 3 excerpts the instructions that carry the relational contract into that intermediate representation: one claim per obligation, a fixed rendering of relational direction, and the identity bindings of Π_q. Instructions concerning surface wording, answer hiding, and output format are omitted, as are the fields that supply the licensed phrases. These constraints act on the candidate query; whether the emitted query satisfies them is decided by the acceptance audit, not asserted by them.

B.8 Matched-Supervision Controls

Two controls support the matched-example comparisons of §3.2 and §3.3: they confirm that the accuracy differences track the task distribution, not the amount of supervision. *Teacher search effort.* On the same three 5k corpora, the shared teacher issues on average 74.2, 71.8, and 68.0 tool invocations per training trajectory for REDSearcher, OpenSeeker-v1, and CARTA (medians 69, 71, and 63); CARTA is, if anything, marginally lower, so its downstream advantage in Table 1 cannot be

Realization Plan — Structure Rules (excerpt)

One obligation, one claim. For each edge, write exactly one short claim that faithfully expresses *source* → *relation* → *target*. Do not merge multiple edges into one claim. Each claim must be a natural atomic sentence.

Direction. Every factual relation has a fixed, known direction: one endpoint is its SOURCE (the one that performs or originates the relation) and the other is its TARGET. Render each relation so that the reader can recover this direction from word order and voice alone, with no reliance on world knowledge: put the SOURCE endpoint first as the grammatical subject, the TARGET after it as the object, in active voice. A passive "was ⟨predicate⟩ by ⟨endpoint⟩" shape is allowed only when the agent is the hidden-answer anchor; it is forbidden whenever both endpoints are clue entities, and especially when they share a broad type (two persons, two places), because then a stray "by" makes their source/target roles indistinguishable. Re-cast a passive relation phrase into active source-first order using only the words already present in the phrase: reorder and re-voice, never add or infer a new word.

Identity. Follow the identity and distinctness constraints when choosing "same", "another", or a repeated role phrase. Where the plan records that several claims mention the same entity, do not introduce it as separate entities; where two entities must stay distinct, use "another X" or "a different X", or introduce them separately.

Figure 3: Structure-relevant rules given to the realization stage of the production pipeline, excerpted from the plan instructions. They govern how the obligations in E_q and the identity bindings in Π_q are lowered into clauses; they are not a check on the result.

	Low	Mixed	High
Depth d	1.94	2.15	2.37
Nested subproblem (%)	0.0	25.2	49.9
$\kappa \geq 2$ (%)	0.8	38.8	76.9
Excess edges	0.01	0.50	1.00

Table 5: Manipulation check for the structural arms: the three subsets separate on the composition axes while matched on the supervision quantities given in §3.3. Nesting is reported as a branching manifestation and excess edges over a spanning tree as a coupling-magnitude proxy, not as the raw b or κ values. Nesting and $\kappa \geq 2$ are percentages; depth and excess edges are counts. Downstream accuracies are in Table 2.

attributed to longer teacher trajectories or more search per example. *Structural-arm manipulation check.* Table 5 reports how the low-, high-, and mixed-structure subsets of §3.3 separate on the composition axes while matched on supervision volume: across the three arms distinct source pages average 5.46, trajectory turns 44.6–44.7, tool calls 25.5–25.8, and supervised tokens per example 36,971–37,175.

B.9 Synthesis Cost

Synthesizing one task takes at least about twenty-five model calls and roughly eighteen tool calls, most of them web searches, with a mean runtime of about four minutes; teacher trajectory generation and SFT are separate from these synthesis totals. Contract-guided synthesis incurs additional overhead relative to single-pass generation through its audit and revision stages. Under a matched planning-and-revision budget, however, CARTA improves whole-query conformance from 72.9% to 86.3% (Table 3). Separately, paired-source-graph training comparisons show downstream gains from contract-guided realization (Table 2).

B.10 Realization Comparison: Setup and Metrics

This appendix specifies the protocol used to measure how much of the composed structure survives query realization, and how that survival depends on the relational rubric $R_q = (E_q, \Pi_q, a)$ and on the workflow that consumes it. The comparison holds the source graph, the realizer model, and the

decoding settings fixed across five conditions (defined below); the variables are whether the rubric is laid on the shared source before realization, and whether the production pipeline's staged binding pass is used. It measures the fidelity of the emitted query text to the composed structure, and is distinct from the trajectory-level alignment analysis in Table 8, which measures a solver's search behavior.

