CAUSALMIX: DEFINING AND BENCHMARKING HYBRID CAUSAL DISCOVERY

Anonymous authors
Paper under double-blind review

ABSTRACT

Classical causal discovery uses numerical observations and interventions, while large language models (LLMs) also bring knowledge associated with variable names. A graph-recovery score alone does not reveal which information source supported the prediction. We introduce CAUSALMIX, a simulation and evaluation suite for hybrid causal discovery that compares names-only, data-present, anonymized, and classical data-only conditions under controlled graph and data budgets. The benchmark separates output validity, name effects, local responses to data, and accurate recovery. Our evaluation spans six reference networks and four synthetic graph configurations with 5–50 variables. On the evaluated reference graphs, recovery is strongly name-mediated, while gains from numerical evidence are small or unstable. These name effects may reflect domain knowledge or benchmark recall; our controls do not distinguish the two. We also post-train Qwen3-4B with supervised fine-tuning (SFT) followed by group relative policy optimization (GRPO). Post-training improves output validity, but adding data improves Sachs recovery only when names are anonymized, and recovery on 20–50-variable graphs remains far below a classical interventional baseline. CAUSALMIX provides a reproducible protocol for testing whether hybrid methods use supplied evidence reliably, rather than inferring evidence use from aggregate scores.

1 INTRODUCTION

Causal discovery in scientific domains rarely starts from numbers alone. Practitioners know what the variables mean, which mechanisms are plausible, and which relations would contradict domain knowledge; at the same time, observed and post-intervention samples provide evidence under causal, sampling, and measurement assumptions. Classical causal-discovery methods exploit the numerical evidence but ignore variable meanings, while LLM-based methods exploit names, descriptions, and domain knowledge but may recover familiar benchmark structure from semantic priors rather than from the supplied data. When such a method outputs a plausible graph, a headline score alone does not reveal whether the method used semantics, numerical evidence, or both. We call this setting *hybrid causal discovery* (HYBRID-CD): the learner is given semantic side information (variable names, descriptions, domain text) and numerical evidence (observational and optionally interventional samples), and must recover a causal graph from either or both.

The regime matters because many proposed uses of LLMs in causal discovery are motivated precisely by the hope that semantic priors and statistical evidence can complement each other. It is hard because the two channels can support the same edge for different reasons, and standard graph-recovery metrics do not attribute performance to the information source that produced it. Thus the scientific question is not only whether a method scores high, but whether it improves beyond what variable names alone provide and whether that improvement survives when names are anonymized.

Recent LLM-assisted methods target this middle ground by adding statistical summaries to prompts, decomposing graph construction into LLM queries, or combining language-model knowledge with data-driven evidence (Abdulaal et al., 2023; Jiralerspong et al., 2024; Takayama et al., 2025; Peng et al., 2026). However, they are evaluated under different graphs, sample counts, prompt formats, variable-name conditions, and classical baselines, so reported gains cannot be attributed to inputs: they may come from memorized variable names, the supplied data, the data representation, or their interaction. The field lacks a controlled protocol that compares semantic-only LLM prompting,

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.

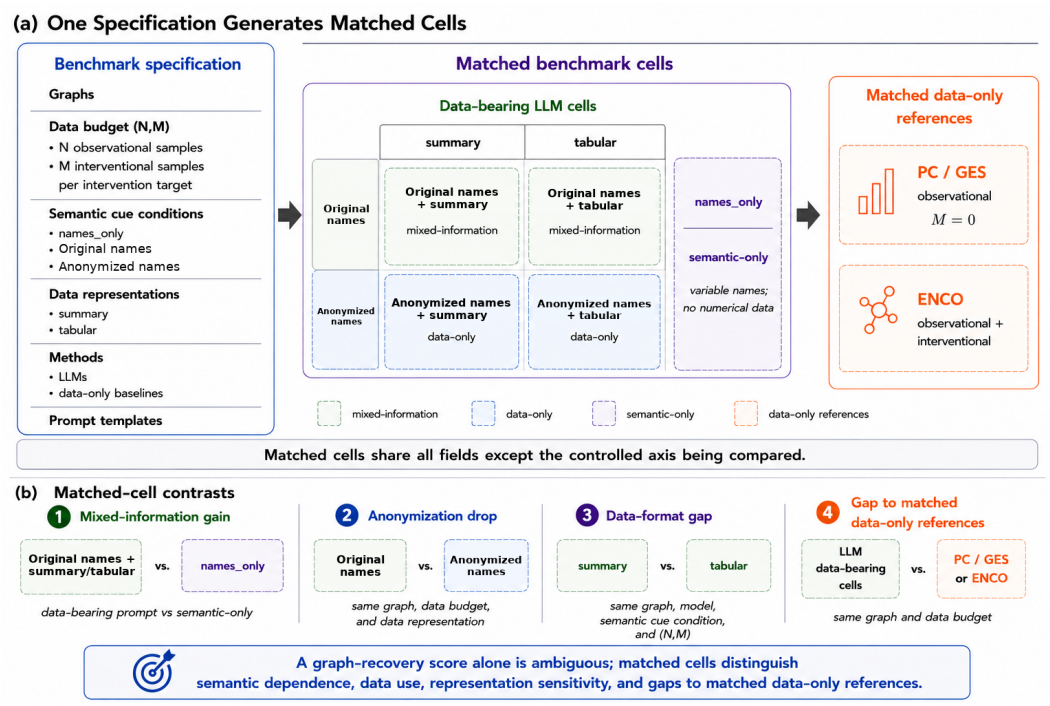

Figure 1: Overview of CAUSALMIX. A benchmark specification expands graphs, data budgets, representations, models, and baselines into matched evaluation cells. Each cell fixes the graph and method while varying one axis (semantic cue, data budget, or data representation), enabling attribution of recovery scores to information sources rather than to confounded experimental setups.

data-only classical baselines, and hybrid methods on the same graph and data. Concretely, such a protocol must isolate four levers—a `names_only` prompt that estimates what an LLM recovers from variable names with no data, original-name and anonymized-name prompts at the same data budget that test dependence on names, two data representations (`summary` and `tabular`) that probe format sensitivity, and classical methods run on the same samples that measure what those algorithms recover under their assumptions—and report all four alongside any headline score.

Contributions. We introduce CAUSALMIX, a dataset and evaluation suite for testing whether LLM-assisted causal discovery uses numerical evidence beyond variable-name cues. It combines graph simulators, prompt builders, matched evidence conditions, and classical causal discovery baselines in a reproducible protocol. We also post-train Qwen3-4B with supervised fine-tuning (SFT) followed by group relative policy optimization (GRPO) and evaluate it under the same controls.

What the evaluation establishes. The evaluation covers ten graph configurations with 5–50 variables. Matched data/no-data pairs test the effect of supplying observational and interventional evidence. SFT+GRPO improves output validity (331/360 versus 298/360 valid graphs across ten graphs), yet on repeated Sachs, adding data raises F1 only with anonymized names ($+0.134$, versus -0.002 with original names). On 20–50-node graphs, its F1 of 0.005–0.115 remains far below ENCO (0.609–0.790), and data-use gains span -0.112 to $+0.079$. Improved validity therefore does not by itself establish consistent numerical-evidence integration; within-label no-data controls and output-validity rates remain necessary.

2 PROBLEM FORMULATION AND RELATED BENCHMARKS

We formalize *hybrid causal discovery* (HYBRID-CD): causal graph recovery given semantic side information and numerical observational or interventional evidence. The formulation makes explicit the attribution problem that hybrid and LLM-assisted pipelines implicitly raise but rarely test.

Setup. Let $X_1, \ldots, X_d$ be random variables and $G = (\mathcal{V}, \mathcal{E})$ a directed acyclic graph (DAG) on $\mathcal{V} = \{1, \ldots, d\}$ with adjacency matrix $A \in \{0,1\}^{d \times d}$ (Peters et al., 2017). A HYBRID-CD method takes *semantic side information* $S = \{s_i\}_{i=1}^{d}$ associated with each X_i (e.g., names or descriptions), *observational data* $\mathcal{D}_{\text{obs}} = \{x^{(k)}\}_{k=1}^{N}$, and optional *interventional data* $\mathcal{D}_{\text{int}} = \{\mathcal{D}_{\text{int}}^{(i)}\}_{i \in \mathcal{I}}$ over intervention targets $\mathcal{I} \subseteq \mathcal{V}$, and returns an estimate $\hat{A} = \mathcal{A}(S, \mathcal{D}_{\text{obs}}, \mathcal{D}_{\text{int}})$. Each $\mathcal{D}_{\text{int}}^{(i)}$ contains samples from the post-intervention distribution induced by replacing the mechanism of X_i with an exogenous P_{int} (a perfect stochastic intervention (Eberhardt & Scheines, 2007)). We write (N, M) for the balanced budget with N observational and M interventional samples per target, where $M = 0$ denotes no interventional data.

Method classes. A HYBRID-CD instance offers two admissible evidence channels: S and $(\mathcal{D}_{\text{obs}}, \mathcal{D}_{\text{int}})$. Methods are classified by which they consume. *Data-only* methods use only $(\mathcal{D}_{\text{obs}}, \mathcal{D}_{\text{int}})$ and include classical algorithms such as PC (Spirtes et al., 2000), GES (Chickering, 2002), and ENCO (Lippe et al., 2022). *Semantic-only* methods use only S and include zero-shot LLM prompting from variable names. *Hybrid* methods use both, and most recent LLM-assisted pipelines fall in this class: they combine semantic priors with statistical summaries, candidate graphs, or edge constraints. Classical causal discovery and prompt-based LLM-CD are therefore limit cases of HYBRID-CD rather than separate settings, and whether a hybrid method actually benefits from the channels it receives is an empirical attribution question.

Identifiability and references. We adopt standard causal-discovery assumptions: causal sufficiency, no selection bias, acyclicity, Markov, and faithfulness (Peters et al., 2017). These define what numerical evidence can identify in principle—observational data identify only the Markov equivalence class of G, while perfect stochastic interventions can break equivalences and, with sufficient target coverage, identify the full DAG in the limit (Eberhardt & Scheines, 2007). Both channels can be informative or misleading: numerical evidence depends on the stated causal and sampling assumptions, while S may encode domain knowledge, benchmark recall, or incorrect associations. The known generating DAG is the scoring target; classical algorithms are references under their assumptions, not gold standards. Attribution on a fixed graph requires three matched references: a semantic-only control at $N = M = 0$, a data-only classical baseline at the same (N, M), and a semantic-cue control replacing S with anonymized placeholders. CAUSALMIX (Section 3) makes these references routine outputs of a single benchmark specification.

Related benchmarks. Existing benchmarks cover endpoints of HYBRID-CD but not the matched attribution problem. Causal-reasoning benchmarks test textual or symbolic causal reasoning (Jin et al., 2023; Zhou et al., 2024; Chen et al., 2024); classical benchmarks test statistical recovery from numerical data (Mooij et al., 2016; Chevalley et al., 2025); and LLM causal-graph benchmarks test graph construction from names or metadata (Babakov et al., 2025; Roy et al., 2025). None of these holds a graph/data condition fixed while jointly varying semantic cues, observational and interventional evidence, data representation, and matched semantic-only, hybrid, and data-only baselines—the joint control that motivates CAUSALMIX. Table 1 in Appendix C summarizes benchmark support for each control.

3 CAUSALMIX: SIMULATION AND EVALUATION SUITE

CAUSALMIX operationalizes HYBRID-CD by turning graphs, data budgets, data representations, models, and baselines into matched evaluation cells. We write CAUSALMIX for the general suite and CAUSALMIX V1 for the fixed released instance used in this paper.

Design principles. CAUSALMIX is built around three principles that make attribution part of every measurement. (i) *Input channels are experimentally controlled*: semantic cues, data budget, and data representation are independently controlled axes, so recovery scores can be compared across names, numerical evidence, and representations. These contrasts measure dependence on inputs, not the mechanism learned in pretraining. (ii) *Cells are matched attribution tests*: within each comparison the graph and scoring code are held fixed and only the controlled axis varies, yielding semantic-only, hybrid, anonymized, and data-only cells under identical scoring. (iii) *Progress is measured against bracketing references*: a hybrid result is informative only when reported alongside the matched names-only reference, the anonymized control, and the matched data-only reference at the same data budget.

Cells and cell families. A benchmark specification contains graphs, semantic cue conditions, data budgets (N, M), data representations, LLMs, data-only baselines, and seeds; a *benchmark cell* is one fixed combination of these fields plus a prompt template. Matched cells share all fields except the axis being compared, so one specification expands into four cell families: *semantic-only* (variable names without data, `names_only`), *hybrid* (original names + data), *anonymized* (placeholder names + the same data), and *data-only* (classical methods on the same data). All cells are scored against the same ground-truth adjacency matrix with the same code, so every contrast changes exactly one information source at a time.

Terminology and a matched example. A *semantic cue* is text associated with a variable, commonly its name or description. A *matched attribution test* changes the selected input factor while fixing the graph, numerical realization and seed, data budget, variable order, model and decoding settings, *output format* (adjacency-matrix or edge-list JSON), and scoring. For example, a Sachs pair with original names and anonymized names contains identical samples: labels such as Raf and Mek become placeholders such as `X1` and `X2`, with the same column order. We call an identifier *neutral* when it carries no domain-specific variable meaning. A data/no-data pair retains its label condition and omits only the numerical evidence; a summary/tabular pair changes its serialization. The *name effect* is the F1 difference between original and anonymized names; it may reflect domain knowledge or benchmark recall. Anonymization does not distinguish these sources. Studies with different output formats are analyzed as distinct protocols, and matching data budgets alone does not establish that two methods received identical sample arrays.

Simulation engine. For each graph G and budget (N, M), the engine samples N observational examples from the parameterized joint distribution of G, and when $M > 0$ samples M interventional examples per target by perfect stochastic intervention (Eberhardt & Scheines, 2007; Lippe et al., 2022): the target's incoming edges are removed, the target is redrawn from a replacement distribution P_{int}, and downstream variables are resampled ancestrally. In CAUSALMIX V1's bnlearn suite, $P_{\text{int}}(X_i) = \text{Uniform}(\mathcal{S}_i)$ over the state space of X_i; this is configurable. Specifications that supply pre-sampled arrays (as in the Synthetic-25 tier) are treated as fixed data artifacts. Every generated cell records its source graph, data path, prompt, budget, and seed for exact regeneration.

Data representations. The sampled data are presented to LLMs in two forms. `summary` encodes each observational or interventional regime as a sparse empirical histogram, while `tabular` prints the raw rows. Sections D.2 and I give templates and a token-scaling analysis.

Released instance. CAUSALMIX V1 fixes graph sources, seeds, generated data, prompt templates, conditions, budgets, reference outputs, and scoring scripts. The release contains 14 graph instances in three tiers: six real reference graphs from bnlearn (Scutari, 2010), six Synthetic-25 topology controls from ENCO (Lippe et al., 2022), and two large challenge graphs. Section D.1 reports the full inventory, graph scale, and prompt-token burden.

4 EVALUATION SETUP

Graphs and studies. The graph suite contains Cancer (5 variables), Earthquake (5), Asia (8), Sachs (11), Child (20), and Alarm (37), plus generated Chain and Jungle graphs at 25 and 50 variables. The 25-node Chain/Jungle source labels are `A`, `B`, ..., `Y`, without domain-specific meanings. Anonymized runs use `X1`, `X2`, ..., `Xd`, where d is the number of variables; labels are assigned after the column-order permutation. Three studies use this suite. The *attribution study* crosses Cancer, Earthquake, Asia, Sachs, Child, and Alarm with ten LLMs from the GPT-5, Llama-3.1, Qwen2.5, and Qwen3 families (OpenAI, 2025b;a; Dubey et al., 2024; Qwen Team, 2024; 2025), names-only prompts, original and anonymized names with data, and summary/tabular representations. We call it an attribution study because we control every factor: any two cells that differ in exactly one factor form a matched attribution test (Section 3), so differences in recovery can be attributed to names, data, or representation. The *post-training study* fine-tunes Qwen3-4B with SFT followed by GRPO and compares it with the pretrained model on the ten graph configurations.

Classical references are observational PC (Spirtes et al., 2000) and GES (Chickering, 2002) and interventional ENCO (Lippe et al., 2022). We also evaluate five benchmark-adapted LLM-CD strategies: Causal-LLM and its data-driven branch (Roy et al., 2025), pairwise prompting (Vashishtha

et al., 2023; Kıcıman et al., 2024), BFS-style querying (Jiralerspong et al., 2024), and SCP (Takayama et al., 2025).

Validity and recovery. The attribution study requests an adjacency matrix; the post-training and scaling studies request an edge list, in which self-loops are invalid. An output is valid if it parses as a complete graph over the expected variables; acyclicity is reported separately. We measure recovery with directed-edge F1 (a valid graph with no true positives scores zero) and SHD. The attribution study reports F1 over valid outputs alongside validity; the other studies score invalid outputs as F1 zero. Input-context rejections and API failures are marked unavailable rather than scored zero. Data-use gain is F1 with data minus F1 without data under the same naming condition (Section E).

