Multi-Agent Causal Discovery Using Large Language Models

Hao Duong Le, Xin Xia, Haijie Xu, Zhang Chen
Department of Industrial Engineering, Tsinghua University
Beijing, China
{lihaoyan25, xu-hj22}@mails.tsinghua.edu.cn
{xxia99, zhangchen01}@tsinghua.edu.cn

Abstract

Causal discovery aims to identify causal relationships between variables and is a fundamental problem across the sciences. Traditional statistical causal discovery (SCD) methods rely solely on observational data and ignore the contextual information available in metadata, whereas recent LLM-based methods exploit metadata but treat the large language model (LLM) as a single agent, leaving its judgments vulnerable to memorized or biased associations. To address this gap, we introduce MAC (Multi-Agent Causal Discovery Framework), which casts causal discovery as a multi-agent debate coupled with the autonomous selection of an SCD algorithm. MAC combines two complementary modules, bridged by a Meta Fusion mechanism: a Debate-Coding Module (DCM) that grounds an initial graph in data by autonomously selecting and executing the best-suited SCD algorithm, and a Meta-Debate Module (MDM) that refines the graph through an adversarial Affirmative–Negative–Judge debate over the metadata. Across five benchmark datasets and three metrics (F1, SHD, NHD), MAC achieves the best aggregate performance among five statistical and four LLM-based baselines, ranking first on 10 of 15 evaluation points with Gemini-2.0-Flash—including a perfect reconstruction of the Earthquake graph—and remains robust across three backbone LLMs.

1 Introduction

Discovering causal relationships between variables is fundamental across scientific fields, with many statistical causal discovery (SCD) methods developed in recent years (Huang et al., 2018; Glymour et al., 2019; Spirtes et al., 2013). However, these methods face two significant challenges. First, they heavily rely on large volumes of structured data for accurate inference. In large-scale systems with thousands of variables (or nodes), obtaining sufficient structured data is often infeasible. Second, SCD methods fail to leverage metadata—additional contextual information such as variable contexts, domain knowledge, and external factors—that can enhance the causal discovery process and improve inference accuracy.

The advent of Large Language Models (LLMs), trained on vast datasets, has opened new avenues for addressing the challenges mentioned above. LLMs possess a wide range of knowledge, encompassing common sense, specialized domains, and advanced reasoning abilities (Wei et al., 2023; Rozière et al., 2024; Zhao et al., 2023b; Yao et al., 2023a). Leveraging LLMs for causal discovery has gained attention due to their ability to replicate expert knowledge in a cost-effective and accessible manner. Recent work (Kıcıman et al., 2023; Choi et al., 2022; Long et al., 2024; Chen et al., 2024a) has explored LLM-based methods for causal discovery, focusing on metadata and knowledge-driven reasoning similar to human experts. However, these methods largely treat LLMs as single-agent systems, which may limit their reasoning capabilities, especially when handling complex causal relationships or large-scale, dense causal graphs. In contrast, multi-agent LLM systems, where multiple models collaborate to collectively discover causal relationships, offer greater potential for tackling complex cases.

This paper introduces MAC (Multi-Agent Causal Discovery Framework), which casts causal-graph discovery as a multi-agent debate coupled with the autonomous selection of a statistical causal discovery (SCD) algorithm. MAC integrates two core components:

- **Meta-Debate Module (MDM):** *Problem:* a single LLM's causal judgments are often unreliable, reflecting memorized or biased associations rather than evidence. *Our answer:* instead of trusting one model, MDM pits three specialized LLM-based agents—a Causal Affirmative Debater, a Causal Negative Debater,

Method / approach	LLM-based method	Statistical method	Agentic ability	Multi-agent	Introduced by
Pairwise causal discovery	✓	✗	✗	✗	(Kıcıman et al., 2023; Zečević et al., 2023)
Various prompting techniques	✓	✗	✗	✗	(Chen et al., 2024a)
Effective LLMs prompting	✓	✓	✗	✗	(Jiralerspong et al., 2024)
Hybrid statistical and LLMs	✓	✓	✗	✗	(Vashishtha et al., 2023; Takayama et al., 2024)
Iterative LLM–SCD refinement	✓	✓	✗	✗	(Ban et al., 2023a)
MAC	✓	✓	✓	✓	**Our approach**

Table 1: Comparison of approaches for using LLMs in causal discovery.

and a Causal Judge—against each other, so that competing causal hypotheses are explicitly surfaced and adjudicated against the metadata rather than accepted unchecked.

- **Debate-Coding Module (DCM):** *Problem:* purely text-based LLM reasoning is not grounded in the observed data, and no single statistical algorithm is best across all datasets. *Our answer:* DCM first uses an embedded MDM to debate and select the SCD algorithm best suited to the data's characteristics, then executes it on the structured data, grounding the initial graph in empirical evidence.

In particular, DCM is first implemented. It embeds an MDM inside, whose aim is to use metadata and a small subset of structured data to select the most suitable Statistical Causal Discovery (SCD) algorithm. The selected SCD algorithm is then implemented using the whole structured data to learn the causal graph, which is further transferred into new causal metadata through a Meta Fusion mechanism. Using this causal metadata and other available metadata as input, MDM is further implemented to refine and optimize causal relationships.

We evaluate MAC on five datasets against five statistical and four LLM-based baselines using F1, NHD, and SHD. MAC delivers the best aggregate performance across the 15 evaluation points (five datasets × three metrics) and remains robust across three backbone LLMs, ranking first on 10 of 15 points with Gemini-2.0-Flash—including a perfect reconstruction of the Earthquake graph—and on 9 and 7 points with DeepSeek-R1 and GPT-4o, respectively. In summary, our contributions are as follows:

- We cast causal discovery as an adversarial multi-agent debate coupled with the autonomous selection of a statistical causal discovery algorithm, integrating data-grounded statistical precision with metadata-driven, expert-level reasoning.
- We propose two complementary modules: the **Meta-Debate Module (MDM)**, which counters unreliable single-LLM judgments by having agents adversarially propose, challenge, and adjudicate causal graphs against the metadata, and the **Debate-Coding Module (DCM)**, which grounds the graph in data by autonomously selecting and executing the best-suited SCD algorithm (e.g., PC, GES).
- Across five benchmarks, three metrics, and three backbone LLMs, MAC achieves the best aggregate performance over both statistical and LLM-based baselines. Beyond causal discovery, the framework provides a general multi-agent template for machine-learning tasks with metadata, such as classification and anomaly detection.

2 Related Work

Causal discovery methods fall into three main categories: constraint-based methods, score-based methods, and functional causal model-based methods. Classical constraint-based methods use conditional independence to reveal causal structures. Some commonly used algorithms include the PC algorithm (Spirtes et al., 2000), the FCI algorithm (Spirtes et al., 2013), etc. However, these methods may encounter the multiple testing problem due to the numerous conditional independence tests required. In contrast, score-based methods conduct causal discovery by constructing a score function that reflects the goodness-of-fit between the causal structure and the data, and select the

causal structure with the highest score. An example is the Greedy Equivalence Search (GES) algorithm (Chickering, 2002), which optimizes a scoring function by iteratively adding or removing edges, with recent advances along these lines including Ogarrio et al. (2016); Huang et al. (2018). Both constraint-based and score-based methods can only identify Markov Equivalence Classes (MECs), which necessitate stronger assumptions for accurate causal structure identification. This limitation has prompted the development of functional causal model-based methods, which assume that the causal relationships between parent and child nodes can be represented as parameterized functional forms, along with specific assumptions regarding the noise distribution. For example, Shimizu et al. (2006) proposes LiNGAM to identify causal structures characterized by linear relationships under non-Gaussian noise, leading to several enhancements in the field (Zhang and Hyvärinen, 2009; Shimizu et al., 2011; Sanchez-Romero et al., 2019).

As a relatively new research area, knowledge-driven causal discovery using LLMs has attracted increasing attention; we refer readers to Wan et al. (2025) for a comprehensive survey. Early work queries LLMs in a pairwise fashion to infer the causal direction between each pair of variables (Kıcıman et al., 2023; Zečević et al., 2023), but this requires a quadratic number of queries with respect to the number of variables and scales poorly. To improve efficiency, Jiralerspong et al. (2024) adopt a breadth-first search (BFS) strategy that reduces the number of queries to a linear scale, while Chen et al. (2024a) systematically study prompting techniques—including in-context learning, zero-shot and manual Chain-of-Thought (CoT), and adversarial prompts—for causal inference. These methods rely primarily on metadata. A complementary line of work injects LLM knowledge into statistical causal discovery: LLMs serve as soft priors for score-based search (Darvariu et al., 2024) or answer conditional-independence queries within constraint-based methods (Cohrs et al., 2024), while iterative hybrids alternate between statistical structure learning and LLM-based edge verification (Ban et al., 2023a,b; Vashishtha et al., 2023; Takayama et al., 2024), achieving state-of-the-art results along this direction.

However, recent studies caution that single-LLM causal judgments are often driven by memorization rather than genuine reasoning and are highly sensitive to the prompt and graph encoding (Feng et al., 2025; Chi et al., 2024; Sheth et al., 2025; Yang et al., 2024). These findings motivate moving beyond a single agent. A few concurrent efforts explore agentic or interactive LLM workflows for graph discovery (Havrilla et al., 2025; Roy et al., 2025; Antonucci et al., 2024), yet none casts causal discovery as a structured multi-agent debate. Building on multi-agent debate, which improves factuality and reasoning by having agents argue and adjudicate competing answers (Du et al., 2023), MAC combines an adversarial Affirmative–Negative–Judge debate with the autonomous selection of a statistical causal discovery algorithm. We summarize and compare representative methods in Table 1.

3 Methodology

3.1 Problem Setup

Given variables $V = \{X_1, \ldots, X_n\}$ and observational data $O \in \mathbb{R}^{d \times n}$ (rows are observations, columns are variables), our goal is to recover a directed acyclic graph (DAG) $G = (V, E)$, where $(X_i, X_j) \in E$ denotes a causal relation $X_i \to X_j$. In addition to O, we optionally use metadata I that describes variable semantics, measurement context, or domain constraints. To reduce LLM cost in lightweight reasoning steps, we also provide a small subset $S \subset O$ (e.g., a few sampled rows); the final causal graph is always estimated using the full data O.