Compared conditions. We run five realization conditions on the shared 240-task evaluation sample, holding the source graph, the realizer model, and the decoding settings fixed across conditions. The first two form a source-material conditioning contrast that isolates the effect of the relational rubric. *(1) Source material only* gives the realizer one line per non-answer entity (its real name and a short source description) plus the hidden target, and asks for one connected multi-step puzzle. *(2) Source material + rubric* adds the licensed obligations E_q as a directed edge list with the answer masked as a variable, together with the identity bindings Π_q as a shared-entity binding note, and instructs the realizer to state exactly those links in those directions. Because both conditions may use the same real names, their contrast measures the effect of making the selected obligations and shared-entity bindings explicit during realization. The remaining three form a full-inventory workflow contrast that varies how the production pipeline consumes the same inventory: *(3) strong graph-to-query* lowers the graph directly without the staged binding pass, *(4) CARTA without explicit bindings* runs the full pipeline but omits the Π_q binding note, and *(5) full CARTA* is the production realizer. Conditions 3–5 attribute the residual fidelity gap to the binding pass rather than to the edge list alone.

Graphs, models, and decoding. All five conditions are evaluated on the same 240 source graphs, sampled once with a fixed seed and matched across conditions; the sample spans graphs that carry coupling and nested subproblems as well as graphs with more edges than the median, so that fidelity is not measured only where the licensed obligations comfortably fit a natural query. The realizer is GPT-5.2 throughout; the structure-fidelity judge (both the extraction and alignment steps) uses GPT-4o. Realization uses standard sampling; the judge is run at temperature 0.

Structure-fidelity judge. The judge reads the query text only, with no access to the source graph, in two steps. In the *extraction* step, the model is given the query and a de-identified roster (stable labels $N_1, \ldots, N_k$, with the answer shown as `[TARGET]`) and recovers a list of (head, relation, tail) triples using only what the query states; it also flags any roster entity the query realizes as two non-coreferring mentions (*split*) and any two distinct roster entities the query collapses into one (*merge*). In the *alignment* step, each licensed obligation in E_q is compared against the extracted triples and receives exactly one verdict.

Obligation verdicts and metrics. The verdicts map directly onto the rubric components. Each obligation is scored *expressed* (an extracted triple asserts the same real-world fact), *direction-reversed* or *relation-changed* (a relational realization error on the same entity pair), or *missing* (an E_q omission). An extracted triple whose endpoint pair is licensed by no obligation is counted as an *addition* (E_q over-generation). Because this test keys on the endpoint pair rather than the individual relation, an extra relation asserted over a pair that some obligation already licenses is not counted; the reported Add rate therefore tracks over-generation onto unlicensed entity pairs and can understate additions relative to the stricter per-relation no-addition reading. Whole-query conformance reported under this operational audit is therefore an estimate, not a guarantee of satisfying every conjunct of Eq. equation 5. A shared entity (degree ≥ 2, other than the answer) is counted as correctly bound when the extractor neither split nor merged it, giving an entity-binding accuracy associated with Π_q. We report, per condition, the Expressed rate, Drop rate (missing), and Add rate, averaged over graphs; direction-reversed and relation-changed obligations are reported together as the expressed-but-altered residual. Relation recall is the per-query (query-macro) mean, whole-query conformance is taken over all 240 tasks, and the identity split/merge rates use the applicable-task counts (144 same-entity and 96 distinctness tasks). Unresolved judgments are not counted as passes, and a missing output counts as zero recall with a failed conformance outcome. The 240-task sample is selected by task ID under a preset random seed, and every realization condition is applied to the same source graphs, so their outputs are paired.

Direction under invertible relations. Many relations denote the same fact when their endpoints are swapped and the converse wording is used: spatial "A is north of B" = "B is south of A";

Realization Prompt — Condition 1: Source material only

You are given reference material about several real entities and must write ONE natural, fluent research-puzzle question (40–100 words) whose answer is a hidden target entity, requiring multi-step reasoning to solve.