5 BENCHMARKING LLM-ASSISTED CAUSAL DISCOVERY

We use CAUSALMIX V1 to benchmark LLMs on the ten graph configurations of Section 4. Unless otherwise noted, reported values are directed-edge F1 with SHD in parentheses; attribution-study averages use valid parsed outputs, whereas post-training analyses score invalid outputs as zero. We refer to $(N, M) = (5000, 200)$ as the primary budget.

The evaluation answers four research questions:

- **RQ1.** How much structure can LLMs recover from names alone, including on larger graphs?
- **RQ2.** Are gains from numerical evidence attributable to data rather than names or representation?
- **RQ3.** Can post-training move LLMs toward combining semantic priors and numerical data?
- **RQ4.** How do LLM-CD strategies compare with semantic-only and matched data-only references?

RQ1, RQ2, and RQ4 draw on the attribution study, with Sachs supporting the denser budget sweeps and adapted-method comparisons; they demonstrate the diagnostic value of the suite (our first contribution). RQ3 applies the same controls to our post-training procedure (our second contribution). Table 5 reports the ten-model parse-validity profile, and the discussion emphasizes the GPT and Qwen2.5 slices with high parse validity.

5.1 RQ1: RECOVERY FROM VARIABLE NAMES ALONE

Under `names_only`, models see variable names but no samples. Figure 2 (left) shows strong recovery on small graphs with original names: across the 15 displayed cells for Cancer, Earthquake, and Asia, seven are exact recoveries and ten have F1 ≥ 0.75. Recovery is weaker on Sachs, where all five models remain below F1 0.52.

The larger graphs reveal differences in both recovery and output reliability (Figure 2, right). On Child-20, GPT-5.2-Pro reaches 0.78 ± 0.03 F1 with 5/5 valid outputs, while GPT-5-Mini reaches 0.51 ± 0.03 with 4/5 valid outputs. On Alarm-37, GPT-5.2-Pro reaches 0.58 ± 0.05 with 3/5 valid outputs, whereas GPT-5-Mini produces no valid graph. These means and standard errors condition on valid outputs. Names alone can therefore support substantial recovery on larger graphs, although valid graph generation remains unreliable in some conditions.

5.2 RQ2: DATA GAINS DEPEND ON SEMANTIC CUES AND REPRESENTATION

At the primary Sachs budget, adding numerical evidence does not produce a stable improvement over the semantic-only baseline. Instead, the measured gain depends on how the same evidence is presented and whether prompts use original names or anonymized names. Figure 3 shows observational and interventional prompts can move F1 in either direction relative to `names_only`, and switching between `summary` and `tabular` formats flips the result between paired cells.

The anonymization test (Figure 10) measures the name effect: replacing original names with generic $X_1, \ldots, X_d$ reduces valid-output F1 in 26 of 33 supported comparisons. Each side must have at least three valid outputs; seven of the 40 candidate comparisons are unavailable at this threshold. Invalid anonymized outputs are not substituted with zero in this valid-only contrast. These results motivate reporting both name controls and output validity alongside any data-grounded recovery claim.

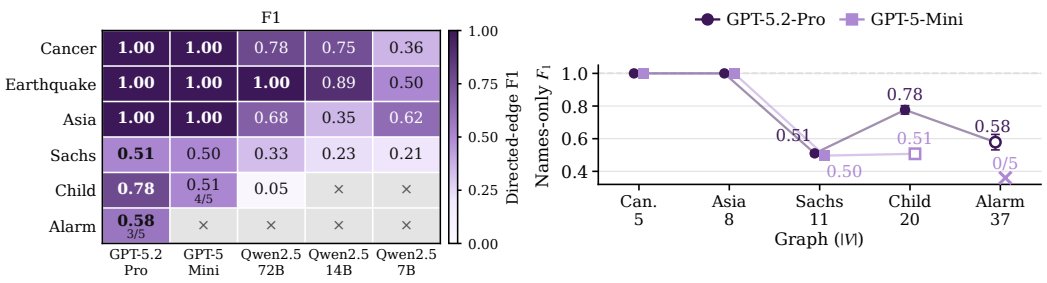

Figure 2: Names alone recover substantial graph structure. Left: mean directed-edge F1 over valid `names_only` outputs on six graphs, with a fixed model roster. Counts identify partial validity; × means no valid graph output and hence no F1 estimate. Attempt counts and failure causes are reported in Appendix G.2. Right: GPT-5.2-Pro and GPT-5-Mini `names_only` F1 across graph sizes, with standard errors over valid runs. Filled markers indicate full validity, hollow markers partial validity, and × no valid output.

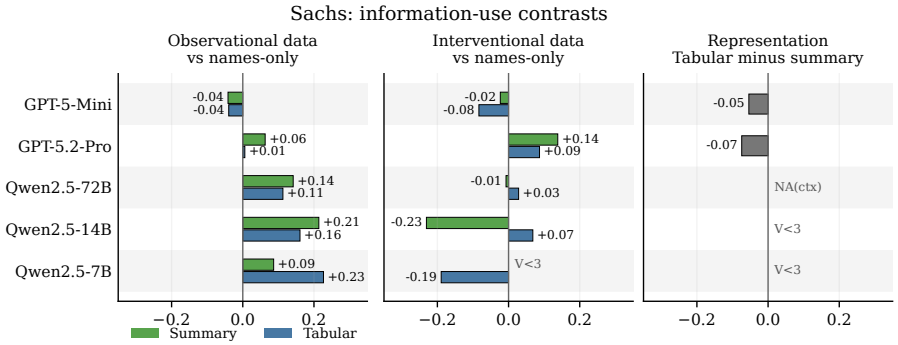

Figure 3: Matched information contrasts on Sachs for pretrained models. Differences for adding samples and changing representation use valid-output condition means, including zero-F1 graphs. `V<3` marks fewer than three valid outputs in a condition; `NA(ctx)` marks input-context overflow. Appendix G.2 separates output failures from graph-recovery errors.

5.3 RQ3: Post-Training as a Step Toward Combining Priors and Data

The controls above identify a concrete repair target for LLM-CD systems: the model must emit a parseable graph, avoid relying solely on familiar names, and respond appropriately to numerical evidence. As a preliminary step beyond zero-shot prompting, we use CausalMix's own simulation engine to construct a verifier-based training curriculum and post-train Qwen3-4B-Thinking on it.

Verifier curriculum. The same generator that produces evaluation cells supplies training data with known causal graphs. We draw observational and interventional datasets at multiple budgets and seeds, using both original and anonymized variable names. Format/graph-target SFT first trains structured graph prediction through next-token supervision on gold adjacency matrices. Starting from this initializer, GRPO samples completions and uses a ground-truth verifier to reward graph recovery, output format, and reasoning structure. The simulator thus supplies both supervised targets and feedback for policy optimization. Appendices F.1 and F.2 give the curricula, objectives, and training settings.

Findings. Figure 4 compares pretrained Qwen3-4B with SFT+GRPO across ten graphs, scoring invalid outputs as zero. Post-training increases mean directed-edge F1 in 11 of the 16 displayed conditions and produces 291/320 valid graphs, compared with 259/320 for the pretrained model. Gains are largest on the smaller graphs with original names, while anonymized-name gains are mixed. We also observe that the improvements in output validity do not transfer: the fine-tuned model outperforms the base model mainly on graphs seen during post-training, while on larger graphs we see little to no improvement. Matched data/no-data comparisons show uneven evidence-use gains, and recovery on larger graphs remains below ENCO. We thus view post-training as a preliminary

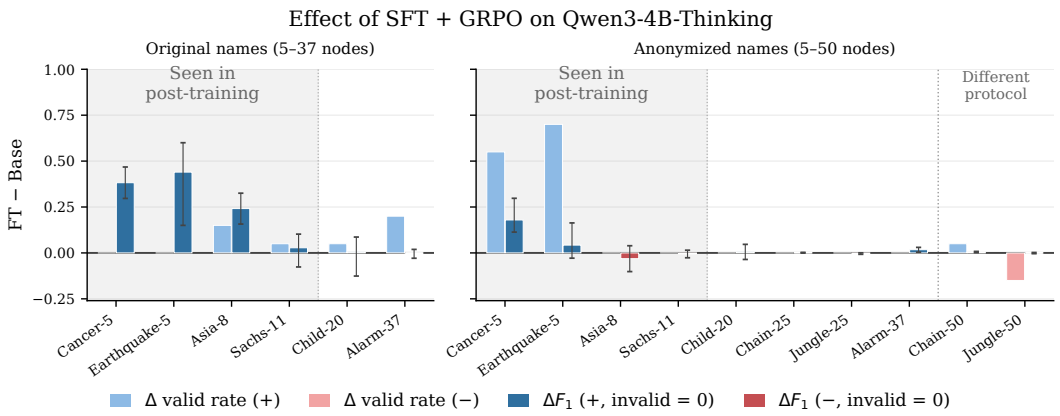

Figure 4: Effect of SFT+GRPO relative to pretrained Qwen3-4B with data summaries at $(N, M) = (1000, 10)$. Paired bars show changes in valid-output rate (light) and directed-edge F1 (dark); blue denotes gains and red declines. Invalid outputs receive F1 zero. Gray backgrounds mark families seen in post-training. Chain/Jungle appear only under anonymized names because their source identifiers are neutral. Each condition uses 20 matched attempts. The 50-node graphs appear only in the anonymized-names panel, beyond the rightmost dotted divider; they use format instructions only, whereas 5–37-node runs enforce format during generation.

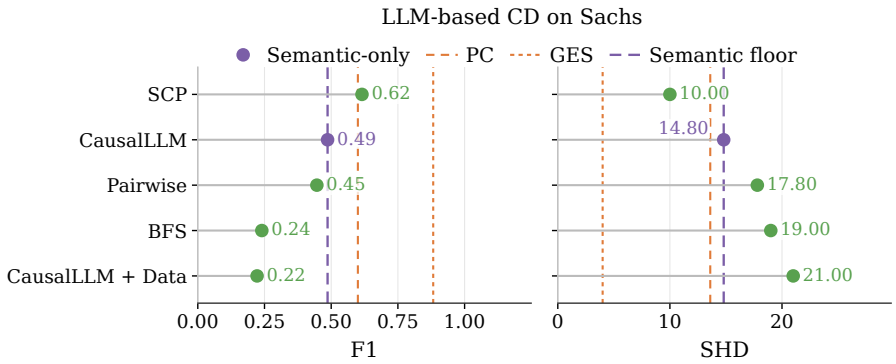

Figure 5: Adapted LLM-CD strategies on Sachs at $N = 5000$ observational samples (Roy et al., 2025; Vashishtha et al., 2023; Kıcıman et al., 2024; Jiralerspong et al., 2024; Takayama et al., 2025), compared with `names_only`, PC, and GES under the same protocol.

step toward combining priors and data, with reliable evidence integration remaining an open problem. Appendices H.4 and H.2 report the detailed evidence-use diagnostics.

5.4 RQ4: Gap to Semantic and Data-Only References

We next ask whether existing LLM-based strategies improve beyond the semantic-only reference under a common protocol, by evaluating five LLM-CD strategies—Causal-LLM and its data-driven branch (Roy et al., 2025), pairwise prompting (Vashishtha et al., 2023; Kıcıman et al., 2024), BFS-style querying (Jiralerspong et al., 2024), and SCP (Takayama et al., 2025)—under our shared graph, data, and scoring protocol. Figure 5 shows that four of the five strategies fail to exceed `names_only`; SCP is the only exception, reaching F1 $= 0.62$ (slightly above PC, below GES). Causal-LLM reaches F1 $= 0.49$, pairwise 0.45, BFS-style 0.24, and the Causal-LLM data-driven branch 0.22.

In the Sachs budget sweeps, classical methods recover more structure when the data match their assumptions. Figure 6 fixes Sachs at $N = 1000$ and increases M: the GPT-5.2-Pro original-name `summary` prompt moves from F1 $= 0.51$ at $M = 0$ only to 0.56–0.57 at $M \in \{50, 100, 200\}$, while ENCO approaches perfect recovery as the intervention budget grows. The relative strengths still

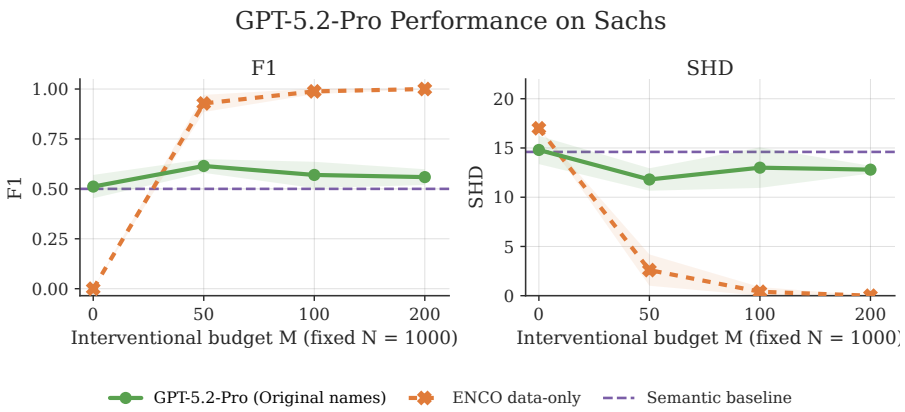

Figure 6: Sachs interventional-budget sweep with $N = 1000$ fixed. Increasing M has little effect on the GPT-5.2-Pro original-name `summary` prompt, while ENCO reaches near-perfect recovery once interventional samples are available.

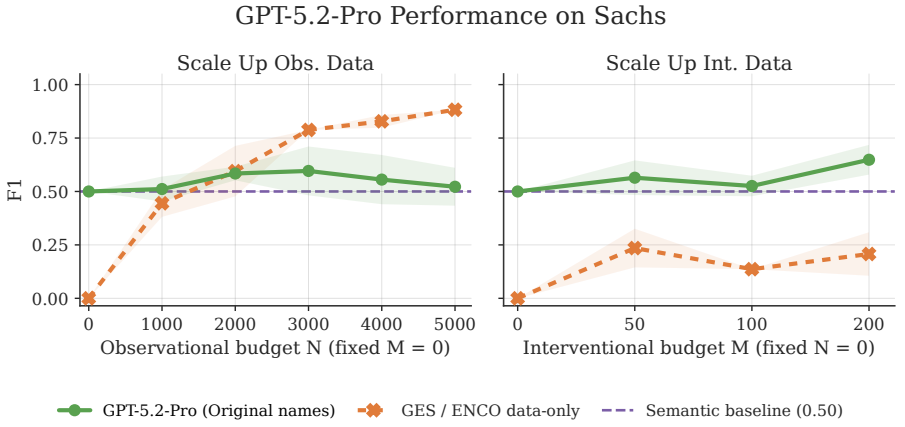

Figure 7: Observational-only and interventional-only sweeps on Sachs. Left ($M = 0$): GES improves with observational sample size, while the GPT-5.2-Pro `summary` prompt stays near `names_only`. Right ($N = 0$): interventional samples lift GPT-5.2-Pro above `names_only`, while ENCO is weak without observational samples.

depend on regime (Figure 7): GES benefits more than GPT-5.2-Pro from observational sample size, while GPT-5.2-Pro gains more than ENCO from intervention-only data—suggesting a complementary role for LLMs when a classical solver is outside its intended regime. CAUSALMIX makes this target measurable: future LLM-CD evaluations should report whether supplied data improve recovery beyond the within-label no-data control and how the result compares with classical references under the stated assumptions.

6 DISCUSSION

The controls in CAUSALMIX recast LLM-CD evaluation as an attribution problem: a high score is informative only if it is paired with the `names_only`, anonymized, and matched data-only references that identify which information source produced it. One immediate implication is that high F1 on named public benchmarks should not, by itself, be read as evidence of data-grounded causal discovery; long-used graphs that are easy to recover from semantics serve as stress tests for that distinction. The post-training experiment in Section 5.3 shows the promise and difficulty of verifier-generated training data: parse validity and recovery improve in several cells, while robust evidence use on harder graphs remains open. We see four research directions that the released suite is designed to support.

Combining evidence channels. We argue that future LLM-CD systems need principled ways to combine statistical outputs, semantic judgments, and expert constraints. Classical methods can supply candidate graphs, separating sets, intervention effects, and uncertainty estimates; LLMs can translate domain knowledge into solver constraints, critique candidate edges, and request follow-up measurements. The open problem is aggregation *with attribution*: a hybrid system should report when expert constraints override statistical evidence, when classical outputs serve as priors, and whether iterative refinement improves reliability under matched controls.