3.2 Design Principles and Overview

MAC is designed to balance two complementary capabilities and mitigate their corresponding failure modes. (1) Statistical causal discovery (SCD) methods are *data-grounded* but cannot directly leverage unstructured metadata and can be brittle in ambiguous regimes. (2) LLM-only causal reasoning may reflect memorized or biased associations ("causal parrots") when used without empirical grounding (Zečević et al., 2023). Accordingly, MAC decomposes causal discovery into two stages: **Stage 1 (DCM):** data-grounded initialization that constructs an initial graph from structured data using an SCD algorithm; **Stage 2 (MDM):** metadata-guided refinement that adjudicates competing causal hypotheses using metadata. The two stages are connected by **Meta Fusion**, which converts the Stage-1 graph into compact textual constraints that can be jointly reasoned over with meta-

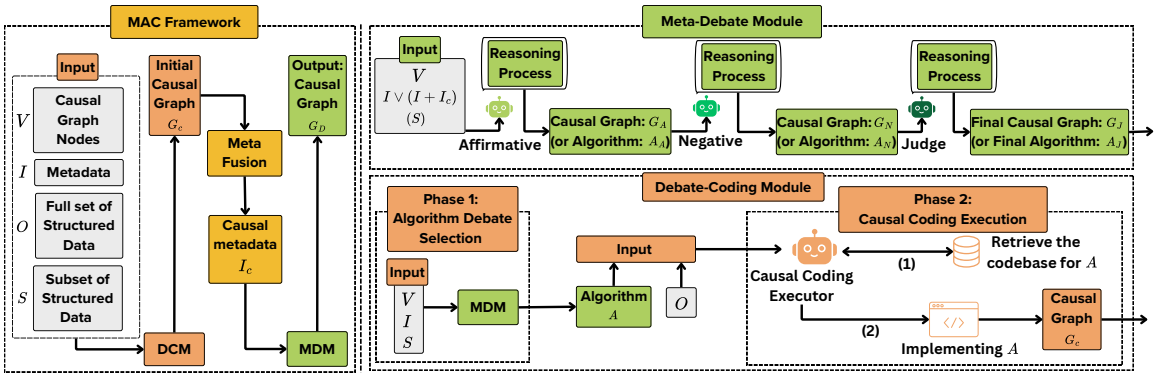

Figure 1: The left side of the image illustrates the overall algorithm of the MAC framework, while the right side details the MDM and DCM. The MDM produces different outputs based on the input: (1) If the input includes metadata I from the original data only, the MDM will output a graph. (2) It can also accept a combination of I and causal metadata I_c from the DCM for further refinement of the graph. (3) Finally, the MDM with SCD algorithm as output, embedded within the DCM, receives an additional input—a subset of the structured data S—and outputs an algorithm.

data. To make the design rationale explicit, we present each component (MDM, DCM, and their integration) with the same structure: the *problem* it addresses, *why* existing approaches fall short, our *design*, and the resulting *procedure*.

3.3 Meta-Debate Module (MDM)

Problem. Inferring causal relations from metadata is a reasoning task with no observable ground truth at decision time: for any pair of variables, several causal explanations are often equally plausible, and the correct one cannot be read off the data alone.

Why prior solutions fall short. Existing LLM-based methods query a single model, which tends to commit to one explanation and propagate memorized or biased associations without scrutiny—a failure mode documented across causal-reasoning benchmarks (Zečević et al., 2023; Feng et al., 2025). A single pass therefore offers no mechanism to surface or contest a wrong-but-confident answer.

Design. MDM refuses to trust a single pass: a candidate hypothesis must survive an explicit challenge before it is accepted. Building on multi-agent debate, which improves reasoning and factuality by having agents argue competing answers (Du et al., 2023), MDM is a reusable *debate-and-judge* operator with three agents: (i) an **Affirmative** agent that proposes a candidate output, (ii) a **Negative** agent that proposes a *plausible alternative* under the same evidence (targeting disputed decisions rather than maximizing arbitrary divergence), and (iii) a **Judge** that scores the disagreements against explicit, evidence-derived criteria and returns the winner. The same operator is reused in two roles with different output spaces: **(a) graph refinement**, producing a DAG over V from context (V, I), and **(b) algorithm selection**, producing an SCD algorithm from a pool $\mathcal{A}$ given (V, I, S).

Algorithm 1 Meta-Debate Module (MDM): Unified Debate-and-Judge Operator

Input: variables V; metadata I; optional subset S; output type $m \in \{\texttt{graph}, \texttt{alg}\}$
Output: y_J ▷ y_J is a graph if $m = \texttt{graph}$, else an algorithm
if $m = \texttt{graph}$ **then**
  $\mathcal{C} \leftarrow (V, I)$; $\mathcal{Y} \leftarrow$ DAGs over V
else if $m = \texttt{alg}$ **then**
  $\mathcal{C} \leftarrow (V, I, S)$; $\mathcal{Y} \leftarrow \mathcal{A}$ ▷ predefined SCD algorithm pool
end if
$y_A \leftarrow \texttt{Affirmative}(\mathcal{C}, \mathcal{Y})$
$y_N \leftarrow \texttt{Negative}(\mathcal{C}, \mathcal{Y}, y_A)$ ▷ plausible alternative under same evidence
$y_J \leftarrow \texttt{Judge}(\mathcal{C}, \mathcal{Y}, y_A, y_N)$
return y_J

Procedure. Algorithm 1 summarizes the unified operator; prompt templates are deferred to Appendix A.1.

3.4 Debate-Coding Module (DCM)

Problem. A causal graph inferred purely from text is not grounded in the observed data, leaving it prone to memorized causal associations ("causal parrots") (Zečević et al., 2023). Statistical causal discovery (SCD) avoids this by learning from data, but no single SCD algorithm is best across datasets.

Why prior solutions fall short. The validity

Algorithm 2 Debate-Coding Module (DCM)

Input: variables V; data O; metadata I; subset S; algorithm pool $\mathcal{A}$
Output: initial graph G_C
$A \leftarrow \texttt{MDM}(V, I, S;\ m = \texttt{alg},\ \mathcal{A})$ ▷ Algorithm 1
$G_C \leftarrow \texttt{Causal\_Coding\_Executor}(A, O)$
return G_C

Algorithm 3 MAC: End-to-End Framework

Input: variables V; data O; metadata I; subset S; algorithm pool $\mathcal{A}$
Output: final graph G_D
$G_C \leftarrow \texttt{DCM}(V, O, I, S, \mathcal{A})$ ▷ Algorithm 2
$I_c \leftarrow \texttt{Meta\_Fusion}(G_C)$
$G_D \leftarrow \texttt{MDM}(V, I, S = \varnothing;\ m = \texttt{graph},\ I \cup I_c)$ ▷ Algorithm 1
return G_D

of each SCD algorithm hinges on assumptions—linearity, Gaussianity, hidden confounders, sample size—that differ from one problem to the next, so a fixed choice is brittle. Selecting the right algorithm is itself a reasoning problem in which metadata is informative, yet existing hybrids commit to a predetermined algorithm rather than reasoning about this trade-off.

Design. DCM provides a *data-grounded initialization* in two steps. *(i) Algorithm selection.* Given (V, I, S) and an algorithm pool $\mathcal{A}$, DCM reuses the unified MDM operator (Algorithm 1) with m = `alg` to argue the trade-offs and commit to the algorithm $A \in \mathcal{A}$ best matched to the data; the small subset S is used only for lightweight reasoning and cost reduction, not as a replacement for the full data. *(ii) Causal coding execution.* A **Causal Coding Executor** runs A on the full structured data O via an existing causal discovery library (e.g., `causal-learn`[1]), handling code generation, execution, and limited debugging to robustly obtain a data-driven graph G_C grounded in empirical evidence.

Procedure. Algorithm 2 summarizes the two steps.

3.5 MAC: Multi-Agent Causal Discovery Framework

Problem. DCM and MDM each address only one half of the task: DCM yields a data-grounded graph G_C but ignores metadata, while MDM reasons over metadata but is not grounded in the observed data. Combining them is not immediate because their evidence lives in different representations—G_C is a numerical adjacency matrix, whereas I and the MDM agents operate in natural language.

Why prior solutions fall short. Feeding a raw adjacency matrix to a language-based reasoner forces it to interpret numerical structure it is not designed to consume, so the data-derived evidence must instead be expressed in the same textual form as the metadata before joint reasoning is possible.

Design. MAC bridges the two stages with **Meta Fusion**, which transforms the adjacency representation of G_C into human-readable causal constraints I_c (directed edges and implied orderings). For instance, a binary adjacency matrix over {Hot, Sales, Swim, Attack} becomes constraints such as Hot→Sales, Hot→Swim, and Swim→Attack. These constraints act as *causal metadata extracted from data* and are appended to the original metadata I. MAC then runs the unified MDM operator (Algorithm 1) with m = `graph` on the combined evidence $(V, I \cup I_c)$ to produce the refined graph G_D.

Procedure. Algorithm 3 summarizes the end-to-end framework.

4 Experiment

In this section, we conduct experiments to address the following research questions: **R1**: How does MAC performance compare with other baselines in different datasets? **R2**: How does each individual module of the MAC perform when independently assessed? **R3**: Does increasing the number of debate rounds in MDM improve performance? **R4**: What is the cost and token analysis of MAC? Can it be optimized?

4.1 Experimental Setup

To answer **R1**, we design our experiments to evaluate the performance of MAC in five datasets: Child, Auto, Earthquake, Cancer, and Survey. Details can be found in Appendix D.

Baselines. To evaluate the performance of MAC, we compare it against nine competitive baselines, comprising five traditional SCD algorithms and four LLM-based methods. The traditional SCD methods include constraint-based methods such as PC (Spirtes et al., 2000) and FCI (Spirtes et al., 2013); score-based methods such as Exact Search

[1] https://causal-learn.readthedocs.io/en/latest/index.html

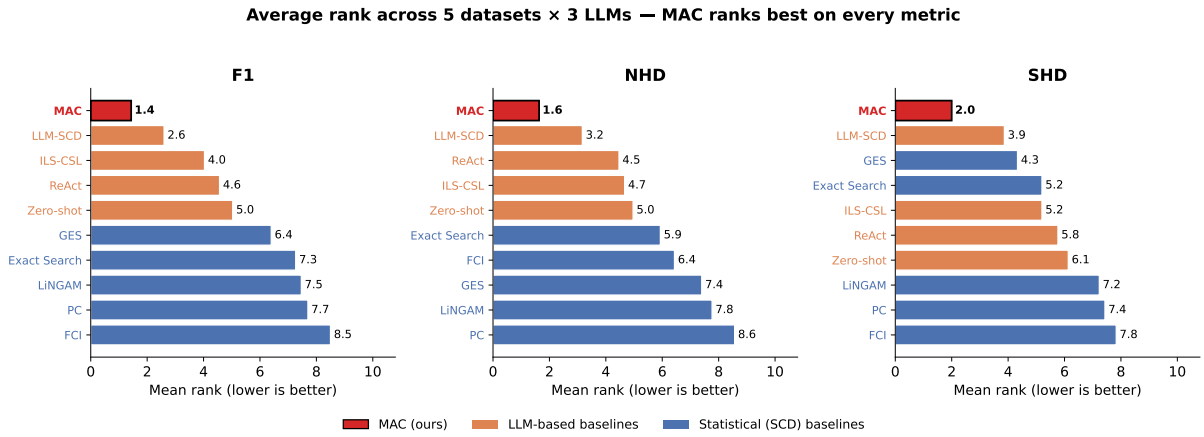

Figure 2: Average rank of each method across the five datasets and three backbone LLMs (lower is better; rank 1 = best). **MAC** (red) attains the best mean rank on all three metrics (F1 1.4, NHD 1.6, SHD 2.0), ahead of both the LLM-based baselines (orange) and the statistical SCD baselines (blue). Full per-dataset results with standard deviations are reported in Table 2 (Appendix E).