Reference material (source snippets): {sources}

Hidden target: The answer is a specific entity; for your own coherence it is {answer}. You MUST NOT print this name or any alias; refer to it only indirectly (e.g. "X", a pronoun, or "the … who …"). The reader must recover it by reasoning.

Rules: (1) Write ONE coherent paragraph ending with "?". (2) Build a connected multi-step puzzle: the entities should chain toward the hidden target, not read as an unrelated list. (3) You may refer to entities by name or by description. (4) Do NOT reveal {answer} or any alias. Output ONLY the question.

Realization Prompt — Condition 2: Source material + rubric

You are given reference material about several real entities AND an explicit relational rubric stating exactly how they connect. Write ONE natural, fluent research-puzzle question (40–100 words) whose answer is a hidden target entity.

Reference material (source snippets): {sources}

Relational rubric (realize EXACTLY these links, preserving each direction): {E_q edge list, head $\rightarrow$ tail, answer shown as X}

Shared entities (each must stay ONE entity across its links): {Π_q bindings}

Hidden target: The answer appears as X in the rubric; for your own coherence it is {answer}. You MUST NOT print this name or any alias.

Rules: (1) Write ONE coherent paragraph ending with "?". (2) Realize EVERY numbered rubric link, preserving its direction (head $\rightarrow$ tail). (3) Add NO relations beyond the rubric. (4) Where an entity appears in several links, keep it as ONE single entity. (5) Do NOT reveal {answer} or any alias. Output ONLY the question.

Figure 4: Source-material realization prompts for Conditions 1 and 2. Both conditions receive the identical {sources} block; Condition 2 additionally receives the licensed obligations E_q as a directed edge list and the identity constraints Π_q as a shared-entity binding note. Conditions 3–5 use the production realizer (Figure 3).

kinship "X is the parent/grandparent of Y" = "Y is the child/grandchild of X"; creation "W created/authored/directed Z" = "Z was created/authored/directed by W"; funding "F funded G" = "G was funded by F"; and active versus passive voice. For such invertible relations, a converse expression is scored as *expressed*, not as a direction error. A direction error is reserved for non-invertible relations where swapping the endpoints changes who-did-what, e.g. "X defeated Y" versus "Y defeated X", or "X is the capital of Y" versus "Y is the capital of X". The direction metric therefore tests preservation of factual role rather than the surface head/tail order chosen by the realizer.

Prompts. The two source-material prompts (Conditions 1 and 2) are shown in Figure 4, and the two-step structure-fidelity judge prompts in Figure 5. Conditions 3–5 use the production realizer, whose structure-relevant instructions are given in Figure 3. Braces mark filled-in fields; the answer field is used only for the realizer's own coherence and is forbidden from appearing in the output. The judge prompts are the measurement instrument and are distinct from the acceptance audit inside the synthesis pipeline: the audit reads the rubric and decides whether a candidate is accepted or discarded, and none of the numbers reported in this paper come from it, whereas the judge's extraction step never sees the rubric and recovers the relational content of a query from the query text alone.

Per-condition fidelity. The obligation-level fidelity breakdown (Expressed/Dropped/Altered/Added and whole-query conformance) and the identity split/merge counts for these conditions are reported

Structure-fidelity judge — Step 1: Blind Triple Extraction

System: You are a precise information extractor. Output only valid JSON.

User: Read ONLY the question and the roster of candidate entities (each has a stable label and short description). Report ONLY what the question itself states; use no outside knowledge, invent no relations.

Question: {question} **Roster:** {$N_1 \dots N_k$ with target as `[TARGET]`}

The question asks the reader to identify ONE hidden target, referred to indirectly (a bare variable like "X"/"Y"; an interrogative "who/what/which…"; or a placeholder). Map EVERY mention of that target to `[TARGET]`; map every other mention to its single best roster label (or `OUT`).

Output {`relations:[{head,relation,tail}], split:[...], merge:[[...]]`}: direction is head → tail as it reads in the question; *split* = a roster entity referred to by two descriptions the question never ties together; *merge* = two distinct roster entities collapsed into one mention.

Structure-fidelity judge — Step 2: Relation-Aware Alignment

System: You align relational triples. Output only valid JSON.

