


# Abduct, Act, Predict: Scaffolding Causal Inference for Automated Failure Attribution in Multi-Agent Systems

Alva West<sup>1</sup>, Yixuan Weng<sup>1</sup>, Minjun Zhu<sup>1</sup>, Zhen Lin<sup>1</sup>, Yue Zhang<sup>1</sup>

<sup>1</sup>Engineering School, Westlake University

Failure attribution in multi-agent systems—pinpointing the exact step where a decisive error occurs—is a critical yet unsolved challenge. Current methods treat this as a pattern recognition task over long conversation logs, leading to critically low step-level accuracy (below 17%), which renders them impractical for debugging complex systems. Their core weakness is a fundamental inability to perform robust counterfactual reasoning: to determine if correcting a single action would have actually averted the task failure. To bridge this *counterfactual inference gap*, we introduce **Abduct-Act-Predict (A2P) Scaffolding**, a novel agent framework that transforms failure attribution from pattern recognition into a structured causal inference task. A2P explicitly guides a large language model through a formal three-step reasoning process within a single inference pass: (1) **Abduction**, to infer the hidden root causes behind an agent’s actions; (2) **Action**, to define a minimal corrective intervention; and (3) **Prediction**, to simulate the subsequent trajectory and verify if the intervention resolves the failure. This structured approach leverages the holistic context of the entire conversation while imposing a rigorous causal logic on the model’s analysis. Our extensive experiments on the Who&When benchmark demonstrate its efficacy. On the Algorithm-Generated dataset, A2P achieves **47.46%** step-level accuracy, a **2.85×** improvement over the 16.67% of the baseline. On the more complex Hand-Crafted dataset, it achieves **29.31%** step accuracy, a **2.43×** improvement over the baseline’s 12.07%. By reframing the problem through a causal lens, A2P Scaffolding provides a robust, verifiable, and significantly more accurate solution for automated failure attribution.

 **Date:** September 13, 2025

 **Code Repository:** <https://github.com/ResearAI/A2P>

 **Contact:** [zhangyue@westlake.edu.cn](mailto:zhangyue@westlake.edu.cn)

## 1. Introduction

The rise of sophisticated multi-agent systems marks a pivotal moment in artificial intelligence, unlocking new frontiers in collaborative problem-solving (Li et al., 2023, Hong et al., 2023) and complex task automation (Wu et al., 2023, Fourney et al., 2024). However, this growing complexity introduces a critical operational bottleneck: debugging. When a system fails, developers are faced with a tangled web of interactions, where a subtle error in an early step can cascade into a catastrophic failure dozens of turns later. Pinpointing the single, decisive error—the task of **failure attribution**—is not merely challenging; it is a labor-intensive, error-prone process that stands as a major barrier to the reliable deployment and iterative improvement of these powerful systems (Zhang et al., 2025).

Current automated approaches to this problem have proven fundamentally inadequate, with step-level accuracy rates hovering below a dismal 17% (Zhang et al., 2025), a figure far too low for practical debugging.

We argue this failure is not a matter of model capability but of methodological paradigm. Existing methods treat failure attribution as a **pattern recognition** task over conversational logs (Zhang et al., 2025, Lightman et al., 2023). They present an entire log to a Large Language Model (LLM) and ask it to "find the mistake," implicitly assuming the model can spot anomalous patterns correlated with failure. This approach fundamentally misses the point. The critical question is not "which step looks wrong?" but rather a causal one: "which single corrective action would have turned failure into success?" This exposes a deep *counterfactual inference gap*: the inability of unstructured, holistic methods to systematically reason about the consequences of hypothetical interventions, a challenge particularly pronounced in multi-turn interactions where cause and effect are obscured (Kiciman et al., 2023, Zevcevic et al., 2023).