(Yuan and Malone, 2013) and Greedy Equivalence Search (GES) (Huang et al., 2018); and a functional causal model, DirectLiNGAM (Shimizu et al., 2011). The LLM-based methods consist of Zero-shot prompting (Kojima et al., 2023), ReAct prompting (Yao et al., 2023b), LLM-SCD (Ban et al., 2023a), and ILS-CSL (Takayama et al., 2024). We assess the final learned adjacency matrix of the causal graph by comparing it with the true adjacency matrix using: F1-Score, Structural Hamming Distance (SHD), Normalized Hamming Distance (NHD). The details of the evaluation metrics can be found at Appendix B.

Implementation Details. We implement our method and LLM baselines using the AutoGen[2] library and employ Gemini-2.0-Flash (Google DeepMind, 2025) via the Gemini API, DeepSeek-R1 (DeepSeek-AI et al., 2025) through TogetherAI, and the OpenAI GPT-4o-2024-08-06 model via the OpenAI API. We set a temperature of 0 to ensure deterministic outputs during causal graph construction and evaluation. For other statistical baselines, we utilize the causal-learn[3] library with its default settings. To assess robustness, we run every method over 10 random seeds and report the mean and standard deviation of each metric; across datasets the standard deviations are small relative to the margin by which MAC improves over the strongest baseline, indicating that the reported gains are stable rather than artifacts of run-to-run variation.

[2] https://microsoft.github.io/autogen/0.2/docs/Getting-Started/

[3] https://causal-learn.readthedocs.io/en/latest/

4.2 Overall Results

Figure 2 summarizes each method's average rank across the five datasets and three backbone LLMs: MAC attains the best mean rank on all three metrics, ahead of every statistical method (PC, Exact Search, GES, FCI, LiNGAM) and LLM-based method (Zero-shot, ReAct, LLM-SCD, ILS-CSL). The full per-dataset numbers are reported in Table 2 (Appendix E). Across the 15 evaluation points derived from the five datasets and three metrics, MAC demonstrates the best overall performance. Specifically, MAC integrated with Gemini-2.0-Flash secures the top position in 10 evaluation points and second place in another 4, including perfect scores (F1 = 1.00, SHD = 0, NHD = 0) on the Earthquake dataset. MAC with DeepSeek-R1 ranks first in 9 points and second in 5, while MAC with GPT-4o is first in 7 and second in another 7. MAC does not dominate every metric on every dataset—for example, GES attains the lowest Structural Hamming Distance (SHD) on Auto and Cancer, LiNGAM the lowest SHD and Normalized Hamming Distance (NHD) on Survey, and Exact Search and ReAct tie for the lowest NHD on Child—but it delivers the strongest aggregate performance and the highest F1 score on most datasets. Notably, although Gemini-2.0-Flash achieves more top rankings, DeepSeek-R1 attains higher F1 scores and lower NHD/SHD on the Child, Cancer, and Survey datasets, suggesting that its stronger reasoning capabilities further enhance results.

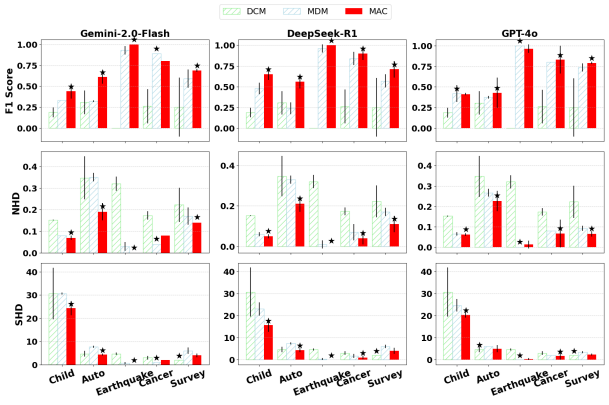

Figure 3: Comparison of F1-score, NHD, and SHD across different datasets for single DCM, single MDM, and MAC. ⋆ indicates first place.

4.3 Quantitative Analysis

To answer **R2**, we evaluate the performance of only using MDM and DCM modules individually. Detailed results can be found in Appendix E. Specifically, MAC performs better than only using MDM or DCM in most cases, except on the Earthquake dataset, where MDM has the best performance. This indicates that within MAC, the causal graph learnt from structure data from DCM is not accurate enough, and it is better to further refined via metadata from MDM. Furthermore, we observe that generally, MDM performs better than DCM, which further demonstrates the necessity of using metadata to learn the causal structure. It also validates our proposal that for most realistic problems, by utilizing their metadata with domain expert's knowledge, the causal relationships between variables can be better discovered. Of course, by combining both metadata and structured data, the performance can be further improved.

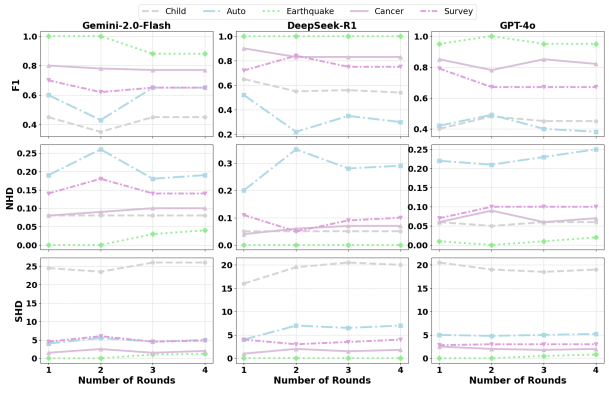

Figure 4: Performance trends of F1-score, NHD, and SHD over five rounds across different datasets.

To answer **R3**, we change the number of debating rounds between Affirmative and Negative agents in MDM, from one to five, and evaluate the corresponding performance of MAC. The hypothesis was that increasing the debate rounds could allow agents to refine their understanding of causal relationships, leading to improved structural accuracy and predictive performance. As shown in Figure 4, the results reveal that MAC effectively converges within the first round: on most datasets the performance stabilizes after a single round and merely fluctuates—rather than steadily improves—with additional rounds, while on the Survey and Auto datasets it even degrades as the round number increases. This indicates that for most datasets in reality, one round of the debating process is sufficient, with extra rounds introducing fluctuation rather than refinement in our causal analysis case. We emphasize that a single round still executes the complete Affirmative–Negative–Judge exchange, so the adversarial structure is preserved; this result therefore shows that one *round* suffices for efficiency, rather than that debate itself is unnecessary (a question we examine directly via the single-agent ablation below).

4.4 Cost Analysis

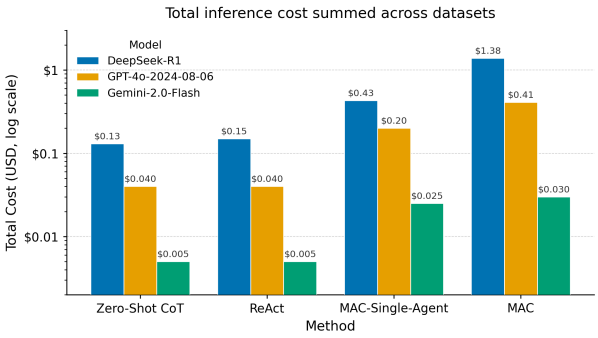

Figure 5: Total inference cost (USD) summed across datasets for four prompting methods (Zero-Shot CoT, ReAct, MAC-Single-Agent, MAC) across three models (DeepSeek-R1, GPT-4o-2024-08-06, Gemini-2.0-Flash).

To answer **R4** we detail the computational costs associated with the MAC framework. The full multi-agent system utilizes 16x to 25x more tokens than standard baselines, with the coding and debugging phase accounting for up to half of this consumption. Efficiency varies by backbone model; GPT-4o proved consistently more token-efficient than Gemini-2.0-Flash or DeepSeek-R1 due to its superior first-pass code generation. However, the framework is adaptable. Our ablation study demonstrates that the MAC-Single-Agent variant provides

a cost-effective alternative. In a rigorous evaluation across 5 datasets and 3 LLMs (45 total data points), the Single-Agent MAC variant (only one agent in the MDM, removing the Negative Agent and Judge) remained competitive with the full system on most datasets at approximately half the cost. The full debate, however, remains essential on harder, larger, or more ambiguous graphs: relative to the single-agent variant, full MAC improves F1 substantially on Survey (e.g., $0.36 \rightarrow 0.79$ with GPT-4o), Auto ($0.06 \rightarrow 0.61$ with Gemini-2.0-Flash), and Child ($0.49 \rightarrow 0.65$ with DeepSeek-R1), whereas the two variants are comparable mainly on the small five-node graphs (Earthquake, Cancer) where the task is easy enough that adversarial refinement adds little. We therefore position Single-Agent MAC as a lightweight option for simple graphs, while the full multi-agent debate is warranted when the causal structure is large or uncertain. Details can be found in Table 6.

5 Conclusion

We introduced MAC, a framework that couples an adversarial multi-agent LLM debate with the autonomous selection of a statistical causal discovery algorithm, leveraging both structured data and metadata. MAC couples a data-grounded initialization stage (DCM), which debates and executes the most suitable SCD algorithm to construct an initial graph, with a metadata-guided refinement stage (MDM), which adjudicates competing causal hypotheses through an Affirmative–Negative–Judge debate; the two stages are bridged by a Meta Fusion mechanism that expresses the data-derived graph as textual causal constraints. Across five benchmark datasets, three metrics, and three backbone LLMs, MAC delivers the best aggregate performance, outperforming both statistical and LLM-based baselines. An ablation further shows that a lightweight single-agent variant recovers much of this gain at roughly half the cost on simple graphs, while the full debate remains essential on larger or more ambiguous ones.