User: You are aligning two lists of triples sharing the same entity labels. GOLD lists the intended relations; FOUND lists relations a reader extracted from the question. Judge purely by whether a FOUND triple asserts the same real-world fact as the gold triple.

Gold: {indexed E_q triples} **Found:** {extracted triples}

Many relations are *invertible*: swapping head/tail with the converse wording ("A north of B" = "B south of A"; "X parent of Y" = "Y child of X"; "W created Z" = "Z created by W"; active vs. passive) expresses the identical fact and is NOT a direction error. For each gold triple choose one: *expressed*; *direction_reversed* (only for non-invertible relations where the swap changes who-did-what, e.g. "X defeated Y" vs. "Y defeated X"); *relation_changed* (same pair, different-meaning relation); *missing* (no FOUND triple connects that pair). Also count *added*: FOUND triples whose pair matches no gold pair in either direction.

Output {`verdicts:[...], added:<int>`}.

Figure 5: Two-step structure-fidelity judge. Step 1 extracts triples from the query text alone against a de-identified roster; step 2 assigns each licensed obligation a verdict, with invertible relations scored direction-agnostically.

in the main text (Table 3); the initial-versus-final query audit is in §B.11. We make no significance claim and impute no uncertainty intervals.

B.11 Revision, Answerability, and Answer Exposure

This subsection reports the per-metric breakdowns for the three quality audits summarized in §3.4: the initial-versus-final revision audit, the answerability audit, and answer exposure. The revision and answerability audits use the same 240-task evaluation sample; the answer-exposure audit covers all 5,000 tasks in the CARTA training corpus.

Revision lifts fidelity on accepted tasks. We audit the final production query and its historical initial version for each task in the common 240-task sample. All initial versions are available, the rubrics contain 1,080 obligations in total, and the applicable binding counts are 144 and 96. The relation/mention reviewer extracts relations and resolves mentions before aligning to the production rubric.

Final production texts preserve 96.3% of the 1,080 relation obligations, and 90.0% of queries satisfy all structural criteria, up from 73.8% for the same full-CARTA queries before the revision pass. Macro recall improves by 6.9 points and whole-query conformance by 16.25 points on the unrounded counts. Conformance sits below relation recall because a single missing relation, broken binding, or unlicensed addition invalidates an otherwise faithful query; production acceptance is an approximate check rather than a guarantee of exact conformance.

Metric	Initial	Final accepted
Relation recall, query-macro	89.1	96.0
Relation recall, obligation-micro	89.4	96.3
Same-entity binding preserved	91.7	97.2
Required distinctions preserved	93.8	97.9
Whole-query structural conformance	73.8	90.0

Table 6: Initial and final versions of the same 240 eventually accepted tasks, in percent over 1,080 obligations and the 144 and 96 applicable bindings. This sample is conditioned on acceptance.

Outcome	Queries	Share
Reference answer supported	228	95.0%
Reference answer contradicted	4	1.7%
Reference status unresolved	8	3.3%
Supported, another satisfying answer found	7	2.9%
Supported, no alternative within budget	221	92.1%

Table 7: Answerability audit. The first three rows partition the 240 tasks; the final two partition the 228 reference-supported cases. Every displayed percentage uses all 240 tasks.

Answerability and ambiguity. An independent evidence-backed solver checks the reference answer and searches for a different satisfying answer on the same 240 final queries under a fixed retrieval budget. Unresolved reference judgments remain in the denominator.

Evidence supports the reference answer on 95.0% of tasks, and a distinct satisfying answer is found on 7 tasks (2.9% of all tasks, 3.1% of the reference-supported cases). The 221 remaining supported cases have no alternative found within budget, which bounds rather than proves uniqueness. Reference support and structural conformance are separate outcomes: a correct answer can survive a structural omission, and a faithfully realized rubric can contain an upstream factual error.

Answer exposure. Across the 5,000 CARTA tasks, no answer name appears verbatim under the specified case-insensitive word-boundary check. Recorded-alias matching finds two degenerate alias strings, a hedge word and a year. A separate no-tool probe recovers 147 reference answers (2.94% of the 5,000); the corpus is not filtered using this probe. These results characterize string exposure and recoverability for one solver.