To bridge this gap, we introduce **Abduct-Act-Predict (A2P)**, a novel prompting framework that reframes failure attribution from pattern recognition into a structured **causal inference** task. Instead of asking for a direct answer, A2P guides an LLM through a formal, three-step counterfactual reasoning process within a single inference pass, operationalizing the logic of Pearl’s structural causal model hierarchy (Pearl, 2009). The framework compels the model to: (1) *Abduct*, inferring hidden factors (e.g., a flawed assumption) that explain a problematic action; (2) *Act*, defining a minimal, concrete corrective intervention; and (3) *Predict*, simulating the subsequent counterfactual trajectory to verify if the intervention would have resolved the overall task failure. This structured process forces the model to move beyond correlation and rigorously test causal hypotheses, transforming the "needle-in-the-haystack" problem (Liu et al., 2024) into a systematic investigation.

Our approach is not just theoretically sound but empirically dominant. Evaluated on the comprehensive Who&When benchmark (Zhang et al., 2025), A2P Scaffolding achieves a step-level accuracy of **47.46%** on the Algorithm-Generated dataset—a **2.85×** improvement over the 16.67% of its direct baseline. On the more challenging Hand-Crafted dataset, it achieves **29.31%** accuracy, a **2.43×** improvement over the baseline’s 12.07%. These results establish a new state-of-the-art and, for the first time, demonstrate a viable path toward reliable automated debugging for multi-agent systems. Rigorous ablation studies further validate our framework, confirming that each causal reasoning component is essential and revealing the surprising, critical role of structural cues like contextual step numbering in enabling fine-grained analysis.

## 2. Related Work

### 2.1. LLM Multi-Agent Systems

The emergence of Large Language Models as capable reasoning agents has catalyzed rapid development in multi-agent system architectures (Wu et al., 2023, Li et al., 2023, Hong et al., 2023). These systems leverage the collaborative potential of multiple specialized agents working together to solve complex tasks that exceed the capabilities of individual models (Park et al., 2023, Liu et al., 2023b). Notable frameworks include AutoGen (Wu et al., 2023), which facilitates multi-agent conversations through customizable agent roles and interaction patterns, CAMEL (Li et al., 2023), which explores role-playing dynamics in collaborative task-solving, and MetaGPT (Hong et al., 2023), which incorporates software development methodologies into multi-agent workflows. Recent work has expanded these foundations to include specialized domains such as scientific research (Ghafarollahi and Buehler, 2024), software development (Kumar et al., 2024), and complex reasoning tasks (Du et al., 2023). However, as these systems grow in sophistication, the challenge of diagnosing failures becomes increasingly complex, with current debugging approaches remaining largely manual and ad-hoc (Wang et al., 2024b). The need for automated failure attribution becomes particularly

acute in production deployments where system reliability directly impacts user experience and operational efficiency (Fourney et al., 2024).

The rapid proliferation of multi-agent systems has outpaced the development of systematic debugging methodologies. While considerable effort has been invested in designing agent architectures and interaction protocols (Qian et al., 2023, Chen et al., 2024), relatively little attention has been paid to post-hoc failure analysis. This gap is particularly problematic given the emergent behaviors that arise from agent interactions, where system failures often result from subtle cascading effects rather than obvious individual errors (Kumar et al., 2024). Our work addresses this critical gap by providing the first systematic framework for automated failure attribution specifically designed for the unique challenges of multi-agent system debugging. Unlike previous approaches that focus on system design or performance evaluation (Wang et al., 2024b), we concentrate on the diagnostic phase that enables iterative improvement and reliable deployment.

## 2.2. LLM-as-a-Judge and Process-Level Evaluation

The paradigm of using LLMs as evaluators has gained significant traction as a scalable alternative to human assessment across diverse domains (Zheng et al., 2023, Wang et al., 2024a). This approach has proven particularly valuable in scenarios where human evaluation is expensive, time-consuming, or requires specialized expertise (Liu et al., 2023a, Dubois et al., 2023). Recent developments have extended LLM-based evaluation to process-level assessment, where models evaluate intermediate reasoning steps rather than only final outputs (Lightman et al., 2023, Wang et al., 2023). Process reward models (Uesato et al., 2022) have shown promise in mathematical reasoning by identifying the specific steps where errors occur, enabling more targeted feedback and improvement strategies. However, these approaches primarily focus on single-agent reasoning chains in well-defined domains like mathematics or coding, where the correctness of individual steps can be objectively determined.