Looking ahead, the core ingredients of MAC—a reusable debate-and-judge operator and a Meta Fusion bridge that turns numerical structure into LLM-readable constraints—are not specific to causal graphs, so the framework may transfer to other structured-output tasks where metadata is informative. Two possibilities are time-series anomaly detection (TSAD) and financial analysis. In TSAD, a statistical detector could flag candidate anomalies and Meta Fusion could express them, together with metadata such as sensor type or known fault modes, as textual evidence for a debate to judge whether an anomaly is genuine or a benign regime shift (e.g., a real outage versus a scheduled job in server telemetry). In finance, similar reasoning could weigh whether a price movement reflects a structural event or normal volatility given contextual metadata. We view adapting MAC's output space and debate criteria to such tasks as a promising, though not yet validated, direction for future work.

6 Limitations

MAC has several limitations. First, we only consider DAGs and do not model confounders or cycles, which are common in practice; extending MAC to such settings (e.g., identifying latent confounders from metadata and adding them to the node set) is left for future work. Second, MAC relies solely on observational data and lacks interventional validation, which is crucial in domains such as medicine, economics, and biology; an agent for intervention policy design could update the graph as new experiments arrive. Third, the multi-agent debate incurs substantial computational overhead, limiting scalability; this could be reduced via a more efficient debate, smaller open-source backbones, or statistical pre-processing. Last, our evaluation uses five established benchmarks with relatively small graphs (5–20 nodes) and does not yet exercise the large-scale, data-scarce regime that motivates our work; scaling MAC to graphs with hundreds of variables is an important direction for future work. Moreover, because several of these networks are canonical and may appear in LLM pre-training corpora, part of MAC's metadata-driven performance could reflect memorized knowledge rather than genuine causal reasoning (Zečević et al., 2023; Feng et al., 2025). Grounding the initial graph in structured data through the DCM partially mitigates this risk, but a rigorous evaluation on novel or synthetically generated graphs that are unlikely to be memorized remains an important direction for future work to fully disentangle reasoning from recall.

References

Alessandro Antonucci, Gregorio Piqué, and Marco Zaffalon. 2024. Zero-shot causal graph extrapolation

from text via LLMs. In *XAI4Sci Workshop at AAAI 2024*.

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

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

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2024. ChatEval: Towards better LLM-based evaluators through multi-agent debate. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Sirui Chen, Bo Peng, Meiqi Chen, Ruiqi Wang, Mengying Xu, Xingyu Zeng, Rui Zhao, Shengjie Zhao, Yu Qiao, and Chaochao Lu. 2024a. Causal evaluation of language models. *Preprint*, arXiv:2405.00622.

Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chi-Min Chan, Heyang Yu, Yaxi Lu, Yi-Hsin Hung, Chen Qian, Yujia Qin, Xin Cong, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2024b. AgentVerse: Facilitating multi-agent collaboration and exploring emergent behaviors. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Yongchao Chen, Jacob Arkin, Yang Zhang, Nicholas Roy, and Chuchu Fan. 2024c. Scalable multi-robot collaboration with large language models: Centralized or decentralized systems? *Preprint*, arXiv:2309.15943.

Haoang Chi, He Li, Wenjing Yang, Feng Liu, Long Lan, Xiaoguang Ren, Tongliang Liu, and Bo Han. 2024. Unveiling causal reasoning in large language models: Reality or mirage? In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

Kristy Choi, Chris Cundy, Sanjari Srivastava, and Stefano Ermon. 2022. Lmpriors: Pre-trained language models as task-specific priors. *Preprint*, arXiv:2210.12530.

Kai-Hendrik Cohrs, Gherardo Varando, Emiliano Diaz, Vasileios Sitokonstantinou, and Gustau Camps-Valls. 2024. Large language models for constraint-based causal discovery. *arXiv preprint arXiv:2406.07378*.

Victor-Alexandru Darvariu, Stephen Hailes, and Mirco Musolesi. 2024. Large language models are effective priors for causal graph discovery. *arXiv preprint arXiv:2405.13551*.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*.

Tao Feng, Lizhen Qu, Niket Tandon, Zhuang Li, Xiaoxi Kang, and Gholamreza Haffari. 2025. On the reliability of large language models for causal discovery. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Clark Glymour, Kun Zhang, and Peter Spirtes. 2019. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524.

Google DeepMind. 2025. Gemini model updates - february 2025. https://blog.google/technology/google-deepmind/gemini-model-updates-february-2025/. Accessed: 2025-02-16.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 8048–8057.

Alex Havrilla, David Alvarez-Melis, and Nicolo Fusi. 2025. IGDA: Interactive graph discovery through large language model agents. *arXiv preprint arXiv:2502.17189*.

Sirui Hong, Mingchen Zhuge, Jiaqi Chen, Xiawu Zheng, Yuheng Cheng, Ceyao Zhang, Jinlin Wang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. 2024. MetaGPT: Meta programming for a multi-agent collaborative framework. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Biwei Huang, Kun Zhang, Mingming Gong, Clark Glymour, and Bernhard Schoelkopf. 2018. Generalized

score functions for causal discovery. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1551–1560.

Thomas Jiralerspong, Xiaoyin Chen, Yash More, Vedant Shah, and Yoshua Bengio. 2024. Efficient causal graph discovery using large language models. *Preprint*, arXiv:2402.01207.

Ehud Karpas, Omri Abend, Yonatan Belinkov, Barak Lenz, Opher Lieber, Nir Ratner, Yoav Shoham, Hofit Bata, Yoav Levine, Kevin Leyton-Brown, Dor Muhlgay, Noam Rozen, Erez Schwartz, Gal Shachaf, Shai Shalev-Shwartz, Amnon Shashua, and Moshe Tenenholtz. 2022. Mrkl systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *Preprint*, arXiv:2205.00445.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2023. Large language models are zero-shot reasoners. *Preprint*, arXiv:2205.11916.

Kevin B Korb and Ann E Nicholson. 2010. *Bayesian artificial intelligence*. CRC press.

Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. 2023. Causal reasoning and large language models: Opening a new frontier for causality. *Preprint*, arXiv:2305.00050.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. CAMEL: Communicative agents for "mind" exploration of large language model society. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *Preprint*, arXiv:2305.19118.

Stephanie Long, Tibor Schuster, and Alexandre Piché. 2024. Can large language models build causal graphs? *Preprint*, arXiv:2303.05279.

Juan Miguel Ogarrio, Peter Spirtes, and Joe Ramsey. 2016. A hybrid causal search algorithm for latent variable models. In *Conference on probabilistic graphical models*, pages 368–379. PMLR.

OpenAI. 2024. Function calling guide. Accessed: 2024-05-20.

Chen Qian, Xin Cong, Wei Liu, Cheng Yang, Weize Chen, Yusheng Su, Yufan Dang, Jiahao Li, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2023. Communicative agents for software development. *Preprint*, arXiv:2307.07924.

R. Quinlan. 1993. Auto MPG. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5859H.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI Technical Report.

Amartya Roy, N Devharish, Shreya Ganguly, and Kripabandhu Ghosh. 2025. Causal-LLM: A unified one-shot framework for prompt- and data-driven causal graph discovery. In *Findings of the Association for Computational Linguistics: EMNLP 2025*.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, and 7 others. 2024. Code llama: Open foundation models for code. *Preprint*, arXiv:2308.12950.

Ruben Sanchez-Romero, Joseph D Ramsey, Kun Zhang, Madelyn RK Glymour, Biwei Huang, and Clark Glymour. 2019. Estimating feedforward and feedback effective connections from fmri time series: Assessments of statistical methods. *Network Neuroscience*, 3(2):274–306.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Preprint*, arXiv:2302.04761.

Marco Scutari and Jean-Baptiste Denis. 2021. *Bayesian networks: with examples in R*. Chapman and Hall/CRC.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Preprint*, arXiv:2303.17580.

Ivaxi Sheth, Bahare Fatemi, and Mario Fritz. 2025. CausalGraph2LLM: Evaluating LLMs for causal queries. In *Findings of the Association for Computational Linguistics: NAACL 2025*.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, and Antti Kerminen. 2006. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7:2003–2030.

Shohei Shimizu, Tomomi Inazumi, Yuichiro Sogawa, Aapo Hyvärinen, Yoshinobu Kawahara, Takashi Washio, Patrik O Hoyer, and Kenneth A Bollen. 2011. Directlingam: A direct method for learning a linear non-gaussian structural equation model. *Journal of Machine Learning Research*, 12:1225–1248.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

David J Spiegelhalter. 1992. Learning in probabilistic expert systems. *Bayesian statistics*, 4:447–465.

Peter Spirtes, Clark Glymour, and Richard Scheines. 2000. *Causation, Prediction, and Search*. MIT Press.

Peter L Spirtes, Christopher Meek, and Thomas S Richardson. 2013. Causal inference in the presence of latent variables and selection bias. *arXiv preprint arXiv:1302.4983*.

Masayuki Takayama, Tadahisa Okuda, Thong Pham, Tatsuyoshi Ikenoue, Shingo Fukuma, Shohei Shimizu, and Akiyoshi Sannai. 2024. Integrating large language models in causal discovery: A statistical causal approach. *Preprint*, arXiv:2402.01454.

Aniket Vashishtha, Abbavaram Gowtham Reddy, Abhinav Kumar, Saketh Bachu, Vineeth N Balasubramanian, and Amit Sharma. 2023. Causal inference using llm-guided discovery. *Preprint*, arXiv:2310.15117.

Guangya Wan, Yunsheng Lu, Yuqi Wu, Mengxuan Hu, and Sheng Li. 2025. Large language models for causal discovery: Current landscape and future directions. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 10687–10695.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Ji-Rong Wen. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6).

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. 2023. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate. *Preprint*, arXiv:2305.11595.

Linying Yang, Vik Shirvaikar, Oscar Clivio, and Fabian Falck. 2024. A critical review of causal reasoning benchmarks for large language models. In *AAAI 2024 Workshop on "Are Large Language Models Simply Causal Parrots?"*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *Preprint*, arXiv:2305.10601.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023c. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Changhe Yuan and Brandon Malone. 2013. Learning optimal bayesian networks: A shortest path perspective. *Journal of Artificial Intelligence Research*, 48:23–65.

Matej Zečević, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. 2023. Causal parrots: Large language models may talk causality but are not causal. *Preprint*, arXiv:2308.13067.

Jintian Zhang, Xin Xu, Ningyu Zhang, Ruibo Liu, Bryan Hooi, and Shumin Deng. 2024. Exploring collaboration mechanisms for LLM agents: A social psychology view. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 14544–14607.