Eliciting causal inductive biases. In our experiments, we found that names-only recovery and anonymization drop performance, which shows that LLMs already bring strong priors. The useful question is then how to convert those priors into testable causal hypotheses—acyclicity, intervention semantics, conditioning, confounding checks, sparsity—rather than unsupported edges. Algorithms can enforce these biases through edge queries, parent search, solver priors, or graph verifiers, and evaluation should test whether the elicited behavior remains stable under row, column, and edge-list permutations while still responding to real changes in sample size and intervention coverage.

Compressing data into prompts. Format sensitivity and context limits make data presentation a central problem. Raw tabular prompts scale with variables, samples, and intervention regimes; in our suite the 37-node Alarm graph reaches 1.4M tokens in the largest data-bearing representation, beyond most current context windows (Table 2). Hence, useful representations may include empirical summaries, retrieval over variable blocks, local subgraph queries, uncertainty-ranked edge lists, and statistical front ends that translate raw samples into causal summaries. The observed summary–tabular gaps may reflect competing mechanisms—compression versus sample-level alignment—which the released format axis is designed to diagnose.

Training with verifiable causal rewards. The evaluated GRPO pipeline rewards graph agreement, format, and trace structure, yet does not produce uniformly beneficial evidence use. This motivates a broader question: what should a causal-discovery verifier reward to improve evidence-conditioned recovery? Single metrics such as F1, SHD, or acyclicity capture only part of the target; so do robustness checks for anonymization, intervention sensitivity, and representation stability. A stronger approach is multi-objective and comparative: train verifiers to choose which of two candidate graphs better explains the supplied evidence, respects expert constraints, and avoids unsupported edges, with curricula that progress across graph size, data budget, and intervention coverage.

Scope. Our attribution study covers Cancer, Earthquake, Asia, and Sachs, all with at most 11 variables, so near-exact `names_only` recovery on them is a stress test rather than a capability claim. The scaling study adds Child, Alarm, and generated Chain/Jungle graphs with 25 or 50 nodes at a single budget, but it does not establish generalization across all models, representations, or graph families. Diabetes and Pigs remain released scaling assets. Anonymized prompts probe name effects but do not separate domain knowledge from benchmark recall. Hence, new named scientific domains are needed to push the diagnosis further. The generated controls test unfamiliar structures but do not eliminate broader pretraining associations. We release our codebase for the community to expand the scope of LLM-CD evaluation.

7 Conclusion

We formalized hybrid causal discovery and released CausalMix, a simulation and evaluation suite that turns the regime into matched attribution tests by varying semantic cues, data budgets, and data representations on a fixed graph alongside semantic-only and matched data-only references. Across ten graph configurations with 5–50 variables, the suite reveals that current LLMs recover substantial structure from variable names alone, that their gains from added data depend on labels and representation, and that classical methods remain the strongest reference when the tabular problem is well specified. The same controls make post-training measurable: SFT+GRPO substantially improves the output validity of Qwen3-4B, and the matched data/no-data contrasts pinpoint evidence use as the capability that training must target next. We also pose a list of open questions that the suite is designed to answer (Section 6). We hope CausalMix becomes a shared standard for measuring whether future LLM-CD methods add useful, attributable, and transferable signal under matched evidence controls.

AI USE STATEMENT

Generative AI tools were used for grammar checking, language polishing, code debugging, visualization refinement, and assistance with collecting and interpreting saved experimental results. The use of LLMs as evaluated causal-discovery systems and to generate training rationales is described in Sections 4, 5.3, F.1 and F.2. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI. Finally, before submission, we used generative AI to clean up the codebase.

REPRODUCIBILITY STATEMENT

The output formats, availability categories, validity rules, and metrics are specified in Sections 4, E and G.2. Section H records the scaling, post-training, and diagnostic study protocols and resampling units; Appendix F.2 documents the SFT+GRPO procedure and Section F.1 documents the SFT stages. The accompanying analysis artifacts retain per-row source paths, protocol identifiers, and count-based F1, distinguishing unavailable attempts from invalid generated graphs. Prompt templates and graph inventories appear in Sections D and I.

REFERENCES

Ahmed Abdulaal, Adamos Hadjivasiliou, Nina Montana-Brown, Tiantian He, Ayodeji Ijishakin, Ivana Drobnjak, Daniel C. Castro, and Daniel C. Alexander. Causal Modelling Agents: Causal Graph Discovery through Synergising Metadata- and Data-driven Reasoning. October 2023. URL https://openreview.net/forum?id=pAoqRlTBtY.

Sawal Acharya, Terry Jingchen Zhang, Andrew Kim, Sun Xianlin, Anahita Haghighat, Maximilian Mordig, Rahul Babu Shrestha, Clijo Jose, Yahang Qi, Pepijn Cobben, Bernhard Schölkopf, Mrinmaya Sachan, and Zhijing Jin. CauSciBench: Assessing LLM causal reasoning for scientific research. In *MATH-AI 2025*, 2025. URL https://openreview.net/forum?id=RHFK3hS7Zv.

Nikolay Babakov, Ehud Reiter, and Alberto Bugarín-Diz. CausalGraphBench: a Benchmark for Evaluating Language Models capabilities of Causal Graph discovery. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 240–258, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.acl-srw.16. URL https://aclanthology.org/2025.acl-srw.16/.

Taiyu Ban, Lyuzhou Chen, Derui Lyu, Xiangyu Wang, and Huanhuan Chen. Causal structure learning supervised by large language model. *arXiv preprint arXiv:2311.11689*, 2023a.

Taiyu Ban, Lyuzhou Chen, Derui Lyu, Xiangyu Wang, Qinrui Zhu, Qiang Tu, and Huanhuan Chen. Integrating large language model for improved causal discovery. *IEEE Transactions on Artificial Intelligence*, 2023b.

Sirui Chen, Mengying Xu, Kun Wang, Xingyu Zeng, Rui Zhao, Shengjie Zhao, and Chaochao Lu. CLEAR: Can Language Models Really Understand Causal Graphs?, June 2024. URL http://arxiv.org/abs/2406.16605. arXiv:2406.16605 [cs.CL].

Mathieu Chevalley, Yusuf H. Roohani, Arash Mehrjou, Jure Leskovec, and Patrick Schwab. A large-scale benchmark for network inference from single-cell perturbation data. *Communications Biology*, 2025. doi: 10.1038/s42003-025-07764-y.

David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554, 2002.

Juncheng Dong, Yiling Liu, Ahmed Aloui, Vahid Tarokh, and David E. Carlson. Care: Turning llms into causal reasoning expert. *arXiv.org*, 2025.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Frederick Eberhardt and Richard Scheines. Interventions and causal inference. *Philosophy of Science*, 74(5):981–995, 2007. ISSN 00318248, 1539767X. URL http://www.jstor.org/stable/10.1086/525638.

Stefan Hegselmann, Alejandro Buendia, Hunter Lang, Monica Agrawal, Xiaoyi Jiang, and David Sontag. TabLLM: Few-shot Classification of Tabular Data with Large Language Models. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pp. 5549–5581. PMLR, April 2023. URL https://proceedings.mlr.press/v206/hegselmann23a.html.

Mingyu Jin, Qinkai Yu, Jingyuan Huang, Qingcheng Zeng, Zhenting Wang, Wenyue Hua, Haiyan Zhao, Kai Mei, Yanda Meng, Kaize Ding, Fan Yang, Mengnan Du, and Yongfeng Zhang. Cladder: Assessing causal reasoning in language models. In *Advances in Neural Information Processing Systems*, 2023.

Thomas Jiralerspong, Xiaoyin Chen, Yash More, Vedant Shah, and Yoshua Bengio. Efficient Causal Graph Discovery Using Large Language Models, July 2024. URL http://arxiv.org/abs/2402.01207. arXiv:2402.01207 [cs].

Elahe Khatibi, Mahyar Abbasian, Zhongqi Yang, Iman Azimi, and Amir M. Rahmani. Alcm: Autonomous llm-augmented causal discovery framework. *arXiv.org*, 2024.

Emre Kıcıman, R. Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language models: Opening a new frontier for causality. *Trans. Mach. Learn. Res.*, 2023.

Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal Reasoning and Large Language Models: Opening a New Frontier for Causality, August 2024. URL http://arxiv.org/abs/2305.00050. arXiv:2305.00050 [cs].

Phillip Lippe, Taco Cohen, and Efstratios Gavves. Efficient Neural Causal Discovery without Acyclicity Constraints, February 2022. URL http://arxiv.org/abs/2107.10483. arXiv:2107.10483 [cs, stat].

Stephanie Long, Tibor Schuster, and Alexandre Piché. Can large language models build causal graphs? In *NeurIPS Workshop on Causal Machine Learning for Real-World Impact*, 2022. arXiv:2303.05279.

Stephanie Long, Alexandre Piché, Valentina Zantedeschi, Tibor Schuster, and Alexandre Drouin. Causal discovery with language models as imperfect experts. *arXiv.org*, 2023.

Joris M. Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *Journal of Machine Learning Research*, 17(32):1–102, 2016. ISSN 1532-4435. URL http://jmlr.org/papers/v17/14-518.html.

OpenAI. Update to GPT-5 System Card: GPT-5.2, December 2025a. URL https://openai.com/index/gpt-5-system-card-update-gpt-5-2. Accessed 2026-05-03.

OpenAI. GPT-5 System Card, August 2025b. URL https://openai.com/index/gpt-5-system-card/. Accessed 2026-05-03.

Bo Peng, Sirui Chen, Lei Xu, and Chaochao Lu. CauScientist: Teaching LLMs to Respect Data for Causal Discovery, January 2026. URL http://arxiv.org/abs/2601.13614. arXiv:2601.13614 [cs].

Jonas Peters, Dominik Janzing, and Bernhard Scholkopf. *Elements of causal inference: foundations and learning algorithms*. MIT press, 2017.

Qwen Team. Qwen2.5 technical report, 2024. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Amartya Roy, N Devharish, Shreya Ganguly, and Kripabandhu Ghosh. Causal-LLM: A Unified One-Shot Framework for Prompt- and Data-Driven Causal Graph Discovery. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 8259–8279, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.439. URL https://aclanthology.org/2025.findings-emnlp.439/.

Marco Scutari. Learning bayesian networks with the bnlearn r package. *Journal of Statistical Software*, 35(3):1–22, 2010. doi: 10.18637/jss.v035.i03.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. doi: 10.48550/arXiv.2402.03300. URL https://arxiv.org/abs/2402.03300.

Peter Spirtes, Clark N. Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT Press, Cambridge, MA, 2 edition, 2000.

Yuni Susanti and Michael Färber. Knowledge graph structure as prompt: Improving small language models capabilities for knowledge-based causal discovery, 2024. URL https://arxiv.org/abs/2407.18752.

Yuni Susanti and Michael Färber. Can llms leverage observational data? towards data-driven causal discovery with llms, 2025. URL https://arxiv.org/abs/2504.10936.

Masayuki Takayama, Tadahisa Okuda, Thong Pham, Tatsuyoshi Ikenoue, Shingo Fukuma, Shohei Shimizu, and Akiyoshi Sannai. Integrating Large Language Models in Causal Discovery: A Statistical Causal Approach, May 2025. URL http://arxiv.org/abs/2402.01454. arXiv:2402.01454 [cs].

Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N. Balasubramanian, and Amit Sharma. Causal order: The key to leveraging imperfect experts in causal inference. *International Conference on Learning Representations*, 2023.

Guangya Wan, Yunsheng Lu, Yuqi Wu, Mengxuan Hu, and Sheng Li. Large language models for causal discovery: Current landscape and future directions. *International Joint Conference on Artificial Intelligence*, 2024.

Anpeng Wu, Kun Kuang, Minqin Zhu, Yingrong Wang, Yujia Zheng, Kairong Han, Baohong Li, Guan-Hong Chen, Fei Wu, and Kun Zhang. Causality for large language models. *arXiv.org*, 2024.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, Junyang Lin, Kai Dang, Kexin Yang, Le Yu, Mei Li, Minmin Sun, Qin Zhu, Rui Men, Tao He, Weijia Xu, Wenbiao Yin, Wenyuan Yu, et al. Qwen2.5-1M technical report, 2025. URL https://arxiv.org/abs/2501.15383.

Linying Yang, Vik Shirvaikar, Oscar Clivio, and Fabian Falck. A critical review of causal reasoning benchmarks for large language models. *arXiv.org*, 2024. doi: 10.48550/arXiv.2407.08029.

Yu Zhou, Xingyu Wu, Beicheng Huang, Jibin Wu, Liang Feng, and Kay Chen Tan. CausalBench: A Comprehensive Benchmark for Causal Learning Capability of LLMs, September 2024. URL http://arxiv.org/abs/2404.06349. arXiv:2404.06349 [cs].

A BROADER IMPACTS

CAUSALMIX is an evaluation benchmark rather than a deployed decision-making system, so its expected impact is mainly on research practice. For causal discovery, the benchmark provides a way to distinguish performance that comes from variable-name semantics or domain priors from performance that comes from statistical evidence in the data. By reporting semantic-only controls, anonymized data-bearing settings, and matched data-only references, it can support more cautious interpretation of LLM-based causal discovery results and encourage future methods to demonstrate data-dependent recovery rather than only plausible graph generation.

For LLM research, the benchmark offers a concrete setting for studying how language models combine textual semantics, memorized domain knowledge, prompt formatting, and numerical data. The resulting comparisons can help identify when an apparent reasoning gain is better explained by shortcut use, sensitivity to data representation, or prior knowledge embedded in variable names. We therefore view the downstream contribution as methodological: improving the evaluation of LLMs on causal discovery tasks and providing diagnostics that may inform the design of future models, prompts, and benchmarks. These results should not be read as evidence that LLM-generated graphs are ready for use as causal conclusions without additional domain-specific validation.

B LIMITATIONS

Scope and familiarity. The evaluation covers six reference networks and four synthetic scaling configurations. Each synthetic size/type cell uses one generating DAG. Coverage therefore extends across 5–50 variables without sampling a broad population of graph families. The scaling and post-training studies use specific budgets and checkpoints; they do not establish universal semantic dominance or model incapability. Neutral synthetic labels test structural scaling rather than semantic priors. Names-only and anonymized contrasts measure name effects without separating domain knowledge from benchmark recall. Anonymization also changes the naturalness of prompts.

Post-training identification. The SFT+GRPO checkpoint is trained on graph families that overlap the small reference graphs, so its gains there need not transfer to unseen graph families; Appendix H.3 tests transfer for the SFT stages only.

Both SFT stages contain gold graph targets, and there is no schema-only supervision control. Output-format learning, target imitation, and pretraining familiarity remain incompletely separated. Higher validity does not establish consistently improved evidence use.

Interfaces and failure causes. A prompt rejected for context length, or an API/runtime failure without an assessable completion, yields no graph-recovery estimate. A generated response may instead hit its output-token limit, violate graph constraints, or fail parsing. Missing stop metadata does not prove that incomplete output finished normally. Increasing an output limit and holding decoding/serialization fixed are necessary follow-ups before attributing a failure to model capability. Enforcing the output format during generation constrains syntax but does not ensure termination, unique edges, acyclicity, or correct recovery. The 50-node Qwen3 runs use format instructions only, while the 20–37-node runs enforce the format during generation; their difference is not a controlled estimate of graph-size effects.

Validity and uncertainty. Valid-only scores describe successful outputs, while invalid-as-zero scores also reflect generation reliability; we report the convention for each analysis. Valid zero-true-positive graphs receive zero F1 under both conventions. Five-realization attribution-study cells and four-order-cluster scale intervals remain descriptive. The 30-pair Sachs replication supports that specified contrast, not every model or graph. Repeated deterministic names-only prompts are not independent variable-order robustness trials. Neither the observed graph set nor the tested prompts constitute a random sample of scientific tasks.

Models, methods, and numerical matching. Broader checkpoint coverage remains conditional on each serving context and output protocol; context-limited checkpoints are not assigned causal-capability scores. The adapted LLM-CD methods test representative query strategies rather than

every engineering detail of the original systems. Hosted aliases may change, so future reruns need not be bitwise comparable. Larger-graph ENCO and LLM runs share nominal graph and budget settings, but exact sample-array identity has not been verified. They therefore provide numerical references, not strictly identical-dataset comparisons.

Causal assumptions. The generated cells assume causal sufficiency, DAGs, and known graph parameterizations. Observational samples generally identify only a Markov equivalence class. Latent confounding, selection bias, measurement error, cycles, and imperfect interventions lie outside this analysis. Semantic knowledge may help under misspecification; neither evidence channel is assumed universally superior.

C Additional Related Work

Prior work on LLMs for causal discovery falls into four groups: *direct inference*, where the LLM outputs a graph or causal order from variable metadata or summaries; *LLMs as prior knowledge*, where the LLM supplies constraints or edge priors to a statistical causal discovery (SCD) procedure; *posterior refinement*, where the LLM critiques or edits a graph produced by SCD; and *iterative graph refinement*, where the LLM repeatedly revises a graph using feedback computed from data. Surveys cover this area from both the causal-discovery and causality-for-LLMs perspectives (Wan et al., 2024; Wu et al., 2024; Kıcıman et al., 2023). We focus on the benchmarks closest to our setting and on why they leave room for a controlled benchmark of graph recovery from data.