Our work extends this process-level evaluation paradigm to the significantly more complex domain of multi-agent system failures. Unlike mathematical reasoning where step correctness is often binary and context-independent, multi-agent failures involve complex interdependencies between agents, temporal dynamics, and emergent behaviors that resist simple classification (Du et al., 2023). While process reward models evaluate individual reasoning steps, our A2P framework must navigate the multi-participant, interactive dynamics of agent systems where the "correctness" of an action depends heavily on the broader conversational context and the ultimate task outcome. This fundamental difference necessitates our novel approach of structured counterfactual reasoning rather than step-by-step correctness assessment (Miller, 2019, Doshi-Velez and Kim, 2017).

## 2.3. Causal Reasoning in LLMs

Recent research has begun exploring the causal reasoning capabilities of large language models, revealing both promising potential and significant limitations (Kiciman et al., 2023, Zevcevic et al., 2023). Benchmarks such as CLadder (Qin et al., 2023) and CausalBench (Jin et al., 2023) have established that while LLMs can perform certain types of causal reasoning, they often struggle with complex counterfactual inference tasks that require systematic manipulation of causal variables (Jin et al., 2024). This limitation is particularly pronounced in scenarios requiring what Pearl terms "Level 3" causal reasoning, answering questions about what would have happened under different circumstances (Pearl, 2009). Studies have shown that structured prompting approaches, such as CausalCoT (Zhang et al., 2024), can significantly enhance LLM performance

on causal tasks by providing explicit reasoning frameworks that guide model inference.

Building on these insights, our A2P Scaffolding framework represents a practical application of structured causal prompting to a real-world diagnostic task. While previous work has focused on synthetic causal reasoning benchmarks or simplified scenarios (Jin et al., 2023, Qin et al., 2023), we tackle the significantly more complex challenge of failure attribution in multi-agent systems where causal relationships are embedded in natural language conversations and span multiple participants over extended time horizons. Our approach operationalizes Pearl’s three-level causal hierarchy (Pearl et al., 2016) into a concrete prompting strategy that enables LLMs to perform sophisticated counterfactual analysis. Unlike previous causal reasoning work that typically evaluates models on isolated causal queries, we demonstrate how structured causal prompting can address practical system debugging challenges where the stakes of accurate causal inference directly impact development efficiency and system reliability (Schölkopf et al., 2021, Peters et al., 2017).

### 3. Method

The challenge of automated failure attribution in multi-agent systems stems from the inherent complexity of causal reasoning over extended, multi-participant conversational sequences. Existing baseline methods, while processing the complete contextual information, treat attribution as a monolithic pattern recognition task, implicitly assuming that LLMs can perform comprehensive counterfactual reasoning within a single, unstructured inference step, an assumption contradicted by recent benchmarks evaluating LLM causal capabilities (Kiciman et al., 2023, Zevcevic et al., 2023). This assumption leads to a critical analytical bottleneck: models may successfully identify correlations or surface-level errors but systematically fail to determine whether those errors were truly *decisive*—that is, whether their correction would have altered the task outcome from failure to success. This *counterfactual inference gap* constitutes the primary cause of the characteristically low step-level accuracy observed in existing attribution systems (Zhang et al., 2025).

To bridge this gap, we introduce Abduct-Act-Predict (A2P) Scaffolding, a novel prompting framework that restructures the failure attribution task into a formal, three-step causal inference process. Our approach is implemented as an enhancement to the All-at-Once method, thereby retaining its key advantage of having access to the complete conversational context. However, instead of a simple instruction, we employ a sophisticated prompt generation function, `construct_causal_prompt`, that guides the LLM through a rigorous analytical sequence inspired by Pearl’s structural causal model framework (Pearl, 2009). This method makes the reasoning process transparent, verifiable, and significantly more accurate without requiring any changes to the underlying model architecture.