Kun Zhang and Aapo Hyvärinen. 2009. Causality discovery with additive disturbances: An information-theoretical perspective. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 570–585. Springer.

Qinlin Zhao, Jindong Wang, Yixuan Zhang, Yiqiao Jin, Kaijie Zhu, Hao Chen, and Xing Xie. 2023a. Competeai: Understanding the competition behaviors in large language model-based agents. *Preprint*, arXiv:2310.17512.

Zirui Zhao, Wee Sun Lee, and David Hsu. 2023b. Large language models as commonsense knowledge for large-scale task planning. *Preprint*, arXiv:2305.14078.

A Prompting of Causal Agent

A.1 Design of MDM with Causal Graph as Output

The prompts for the Meta-Debate Module (MDM) agents are carefully crafted to clarify their roles in the causal discovery process; the full templates are shown in Figure 6. Each prompt begins with a **Role**, explicitly assigning the agent's task within the framework. This clear role assignment is crucial, as experiments have shown that agents sometimes deviate from their roles. For instance, Causal Judge might inadvertently engage in debating instead of making the final decision, which necessitated clearer role prompts.

The **Your Goal** section positions the agents within a competitive debate format, highlighting the dynamics of proposing, critiquing, and evaluating causal graphs. For example, the Affirmative Debater acts as an expert in causal discovery, proposing an initial causal graph based on the provided datasets and causal principles. Conversely,

the Negative Debater critically evaluates this graph by identifying flaws and offering alternative perspectives. The judge is explicitly instructed to act as a neutral evaluator, focusing solely on assessing the factual and logical rigor of the graphs and deciding on the most plausible causal structure.

Regarding the **Causal Principles**, the most important part of the design, these principles aim to replicate the step-by-step process of an expert conducting causal discovery. The first principle prompts agents to identify whether there is a direct causal relationship between any two points. Under the assumption of causal sufficiency and no cycles, a direct causal relationship exists between two nodes if and only if these nodes cannot be conditionally independent given any other nodes. Given a list of nodes: Gene (G), Smoking (S) and Lung Cancer (C), the agents infer edges forming between S and C, and G and C, because they are not conditionally independent. No edge forms between S and G as they are assumed to be independent. Once the skeletons are identified, the second principle aims to discover the directions they take. Specifically, if there is no edge between A and C but there are edges between A and B and B and C, then B is a collider (A $\rightarrow$ B $\leftarrow$ C) if and only if A and C, though marginally independent, become dependent when conditioned on B. Following the previous example, S and G are independent, but if we know someone has Lung Cancer (C), learning about their smoking status (S) gives information about their likely genetic predisposition (G), and vice-versa. This means C is a collider, so we have S $\rightarrow$ C $\leftarrow$ G. In other situations, the causal direction cannot be identified by conditional independence alone; it can only be determined by the acyclic constraint and the causal order derived from the combination of metadata (Principle 3). Back to previous example, the collider rule directed all edges and the structure S $\rightarrow$ C $\leftarrow$ G has no cycles, therefore, this is a valid DAG.

The **Procedure** section outlines essential guidelines for agents to apply the causal principles in a debating format. Specifically, after Causal Affirmative Agent proposes the first graph, Causal Negative Agent is prompted to propose different causal networks while adhering to the principles and incorporating metadata as much as possible. The differences are primarily reflected in three aspects: the skeleton, colliders, and the direction of other edges. This results in significant differences between the agents representing the affirmative and negative sides of the debate. Lastly, Causal Judge uses a quantifiable metric to determine which causal graph presented by the affirmative and negative debaters is more reasonable. It also considers all the differences between the causal graphs proposed by both sides in terms of their skeleton, colliders, and the direction of other causal edges. The overall comparison will reveal which graph aligns more closely with the metadata, leading to the final conclusion.

Finally, the **Output Format** ensures that the agents output in a desired format for parsing the results.

A.2 Design of MDM with SCD Algorithm as Output

Similar to the prompt design for MDM with causal graph as output, the MDM framework with Statistical Causal Discovery (SCD) algorithm as output also begins by explicitly identifying the **Role** of each agent (full templates in Figure 7). However, it accepts additional inputs—such as S—and pursues a different goal. Specifically, in the **Your Goal** section, each agent is asked to select the most suitable statistical causal discovery algorithm based on dataset information (V, I, S).

Each agent is pre-prompted with several **Statistical Causal Discovery Algorithms**, such as the PC algorithm, FCI, and GES. Based on the dataset characteristics (V, I, S), agents propose different algorithms. For instance, Causal Affirmative Debater might argue that for 20 continuous temperature variables (V, I, S), GES is the best choice because it is well-suited for continuous data, can handle 20 variables, and aims to provide a clear, affirmative causal model. In contrast, Causal Negative Debater may counter that while GES is an option, temperature data (S) often includes unmeasured confounders, which GES does not address. To be more cautious with these 20 variables (V), Causal Negative Debater suggests FCI, which is designed to be robust against hidden confounders and therefore yields more reliable, though potentially less specific, causal insights. Causal Judge then evaluates these perspectives. Causal Affirmative Debater correctly points out GES's strengths for the specified continuous 20-variable data (V, I, S), while Causal Negative Debater brings up a valid concern about potential hidden confounders within the temperature data subset (S), justifying a preference for FCI's robustness. Causal Judge's decision ultimately depends on the metadata (I) associated

with subset S. If I provides strong evidence that there are no significant unmeasured confounders among the 20 variables (V) in S, then GES is chosen. Conversely, if the metadata (I) is ambiguous regarding confounders or suggests they might be present in S, FCI is selected to prioritize safety and reliability. In the absence of strong evidence from I ruling out confounders for this specific data subset (S), Causal Judge defaults to selecting FCI.

Finally, the **Output Format** also ensures that the agents output in a desired format for parsing the results.

B Evaluation Metrics

We assess the final learned adjacency matrix of the causal graph by comparing it with the true adjacency matrix using the following metrics: Structural Hamming Distance (SHD) (Takayama et al., 2024), Normalized Hamming Distance (NHD) (Kıcıman et al., 2023), and F1-Score.

B.1 Structural Hamming Distance (SHD)

The Structural Hamming Distance (SHD) for two directed graphs G and G' with m nodes is given by:

$$\mathrm{SHD}(G, G') = \sum_{1 \le i \ne j \le m} \mathbb{1}_{G_{ij} \ne G'_{ij}}$$

where $\mathbb{1}_{G_{ij} \ne G'_{ij}}$ is an indicator function that equals 1 if the directed edge from node i to node j is present in one graph but not the other, and 0 otherwise.

This metric counts the total number of directed edge disagreements between the two graphs, providing a measure of structural difference without normalization.

B.2 Normalized Hamming Distance (NHD)

The Normalized Hamming Distance (NHD) for two undirected graphs G and G' with m nodes is given by:

$$\mathrm{NHD}(G, G') = \frac{1}{\binom{m}{2}} \sum_{1 \le i < j \le m} \mathbb{1}_{G_{ij} \ne G'_{ij}}$$

where $\binom{m}{2} = \frac{m(m-1)}{2}$ is the total number of possible edges in an undirected graph with m nodes. $\mathbb{1}_{G_{ij} \ne G'_{ij}}$ is an indicator function that equals 1 if the edge between nodes i and j exist in one graph but not the other, and 0 otherwise.

This metric measures the proportion of differing edges relative to the total number of possible edges, providing a value between 0 and 1, where 0 indicates identical graphs and 1 indicates completely dissimilar graphs.

B.3 F1 Score

The F1 Score for comparing the edges of two graphs G and G' is given by:

$$\mathrm{F1}(G, G') = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

where:

- For directed graphs:
  - True Positives (TP): $\sum_{i \ne j} \mathbb{1}_{G_{ij}=1 \text{ and } G'_{ij}=1}$
  - False Positives (FP): $\sum_{i \ne j} \mathbb{1}_{G_{ij}=0 \text{ and } G'_{ij}=1}$
  - False Negatives (FN): $\sum_{i \ne j} \mathbb{1}_{G_{ij}=1 \text{ and } G'_{ij}=0}$
- Precision: $\frac{\mathrm{TP}}{\mathrm{TP+FP}}$
- Recall: $\frac{\mathrm{TP}}{\mathrm{TP+FN}}$

The F1 Score provides a balanced measure of precision and recall, adapted to the context of graph edge comparison.

C More Related Work

C.1 Prompt Engineering

Several advanced techniques for leveraging large language models (LLMs) have been identified. Zero-shot prompting, introduced by (Radford et al., 2019), guides LLMs to perform novel tasks using carefully crafted prompts without the need for training data, allowing the model to leverage its existing knowledge to generate predictions. Few-shot prompting, as described by (Brown et al., 2020), enhances model performance by providing a few input-output examples, though it requires more tokens and careful example selection to mitigate biases. For reasoning and logic, (Wei et al., 2023) introduced Chain-of-Thought (CoT) prompting, which guides LLMs through step-by-step reasoning processes, significantly improving accuracy in complex tasks such as math and commonsense reasoning. Building on this, (Wang et al., 2023) proposed self-consistency, a strategy that generates diverse reasoning chains and identifies the most consistent final answer, further enhancing accuracy. Additionally, (Yao et al., 2023c) developed ReAct, enabling LLMs to generate reasoning traces

and task-specific actions concurrently, thereby improving performance in question answering, fact verification, and interactive decision-making by enhancing the synergy between reasoning and action. In this work, we design the prompt to adhere to key principles, ensuring the correct temporal order of variables and maintaining an acyclic structure.

C.2 LLMs' Agentic Workflow

A general LLM agent framework consists of core components: user request, agent/brain, planning, memory, and tools. The agent/brain acts as the main coordinator, activated by a prompt template. It can be profiled with specific details to define its role, using handcrafted, LLM-generated, or data-driven strategies. Planning employs techniques like Chain of Thought and Tree of Thoughts, and for complex tasks, feedback mechanisms like ReAct (Yao et al., 2023c) and Reflexion (Shinn et al., 2023) refine plans based on past actions and observations. Memory stores the agent's logs, with short-term memory for the current context and long-term memory for past behaviors. Hybrid memory combines both to enhance reasoning and experience accumulation. Tools enable interaction with external environments, such as APIs and code interpreters. Frameworks like MRKL (Karpas et al., 2022), Toolformer (Schick et al., 2023), Function Calling (OpenAI, 2024), and HuggingGPT (Shen et al., 2023) integrate tools to solve tasks effectively.