C.1 Comparison with Existing Benchmarks

Table 1 compares benchmarks for causal reasoning, classical causal discovery, and LLM graph prediction. None holds a graph and data condition fixed while jointly varying semantic cues, data availability, and data format, with matched semantic-only, hybrid, and data-only baselines.

Table 1: Benchmark controls for Hybrid-CD evaluation. ✓denotes substantial support for a control, and ✗denotes absence. Columns indicate whether a benchmark supports full graph discovery, semantic cues or controls, observational data, interventional data, multiple data formats, and matched semantic-only, hybrid, and data-only baselines.

Benchmark / Dataset	Graph CD	Semantic Ctrl.	Obs. Data	Int. Data	Data Format Ctrl.	Matched Baselines
CLadder (Jin et al., 2023)	✗	✓	✗	✗	✗	✗
Tübingen Cause–Effect Pairs (Mooij et al., 2016)	✗	✗	✓	✗	✗	✗
CauSciBench (Acharya et al., 2025)	✗	✓	✓	✗	✗	✗
CausalGraphBench (Babakov et al., 2025)	✓	✓	✗	✗	✗	✗
Single-cell CausalBench (Chevalley et al., 2025)	✓	✗	✓	✓	✗	✗
LLM CausalBench (Zhou et al., 2024)	✓	✓	✓	✗	✓	✗
Causal-LLM suite (Roy et al., 2025)	✓	✓	✓	✗	✓	✗
CausalMix	✓	✓	✓	✓	✓	✓

Causal-reasoning benchmarks. CLadder (Jin et al., 2023) textualizes symbolic causal queries and evaluates formal causal inference in natural language. The LLM-oriented CausalBench (Zhou et al., 2024) evaluates correlation, skeleton recovery, and causality identification, and CLEAR (Chen et al., 2024) tests understanding of causal graphs across tasks and complexity levels. Yang et al. (2024) argue that many such benchmarks can be solved by retrieving background knowledge rather than by causal reasoning, and call for benchmarks grounded in interventions and counterfactuals. These benchmarks evaluate textual causal reasoning, not reconstruction of an explicit graph from observational and interventional data under controlled formats.

LLM causal-graph benchmarks. The closest benchmark is CausalGraphBench (Babakov et al., 2025), which assembles 35 graphs with domain descriptions and node dictionaries and compares LLM graph-construction strategies under one evaluation. Its methods operate from names and metadata rather than matched data conditions, and it is released as a fixed collection rather than a generator of new settings. Roy et al. (2025) present a one-shot framework for prompt-based and data-driven graph discovery with synthetic data generation; it is a method and pipeline rather than a benchmark for attributing performance to information sources. CausalMix instead holds the causal

task fixed while varying data budget, data format, and semantic cue condition (original, anonymized, and names-only), with matched classical baselines on real and synthetic graph families.

Classical causal-discovery benchmarks. The Tübingen cause–effect pairs (Mooij et al., 2016) evaluate bivariate direction rather than full-DAG recovery. The single-cell benchmark of Chevalley et al. (2025) evaluates network inference from perturbation data; because the true graph is unavailable, it relies on biological and distributional proxy metrics, and it is not designed around LLM data formats.

C.2 LLM-Assisted Causal Discovery Methods

Table 3 (Section E.2) lists the information channels used by each method below.

Direct inference. Long et al. (2022) introduced pairwise LLM causal discovery, querying GPT-3 to score directed edges in medical DAGs; Kıcıman et al. (2024) benchmark LLMs on pairwise tasks, and Susanti & Färber (2024) add knowledge graphs to the prompt. Causal-LLM (Roy et al., 2025) has a prompting variant that lists edges from dataset metadata and a data variant that scores edges with a frozen LLM backbone and learned projection heads, followed by pruning and acyclicity enforcement. Closest to our setup, Susanti & Färber (2025) place observational data in the prompt and build the graph with pairwise or breadth-first queries; the breadth-first variant resembles Jiralerspong et al. (2024), differing in how evidence is represented. See also Vashishtha et al. (2023).

LLMs as prior knowledge. Ban et al. (2023b) ask the LLM for richer variable metadata, extract high-confidence causal statements, and use them as soft ancestral constraints in score-based structure learning; see also Ban et al. (2023a).

Posterior refinement. Long et al. (2023) use LLMs as imperfect experts to orient edges within a Markov equivalence class. ALCM (Khatibi et al., 2024) runs PC, LiNGAM, or NOTEARS and then prompts an LLM with metadata to validate, reorient, remove, or add edges. Dong et al. (2025) show that pretrained LLMs over-rely on variable-name semantics, producing graphs that change under name permutation or column reordering, and propose CARE, a supervised fine-tuning method that integrates SCD outputs with world knowledge.

Iterative graph refinement. The Causal Modelling Agent (Abdulaal et al., 2023) alternates LLM graph proposals from metadata with fits of a deep structural causal model. Takayama et al. (2025) run SCD, prompt the LLM with its outputs to obtain pairwise causal probabilities (statistical causal prompting, SCP), convert these into constraints, and rerun SCD. CauScientist (Peng et al., 2026) initializes from either an LLM graph or an SCD graph by an intervention-aware BIC, then accepts single LLM-proposed edge edits only if a verifier confirms validity and BIC improvement.

D Benchmark Data and Prompts

D.1 Released Dataset

CausalMix v1 fixes graph sources, random seeds, generated observational and interventional data, prompt templates, semantic cue conditions, data representations, data budgets, reference outputs, and scoring scripts, so that reported scores are reproducible and matched-cell contrasts are comparable.

The release contains 14 graph instances in three tiers: six real reference graphs from bnlearn (Scutari, 2010), six Synthetic-25 topology controls from ENCO exports (Lippe et al., 2022), and two large challenge graphs (Table 2). The attribution study uses Cancer, Earthquake, Asia, and Sachs, with Sachs supporting the densest budget and method grids. The scaling study (Section H.2) adds Child, Alarm, and generated Chain/Jungle instances at 25 and 50 nodes; the generated instances are separate from the 14 released graphs. Diabetes and Pigs are graph- and prompt-scaling stress tests, not primary evaluation cells. The release includes prompt CSVs, manifests, source-graph references, generated data, prompt templates, scoring code, and matched PC, GES, and ENCO outputs. Future challenge splits can instantiate the same generator on held-out graphs, seeds, or graph families.

Table 2: Graph scale and prompt-token burden in CAUSALMIX V1. Density is $|E|/[d(d-1)]$ for a directed graph with d nodes. Tok. mean and Tok. max summarize prompt-token counts for the largest data-bearing representation.

Dataset	Nodes	Edges	Density	Tok. mean	Tok. max
Real reference suite					
Cancer	5	4	0.200	14.1K	84.6K
Earthquake	5	4	0.200	13.8K	84.6K
Asia	8	8	0.143	25.4K	152.7K
Sachs	11	17	0.154	62.7K	232.2K
Child	20	25	0.066	193.0K	541.3K
Alarm	37	46	0.035	433.7K	1.4M
Synthetic topology suite					
Chain	25	24	0.040	137.1K	781.4K
Collider	25	24	0.040	137.1K	781.4K
Bidiag	25	47	0.078	137.1K	781.4K
Jungle	25	46	0.077	137.1K	781.4K
Random	25	96	0.160	137.1K	781.4K
Full	25	300	0.500	137.1K	781.4K
Large challenge suite					
Diabetes	413	602	0.004	60.2M	336.8M
Pigs	441	592	0.003	46.5M	126.1M

Each benchmark cell fixes a graph and chooses a semantic cue condition (original or anonymized names, or names only), a data representation (`summary` or `tabular`), and a data budget (N, M) with N observational samples and M interventional samples per target. The generator supports no data $(0, 0)$; observational-only $(1000, 0)$ and $(5000, 0)$; interventional-only $(0, 200)$; and mixed $(1000, 50)$, $(1000, 200)$, $(5000, 50)$, and $(5000, 200)$.

D.2 PROMPT CONDITIONS AND REPRESENTATIONS

Figure 8 shows what each prompt exposes. The `names_only` prompt gives variable names and the task with no data; it defines the semantic-only reference against which data gains are measured. Data-bearing prompts use original or anonymized names with the same data, in one of two representations. The `summary` representation encodes each observational or interventional regime as a sparse empirical histogram over full variable assignments, mapping labels such as `do(X=v)` to the corresponding post-intervention histogram (Section D.3). The `tabular` representation prints raw rows, one sample per row and one variable per column; it audits representation sensitivity on small graphs and becomes a stress test as graph or sample size grows. Full examples appear in Section I.

We also tried a TabLLM-style serialization (Hegselmann et al., 2023) that renders each row as a natural-language sentence. It used more tokens than `tabular` and degraded graph recovery, so the release uses the compact tabular form, which preserves row/column alignment and fits more observations per token.

D.3 SUMMARY SERIALIZATION AND TOKEN SCALING

Each `summary` histogram entry lists an assignment in the declared variable order with its count. Let d be the number of variables, k_j the number of states of variable j, $K = \prod_{j=1}^{d} k_j$ the number of full assignments, n_r the number of samples in regime r, $h_r \leq \min(n_r, K)$ the number of distinct assignments observed in regime r, and R the number of regimes shown (one observational regime plus the included interventional regimes). The `summary` prompt then scales as $\mathcal{O}(d \sum_{r=1}^{R} h_r)$, whereas `tabular` scales as $\mathcal{O}(d \sum_{r=1}^{R} n_r)$. When $K \gg N, M$, as for large graphs, almost every observed assignment is unique, so $h_r \approx n_r$ and the two coincide; a cap of H entries per regime bounds the `summary` prompt by $\mathcal{O}(dRH)$.

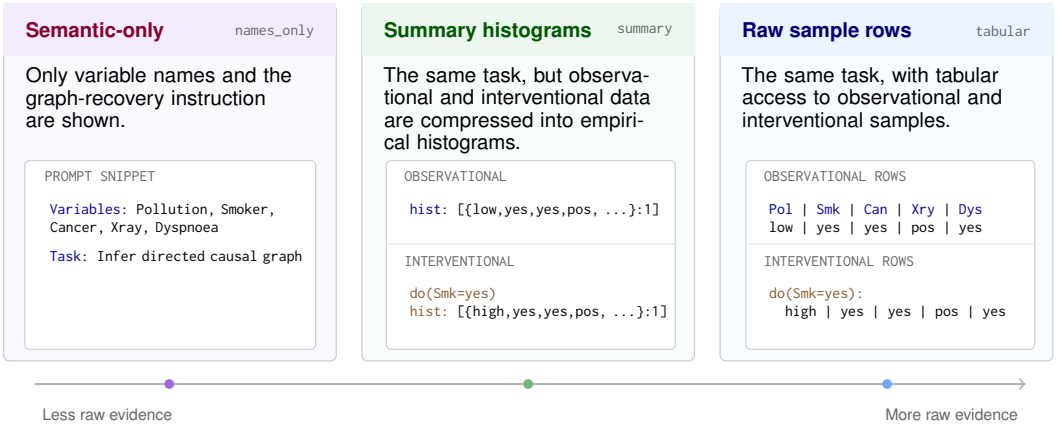

Figure 8: Prompt representations in CAUSALMIX. All three request the same graph-recovery output while varying the exposed evidence: variable names only, summary histograms, or raw sample rows. `do(X=v)` marks post-intervention regimes. Full prompts are in Section I.

D.4 PROMPT WRAPPERS

The four-graph sweep uses two wrapper suites with identical task and data content. The `plain` suite sends the base prompt unchanged. The `chat` suite appends a formatting requirement (the last line of the user turn below), wraps the result in the following transcript, and pre-fills the assistant turn with `<think>`:

```
system
A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant
↪  first thinks about the solution and then provides the final answer.
user
<BASE_PROMPT>

Formatting requirement: Output exactly: <think>...</think><answer>{"adjacency_matrix":
↪  [[0,1,...],[0,0,...],...]}</answer> where the matrix is N×N with integer entries 0 or 1 in VARIABLES order,
↪  and [i][j]=1 means variable i directly causes variable j.
assistant
<think>
```

E MODELS, BASELINES, AND EVALUATION PROTOCOL

E.1 MODELS AND DECODING

The attribution study evaluates ten models from one hosted and three open-weight families, with several sizes per open-weight family. Additional families appear in Section H.5.

- **OpenAI:** GPT-5-Mini (`gpt-5-mini`) and GPT-5.2-Pro (`gpt-5.2-pro`) (OpenAI, 2025b;a). The released manifests record model identifiers and per-call access dates.
- **Qwen3** (Qwen Team, 2025): Qwen3-4B (`Qwen/Qwen3-4B-Thinking-2507`) and Qwen3-30B-A3B (`Qwen/Qwen3-30B-A3B-Thinking-2507`).
- **Qwen2.5:** Qwen2.5-7B (`Qwen/Qwen2.5-7B-Instruct-1M`) and Qwen2.5-14B (`Qwen/Qwen2.5-14B-Instruct-1M`) (Yang et al., 2025); Qwen2.5-72B (`Qwen/Qwen2.5-72B-Instruct-AWQ`) (Qwen Team, 2024).
- **Llama-3.1** (Dubey et al., 2024): Llama-3.1-8B (`meta-llama/Meta-Llama-3.1-8B`), Llama-3.1-8B-Instruct (`meta-llama/Meta-Llama-3.1-8B-Instruct`), and Llama-3.1-70B-Instruct (`meta-llama/Llama-3.1-70B-Instruct`).

LLM calls use temperature 0 when the endpoint supports it; otherwise the endpoint's fixed decoding setting is held constant across matched cells. For data-bearing prompts, the five replicates are independently sampled datasets, not repeated decodes of one prompt. All local experiments run on two H200 GPUs.

Table 3: Information channels used by the causal-discovery methods discussed in Section C.

Method	Semantic Info	Obs. Data	Int. Data
Data-only methods			
PC (Spirtes et al., 2000)	✗	✓	✗
GES (Chickering, 2002)	✗	✓	✗
ENCO (Lippe et al., 2022)	✗	✓	✓
Causal-LLM (data-driven) (Roy et al., 2025)	✗	✓	✗
Semantic-only methods			
`names_only`	✓	✗	✗
Causal-LLM (prompting) (Roy et al., 2025)	✓	✗	✗
Knowledge-graph structure prompting (Susanti & Färber, 2024)	✓	✗	✗
Pairwise causal prompting (Vashishtha et al., 2023; Kıcıman et al., 2024)	✓	✗	✗
Mixed-information methods			
Causal graph prompting with observational data (Susanti & Färber, 2025)	✓	✓	✗
Efficient BFS-style querying (Jiralerspong et al., 2024)	✓	✓	✗
LLM-derived structured prior knowledge (Ban et al., 2023b)	✓	✓	✗
MEC posterior refinement (Long et al., 2023)	✓	✓	✗
Statistical causal prompting (SCP) (Takayama et al., 2025)	✓	✓	✗
CauScientist (Peng et al., 2026)	✓	✓	✓

E.2 Baselines

Table 3 groups methods by the information they use. Data-only methods provide matched references, and `names_only` defines the semantic-only reference.

Data-only references. PC (Spirtes et al., 2000) removes edges with conditional-independence tests and orients the rest with graphical rules; GES (Chickering, 2002) performs greedy search over Markov equivalence classes. Both use observational data. ENCO (Lippe et al., 2022) parameterizes edge existence and orientation separately and uses observational plus interventional data to recover a fully directed graph. Classical methods run with five seeds per (N, M) cell, and we report the mean F1 and SHD.

Adapted LLM methods. Causal-LLM prompting receives only variable names and the task and predicts the full DAG in one shot. The BFS-style baseline receives observational `summary` prompts, asks for root variables, and iteratively queries for direct children. SCP (Takayama et al., 2025) runs PC, converts LLM pairwise judgments into required or forbidden edges, and reruns constrained PC. BFS-style querying and SCP therefore use semantic priors and observational data, but no interventional data.

E.3 Output Validity and Scoring

Output formats and validity. The attribution study and the small-graph post-training comparison request a binary $d \times d$ adjacency matrix in JSON; the repeated Sachs and scaling evaluations request an edge list in JSON, with format either enforced during generation or only instructed (stated per analysis). Complete graph JSON may be extracted from surrounding reasoning or Markdown. A valid output is a complete graph over the expected variables with admissible entries or endpoints. In edge lists, self-loops and malformed edges are invalid, and exact duplicate edges are collapsed because the list represents a set. The matrix scorer checks parsing and shape and masks diagonal entries rather than invalidating the matrix. No cycle removal or other repair is applied. Acyclicity is reported separately from validity.