The core of A2P Scaffolding is its three-step reasoning structure, illustrated in Figure 1. **(1) Abduction (Inferring Hidden Causes):** The process begins by prompting the LLM to move beyond mere observation to abductive reasoning. Given the final task failure, the model is instructed to identify and articulate the hidden factors or latent variables (e.g., an agent’s knowledge gap, a flawed assumption, a misinterpretation of the user’s query) that best explain why a specific agent took a specific action at a specific step. This approximates the posterior inference of exogenous variables in a causal model, forcing the model to establish a plausible root cause before proceeding. **(2) Action (Defining an Intervention):** Once a potential root cause and erroneous action are hypothesized, the framework prompts the LLM to define a minimal, concrete intervention. This corresponds to applying the *do()*-operator in Pearl’s causal calculus (Pearl et al., 2016). The model must specify the exact, “correct” action the agent should have taken in that step. This step is crucial as it translates the abstract hypothesis into a testable, operationalized counterfactual. **(3) Prediction**

**(Simulating the Counterfactual Trajectory):** With the intervention defined, the final step is to predict its consequences. The LLM is instructed to simulate the subsequent 3-5 turns of the conversation under the counterfactual condition that the correct action was taken. It must then predict whether this new, simulated trajectory would lead to the successful completion of the original task. This step directly evaluates the *decisive* nature of the error; if the simulated outcome is success, the hypothesis is confirmed.

Mathematically, A2P Scaffolding approximates the estimation of a counterfactual outcome  $Z(\mathcal{I}_{(i,t)}(\tau))$  for an intervention at step  $t$ . We formalize the failure attribution task within Pearl’s SCM framework where a trajectory  $\tau$  is generated by structural equations with states evolving as  $s_{t+1} = f(s_t, a_t, \epsilon_t)$ , where  $\epsilon_t$  represents unobserved exogenous variables (e.g., agent’s internal knowledge state). The final outcome  $Z(\tau)$  is a function of the full trajectory. Our objective is to find the earliest pair  $(i^*, t^*) = \arg \min_{(i,t)} t$  such that the LLM’s guided simulation predicts  $Z(\mathcal{I}_{(i,t)}(\tau)) = 0$  (success). The A2P framework guides the LLM through three approximations:

$$\text{Abduction: } \epsilon_t \leftarrow \arg \max_{\epsilon} P(\epsilon | s_{0:t}, a_t, Z(\tau) = 1) \quad (1)$$

$$\text{Action: } do(a_t \leftarrow a_t^*) \quad (2)$$

$$\text{Prediction: } Z(\tau^*) = g(s_0, \dots, s_t, s_{t+1}^*, \dots) \text{ where } s_{t+1}^* = f(s_t, a_t^*, \epsilon_t) \quad (3)$$

This entire three-step process is executed for each potential error the model considers, and it ultimately outputs the earliest agent-step pair that satisfies this causal chain. To support this fine-grained temporal reasoning, our method incorporates a critical structural component: **Contextual Step Numbering**. Before being passed to the model, the entire conversation log is pre-processed to prefix each turn with an explicit, formatted identifier like `Step {idx} - Agent_Name:`. Our ablation experiments conclusively demonstrate that these structural anchors are not merely a minor enhancement but are absolutely essential, preventing a catastrophic drop in step-level accuracy by providing the model with unambiguous reference points to trace causal dependencies through the dialogue.