However, for more complex problems where a single LLM agent may struggle, LLM-based multi-agent (LLM-MA) systems excel (Guo et al., 2024; Wang et al., 2024). Current LLM-MA systems primarily employ three communication paradigms: Cooperative, Competitive, and Debating. In the Cooperative paradigm, agents collaborate towards a shared goal, typically exchanging information to enhance a collective solution; role-specialized frameworks such as MetaGPT (Hong et al., 2024) and AgentVerse (Chen et al., 2024b) show that structured cooperation reduces cascading errors and outperforms monolithic models (Qian et al., 2023; Chen et al., 2024c). In the Competitive paradigm, agents work towards their own goals, which might conflict with those of other agents (Zhao et al., 2023a). The Debating paradigm involves agents engaging in argumentative interactions, where they present and defend their viewpoints while critiquing those of others; this approach is effective for reaching a consensus or a more refined solution and has been shown to mitigate hallucination and improve collective reasoning (Du et al., 2023; Zhang et al., 2024; Chan et al., 2024; Li et al., 2023; Liang et al., 2023; Xiong et al., 2023). In this work, the debating paradigm is adopted, as the nature of causal discovery requires diverse and potentially conflicting hypotheses to be adjudicated before converging on the truth.

D Dataset

We measure the performance of our MAC framework using five different datasets. The details of each dataset are as follows:

- **Auto** (Quinlan, 1993): It is from the UCI Machine Learning Repository and is a commonly used benchmark for causal inference. It includes variables related to car fuel consumption with five variables selected, "Weight," "Displacement," "Horsepower," "Acceleration," and "MPG" (miles per gallon) for causal inference. It has 392 continuous observations.

- **Child** (Spiegelhalter, 1992): It is a moderately sized dataset that focuses on congenital heart disease in newborns. The ground-truth graph consists of 20 nodes with 25 edges. Some key variables in this graph include Birth Asphyxia, Lung Flow, and Chest X-Ray. It has 10,000 integer observations.
- **Cancer** (Korb and Nicholson, 2010): It is a medical dataset, in the context of metastatic cancer. The ground-truth graph consists of five nodes, Pollution, Cancer, Smoker, X-ray, and Dyspnoea, with four edges. It has 10,000 binary observations.
- **Earthquake** (Korb and Nicholson, 2010): It is a dataset related to earthquake detection. The ground-truth graph consists of five nodes, Burglary, Alarm, Earthquake, John Calls, and Mary Calls, with four edges. It includes 10,000 binary observations.
- **Survey** (Scutari and Denis, 2021): It is a dataset related to demographic and social behaviors. The ground-truth graph consists of six nodes, Age, Education, Sex, Occupation, Residence, and Travel, with six edges. It includes 10,000 integer observations.

E Additional Experiment Results

This section comprises the detailed experiment results for all of the models. We note that the DCM

rows are identical across the three backbones: for each dataset, all three LLMs selected the same SCD algorithm during the debate, and since the selected algorithm is then executed deterministically on the full data, the resulting DCM graphs—and hence their metrics—coincide.

Causal Affirmative Debater

<Role> You are an expert in causal discovery for <metadata: field> participating in a debate-style competition to construct a causal graph. You are Causal Affirmative Debater.
<Your Goal> Determine causal edges and their directions between given <causal_graph_nodes> using <metadata>. If an opponent's answer is provided, critique it and offer a distinct, superior solution.
<Causal Principles>
1. Edge Existence: No edge exists between nodes A and B if they are conditionally independent given a third set of nodes C. Otherwise, an edge exists.
2. Colliders: For any three nodes A, B and C, if A-B exists (either A→B or A←B), and B-C exists, but A-C does not, and A and C are independent only when not conditioned on B, then they form a collider A→B←C.
3. Other Edges: For remaining edges, assign direction based on higher probability (e.g., if P(A→B) > P(B→A), then A→B), ensuring no directed cycles are formed.
<Procedure>
Repeat until all directions are set.
<Output Format>
Provide your step-by-step reasoning and the final graph G_A as a comma-separated list of directed edges (e.g., A→B,B→C), followed by #answer#[(A,B),(B,C)]#answer#.

Here are nodes: <causal_graph_nodes>
Here are metadata: <metadata>

Causal Negative Debater

<Role>: You are an expert in causal discovery for <metadata: field> participating in a debate-style competition to construct a causal graph. Your are Causal Negative Debater.
<Your Goal> Debunk output G_A from the Causal Affirmative Debater and generate a causal network G_N according to below principles.
<Causal Principles>
1. Edge Existence: No edge exists between nodes A and B if they are conditionally independent given a third set of nodes C. Otherwise, an edge exists.
2. Colliders: For any three nodes A, B and C, if A-B exists (either A→B or A←B), and B-C exists, but A-C does not, and A and C are independent only when not conditioned on B, then they form a collider A→B←C.
3. Other Edges: For remaining edges, assign direction based on higher probability (e.g., if P(A→B) > P(B→A), then A→B), ensuring no directed cycles are formed.
<Procedure> Utilizing metadata to ensure the conditional independence and the logical direction of causal influence, the following premises should be introduced:
- Strive to ensure that the skeleton of the G_N differs from G_A as much as possible.
- Aim to make the structure of the colliders in G_N different from that in G_A.
- Seek to ensure that, aside from colliders, the directions of other edges in G_N differ from those in G_A as much as possible.

<Output Format>
Provide your step-by-step reasoning and the final graph as a comma-separated list of directed edges (e.g., A→B,B→C), followed by #answer#[(A,B),(B,C)]#answer#.

Here are nodes: <causal_graph_nodes>
Here are metadata: <metadata>
Here are the causal graph of Causal Affirmative Debater: <G_A>

Causal Judge

<Role> You are an expert in causal discovery for <metadata: field> participating in a debate-style competition to construct a causal graph. Your are Causal Jugde of this competition.
<Your Goal> Scoring Causal Graphs G_A (Affirmative) and G_N (Negative) based on these principles.
<Causal Principles>
1. Edge Existence: No edge exists between nodes A and B if they are conditionally independent given a third set of nodes C. Otherwise, an edge exists.
2. Colliders: For any three nodes A, B and C, if A-B exists (either A→B or A←B), and B-C exists, but A-C does not, and A and C are independent only when not conditioned on B, then they form a collider A→B←C.
3. Other Edges: For remaining edges, assign direction based on higher probability (e.g., if P(A→B) > P(B→A), then A→B), ensuring no directed cycles are formed.
<Procedure>: Initialize scores: S_A=0, S_N=0
1. Skeleton Divergences (Principle 1): For each difference in the underlying undirected structure (skeleton) between G_A and G_N, add 1 to either S_A or S_N, favoring the graph whose structure more accurately reflects conditional independence relations and logically inferred causal influence per metadata.
2. Collider Divergences (Principle 2): For each differing collider, award +1 to the score of the graph whose collider structure better aligns with metadata.
3. Direction Divergences (Non-Colliders, Principle 3): For edges present in both graphs but with opposite directions (excluding those part of differing colliders), award +1 to the score of the graph whose edge direction better aligns with metadata.
4. Final Decision: If S_A > S_N, G_A is superior and is the final output. Otherwise, G_N is the final output.

<Output Format>
Provide your step-by-step reasoning and the final graph as a comma-separated list of directed edges (e.g., A→B,B→C), followed by #answer#[(A,B),(B,C)]#answer#.

Here are nodes: <causal_graph_nodes>
Here are metadata: <metadata>
Here are <G_A> and <G_N>

Figure 6: Prompt templates for the three MDM agents—Causal Affirmative Debater, Causal Negative Debater, and Causal Judge—when MDM outputs a causal graph (`graph` mode). Each prompt specifies the agent's **Role**, **Goal**, the shared **Causal Principles** (edge existence, colliders, and edge orientation), the debate **Procedure**, and the **Output Format**.

Causal Affirmative Debater

<Role> You are an expert in causal discovery for <metadata: field> participating in a debate-style competition to propose an algorithm for constructing a causal graph. Your are Causal Affirmative Debater.
<Your Goal> Determine an causal algorithm given <causal_graph_nodes>, <metadata>, and <subset_structured_data>. If an opponent's answer is provided, critique it and offer a distinct, superior solution.
<Statistical Causal Discovery Algorithms> (3-5 algorithms within the codebase)
<Output Format>
Provide your step-by-step, analyzing pros and cons of the algorithm, ouput in this format #answer#<algorithm>#answer#.

Here are nodes: <causal_graph_nodes>
Here are metadata: <metadata>
Here are subset of structured data: <subset_structured_data>

Causal Negative Debater

<Role> You are an expert in causal discovery for <metadata: field> participating in a debate-style competition to propose an algorithm for constructing a causal graph. Your are Causal Negative Debater.
<Your Goal> Determine an causal algorithm given <causal_graph_nodes>, <metadata>, and <subset_structured_data>. If an opponent's answer is provided, critique it and offer a distinct, superior solution.
<Statistical Causal Discovery Algorithms> (3-5 algorithms within the codebase)
<Output Format>
Provide your step-by-step, analyzing pros and cons of the algorithm, ouput in this format #answer#<algorithm>#answer#.

Here are nodes: <causal_graph_nodes>
Here are metadata: <metadata>
Here are subset of structured data: <subset_structured_data>
Here are the algorithm of Causal Affirmative Agent: <A_A>

Causal Judge

<Role> You are an expert in causal discovery for <metadata: field> participating in a debate-style competition to propose an algorithm for constructing a causal graph. Your are the Causal Juge of this competition.
<Your Goal> Given <causal_graph_nodes>, <metadata>, and <subset_structured_data>, determine which algorithm of Causal Graphs A_A (Affirmative) and A_N (Negative).
<Statistical Causal Discovery Algorithms> (3-5 algorithms within the codebase)
<Output Format>
Provide your step-by-step, analyzing pros and cons of the algorithm, ouput in this format #answer#<algorithm>#answer#.

Here are nodes: <causal_graph_nodes>
Here are metadata: <metadata>
Here are subset of structured data: <subset_structured_data>
Here are the algorithm of Causal Affirmative Agent: <A_A>
Here are the algorithm of Causal Negative Agent: <A_N>

Figure 7: Prompt templates for the three MDM agents when MDM outputs an SCD algorithm (`alg` mode), used inside DCM for algorithm selection. In addition to the graph nodes V and metadata I, each agent receives a small subset of structured data S and a pool of candidate SCD algorithms, then argues for and adjudicates the most suitable choice.

Table 2: Overall results across five datasets and three backbone LLMs (mean ± std over 10 seeds). For each metric within a model, **bold** marks the best method and underline the second best. Lower SHD/NHD and higher F1 are better. The five statistical baselines (Exact Search, GES, PC, FCI, LiNGAM) do not use an LLM, so their values are identical across the three models.