Scores and unavailable attempts. Directed-edge $F_1 = 2\mathrm{TP}/(2\mathrm{TP} + \mathrm{FP} + \mathrm{FN})$ for every valid graph, so a valid graph with no true positives scores zero. The attribution study reports F1 over valid outputs alongside validity; the post-training and scaling analyses score invalid outputs as zero and also report validity and valid-only summaries. Input-context rejections and API or runtime failures without an assessable completion are *unavailable*: they are reported separately and are not evidence of zero recovery. The same availability rule applies to both members of a contrast. SHD, skeleton F1,

ancestor F1, and acyclicity are also reported; a metric undefined for the ground truth (e.g., v-structure F1 without an unshielded collider) remains undefined.

Replicate summaries. Because data-bearing prompts are instantiated on several sampled datasets, each cell also records the acyclic-output fraction, dispersion of per-run F1 and SHD, edge stability, and a consensus graph from empirical edge frequencies at a fixed threshold, scored with the same metrics. Where edge frequencies are available, directed and skeleton Brier scores are reported.

Pairing and intervals. A difference of condition means and a mean paired difference are distinct estimators when outputs are missing. Figure 3 subtracts valid-output condition means and requires at least three valid outputs per condition. The primary-budget uncertainty tables (Tables 9 and 10) use paired-valid seed-level differences, retain valid zero-F1 graphs, require at least two pairs, and use 20,000 percentile-bootstrap resamples; their estimates need not equal the figure's. Five-draw intervals are descriptive. The post-training and scaling analyses state their pairing keys and resampling units in Section H.

E.4 Matched Contrasts

Let $S_{g,\theta}(c, f, N, M)$ be a condition-level score (e.g., mean directed-edge F1) for graph g, model θ, cue condition $c \in \{\text{orig}, \text{anon}\}$ (original or anonymized names), data representation $f \in \{\text{summary}, \text{tabular}\}$, observational budget N, and interventional budget M per target, and let $S^{\text{name}}_{g,\theta}$ be the `names_only` score. Each contrast holds g and θ fixed and is computed only when both cells are available. The mixed-information gain, anonymization drop, format gap, and observational and interventional uplifts are, in order,

$$\begin{aligned}
\Delta_{\text{mix}}(c, f, N, M) &= S_{g,\theta}(c, f, N, M) - S^{\text{name}}_{g,\theta}, \\
\Delta_{\text{name}}(f, N, M) &= S_{g,\theta}(\text{orig}, f, N, M) - S_{g,\theta}(\text{anon}, f, N, M), \\
\Delta_{\text{fmt}}(c, N, M) &= S_{g,\theta}(c, \text{summary}, N, M) - S_{g,\theta}(c, \text{tabular}, N, M), \\
\Delta_{\text{obs}}(c, f; n \mid M) &= S_{g,\theta}(c, f, n, M) - S_{g,\theta}(c, f, 0, M), \\
\Delta_{\text{int}}(c, f; m \mid N) &= S_{g,\theta}(c, f, N, m) - S_{g,\theta}(c, f, N, 0).
\end{aligned}$$

The *data-use gain* used in the post-training analyses is the within-label contrast $F_1(c, \text{data}) - F_1(c, \text{no data})$. For anonymized prompts, Δ_{mix} changes both names and data relative to the original-name `names_only` reference, so it is not a pure data-use estimate.

F Post-Training Details

Our pipeline applies format/graph-target SFT to `Qwen/Qwen3-4B-Thinking-2507` and then GRPO (Section F.2). A reasoning-SFT continuation from the same initializer (*two-stage SFT*) serves as a supervised reference.

F.1 Supervised Fine-Tuning Stages

Both supervised stages use PEFT LoRA with rank 16, $\alpha = 16$, dropout 0.0, no bias terms, and adapters on `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, and `down_proj`. Both learn verifier-checked graph targets through next-token supervision.

Stage 1: format and graph-target SFT. Training samples cover Cancer, Earthquake, Asia, and Sachs with budgets $(N, M) \in \{100, 1000\} \times \{10, 50\}$. Each completion follows the contract `<think> ... </think><answer>{"adjacency_matrix": ...}</answer>`, where the answer is the simulator's ground-truth DAG. We train for 2 epochs on two GPUs in bfloat16 with fused AdamW, learning rate 2×10^{-5}, no warmup or weight decay, max gradient norm 1.0, per-device batch size 1, and gradient accumulation 4. Checkpoints are evaluated every 100 optimizer steps on a held-out split, and the lowest-validation-loss checkpoint is kept. This stage trains output format and graph content together; it is not a schema-only control.

Stage 2: reasoning SFT continuation. Starting from the Stage 1 adapter, we build 768 prompt–answer pairs over Earthquake, Asia, and Sachs; `summary` and `matrix` prompts; budgets $(N, M) \in \{100, 1000\} \times \{10, 50\}$; original and anonymized names; random row and column order with four column permutations; and prompt-level reasoning guidance `none` or `concise`. Each pair is expanded across five answer/rationale styles, giving 3840 records, all containing the gold graph. `Qwen/Qwen2.5-72B-Instruct-AWQ` generates the rationales; the `<answer>` payload remains the gold adjacency matrix, and rationales failing format or edge-consistency checks are retried or discarded. With a 5% validation split (3648/192 records), we train on two GPUs in bfloat16 with fused AdamW, learning rate 5×10^{-6} with linear decay and no warmup, no weight decay, max gradient norm 1.0, per-device batch size 1, and gradient accumulation 8, evaluating every 100 steps. The lowest validation loss occurs at step 200 (0.39997); this checkpoint is used for the repeated Sachs and larger-graph evaluations.

F.2 SFT+GRPO

Data and initialization. GRPO (Shao et al., 2024) continues from the Stage 1 adapter. It uses 4800 prompts with original names and 4800 with anonymized names over Earthquake, Asia, and Sachs, with joint histogram summaries and $(N, M) \in \{100, 1000\} \times \{10, 50\}$. A seeded 90/10 split gives 8640 training and 960 reserved prompts. Although the GRPO data exclude Cancer, the Stage 1 initializer includes it, so Cancer is not held out from the full pipeline.

Verifier reward. Each completion contains a reasoning trace and a final adjacency matrix. The scalar reward is

$$\begin{aligned} R = {} & 0.05 I_{\text{fmt}} + 0.05 C_{\text{stage}} + 0.10(F_{\text{skel}} + F_{\text{coll}} + F_{\text{orient}}) \\ & + 0.30 F_{\text{edge}} + 0.20(1 - h) + 0.50\big(F_{\text{edge}} - 0.10 I_{\text{cycle}}\big). \end{aligned} \tag{1}$$

I_{fmt} checks the required tags and a parseable answer; C_{stage} is the fraction of the three requested reasoning stages present; and F_{skel}, F_{coll}, and F_{orient} score the skeleton, unshielded colliders, and oriented edges parsed from the trace against the gold graph. F_{edge} is directed-edge F1 of the final matrix, h its off-diagonal Hamming error divided by $d(d-1)$, and I_{cycle} indicates a directed cycle. Final-matrix terms are zero when no binary $d \times d$ matrix can be parsed. The reward measures agreement with the generating graph and the requested trace structure, not whether the model used the supplied data.

Policy update. For each prompt x, the policy samples $G = 4$ completions with advantages $A_i = (R_i - \overline{R})/(s_R + 10^{-4})$ standardized within the group. With $\rho_{i,t} = \pi_\theta(y_{i,t} \mid x, y_{i,<t}) / \pi_{\text{old}}(y_{i,t} \mid x, y_{i,<t})$, training maximizes the token-normalized clipped objective

$$\mathcal{J}(\theta) = \frac{1}{T} \sum_{i,t} m_{i,t} \left[\min(\rho_{i,t} A_i, \text{clip}(\rho_{i,t}, 0.8, 1.2) A_i) - 0.05\, k_{i,t} \right], \tag{2}$$

where $m_{i,t}$ selects completion tokens, $T = \sum_{i,t} m_{i,t}$, and the sum covers the 16 completions from four prompts per optimizer update. The KL estimator is $k_{i,t} = u_{i,t} - \log u_{i,t} - 1$ with $u_{i,t} = \pi_{\text{ref}}(y_{i,t} \mid x, y_{i,<t}) / \pi_\theta(y_{i,t} \mid x, y_{i,<t})$, where the reference policy is the base model with the LoRA adapter disabled. We use one update per rollout batch, no value model, and the `dapo` loss, which normalizes over completion tokens rather than per sequence.

Training settings. Training keeps rank-16 LoRA on two GPUs in bfloat16 with per-device batch size 1, gradient accumulation 8, fused AdamW (learning rate 5×10^{-6}, $(\beta_1, \beta_2) = (0.9, 0.999)$, $\epsilon = 10^{-8}$, no weight decay), linear decay without warmup, and gradient clipping at 1.0. Rollouts use temperature 1 and top-$p = 1$ with `</answer>` as the stop string, and prompt and completion limits of 8192 and 1024 tokens; overlong prompts keep their first and last halves. The training seed is 42 and the split seed is 1234. We evaluate the checkpoint at step 1200 (about 0.56 epochs of a 2160-step schedule; recorded training reaches step 1659). No validation-based selection criterion was recorded for this checkpoint.

G Additional Attribution-Study Results

Values are directed-edge F1 over valid outputs unless stated otherwise.

Table 4: Coverage of the four-graph LLM sweep. The two prompt-wrapper suites (Section D.4) use the same matched cells.

Axis	Coverage	#
Graphs	Cancer, Earthquake, Asia, Sachs	4
Models	GPT-5-Mini, GPT-5.2-Pro; Qwen2.5-7B/14B/72B; Qwen3-4B/30B-A3B; Llama-3.1-8B/8B-Instruct/70B-Instruct	10
Prompt-wrapper suites	chat + format hint; plain, no hint	2
Matched conditions per suite	names-only $\cup$ {orig, anon}$\times${summary, tabular}$\times$budgets	17
Data budgets (N, M)	$(5000, 200)$, $(5000, 0)$, $(0, 200)$, $(1000, 0)$	4
Graph $\times$ model $\times$ suite $\times$ condition cells	$4 \times 10 \times 2 \times 17 =$	**1,360**

Table 5: Valid responses ratio after top-valid-rate prompt-config selection at $N = 5000$, $M = 200$. Each graph cell reports Names/Sum/Mat, where entries are valid/total runs. "–" denotes the prompts exceed context-window error.

Model	Cancer			Earthquake			Asia			Sachs		
	Names	Sum	Mat	Names	Sum	Mat	Names	Sum	Mat	Names	Sum	Mat
GPT-5.2 Pro	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5
GPT-5 Mini	5/5	5/5	5/5	5/5	5/5	5/5	5/5	3/5	0/5	5/5	5/5	5/5
Qwen3-4B	5/5	0/5	0/5	5/5	0/5	0/5	0/5	0/5	0/5	5/5	5/5	5/5
Qwen3-30B-A3B	5/5	0/5	0/5	5/5	0/5	1/5	0/5	0/5	0/5	5/5	2/5	1/5
Qwen2.5-7B	5/5	4/5	5/5	5/5	5/5	4/5	5/5	5/5	2/5	5/5	0/5	3/5
Qwen2.5-14B	5/5	5/5	5/5	5/5	5/5	4/5	5/5	5/5	5/5	5/5	5/5	1/5
Qwen2.5-72B	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	5/5	0/5
Llama-3.1-8B	5/5	4/5	5/5	5/5	2/5	5/5	0/5	4/5	5/5	0/5	1/5	–
Llama-3.1-8B-Inst.	5/5	5/5	4/5	0/5	5/5	4/5	0/5	5/5	0/5	0/5	1/5	–
Llama-3.1-70B	5/5	5/5	5/5	5/5	5/5	5/5	0/5	5/5	0/5	0/5	0/5	–

G.1 Coverage and Output Validity

Table 4 summarizes the four-graph sweep. Table 5 reports parse validity at the primary budget after selecting the highest-validity prompt configuration per graph, model, and representation, and Figure 9 extends the `names_only` heatmap of Figure 2 to all ten models.

G.2 Diagnosing Invalid and Unavailable Outputs

We distinguish input-context rejection, output-limit termination, completed malformed graphs, and outputs whose stop metadata is unavailable (Table 6). Context rejection is an interface limit, and an output cutoff does not show that a model would fail with a larger generation budget.

Names-only Qwen2.5 on Child and Alarm. The cells in Figure 2 use five repetitions of one names-only prompt: chat wrapper with format hint, concise reasoning, columns randomized with seed 42, temperature 0, matrix output, a 16,384-token context, and a 4,096-token output cap. The inputs are short (365 tokens for Child, 490 for Alarm). Qwen2.5-7B emits 17 ragged rows for the 20-node Child graph and 14B a 20×19 matrix, both without reaching the output cap. The 72B model produces valid Child matrices but recovers one true edge (15 false positives, 24 false negatives; F1 0.049). All 15 Alarm attempts reach the output cap. A $\times$ in the figure therefore marks an invalid matrix or truncation, not context overflow. These deterministic repetitions use one variable order and are not independent robustness trials.

SFT+GRPO on larger graphs. In the scaling study, 171/200 data-present and 160/200 no-data outputs are valid. The 29 invalid data-present outputs comprise five Alarm output-limit terminations, two Alarm self-loop violations, and 22 unparsed 50-node outputs whose stop metadata is unavailable. All 40 invalid no-data outputs occur at 50 nodes and contain endpoints outside the variable set. Pooled data-present F1 is 0.068 with invalid outputs scored as zero and 0.079 over valid outputs, so

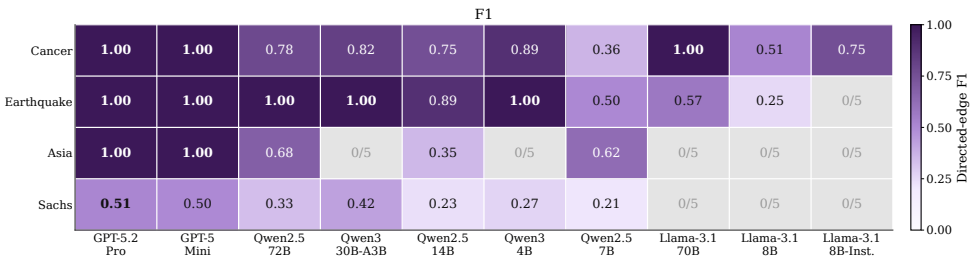

Figure 9: `names_only` directed-edge F1 for all ten models, with no observational or interventional data. Gray cells marked `0/5` indicate that none of the five runs produced a valid graph.

Table 6: Failure audit for Figure 2 Qwen cells and Figure 15. Counts distinguish valid graphs (V), input-context rejection (C), confirmed output-limit termination (L), observed structural violations (S), and unparsed output with unavailable stop metadata (U). Categories are mutually exclusive; output-limit termination is recorded before structural checks. S does not imply known termination status, and U does not establish that generation completed. The larger-graph rows use the main SFT+GRPO checkpoint.

Graph	Model	Labels	Attempts	V	C	L	S	U
Names only: matrix output								
Alarm	Qwen2.5 14B	Original names	5	0	0	5	0	0
Alarm	Qwen2.5 72B	Original names	5	0	0	5	0	0
Alarm	Qwen2.5 7B	Original names	5	0	0	5	0	0
Child	Qwen2.5 14B	Original names	5	0	0	0	5	0
Child	Qwen2.5 72B	Original names	5	5	0	0	0	0
Child	Qwen2.5 7B	Original names	5	0	0	0	5	0
Numerical evidence: SFT+GRPO edge-list JSON								
Child-20	Qwen3-4B GRPO	Original names	20	20	0	0	0	0
Child-20	Qwen3-4B GRPO	Anonymized names	20	20	0	0	0	0
Alarm-37	Qwen3-4B GRPO	Original names	20	13	0	5	2	0
Alarm-37	Qwen3-4B GRPO	Anonymized names	20	20	0	0	0	0
Chain-25	Qwen3-4B GRPO	Original names	20	20	0	0	0	0
Chain-25	Qwen3-4B GRPO	Anonymized names	20	20	0	0	0	0
Jungle-25	Qwen3-4B GRPO	Original names	20	20	0	0	0	0
Jungle-25	Qwen3-4B GRPO	Anonymized names	20	20	0	0	0	0
Chain-50	Qwen3-4B GRPO	Anonymized names	20	9	0	0	0	11
Jungle-50	Qwen3-4B GRPO	Anonymized names	20	9	0	0	0	11
No data: SFT+GRPO, format instructions only								
Chain-50	Qwen3-4B GRPO	Anonymized names	20	0	0	0	20	0
Jungle-50	Qwen3-4B GRPO	Anonymized names	20	0	0	0	20	0

low recovery is not an artifact of invalid graphs. Format enforcement controls syntax but does not guarantee termination, absence of self-loops, or correct structure.