B.12 Solver Alignment

The audits above ask whether a query expresses its contract. This subsection asks a different question: whether a *trajectory* that solves the task traverses the relations the contract records. We examine a stratified sample of teacher trajectories with a frozen light judge, measuring whether rollouts reach the intended waypoints and follow the required relations, and recording the same quantities separately for failed long rollouts. This is a conditional validity check, not a claim that a task admits a unique correct search trajectory.

Correct trajectories cover 91% of intended waypoints and 93% of required edges on the model side, while failed long rollouts cover only 69% and 62%. The same solver therefore behaves differently depending on whether it succeeds, which is consistent with the composed relational structure influencing the search process rather than merely decorating the query. Two limits are worth stating. The rates are conditioned on outcome, so they diagnose a correlation between alignment and success rather than establishing that the tasks are intrinsically unshortcuttable. And evidence-side coverage is markedly lower than model-side coverage even among correct trajectories (64% against 93% for required edges), so a trajectory discusses an obligation more often than retrieval demonstrably covers it.

Conformance metric	Correct	Failed†
Waypoint traversal (model)	91	69
Entity recall (evidence)	76	51
Edge-followed (model)	93	62
Required-edge recall (evidence)	64	34
Answer-edge followed (model)	94	46
Zero-waypoint trajectories	0	18
Zero-required-edge trajectories	2	26
Trajectories / waypoints / edges	200/809/887	200/795/855

Table 8: Solver alignment, in percent. Model-side rates read the trajectory's reasoning; evidence-side rates require the relation to be covered by retrieved content. Rates are conditioned on outcome and are solver-relative diagnostics. †Failed rollouts ran for at least 30 tool calls.

Audit	Automated adjudicator	N	Agree (n)	Agree %
Whole-query conformance (Tab. 3)	structure-fidelity judge, GPT-4o	50	48	96.0
Revision fidelity (Tab. 6)	relation/mention reviewer	50	47	94.0
Solver alignment (Tab. 8)	light rubric-hit judge	50	45	90.0
Factual support† (§B.13)	unit-support judge	50	40	80.0

Table 9: Human verification of the automated audits. On a blind calibration subset of 50 items for each audit, we report the number of items where an independent human annotation agrees with the automated verdict and the resulting agreement rate. The factual-support row (†) is author adjudication (§B.13), not independent annotation.

B.13 Factual-Support Calibration

The factual-support audit is calibrated rather than assumed. On the identical stored evidence, a second judge from the construction model family agrees on 89.2% of 611 units (Cohen's $\kappa = 0.606$) and would itself report a lower unsupported rate (0.81% against 1.64%), so the reported headline is the more conservative of the two. On 50 units re-adjudicated blind by an author, agreement is 80.0% ($\kappa = 0.625$), and all ten disagreements run in one direction, with the judge stricter than the author.

One retrieval-sensitivity pass was run. For each initially indeterminate textual unit we issued three additional retrieval queries, leaving the adjudication prompt untouched and never re-running a decided unit. This resolved 49 of 64 indeterminates and raised the reported unsupported rate from 1.64% to 2.01%, while tightening the two quantities that are easier to contest: the conservative bound from 11.93% to 4.41%, and the strict per-query rate from 63.8% to 83.2%. Both passes are retained, and the headline uses the second.

The audit covers 670 construction units drawn from 150 sampled queries. By unit type, no coordinate-derived or shared-attribute unit is unsupported, but those samples (58 and 48 units) are too small to claim lower risk than ordinary graph edges; the 2.01% headline is over the textual units, which dominate the sample.

B.14 Human Verification of the Automated Audits

The trustworthiness audits in §3.4 and above rest on automated adjudicators—an LLM structure-fidelity judge for conformance, a relation/mention reviewer for revision fidelity, and a light judge for solver alignment. A model-based judge *estimates* conformance rather than deciding it, so we bound these judges against human annotation on a blind calibration subset and report the per-item human–automated agreement. The factual-support row is the author adjudication already described in §B.13; the remaining rows use independent annotators. Agreement is the fraction of subset items where the human label matches the automated verdict.

Each audit uses a prespecified sample of 50 items. Annotators assign one categorical label per item using the same rubric as the automated judge, without seeing its verdict; we report agreement between human and automated labels. Whole-query conformance is assessed on a sample spanning the two realization arms and the composition buckets; revision fidelity re-audits the initial and final versions of accepted tasks; and solver alignment re-scores a stratified sample of teacher trajectories against the rubric.