Dataset	Method	Gemini-2.0-Flash			DeepSeek-R1			GPT-4o		
		SHD↓	NHD↓	F1↑	SHD↓	NHD↓	F1↑	SHD↓	NHD↓	F1↑
Child	Exact Search	25.00 ± 0.00	**0.06** ± 0.00	0.00 ± 0.00	25.00 ± 0.00	0.06 ± 0.00	0.00 ± 0.00	25.00 ± 0.00	0.062 ± 0.000	0.00 ± 0.00
	GES	39.33 ± 1.70	0.15 ± 0.00	0.20 ± 0.05	39.33 ± 1.70	0.15 ± 0.00	0.20 ± 0.05	39.33 ± 1.70	0.150 ± 0.000	0.20 ± 0.05
	PC	30.67 ± 2.05	0.12 ± 0.00	0.36 ± 0.03	30.67 ± 2.05	0.12 ± 0.00	0.36 ± 0.03	30.67 ± 2.05	0.120 ± 0.000	0.36 ± 0.03
	FCI	39.67 ± 0.94	0.17 ± 0.00	0.33 ± 0.02	39.67 ± 0.94	0.17 ± 0.00	0.33 ± 0.02	39.67 ± 0.94	0.170 ± 0.000	0.33 ± 0.02
	LiNGAM	64.33 ± 2.62	0.19 ± 0.01	0.26 ± 0.01	64.33 ± 2.62	0.19 ± 0.01	0.26 ± 0.01	64.33 ± 2.62	0.190 ± 0.010	0.26 ± 0.01
	Zero-shot	27.00 ± 0.00	0.08 ± 0.00	0.33 ± 0.00	21.33 ± 1.70	0.06 ± 0.01	0.57 ± 0.06	28.67 ± 2.36	0.080 ± 0.010	0.35 ± 0.07
	ReAct	27.67 ± 2.36	**0.06** ± 0.01	0.40 ± 0.05	21.67 ± 5.73	0.06 ± 0.01	0.55 ± 0.12	23.00 ± 2.94	**0.061** ± 0.010	0.35 ± 0.08
	LLM-SCD	25.15 ± 2.50	0.08 ± 0.01	0.41 ± 0.08	16.82 ± 3.15	0.06 ± 0.02	0.61 ± 0.08	21.15 ± 1.30	0.070 ± 0.020	0.39 ± 0.03
	ILS-CSL	26.50 ± 3.00	0.09 ± 0.02	0.39 ± 0.10	17.54 ± 3.50	0.07 ± 0.03	0.59 ± 0.10	22.40 ± 1.45	0.080 ± 0.020	0.37 ± 0.04
	MAC	**24.33** ± 3.09	0.07 ± 0.01	**0.44** ± 0.09	**15.67** ± 3.09	**0.05** ± 0.01	**0.65** ± 0.07	**20.33** ± 1.25	**0.061** ± 0.010	**0.41** ± 0.02
Auto	Exact Search	7.00 ± 0.00	0.44 ± 0.00	0.15 ± 0.00	7.00 ± 0.00	0.44 ± 0.00	0.15 ± 0.00	7.00 ± 0.00	0.44 ± 0.00	0.15 ± 0.00
	GES	**3.00** ± 0.00	0.32 ± 0.00	0.57 ± 0.00	**3.00** ± 0.00	0.32 ± 0.00	**0.57** ± 0.00	**3.00** ± 0.00	0.32 ± 0.00	**0.57** ± 0.00
	PC	8.00 ± 0.00	0.48 ± 0.00	0.14 ± 0.00	8.00 ± 0.00	0.48 ± 0.00	0.14 ± 0.00	8.00 ± 0.00	0.48 ± 0.00	0.14 ± 0.00
	FCI	5.00 ± 0.00	0.24 ± 0.00	0.25 ± 0.00	5.00 ± 0.00	0.24 ± 0.00	0.25 ± 0.00	5.00 ± 0.00	0.24 ± 0.00	0.25 ± 0.00
	LiNGAM	8.00 ± 0.00	0.48 ± 0.00	0.14 ± 0.00	8.00 ± 0.00	0.48 ± 0.00	0.14 ± 0.00	8.00 ± 0.00	0.48 ± 0.00	0.14 ± 0.00
	Zero-shot	7.00 ± 0.00	0.32 ± 0.00	0.20 ± 0.00	7.00 ± 0.82	0.33 ± 0.05	0.29 ± 0.09	6.75 ± 1.09	0.32 ± 0.06	0.26 ± 0.08
	ReAct	7.67 ± 0.47	0.35 ± 0.02	0.32 ± 0.01	6.67 ± 0.47	0.32 ± 0.03	0.33 ± 0.02	6.33 ± 0.47	0.28 ± 0.03	0.37 ± 0.03
	LLM-SCD	5.10 ± 0.50	0.22 ± 0.05	0.58 ± 0.07	4.95 ± 1.10	0.23 ± 0.05	0.53 ± 0.09	5.55 ± 1.70	0.25 ± 0.06	0.41 ± 0.19
	ILS-CSL	5.50 ± 0.60	0.25 ± 0.06	0.55 ± 0.09	5.41 ± 1.25	0.25 ± 0.06	0.51 ± 0.11	5.95 ± 1.85	0.27 ± 0.07	0.39 ± 0.21
	MAC	4.33 ± 0.47	**0.19** ± 0.04	**0.61** ± 0.08	4.33 ± 0.94	**0.21** ± 0.04	0.56 ± 0.08	5.00 ± 1.63	**0.23** ± 0.05	0.43 ± 0.18
Earthquake	Exact Search	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00
	GES	4.00 ± 0.00	0.32 ± 0.00	0.00 ± 0.00	4.00 ± 0.00	0.32 ± 0.00	0.00 ± 0.00	4.00 ± 0.00	0.32 ± 0.00	0.00 ± 0.00
	PC	4.67 ± 0.00	0.35 ± 0.00	0.00 ± 0.00	4.67 ± 0.00	0.35 ± 0.00	0.00 ± 0.00	4.67 ± 0.00	0.35 ± 0.00	0.00 ± 0.00
	FCI	4.33 ± 0.00	0.25 ± 0.00	0.00 ± 0.00	4.33 ± 0.00	0.25 ± 0.00	0.00 ± 0.00	4.33 ± 0.00	0.25 ± 0.00	0.00 ± 0.00
	LiNGAM	4.00 ± 0.00	0.28 ± 0.00	0.22 ± 0.00	4.00 ± 0.00	0.28 ± 0.00	0.22 ± 0.00	4.00 ± 0.00	0.28 ± 0.00	0.22 ± 0.00
	Zero-shot	1.00 ± 0.00	0.04 ± 0.00	0.89 ± 0.00	1.00 ± 0.00	0.04 ± 0.00	0.89 ± 0.00	1.00 ± 0.00	0.04 ± 0.00	0.89 ± 0.00
	ReAct	1.00 ± 0.00	0.04 ± 0.00	0.89 ± 0.00	1.00 ± 0.00	0.04 ± 0.00	0.89 ± 0.00	1.00 ± 0.00	0.04 ± 0.00	0.89 ± 0.00
	LLM-SCD	0.50 ± 0.10	0.01 ± 0.01	0.95 ± 0.02	0.45 ± 0.15	0.01 ± 0.01	0.96 ± 0.03	0.40 ± 0.50	0.02 ± 0.02	0.94 ± 0.06
	ILS-CSL	0.80 ± 0.20	0.02 ± 0.01	0.92 ± 0.03	0.78 ± 0.25	0.02 ± 0.01	0.93 ± 0.04	0.50 ± 0.60	0.03 ± 0.03	0.92 ± 0.07
	MAC	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00	0.33 ± 0.47	0.01 ± 0.02	0.96 ± 0.05
Cancer	Exact Search	2.33 ± 0.00	0.17 ± 0.00	0.44 ± 0.00	2.33 ± 0.00	0.17 ± 0.00	0.44 ± 0.00	2.33 ± 0.00	0.17 ± 0.00	0.44 ± 0.00
	GES	**1.33** ± 0.00	0.17 ± 0.00	0.62 ± 0.00	1.33 ± 0.00	0.17 ± 0.00	0.62 ± 0.00	1.33 ± 0.00	0.17 ± 0.00	0.62 ± 0.00
	PC	2.00 ± 0.00	0.16 ± 0.00	0.50 ± 0.00	2.00 ± 0.00	0.16 ± 0.00	0.50 ± 0.00	2.00 ± 0.00	0.16 ± 0.00	0.50 ± 0.00
	FCI	4.00 ± 0.00	0.16 ± 0.00	0.00 ± 0.00	4.00 ± 0.00	0.16 ± 0.00	0.00 ± 0.00	4.00 ± 0.00	0.16 ± 0.00	0.00 ± 0.00
	LiNGAM	2.33 ± 0.00	0.17 ± 0.00	0.43 ± 0.00	2.33 ± 0.00	0.17 ± 0.00	0.43 ± 0.00	2.33 ± 0.00	0.17 ± 0.00	0.43 ± 0.00
	Zero-shot	3.00 ± 0.00	0.12 ± 0.00	0.73 ± 0.00	1.33 ± 0.47	0.05 ± 0.02	0.86 ± 0.04	2.00 ± 0.00	0.08 ± 0.00	0.80 ± 0.00
	ReAct	2.00 ± 0.00	**0.08** ± 0.00	**0.80** ± 0.00	**0.67** ± 0.47	**0.03** ± 0.02	**0.93** ± 0.05	**1.00** ± 0.00	**0.04** ± 0.00	**0.89** ± 0.00
	LLM-SCD	2.50 ± 0.10	0.10 ± 0.01	0.75 ± 0.02	1.45 ± 0.90	0.06 ± 0.04	0.87 ± 0.09	1.85 ± 1.75	0.08 ± 0.08	0.81 ± 0.18
	ILS-CSL	2.80 ± 0.20	0.12 ± 0.02	0.72 ± 0.03	1.83 ± 1.10	0.07 ± 0.05	0.84 ± 0.11	2.10 ± 1.90	0.09 ± 0.09	0.79 ± 0.20
	MAC	2.00 ± 0.00	**0.08** ± 0.00	**0.80** ± 0.00	1.00 ± 0.82	0.04 ± 0.03	0.90 ± 0.08	1.67 ± 1.70	0.07 ± 0.07	0.83 ± 0.17
Survey	Exact Search	4.33 ± 0.00	0.21 ± 0.00	0.28 ± 0.00	4.33 ± 0.00	0.21 ± 0.00	0.28 ± 0.00	4.33 ± 0.00	0.21 ± 0.00	0.28 ± 0.00
	GES	3.00 ± 0.00	0.17 ± 0.00	0.54 ± 0.00	3.00 ± 0.00	0.17 ± 0.00	0.54 ± 0.00	3.00 ± 0.00	0.17 ± 0.00	0.54 ± 0.00
	PC	4.33 ± 0.00	0.22 ± 0.00	0.42 ± 0.00	4.33 ± 0.00	0.22 ± 0.00	0.42 ± 0.00	4.33 ± 0.00	0.22 ± 0.00	0.42 ± 0.00
	FCI	5.00 ± 0.00	0.19 ± 0.00	0.22 ± 0.00	5.00 ± 0.00	0.19 ± 0.00	0.22 ± 0.00	5.00 ± 0.00	0.19 ± 0.00	0.22 ± 0.00
	LiNGAM	**2.67** ± 0.00	**0.11** ± 0.00	0.65 ± 0.00	**2.67** ± 0.00	**0.11** ± 0.00	0.65 ± 0.00	2.67 ± 0.00	0.11 ± 0.00	0.65 ± 0.00
	Zero-shot	4.00 ± 0.00	**0.11** ± 0.00	0.67 ± 0.00	7.00 ± 1.63	0.19 ± 0.05	0.55 ± 0.13	7.00 ± 0.82	0.19 ± 0.02	0.58 ± 0.10
	ReAct	6.00 ± 0.00	0.17 ± 0.00	0.60 ± 0.04	8.00 ± 0.82	0.22 ± 0.02	0.45 ± 0.06	7.33 ± 1.25	0.20 ± 0.03	0.45 ± 0.04
	LLM-SCD	4.50 ± 0.80	0.16 ± 0.01	0.66 ± 0.02	4.67 ± 1.55	0.13 ± 0.05	0.68 ± 0.11	2.65 ± 0.50	0.07 ± 0.02	0.77 ± 0.02
	ILS-CSL	4.90 ± 0.90	0.18 ± 0.02	0.63 ± 0.03	5.21 ± 1.80	0.15 ± 0.06	0.65 ± 0.13	2.90 ± 0.60	0.08 ± 0.02	0.75 ± 0.03
	MAC	4.00 ± 0.82	0.14 ± 0.00	**0.69** ± 0.02	4.00 ± 1.41	**0.11** ± 0.04	**0.71** ± 0.10	**2.33** ± 0.47	**0.06** ± 0.01	**0.79** ± 0.01