Missing scores. A valid graph with no correct edges has F1 zero and is retained. `V<3` marks fewer than three valid outputs in a condition, and `NA(ctx)` a verified input-context rejection; neither is evidence of zero recovery. For example, Sachs `tabular` prompts at the primary budget contain about 160.8K tokens, exceeding serving limits of 128K or 131,072 tokens.

G.3 Cross-Graph Controls and Classical Baselines

Table 7 reports GPT-5-Mini across the four graphs under each cue condition and representation, with the strongest matched data-only reference. Anonymization lowers `summary` F1 on Cancer, Asia, and Earthquake, while on Sachs all LLM conditions remain at or below 0.50 and ENCO reaches 1.00. Table 8 lists every available PC, GES, and ENCO cell.

Table 7: Available cross-graph matched controls at the selected paper budget. Values are directed-edge F1. Rows hold graph and model fixed while varying semantic cue and data format; the final column reports the strongest matched data-only reference when available.

Graph	Model	Names	Original names summary	Anonymized names summary	Original names tabular	Anonymized names tabular	Data-only
Cancer	GPT-5 Mini	1.00	1.00	0.51	1.00	1.00	ENCO 0.91
Earthquake	GPT-5 Mini	1.00	1.00	0.91	1.00	0.86	ENCO 1.00
Asia	GPT-5 Mini	1.00	1.00	0.58	1.00	0.53	ENCO 0.96
Sachs	GPT-5 Mini	0.50	0.45	0.46	0.40	0.19	ENCO 1.00

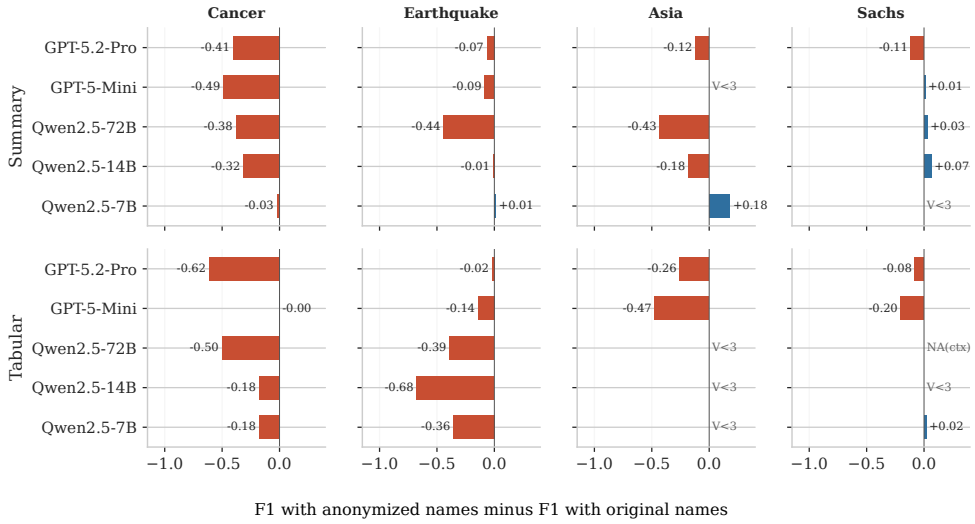

Figure 10: Anonymized-name F1 minus original-name F1 for paired cells. Negative values mean original names helped. `NA(ctx)` denotes a verified context rejection; `V<3` marks fewer than three valid outputs on at least one side. Invalid outputs are excluded rather than scored as zero.

G.4 Name Effects and Evidence Ladders

Figure 10 reports the anonymization contrast summarized in Section 5.2. Figures 11 and 12 show Sachs recovery by condition for six pretrained models with original and anonymized names, and Figure 13 shows Cancer, Earthquake, and Asia with original names.

G.5 Sachs Uncertainty and Budget Sweeps

Tables 9 and 10 give percentile-bootstrap intervals for matched Sachs contrasts at the primary budget. Tables 11 and 12 give the values plotted in Figures 6 and 7.

Table 8: All available classical-baseline cells for the four attribution-study graphs: directed-edge F1 and SHD for PC, GES, and ENCO at each evaluated observational budget N and per-target interventional budget M. Values are means over five seeds; valid predictions with no correct directed edge receive F1 zero.

Method	N	M	F1	SHD
Cancer				
ENCO	0	3	0.18	9.0
ENCO	0	50	0.29	5.0
ENCO	0	100	0.25	6.0
ENCO	0	200	0.24	7.6
ENCO	200	0	0.00	4.0
ENCO	200	3	0.00	7.0
ENCO	200	5	0.00	7.0
ENCO	500	5	0.40	3.0
ENCO	1,000	0	0.00	4.0
GES	1,000	0	0.17	5.2
PC	1,000	0	0.20	4.6
ENCO	1,000	50	0.56	3.2
ENCO	1,000	100	0.64	2.4
ENCO	1,000	200	0.62	2.2
ENCO	2,000	30	0.57	3.0
ENCO	5,000	0	0.00	4.0
GES	5,000	0	0.71	2.2
PC	5,000	0	0.43	4.2
ENCO	5,000	200	0.91	0.6
ENCO	50,000	512	1.00	0.0
ENCO	50,000	2000	0.86	1.0
Earthquake				
ENCO	0	200	0.26	6.8
ENCO	1,000	0	0.00	4.0
GES	1,000	0	0.95	0.4
PC	1,000	0	0.23	4.6
ENCO	1,000	50	0.89	1.0
ENCO	1,000	100	0.98	0.2
ENCO	1,000	200	0.98	0.2
ENCO	5,000	0	0.00	4.0
GES	5,000	0	1.00	0.0
PC	5,000	0	1.00	0.0
ENCO	5,000	200	1.00	0.0
Asia				
ENCO	0	200	0.23	16.8
ENCO	200	3	0.50	10.0
ENCO	1,000	0	0.00	8.0
GES	1,000	0	0.77	3.4
PC	1,000	0	0.38	8.2
ENCO	1,000	50	0.83	3.0
ENCO	1,000	100	0.86	2.4
ENCO	1,000	200	0.91	1.4
ENCO	5,000	0	0.00	8.0
GES	5,000	0	0.76	3.8
PC	5,000	0	0.70	4.2
ENCO	5,000	200	0.96	0.6
ENCO	50,000	512	1.00	0.0

Method	N	M	F1	SHD
Sachs				
ENCO	0	50	0.24	30.8
ENCO	0	100	0.14	38.0
ENCO	0	200	0.21	32.4
ENCO	0	500	0.13	40.0
ENCO	1	0	0.00	17.0
ENCO	1	50	0.11	17.0
ENCO	1	100	0.11	16.0
ENCO	1	200	0.11	17.0
ENCO	100	0	0.00	17.0
GES	100	0	0.38	15.6
PC	100	0	0.33	14.2
ENCO	100	10	0.55	13.0
ENCO	100	50	0.67	9.0
ENCO	100	100	0.64	9.0
ENCO	100	200	0.69	8.0
ENCO	100	500	0.58	10.0
GES	500	0	0.45	15.4
PC	500	0	0.34	17.4
ENCO	500	20	0.74	9.0
ENCO	1,000	0	0.00	17.0
GES	1,000	0	0.44	17.0
PC	1,000	0	0.42	16.8
ENCO	1,000	50	0.93	2.6
ENCO	1,000	100	0.99	0.4
ENCO	1,000	200	1.00	0.0
ENCO	1,000	500	0.97	1.0
GES	2,000	0	0.60	13.0
PC	2,000	0	0.50	16.0
GES	3,000	0	0.79	7.0
PC	3,000	0	0.48	17.6
GES	4,000	0	0.83	5.8
PC	4,000	0	0.60	13.6
ENCO	5,000	0	0.00	17.0
GES	5,000	0	0.88	4.0
PC	5,000	0	0.60	13.6
ENCO	5,000	50	0.95	1.8
ENCO	5,000	100	0.97	1.0
ENCO	5,000	200	1.00	0.0
ENCO	5,000	500	1.00	0.0
ENCO	8,000	0	0.00	17.0
ENCO	8,000	50	0.97	1.0
ENCO	8,000	100	0.97	1.0
ENCO	8,000	200	1.00	0.0
ENCO	8,000	500	1.00	0.0

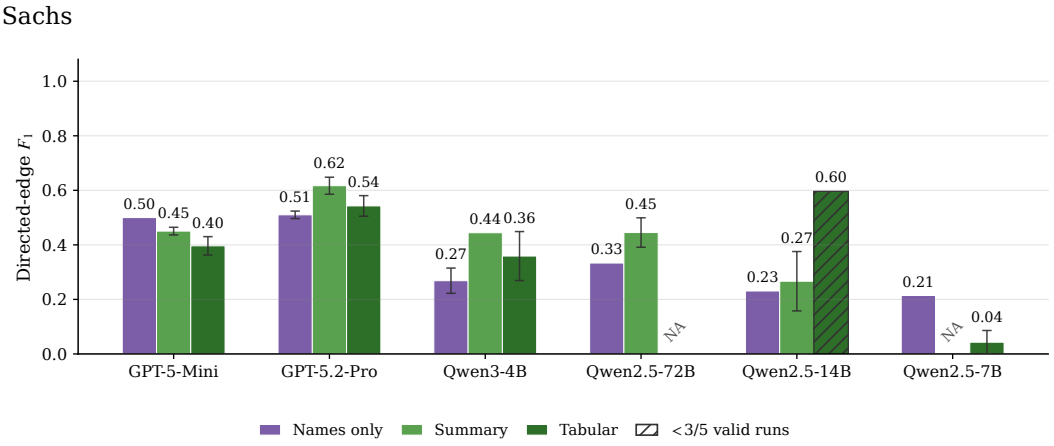

Figure 11: Sachs directed-edge F1 for six pretrained models with original names under `names_only`, `summary`, and `tabular` prompts at the primary budget. Error bars are standard errors over valid runs; hatching marks fewer than three of five valid runs, and NA no estimate.

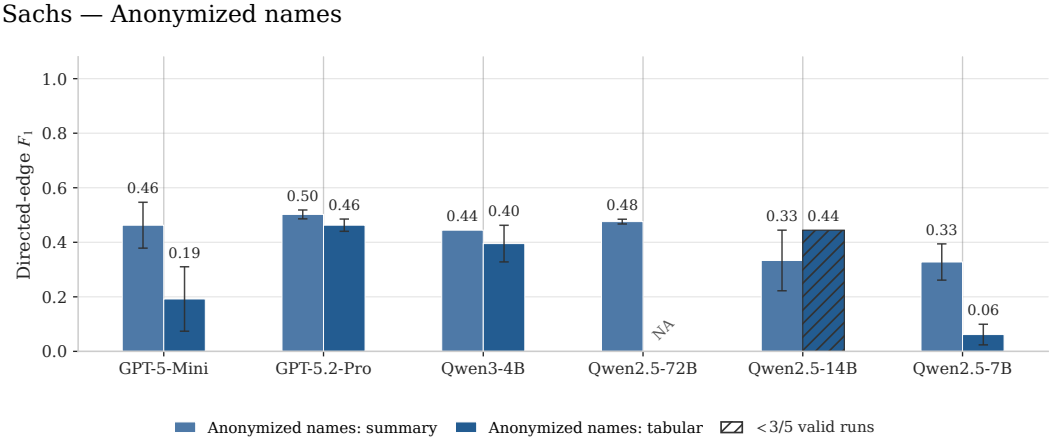

Figure 12: As Figure 11, with anonymized names in the data-bearing prompts (`names_only` is not shown).

Table 9: Bootstrap uncertainty for primary Sachs matched contrasts for GPT-5.2-Pro at $(N, M) = (5000, 200)$. Intervals are percentile bootstrap intervals over paired valid runs using 20,000 resamples. Contrasts pair runs by the same data-realization index; with five seeds, these intervals are descriptive uncertainty annotations rather than high-powered significance tests. In formulas, orig. and anon. denote original names and anonymized names, respectively.

Value name	Value formula	Paired runs	Mean ΔF_1	95% bootstrap interval
Data gain	`orig. summary − names-only`	5	$+0.107$	$[+0.057, +0.158]$
Anonymization drop	`orig. summary − anon. summary`	5	$+0.115$	$[+0.055, +0.167]$
Format gap	`orig. summary − orig. tabular`	5	$+0.074$	$[-0.014, +0.189]$

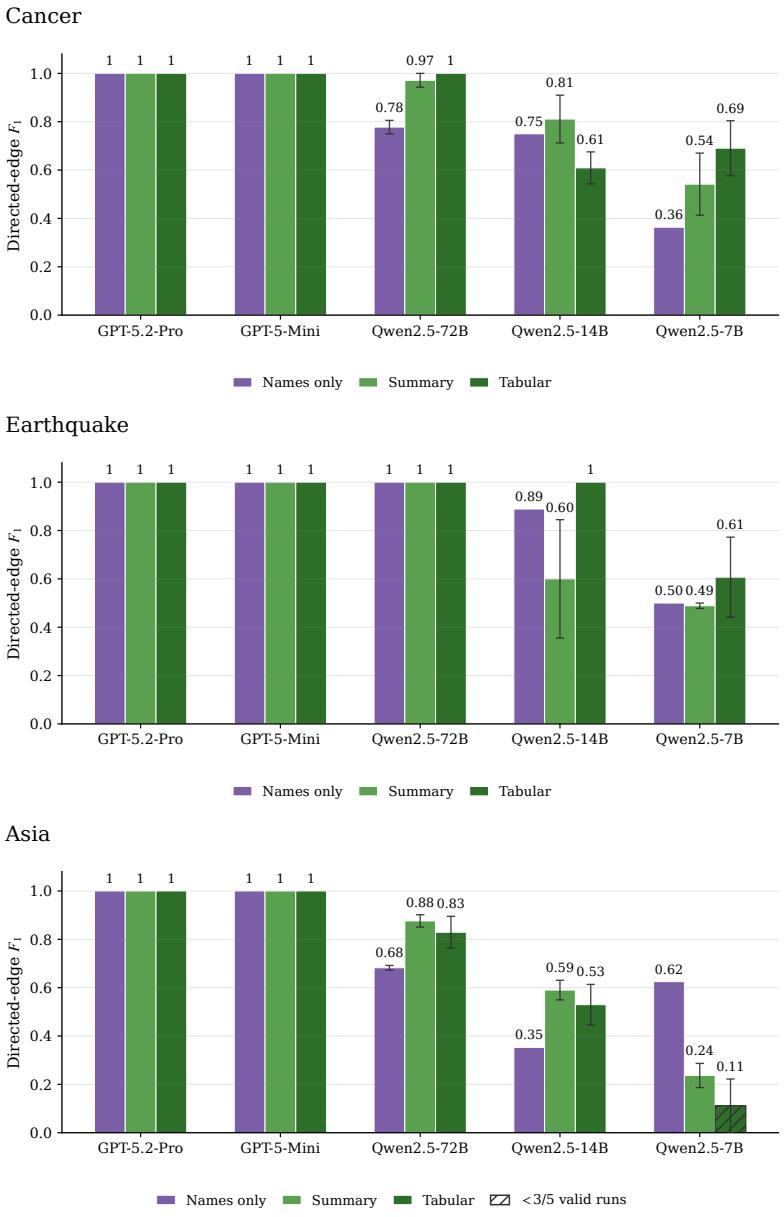

Figure 13: As Figure 11, for Cancer, Earthquake, and Asia with the five-model roster.

Table 10: Bootstrap uncertainty for primary Sachs matched contrasts for the remaining nine evaluated models. The GPT-5.2-Pro companion table is shown in Table 9. Intervals use 20,000 percentile-bootstrap resamples over paired valid seed-level F1 differences. Valid zero-F1 outputs are retained. Cells with fewer than two paired valid rows are not estimable and are marked "–". In formulas, orig. and anon. denote original names and anonymized names, respectively.