B.15 CASE STUDY

Where §B.3 makes the definitions checkable on a structure small enough to verify by eye, this subsection shows two items taken unchanged from the runs reported in the paper. Figure 6 follows one item of the 5k training corpus through the three views that the construction produces: the selected obligations, the surface descriptions they are allowed to use, and the realized query. Figure 7 contrasts the relational content of questions realized from the same source graph with and without the rubric. The first item comes from the 5k corpus used for the controlled comparison in Table 1, the second from the 240-graph GPT-5.2 comparison in Table 3.

What the production item illustrates. One thing is visible in Figure 6 that neither the aggregate numbers nor a small illustration can show: the composition is carried by the obligation set rather than by the query's length. The query runs to several sentences, yet the structure it is required to state is six obligations, three of them pointing at the answer, with the longest path reaching it in three relational steps.

What the contrast illustrates. In Figure 7 both arms received the identical real-name source block, so the difference is the rubric alone. The source-only query retains Baylson's school attendance but replaces the other intended links with clues about birth cities, a father's graduate school, and a neighbouring township. The rubric query expresses all four selected relations and binds Finck's and Baylson's school references to the same institution. The contrast illustrates why preserving the answer target alone is insufficient: realization can change the relational task rather than merely rephrase it.

Figure 8 presents one representative CARTA item through the complete synthesis process: starting from the hidden answer node X, the system applies EXTEND, COUPLE, and BRANCH to verified relation units and obtains the final answer-rooted graph; it then compiles the graph into a relational contract containing six obligations E_q, identity constraints Π_q, and the hidden answer a, where obligation 2 connects two existing entities and closes a triangle in the undirected projection, yielding $\phi(G) = (b, d, \kappa) = (3, 3, 2)$; during realization, licensed surface descriptions replace entity names, while definite expressions such as "the hip-hop-musician actress," "that film," and "the legal-drama actress" preserve cross-clause identity bindings and keep the answer hidden; finally, the audit checks relations and directions, identity and distinctness constraints, answer masking, and unlicensed additions, producing the answer-hidden query shown in the figure.

(a) Selected obligations E_q (answer masked as X)

1. *Billie Rose Prichard* → *Monkey Puzzle (film)* acted in
2. *Billie Rose Prichard* → *Ella Scott Lynch* appeared alongside
3. *Monkey Puzzle (film)* → *Ella Scott Lynch* stars
4. *Ella Scott Lynch* → X attended
5. *Rebecca Smart* → X educated at
6. *Eirene Mort* → X attended

Π_q: *Billie Rose Prichard* is one entity across 1–2, *Monkey Puzzle* across 1 and 3, *Ella Scott Lynch* across 2–4. Profile: b=3 (obligations into X), d=3 (*Billie Rose Prichard* → *Monkey Puzzle* → *Ella Scott Lynch* → X), κ=2 (the cycle 1–2–3), two-level convergence at *Ella Scott Lynch*. Answer a: *St Catherine's School, Waverley*.

(b) Surface descriptions licensed for those entities

Monkey Puzzle (film): "2008 Australian feature film about a group of friends who get lost in the wilderness" *Billie Rose Prichard*: "Australian actress on stage and screen, and a hip-hop musician" *Ella Scott Lynch*: "Australian actress with notable roles in legal drama series and a Channel Nine mini-series" *Rebecca Smart*: "Australian actor" *Eirene Mort*: "educator, born in the 1870s"

(c) Realized query (verbatim)

An Australian actress on stage and screen, and a hip-hop musician (Actress A) acted in a 2008 Australian feature film about a group of friends who get lost in the wilderness (Film A). Actress A appeared alongside an Australian actress with notable roles in legal drama series and a Channel Nine mini-series (Actress B). Film A stars Actress B. Actress B attended an institution. An Australian actor was educated at that institution. An educator, born in the 1870s attended that institution. Which institution was this?