Table 3: Comparison of methods on Child, Auto, Earthquake, Cancer, and Survey (with standard deviations) with other modules of MAC framework (Gemini Flash 2.0).

Dataset	**Method**	**SHD** (↓)	**NHD** (↓)	**F1** (↑)
Child	DCM	30.67 ± 11.09	0.15 ± 0.00	0.19 ± 0.06
	MDM	30.67 ± 0.47	0.08 ± 0.00	0.33 ± 0.00
	MAC	**24.33** ± 3.09	**0.07** ± 0.01	**0.44** ± 0.09
Auto	DCM	**4.67** ± 1.25	0.35 ± 0.10	0.31 ± 0.14
	MDM	7.67 ± 0.47	0.35 ± 0.02	0.32 ± 0.01
	MAC	4.33 ± 0.47	**0.19** ± 0.04	**0.61** ± 0.08
Earthquake	DCM	4.67 ± 0.47	0.32 ± 0.03	0.00 ± 0.00
	MDM	0.67 ± 0.47	0.03 ± 0.02	0.93 ± 0.05
	MAC	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00
Cancer	DCM	3.00 ± 0.82	0.17 ± 0.02	0.26 ± 0.20
	MDM	**1.00** ± 0.00	**0.04** ± 0.00	**0.89** ± 0.00
	MAC	2.00 ± 0.00	0.08 ± 0.00	0.80 ± 0.00
Survey	DCM	**2.00** ± 0.00	0.22 ± 0.08	0.25 ± 0.35
	MDM	6.00 ± 1.41	0.17 ± 0.04	0.59 ± 0.11
	MAC	4.00 ± 0.82	**0.14** ± 0.00	**0.69** ± 0.02

Table 4: Comparison of methods on Child, Auto, Earthquake, Cancer, and Survey (with standard deviations) with other modules of MAC framework (DeepSeek-R1).

Dataset	**Method**	**SHD** (↓)	**NHD** (↓)	**F1** (↑)
Child	DCM	30.67 ± 11.09	0.15 ± 0.00	0.19 ± 0.06
	MDM	23.00 ± 2.94	0.06 ± 0.01	0.48 ± 0.07
	MAC	**15.67** ± 3.09	**0.05** ± 0.01	**0.65** ± 0.07
Auto	DCM	**4.67** ± 1.25	0.35 ± 0.10	0.31 ± 0.14
	MDM	7.33 ± 0.47	0.33 ± 0.02	0.24 ± 0.07
	MAC	4.33 ± 0.94	**0.21** ± 0.04	**0.56** ± 0.08
Earthquake	DCM	4.67 ± 0.47	0.32 ± 0.03	0.00 ± 0.00
	MDM	0.33 ± 0.47	0.01 ± 0.02	0.96 ± 0.05
	MAC	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00
Cancer	DCM	3.00 ± 0.82	0.17 ± 0.02	0.26 ± 0.20
	MDM	1.67 ± 0.94	0.07 ± 0.04	0.84 ± 0.08
	MAC	**1.00** ± 0.82	**0.04** ± 0.03	**0.90** ± 0.08
Survey	DCM	**2.00** ± 0.00	0.22 ± 0.08	0.25 ± 0.35
	MDM	6.00 ± 0.82	0.17 ± 0.02	0.57 ± 0.08
	MAC	4.00 ± 1.41	**0.11** ± 0.04	**0.71** ± 0.10

Table 5: Comparison of Methods on Child, Auto, Earthquake, Cancer, and Survey (with standard deviations) with other modules of MAC framework (GPT-4o).

Dataset	**Method**	**SHD** (↓)	**NHD** (↓)	**F1** (↑)
Child	DCM	30.67 ± 11.09	0.15 ± 0.00	0.19 ± 0.06
	MDM	24.67 ± 2.87	0.07 ± 0.01	0.42 ± 0.10
	MAC	**20.33** ± 1.25	**0.06** ± 0.01	**0.41** ± 0.02
Auto	DCM	**4.67** ± 1.25	0.35 ± 0.10	0.31 ± 0.14
	MDM	6.00 ± 0.00	0.27 ± 0.02	0.38 ± 0.02
	MAC	5.00 ± 1.63	**0.23** ± 0.05	**0.43** ± 0.18
Earthquake	DCM	4.67 ± 0.47	0.32 ± 0.03	0.00 ± 0.00
	MDM	**0.00** ± 0.00	**0.00** ± 0.00	**1.00** ± 0.00
	MAC	0.33 ± 0.47	0.01 ± 0.02	0.96 ± 0.05
Cancer	DCM	3.00 ± 0.82	0.17 ± 0.02	0.26 ± 0.20
	MDM	2.00 ± 0.00	0.08 ± 0.00	0.80 ± 0.00
	MAC	**1.67** ± 1.70	**0.07** ± 0.07	**0.83** ± 0.17
Survey	DCM	**2.00** ± 0.00	0.22 ± 0.08	0.25 ± 0.35
	MDM	3.33 ± 0.47	0.09 ± 0.01	0.74 ± 0.05
	MAC	2.33 ± 0.47	**0.06** ± 0.01	**0.79** ± 0.01

Table 6: Performance comparison of Single-Agent MAC vs. MAC (full multi-agent) across different LLMs. Lower is better for SHD and NHD; higher is better for F1.

Dataset	Method	SHD (↓)	NHD (↓)	F1 (↑)
Gemini-2.0-Flash				
Child	Single-Agent MAC	25.33 ±0.58	0.07 ±0.01	0.15 ±0.26
	MAC	24.33 ±3.09	0.07 ±0.01	0.44 ±0.09
Auto	Single-Agent MAC	5.33 ±0.58	0.25 ±0.09	0.06 ±0.10
	MAC	4.33 ±0.47	0.19 ±0.04	0.61 ±0.08
Earthquake	Single-Agent MAC	0.00 ±0.00	0.00 ±0.00	1.00 ±0.00
	MAC	0.00 ±0.00	0.00 ±0.00	1.00 ±0.00
Cancer	Single-Agent MAC	1.67 ±2.08	0.07 ±0.08	0.63 ±0.55
	MAC	2.00 ±0.00	0.08 ±0.00	0.80 ±0.00
Survey	Single-Agent MAC	9.00 ±1.41	0.28 ±0.04	0.45 ±0.04
	MAC	4.00 ±0.82	0.14 ±0.00	0.69 ±0.02
DeepSeek-R1				
Child	Single-Agent MAC	19.18 ±3.98	0.05 ±0.01	0.49 ±0.21
	MAC	15.67 ±3.09	0.05 ±0.01	0.65 ±0.07
Auto	Single-Agent MAC	6.25 ±1.50	0.35 ±0.05	0.16 ±0.12
	MAC	4.33 ±0.94	0.21 ±0.04	0.56 ±0.08
Earthquake	Single-Agent MAC	0.25 ±0.90	0.02 ±0.07	0.95 ±0.21
	MAC	0.00 ±0.00	0.00 ±0.00	1.00 ±0.00
Cancer	Single-Agent MAC	0.77 ±1.51	0.03 ±0.06	0.87 ±0.30
	MAC	1.00 ±0.82	0.04 ±0.03	0.90 ±0.08
Survey	Single-Agent MAC	8.77 ±1.30	0.26 ±0.04	0.28 ±0.12
	MAC	4.00 ±1.41	0.11 ±0.04	0.71 ±0.10
GPT-4o				
Child	Single-Agent MAC	28.67 ±1.53	0.09 ±0.00	0.34 ±0.01
	MAC	20.33 ±1.25	0.06 ±0.01	0.41 ±0.02
Auto	Single-Agent MAC	6.67 ±1.15	0.41 ±0.02	0.11 ±0.10
	MAC	5.00 ±1.63	0.23 ±0.05	0.43 ±0.18
Earthquake	Single-Agent MAC	0.00 ±0.00	0.00 ±0.00	1.00 ±0.00
	MAC	0.33 ±0.47	0.01 ±0.02	0.96 ±0.05
Cancer	Single-Agent MAC	0.00 ±0.00	0.00 ±0.00	1.00 ±0.00
	MAC	1.67 ±1.70	0.07 ±0.07	0.83 ±0.17
Survey	Single-Agent MAC	11.67 ±0.58	0.33 ±0.00	0.36 ±0.04
	MAC	2.33 ±0.47	0.06 ±0.01	0.79 ±0.01