Model	Value name	Value formula	Paired runs	Mean ΔF_1	95% bootstrap interval
GPT-5-Mini	Data gain	`orig. summary − names-only`	5	-0.050	$[-0.074, -0.025]$
	Anonymization drop	`orig. summary − anon. summary`	5	-0.012	$[-0.148, +0.128]$
	Format gap	`orig. summary − orig. tabular`	5	$+0.054$	$[-0.022, +0.139]$
Qwen3-4B	Data gain	`orig. summary − names-only`	5	$+0.176$	$[+0.090, +0.249]$
	Anonymization drop	`orig. summary − anon. summary`	4	$+0.000$	$[+0.000, +0.000]$
	Format gap	`orig. summary − orig. tabular`	5	$+0.085$	$[-0.010, +0.267]$
Qwen3-30B-A3B	Data gain	`orig. summary − names-only`	2	-0.431	$[-0.462, -0.400]$
	Anonymization drop	`orig. summary − anon. summary`	2	-0.222	$[-0.444, +0.000]$
	Format gap	`orig. summary − orig. tabular`	0	$--$	$--$
Qwen2.5-7B	Data gain	`orig. summary − names-only`	0	$--$	$--$
	Anonymization drop	`orig. summary − anon. summary`	0	$--$	$--$
	Format gap	`orig. summary − orig. tabular`	0	$--$	$--$
Qwen2.5-14B	Data gain	`orig. summary − names-only`	5	$+0.036$	$[-0.142, +0.214]$
	Anonymization drop	`orig. summary − anon. summary`	4	$+0.000$	$[-0.333, +0.333]$
	Format gap	`orig. summary − orig. tabular`	1	$--$	$--$
Qwen2.5-72B	Data gain	`orig. summary − names-only`	5	$+0.112$	$[+0.021, +0.203]$
	Anonymization drop	`orig. summary − anon. summary`	5	-0.031	$[-0.121, +0.057]$
	Format gap	`orig. summary − orig. tabular`	0	$--$	$--$
Llama-3.1-8B	Data gain	`orig. summary − names-only`	0	$--$	$--$
	Anonymization drop	`orig. summary − anon. summary`	0	$--$	$--$
	Format gap	`orig. summary − orig. tabular`	0	$--$	$--$
Llama-3.1-8B-Instruct	Data gain	`orig. summary − names-only`	0	$--$	$--$
	Anonymization drop	`orig. summary − anon. summary`	1	$--$	$--$
	Format gap	`orig. summary − orig. tabular`	0	$--$	$--$
Llama-3.1-70B Instruct	Data gain	`orig. summary − names-only`	0	$--$	$--$
	Anonymization drop	`orig. summary − anon. summary`	0	$--$	$--$
	Format gap	`orig. summary − orig. tabular`	0	$--$	$--$

Table 11: Sachs observational-budget sweep for gpt-5-mini at fixed $M = 0$. Values are F1 (SHD). The table asks whether observational samples alone improve graph recovery beyond the semantic-only floor, with PC and GES as matched observational references.

N	Names	Original names summary	Anonymized names summary	PC	GES
0	0.50 (14.60)	–	–	–	–
1000	–	0.51 (14.60)	0.52 (13.20)	0.42 (16.80)	0.44 (17.00)
5000	–	0.47 (16.40)	0.62 (12.20)	0.60 (13.60)	0.88 (4.00)

Table 12: Sachs intervention-budget sweep for gpt-5.2-pro at fixed $N = 1000$. Values are F1 (SHD). Increasing intervention budget does not yield monotonic LLM improvement, while the matched data-only reference reaches the classical ceiling once interventions are available.

M	Original names summary	Anonymized names summary	Data-only ref.	Data-only F1 (SHD)
0	0.51 (14.80)	0.47 (15.20)	GES	0.44 (17.00)
50	0.57 (13.20)	0.43 (15.40)	ENCO	0.93 (2.60)
100	0.57 (13.00)	0.48 (15.00)	ENCO	0.99 (0.40)
200	0.56 (12.80)	0.47 (14.60)	ENCO	1.00 (0.00)

H POST-TRAINING, SCALING, AND EVIDENCE-USE DIAGNOSTICS

These analyses test whether post-training improves recovery, whether gains transfer beyond the training graph families, and whether models respond to supplied data. Three conventions apply throughout. (i) For Qwen3 checkpoints, edge-list format is enforced during generation at 5–37 nodes but only instructed at 50 nodes, so comparisons across the 50-node boundary confound graph size with output protocol; other models use format instructions only, and runs with different output protocols are never paired. (ii) Generated invalid outputs receive F1 zero, while unavailable attempts (context rejections, API failures) are excluded and counted separately. (iii) Where four column orders are used, 95% intervals enumerate all $4^4 = 256$ bootstrap resamples of the order-cluster means, each averaging five data draws; these intervals are descriptive and condition on the evaluated checkpoints and graphs.

H.1 PRETRAINED VERSUS SFT+GRPO ACROSS TEN GRAPHS

Table 13 extends Figure 4 to all 18 graph/naming conditions at $(N, M) = (1000, 10)$: 360 data-present responses per checkpoint, with four order seeds and five draws per seed. Pairs match graph, variable order, data draw, naming, budget, and decoding (temperature 0, 8192-token output cap; 131,072-token context at 5–37 nodes and 262,144 at 50 nodes). The 50-node graphs use anonymized names only.

On Cancer and Earthquake with original names, both checkpoints produce 20/20 valid graphs, so the F1 gains there reflect better recovery on training families rather than validity. Mean F1 decreases in five conditions: Alarm with original names, and Asia, Sachs, Jungle-25, and Jungle-50 with anonymized names.

Supervised reference. Table 14 places two-stage SFT beside SFT+GRPO on repeated Sachs and the six scaling graphs. The two pipelines differ in training data, objective, duration, and checkpoint selection, so the table is a reference point rather than an estimate of the effect of GRPO. Its bottom block uses matrix outputs on the four small graphs.

Matrix-output comparison. Figure 14 compares SFT+GRPO with pretrained Qwen3-4B using matrix outputs, per-condition wrappers, and five draws per condition. Its Sachs decreases come from this different protocol and need not match the edge-list comparison in Figure 4. All four graphs overlap post-training.

H.2 SCALING STUDY

We evaluate the SFT+GRPO checkpoint on Child, Alarm, and generated Chain and Jungle graphs with 25 or 50 nodes at $(N, M) = (1000, 10)$, using four column-order seeds (1103, 2207, 3301, 4409) and five data draws per order. Data and no-data outputs are paired by graph, protocol, naming condition, order, and draw. The 25-node Chain/Jungle source labels are `A`, ..., `Y`, which identify variables without describing real quantities; anonymization replaces them with `X1`, ..., `X25` in permuted order. The 50-node graphs use only `X1`, ..., `X50`. On these graphs, data/no-data contrasts measure responsiveness to data, and letter-versus-placeholder contrasts concern label encoding rather than domain knowledge.

Table 15 reports the conditions plotted in Figure 15. At 50 nodes, all 40 no-data outputs are invalid, so the positive data-use gains there compare against zero and do not show successful use of data (Section G.2).

H.3 HELD-OUT-FAMILY TRANSFER OF THE SFT STAGES

Because SFT+GRPO is trained on the small reference graph families, we test whether supervised output reliability transfers beyond them using leave-one-graph-out (LOGO) training of Base, format SFT, and reasoning SFT. Cancer, Earthquake, Asia, and Sachs are held out in turn and excluded from both supervised stages. Each fold uses training seeds 42, 314, and 2718 and crosses original/anonymized names with data/no-data conditions; four column orders and five draws per order give 20 paired realizations per seed and naming condition. Fold-specific GRPO is not evaluated.

Table 13: Pretrained Qwen3-4B versus SFT+GRPO across all ten graphs at $(N, M) = (1000, 10)$. Each model/naming cell has 20 data-present attempts, matched by graph, order, draw, and recorded settings. Cells show directed-edge F1 with invalid generated outputs scored as zero, followed by valid/attempted counts. All attempts are available. Original names preserve source identifiers; Chain/Jungle source identifiers are neutral. Original-name 50-node cells were not run. Format is enforced during generation at 5–37 nodes; 50-node runs use format instructions only. The four smallest graph families overlap post-training. ENCO values retain the existing 20-run scaling reference at the same nominal budget; sample arrays are not verified identical. A dash in its small-graph cells means no matched reference is reported.

	Original names		Anonymized names		ENCO
Graph	Pretrained	SFT+GRPO	Pretrained	SFT+GRPO	F1
Cancer-5	0.544 (20/20)	0.926 (20/20)	0.153 (9/20)	0.332 (20/20)	—
Earthquake-5	0.550 (20/20)	0.990 (20/20)	0.079 (6/20)	0.121 (20/20)	—
Asia-8	0.264 (17/20)	0.505 (20/20)	0.236 (20/20)	0.205 (20/20)	—
Sachs-11	0.259 (19/20)	0.287 (20/20)	0.216 (20/20)	0.211 (20/20)	—
Child-20	0.081 (19/20)	0.086 (20/20)	0.109 (20/20)	0.115 (20/20)	0.652
Chain-25	0.052 (20/20)	0.081 (20/20)	0.074 (20/20)	0.075 (20/20)	0.609
Jungle-25	0.023 (19/20)	0.107 (20/20)	0.119 (20/20)	0.115 (20/20)	0.703
Alarm-37	0.035 (9/20)	0.033 (13/20)	0.030 (20/20)	0.048 (20/20)	0.677
Chain-50	not run		0.006 (8/20)	0.011 (9/20)	0.754
Jungle-50	not run		0.006 (12/20)	0.005 (9/20)	0.790

Table 14: Data-present recovery for SFT+GRPO, with two-stage SFT as a supervised reference. Entries report directed-edge F1 with invalid generated outputs scored as zero, and valid/generated counts. Sachs uses 30 realizations per naming condition at $(N, M) = (5000, 200)$; scaling uses 20 at $(1000, 10)$. Format is enforced during generation for Sachs and 20–37-node graphs; 50-node graphs use format instructions only. Training data, objectives, duration, and checkpoint selection differ between pipelines, so these comparisons do not isolate the training algorithm. They do not establish generalization to unseen graph families. The bottom block uses matrix outputs with the per-cell settings selected for GRPO and reused for SFT; its wrappers and output budgets differ from the edge-list rows, and repeated deterministic names-only prompts are not independent training replicates.

		Two-stage SFT		SFT+GRPO	
Graph	Names	F1	Valid	F1	Valid
Sachs-11	Original names	0.219	26/30	0.253	27/30
Sachs-11	Anonymized names	0.159	30/30	0.233	30/30
Child-20	Original names	0.070	20/20	0.086	20/20
Child-20	Anonymized names	0.104	20/20	0.115	20/20
Chain-25	Original names	0.049	18/20	0.081	20/20
Chain-25	Anonymized names	0.069	20/20	0.075	20/20
Jungle-25	Original names	0.044	19/20	0.107	20/20
Jungle-25	Anonymized names	0.095	20/20	0.115	20/20
Alarm-37	Original names	0.022	12/20	0.033	13/20
Alarm-37	Anonymized names	0.031	18/20	0.048	20/20
Chain-50	Anonymized names	0.024	19/20	0.011	9/20
Jungle-50	Anonymized names	0.011	14/20	0.005	9/20
Matrix outputs: four small graphs, five realizations each					
Names only		0.742	20/20	0.584	20/20
Original names + tabular data		0.254	7/20	0.488	16/20
Original names + summaries		0.592	19/20	0.678	19/20

For checkpoint t and naming condition c, let $D_t(c) = F_1(t, c, \text{data}) - F_1(t, c, \text{no data})$; the additional data-use effect of reasoning SFT is $D_{\text{reasoning}}(c) - D_{\text{format}}(c)$. Both learned stages produce 100% valid outputs in every fold (Table 16). For the eight reasoning-minus-format contrasts, we

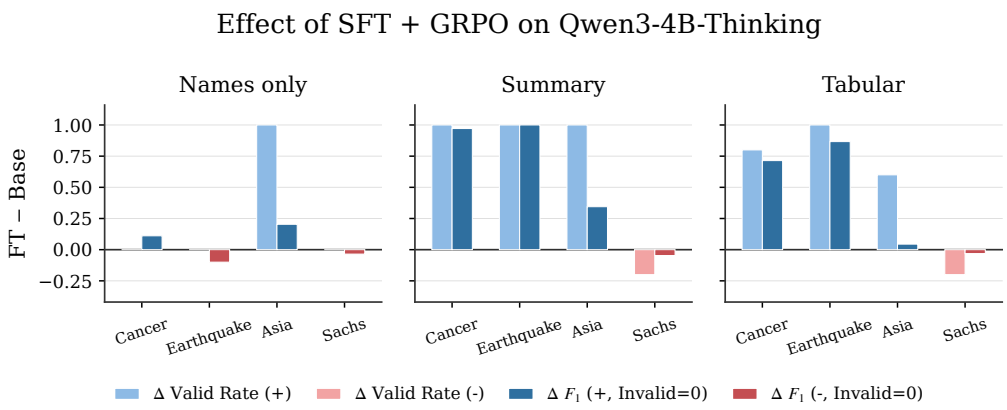

Figure 14: SFT+GRPO minus pretrained Qwen3-4B with matrix outputs on graphs that overlap post-training. Δ valid rate is the change in parseable-output rate; ΔF_1 is the change in directed-edge F1 with invalid outputs scored as zero. These gains do not establish transfer to unseen graph families or improved evidence use.

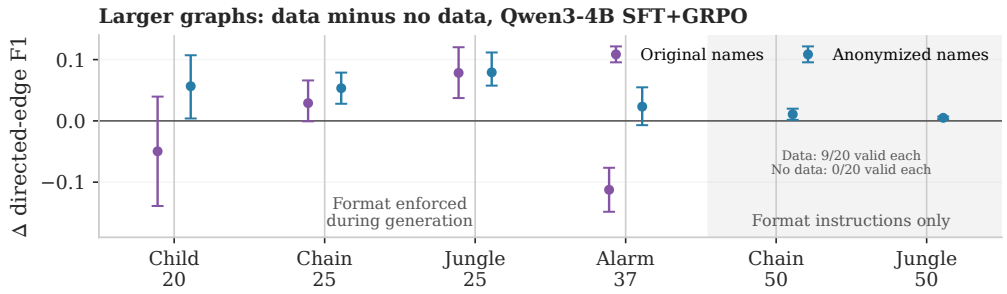

Figure 15: Data-use gains on larger graphs after post-training: data-minus-no-data F1 for SFT+GRPO at $N = 1000$, $M = 10$ per target, with invalid outputs scored as zero. Each point uses 20 matched pairs (four column orders, five draws each), with descriptive 95% bootstrap intervals over order clusters. Shaded 50-node runs use format instructions only and anonymized names only; all their no-data outputs are invalid, so positive contrasts there do not imply meaningful recovery.

average the five draws per seed/order cell and resample the three seeds and four orders independently (10,000 replicates). The effects range from -0.071 to $+0.037$, and all eight 95% intervals include zero. Equal-fold means give name effects of $+0.341/+0.356$ for format SFT and $+0.355/+0.315$ for reasoning SFT (no data/data), and within-label data gains of $+0.017/+0.002$ and $-0.016/+0.024$ (original/anonymized names). Output reliability therefore transfers to held-out families, reasoning SFT adds no detectable data use over format SFT, and because both stages contain gold targets, the contribution of target imitation remains unresolved.

H.4 Repeated Sachs Comparison

To reduce replication noise, we compare each data-bearing Sachs response with its no-data control at the same naming condition and replicate index over 30 order/data realizations (versus five in the attribution study). All five models have 120 available attempts; intervals resample the 30 clusters.

Table 17 shows negative data-use gains with original names and positive gains with anonymized names for Base, both supervised stages, and GPT-5-Mini, with all eight intervals excluding zero. SFT+GRPO has near-zero gain with original names (-0.002; 27/30 valid) and positive gain with anonymized names ($+0.134$; 30/30 valid), so its response to data also depends on the labels. Section H.6 examines whether the changed edges move toward the true graph.

Table 15: Scaling study: paired data/no-data results for SFT+GRPO. F1 assigns zero to invalid generated outputs; validity is reported separately. Each row contains 20 matched data/no-data pairs. $\times$ indicates no valid graph; its zero contribution is retained only in the invalid-aware gain. Thus the 50-node gains are descriptive contrasts against entirely invalid no-data outputs, not evidence of improved recovery from data. Intervals in Figure 15 resample four column-order clusters.

Graph	Names	No-data F1	Data F1	Gain	Data valid	No-data valid
Child-20	Original names	0.136	0.086	-0.050	20/20	20/20
Child-20	Anonymized names	0.058	0.115	+0.057	20/20	20/20
Chain-25	Original names	0.052	0.081	+0.029	20/20	20/20
Chain-25	Anonymized names	0.021	0.075	+0.053	20/20	20/20
Jungle-25	Original names	0.028	0.107	+0.078	20/20	20/20
Jungle-25	Anonymized names	0.036	0.115	+0.079	20/20	20/20
Alarm-37	Original names	0.146	0.033	-0.112	13/20	20/20
Alarm-37	Anonymized names	0.025	0.048	+0.023	20/20	20/20
Chain-50	Anonymized names	$\times$	0.011	+0.011	9/20	0/20
Jungle-50	Anonymized names	$\times$	0.005	+0.005	9/20	0/20

Table 16: Held-out-family output validity (percent; valid/available attempts) and reasoning-minus-format change in data-use F1 gain. Validity pools all four information-source conditions (240 attempts per learned stage). Each naming-condition gain contrast has 60 paired realizations (three training seeds, four order seeds, five draws per order). Intervals resample training seeds and order clusters independently after averaging the five within-order draws; all eight intervals include zero. Edge-list format was enforced during generation. Base Earthquake has three recorded seed-labelled evaluation repetitions; other Base folds have one.