Figure 6: **One item of the 5k training corpus in three views.** The obligations in (a) are the contract obligations E_q; the descriptions in (b) are the only surface material the realizer may use for the non-answer entities, which is why no entity is named in the query in (c) and the answer is referred to only as "an institution". The structural coordinates are computed on the constructed graph. The item passed edge-level fact verification (verdict *sound*) and the query-level audit (verdict *keep*). The rollout model answered it correctly in 65 tool calls (47 searches, 16 page opens, 2 code executions): six pages, among them two other 2008 Australian films, were opened before the licensed film at turn 18, the two actresses at turns 20 and 22, and the answer's page at turn 24, after which the answer page's alumnae listings were opened again at turns 28–35. This rollout illustrates multi-step evidence gathering and cross-checking.

Source graph (4 obligations, answer masked as X)

1. *Steve Kuhn* → *David Finck* appeared with 2. *Symphonica Tour* → *David Finck* included bassist
3. *Michael Baylson* → X attended 4. *David Finck* → X graduated from Answer a: *Cheltenham High School*.

Condition 1: Source material only

Starting from Steve Kuhn, identify the city where he was born, then move to David E. Finck, who was born in another city while his father attended graduate school at that city's university and was later raised in the same city where Judge Michael Morris Baylson was born. Baylson attended a particular high school in a neighboring township—what is the name of that school?

Condition 2: Source material + rubric

Which secondary school is jointly pointed to by these clues: the jazz pianist Steve Kuhn appeared with the bassist David Finck; the Symphonica Tour included David Finck as its bassist; David Finck graduated from the same school that a senior U.S. district judge, Michael Baylson, attended—so what is the name of the shared school?

Figure 7: **The same source graph realized with and without the rubric.** Both conditions use GPT-5.2 and the identical real-name source block from one graph in the 240-graph comparison (Table 3). Without the rubric, the question shifts to birthplaces, family education, and upbringing instead of retaining the selected relations. With the rubric, it expresses all four relations, and "the same school" and "the shared school" bind the two school references to one entity. Entity names are retained in these conditions; production items are de-identified as in Figure 6.

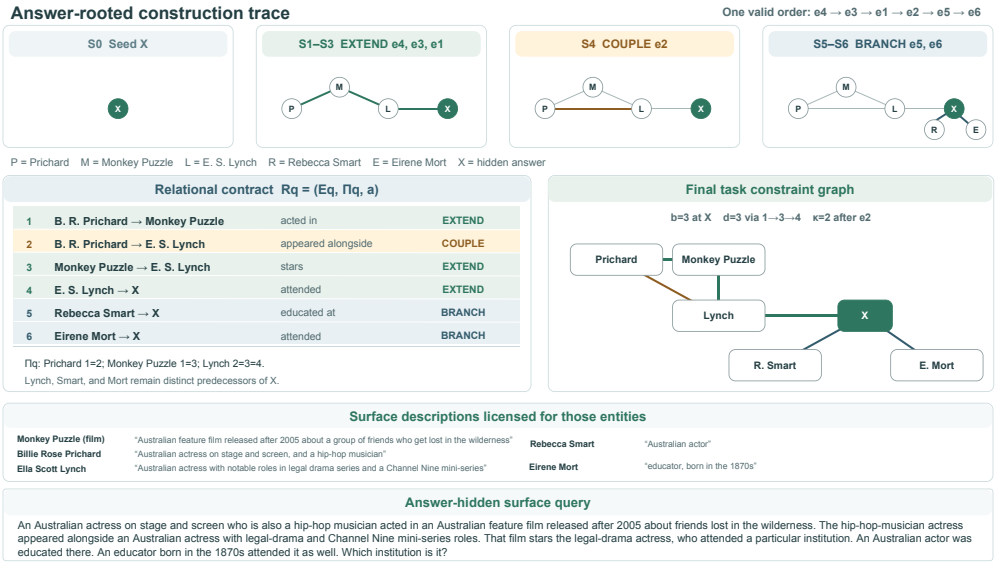


Figure 8: **A representative CARTA item in aligned construction, contract, graph, and query views.** The top row shows one valid answer-rooted construction order, $e_4 \to e_3 \to e_1 \to e_2 \to e_5 \to e_6$. The contract lists the six verified obligations E_q, the identity constraints Π_q, and the hidden answer a. The final graph has $\phi(G) = (b, d, \kappa) = (3, 3, 2)$. The bottom rows show the licensed surface descriptions and the answer-hidden query.