Held-out graph	Base validity	Format validity	Reasoning validity	Δ gain: Original names	Δ gain: Anonymized names
Asia	71.2 (57/80)	100.0 (240/240)	100.0 (240/240)	-0.071 [-0.180, +0.027]	+0.028 [-0.042, +0.101]
Cancer	86.2 (69/80)	100.0 (240/240)	100.0 (240/240)	+0.003 [-0.161, +0.203]	+0.004 [-0.171, +0.263]
Earthquake	76.2 (183/240)	100.0 (240/240)	100.0 (240/240)	-0.025 [-0.119, +0.062]	+0.037 [-0.019, +0.126]
Sachs	73.8 (59/80)	100.0 (240/240)	100.0 (240/240)	-0.036 [-0.097, +0.014]	+0.018 [-0.021, +0.064]

H.5 Cross-Family Coverage on Larger Graphs

Table 18 evaluates additional model families on the six scaling graphs with anonymized names, each under its own serving and output constraints. Context rejections and API failures are marked, not scored as zero. Across these graphs, ENCO attains F1 0.609–0.790 (Table 13), while SFT+GRPO reaches 0.005–0.115 with data-use gains from -0.112 to $+0.079$. ENCO shares the nominal budgets but not verified identical sample arrays. GPT-5-Mini has 120 evaluable anonymized-name attempts; its original-name Child/Alarm slice (not shown) has 59 API failures, including all 40 no-data attempts. Qwen2.5-7B has no evaluable data-present attempts, and several checkpoints are context-limited at 50 nodes; these missing scores do not show an inability to recover the graphs.

H.6 Edge-Level Responses to Data

For paired-valid outputs, a *correction* moves an incorrect edge state to the ground truth and a *regression* moves a correct state away from it; the net correction rate is their difference. Predictions can change without improving, so responsiveness and beneficial use of data are reported separately. These rates cover all unordered node pairs, including absent edges, and use paired-valid denominators, so they measure a different aspect of recovery from invalid-as-zero edge F1.

Table 19 shows that adding data makes SFT+GRPO change more edges harmfully than beneficially on both Sachs and the 20–37-node graphs. On Sachs, net correction rates are -0.160 with original names and -0.398 with anonymized names. Higher validity can therefore coexist with changes that move away from the generating graph. The LOGO rows use the fold-specific SFT checkpoints of Section H.3.

Table 17: Sachs replication: paired change in directed-edge F1 when adding data. Each listed model has 120 available attempts and 30 matched pairs per naming condition at $(N, M) = (5000, 200)$. All models were asked for edge lists. Qwen checkpoints used format enforced during generation; GPT-5-mini used format instructions only. Invalid generated outputs contribute F1 zero; unavailable attempts are excluded. Intervals bootstrap the 30 order/data-realization clusters. Absolute recovery and validity for SFT+GRPO and two-stage SFT are reported in Table 14.

Model	Original names ΔF1	Pairs	Anonymized names ΔF1	Pairs
Base	-0.183 [-0.234, -0.130]	30	+0.230 [+0.207, +0.252]	30
GPT-5-mini	-0.087 [-0.104, -0.069]	30	+0.039 [+0.017, +0.062]	30
Format SFT	-0.091 [-0.142, -0.039]	30	+0.061 [+0.023, +0.101]	30
Two-stage SFT	-0.089 [-0.167, -0.008]	30	+0.061 [+0.025, +0.096]	30
SFT+GRPO	-0.002 [-0.053, +0.047]	30	+0.134 [+0.106, +0.164]	30

Table 18: Cross-family data-present recovery with anonymized names at the main budget (1,000/10), using edge lists with format instructions only, except for Qwen3 at 20–37 nodes, where the format was enforced during generation. Cells show F1 (evaluable/attempted). CTX denotes a verified input-context rejection; RATE denotes an API token-quota rejection; Pending denotes an unfinished evaluation. These are not F1 zero. An asterisk marks a partial result. Generated invalid outputs remain zero within the evaluable denominator. Full naming-condition validity and failure counts are in the accompanying CSV. The SFT+GRPO row uses the same checkpoint as the scaling study; its data-present validity is 20/20 on each 20–37-node graph and 9/20 on each 50-node graph.

Model	Child	Chain25	Jungle25	Alarm	Chain50	Jungle50
DeepSeek-V2-Lite	0.092 (20/20)	0.057 (20/20)	0.032 (20/20)	0.017 (20/20)	CTX (0/20)	CTX (0/20)
Gemma3-12B	0.021 (20/20)	0.016 (20/20)	0.003 (20/20)	0.001 (20/20)	CTX (0/20)	CTX (0/20)
GPT-5.2-Pro	RATE (0/20)	RATE (0/20)	RATE (0/20)	RATE (0/20)	RATE (0/20)	RATE (0/20)
GPT-5-mini	0.000 (20/20)	0.006 (20/20)	0.003 (20/20)	0.000 (20/20)	0.000 (20/20)	0.001 (20/20)
Granite3.2-8B	0.076 (20/20)	0.075 (20/20)	0.128 (20/20)	0.062 (20/20)	CTX (0/20)	CTX (0/20)
Llama3.1-8B	0.054 (20/20)	0.062 (20/20)	0.068 (20/20)	0.056 (20/20)	CTX (0/20)	CTX (0/20)
Ministral-8B	0.000 (20/20)	0.000 (20/20)	0.000 (20/20)	0.000 (20/20)	0.000 (20/20)	0.000 (20/20)
Qwen2.5-7B	CTX (0/20)	CTX (0/20)	CTX (0/20)	CTX (0/20)	CTX (0/20)	CTX (0/20)
Qwen3 format (100)	0.079 (20/20)	0.070 (20/20)	0.094 (20/20)	0.031 (20/20)	0.027 (20/20)	0.006 (20/20)
Qwen3 two-stage SFT	0.104 (20/20)	0.069 (20/20)	0.095 (20/20)	0.031 (20/20)	0.024 (20/20)	0.011 (20/20)
Qwen3 base	0.109 (20/20)	0.074 (20/20)	0.119 (20/20)	0.030 (20/20)	0.006 (20/20)	0.006 (20/20)
Qwen3 SFT+GRPO	0.115 (20/20)	0.075 (20/20)	0.115 (20/20)	0.048 (20/20)	0.011 (20/20)	0.005 (20/20)

H.7 Name–Data Conflict

Using the two-stage SFT checkpoint, we keep the Child variable names but rewire the generating graph (Table 20). Retention of removed edges and detection of added edges both remain low (about 0.3) under original and anonymized names. This diagnostic concerns the SFT checkpoint and does not establish conflict resolution by SFT+GRPO.

I Prompt Examples

We give one complete prompt per condition for the Cancer graph with a small sample size for readability. Benchmark runs use the same templates with the budgets in the released manifests; the released templates mark dataset name, variables, state counts, and data payloads as placeholders.

I.1 `names_only`

```
ROLE: You are an expert in causal discovery from variable semantics and background knowledge.
TASK: Infer the directed causal graph over the variables.
We are studying a system called 'cancer'.
No observational or interventional data are provided for this case. Use only the variable meanings and relevant
↪  background causal knowledge.
ASSUMPTIONS:
- The true graph is a DAG (no directed cycles).
- Causal sufficiency holds (no unobserved confounders among these variables).

--- VARIABLE ORDER (ORDER MATTERS) ---
0: Pollution
1: Smoker
```

Table 19: Edge-level changes from no data to observational and interventional evidence, with edge-list format enforced during generation. Rates average unordered-node-pair state transitions over pairs for which both outputs are valid (valid/attempted-pair counts shown). Beneficial and harmful transitions correct and introduce state errors, respectively; Net is their difference. These rates concern all unordered pairs, including absent edges, and are distinct from directed-edge F1. Leave-one-graph-out (LOGO) rows use fold-specific two-stage SFT checkpoints; no GRPO LOGO result is available.

Evaluation	Method	Names	Valid pairs	Changed	Beneficial	Harmful	Net
Scale (20–37)	SFT+GRPO	Original names	73/80	0.315	0.056	0.227	-0.171
Scale (20–37)	SFT+GRPO	Anonymized names	80/80	0.680	0.056	0.585	-0.529
Scale (20–37)	Two-stage SFT	Original names	69/80	0.187	0.057	0.114	-0.057
Scale (20–37)	Two-stage SFT	Anonymized names	78/80	0.340	0.050	0.267	-0.217
Sachs (30)	SFT+GRPO	Original names	27/30	0.494	0.119	0.279	-0.160
Sachs (30)	SFT+GRPO	Anonymized names	30/30	0.771	0.128	0.525	-0.398
Sachs (30)	Two-stage SFT	Original names	26/30	0.338	0.112	0.171	-0.059
Sachs (30)	Two-stage SFT	Anonymized names	30/30	0.422	0.108	0.255	-0.147
Strict LOGO	Two-stage SFT	Original names	240/240	0.282	0.114	0.150	-0.036
Strict LOGO	Two-stage SFT	Anonymized names	240/240	0.255	0.098	0.123	-0.025

Table 20: Name–data conflict on five rewired Child graphs: reasoning-SFT checkpoint, observational and interventional evidence, edge lists with format enforced during generation, 20 attempts per graph and naming condition. Removed edges are adjacent in Child but not in the generating DAG; added edges are the reverse (8–12 of each per graph). Entries are the fraction of such pairs predicted adjacent, averaged over valid outputs and then over graphs; parentheses give the range across graphs. Orientation-only disagreements (1% of disputed pairs) are omitted.

Names	Valid/attempted	Removed-edge retention	Added-edge detection
Original names	99/100	0.290 (0.206–0.375)	0.310 (0.244–0.406)
Anonymized names	100/100	0.330 (0.280–0.404)	0.296 (0.200–0.412)

```
2: Cancer
3: Xray
4: Dyspnoea

--- AVAILABLE EVIDENCE ---
No sampled data are provided. The graph must be inferred from the variable names and domain knowledge alone.

--- OUTPUT INSTRUCTIONS ---
Output exactly: <think>...</think><answer>...</answer>.
Keep <think> concise (minimal necessary reasoning only).
Inside <answer>, output exactly one JSON object with key "adjacency_matrix".
- "adjacency_matrix": N x N 0/1 matrix in declared variable order.
No extra text before, between, or after the two blocks.
The JSON in <answer> must start with "{" and end with "}".
```

I.2 summary

```
ROLE: You are an expert in causal discovery from observational and interventional data.
TASK: Infer the directed causal graph over the variables.
The following are empirical distributions computed from data sampled from a Bayesian network named cancer.
ASSUMPTIONS:
- The true graph is a DAG (no directed cycles).
- Causal sufficiency holds (no unobserved confounders among these variables).
- Interventions are perfect do-interventions (surgical): do(X=v) cuts all incoming edges into X.

--- VARIABLE ORDER (ORDER MATTERS) ---
0: Pollution
1: Smoker
2: Cancer
3: Xray
4: Dyspnoea

--- DATA FORMAT ---
Each payload summarizes empirical samples in the declared variable order.
num_states=[2,2,2,2,2].
hist entries use [assignment, count, prob]; assignments follow the declared variable order.
observational_data summarizes samples with no intervention.
interventional_data maps each do(X=v) regime to its empirical summary.
Unlisted assignments may be absent due to finite sampling.
```

```
--- STATE LEGEND ---
Pollution: 0->low, 1->high
Smoker: 0->False, 1->True
Cancer: 0->False, 1->True
Xray: 0->negative, 1->positive
Dyspnoea: 0->True, 1->False

--- OBSERVATIONAL DATA ---
observational_data={
  "n": 10,
  "hist": [[[0,0,0,0,1],2,0.2],[[0,0,0,1,1],2,0.2],[[0,0,0,0,0],1,0.1],[[0,0,0,1,0],1,0.1],[[1,0,0,1,0],1,0.1],[
 ↪  [1,0,0,0,1],1,0.1],[[0,1,0,1,1],1,0.1]]
}

--- INTERVENTIONAL DATA ---
interventional_data={
  "do(Cancer=False)": {"n":3,"hist":[[[0,1,0,0,0],1,0.3333],[[0,1,0,0,1],1,0.3333],[[1,1,0,0,0],1,0.3333]]},
  "do(Cancer=True)": {"n":3,"hist":[[[0,0,1,1,1],1,0.3333],[[0,1,1,1,0],1,0.3333],[[0,0,1,1,0],1,0.3333]]},
  "do(Dyspnoea=False)": {"n":3,"hist":[[[0,0,0,0,1],2,0.6667],[[0,1,0,0,1],1,0.3333]]},
  "do(Dyspnoea=True)": {"n":3,"hist":[[[0,1,0,1,0],1,0.3333],[[1,1,0,0,0],1,0.3333],[[0,0,0,0,0],1,0.3333]]},
  "do(Pollution=high)": {"n":3,"hist":[[[1,1,0,0,0],1,0.3333],[[1,0,0,1,0],1,0.3333],[[1,0,0,0,1],1,0.3333]]},
  "do(Pollution=low)": {"n":3,"hist":[[[0,0,0,0,1],1,0.3333],[[0,1,0,0,0],1,0.3333],[[0,0,0,1,1],1,0.3333]]},
  "do(Smoker=False)": {"n":3,"hist":[[[0,0,0,0,1],1,0.3333],[[1,0,0,0,0],1,0.3333],[[0,0,0,1,1],1,0.3333]]},
  "do(Smoker=True)": {"n":3,"hist":[[[0,1,0,0,1],2,0.6667],[[0,1,0,0,0],1,0.3333]]},
  "do(Xray=negative)": {"n":3,"hist":[[[1,0,0,0,1],1,0.3333],[[0,1,0,0,1],1,0.3333],[[0,0,0,0,1],1,0.3333]]},
  "do(Xray=positive)": {"n":3,"hist":[[[0,0,0,1,1],1,0.3333],[[0,1,0,1,1],2,0.6667]]}
}

--- END OF DATA ---

--- OUTPUT INSTRUCTIONS ---
Output exactly: <think>...</think><answer>...</answer>.
Keep <think> concise (minimal necessary reasoning only).
Inside <answer>, output exactly one JSON object with key "adjacency_matrix".
- "adjacency_matrix": N x N 0/1 matrix in declared variable order.
No extra text before, between, or after the two blocks.
The JSON in <answer> must start with "{" and end with "}".
```

I.3 tabular

```
ROLE: You are an expert in causal discovery from observational and interventional data.
TASK: Infer the directed causal graph over the variables.
The following are empirical samples from a Bayesian network named cancer.
ASSUMPTIONS:
- The true graph is a DAG (no directed cycles).
- Causal sufficiency holds (no unobserved confounders among these variables).
- Interventions are perfect do-interventions (surgical): do(X=v) cuts all incoming edges into X.

--- VARIABLE ORDER (ORDER MATTERS) ---
0: Pollution
1: Smoker
2: Cancer
3: Xray
4: Dyspnoea

--- OBSERVATIONAL DATA (no intervention) ---
Each row is one observed case.
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | False | False | negative | True
low | False | False | negative | False
low | False | False | positive | True
low | False | False | positive | False
high | False | False | positive | True
low | False | False | positive | False
high | False | False | negative | False
low | False | False | negative | False
low | False | False | negative | True
low | True | False | positive | False

--- INTERVENTIONAL DATA ---
Each block corresponds to samples collected under a perfect intervention do(X = value).

[Intervention: do(Cancer = False)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | True | False | negative | True
low | True | False | negative | False
high | True | False | negative | True

[Intervention: do(Cancer = True)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | False | True | positive | False
```

```
low | True | True | positive | True
low | False | True | positive | True

[Intervention: do(Dyspnoea = False)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | False | False | negative | False
low | False | False | negative | False
low | True | False | negative | False

[Intervention: do(Dyspnoea = True)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | True | False | positive | True
high | True | False | negative | True
low | False | False | negative | True

[Intervention: do(Pollution = high)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
high | True | False | negative | True
high | False | False | positive | True
high | False | False | negative | False

[Intervention: do(Pollution = low)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | False | False | negative | False
low | True | False | negative | True
low | False | False | positive | False

[Intervention: do(Smoker = False)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | False | False | negative | False
high | False | False | negative | True
low | False | False | positive | False

[Intervention: do(Smoker = True)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | True | False | negative | False
low | True | False | negative | True
low | True | False | negative | False

[Intervention: do(Xray = negative)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
high | False | False | negative | False
low | True | False | negative | False
low | False | False | negative | False

[Intervention: do(Xray = positive)]
Pollution | Smoker | Cancer | Xray | Dyspnoea
low | False | False | positive | False
low | True | False | positive | False
low | True | False | positive | False

--- END OF DATA ---

--- OUTPUT INSTRUCTIONS ---
Output exactly: <think>...</think><answer>...</answer>.
Keep <think> concise (minimal necessary reasoning only).
Inside <answer>, output exactly one JSON object with key "adjacency_matrix".
- "adjacency_matrix": N x N 0/1 matrix in declared variable order.
No extra text before, between, or after the two blocks.
The JSON in <answer> must start with "{" and end with "}".
```