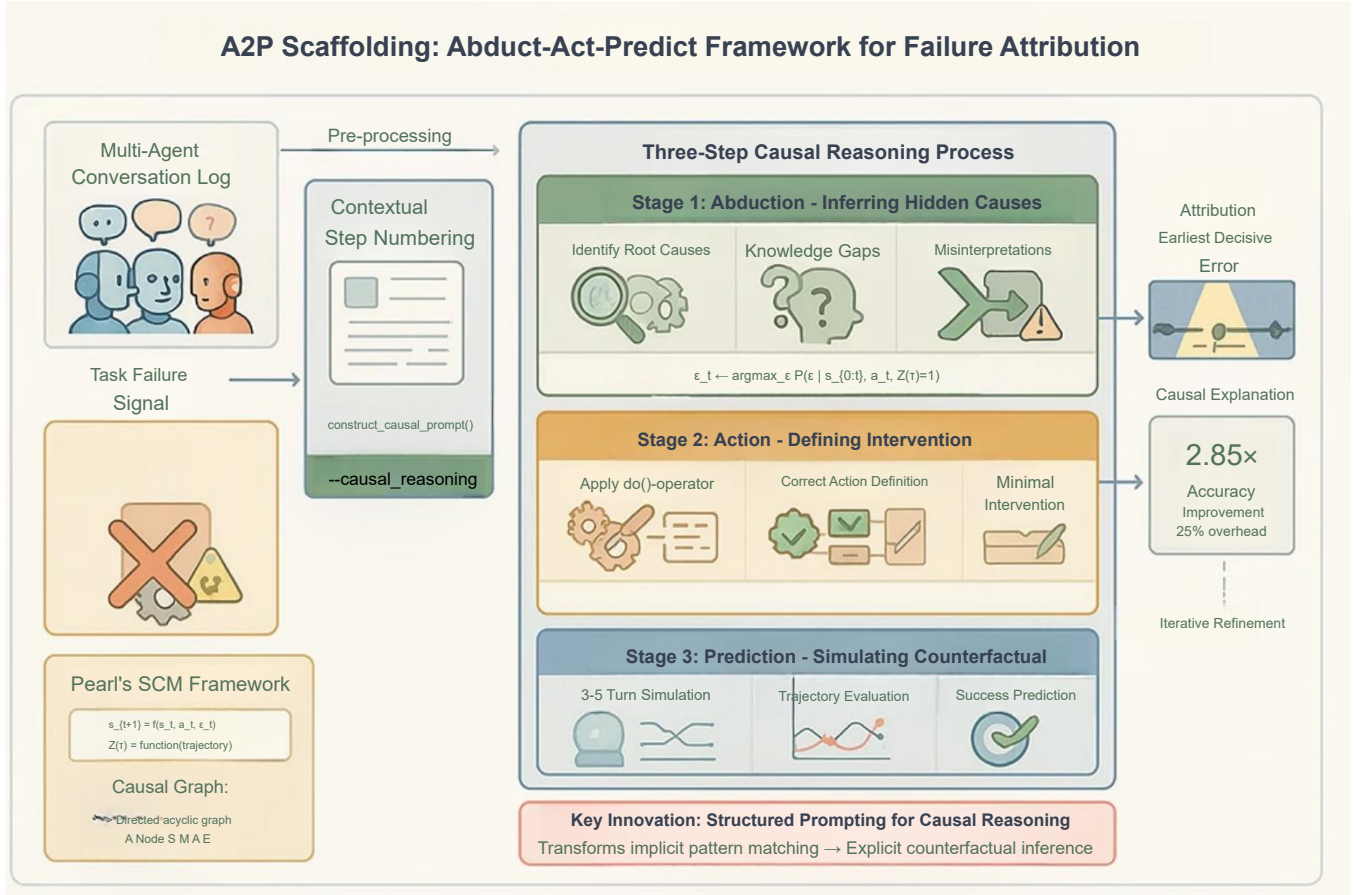
The implementation is seamlessly integrated into the existing codebase through a command-line flag `-causal_reasoning` that activates the `construct_causal_prompt` function within the `all_at_once` and `all_at_once_async` methods. This design ensures full backward compatibility while making our advanced causal analysis easily accessible. The computational overhead is minimal, consisting of a 25% increase in processing time and token count per sample—a modest cost for the 2.85× improvement in accuracy achieved by our method.

Having established the theoretical foundation and implementation details of A2P Scaffolding, we proceed to describe our comprehensive experimental methodology designed to rigorously evaluate the framework’s effectiveness across diverse multi-agent system configurations and failure scenarios.

## 4. Experimental Setup

All experiments were conducted on the Who&When benchmark (Zhang et al., 2025), a comprehensive dataset specifically designed for automated failure attribution in multi-agent systems. The benchmark comprises two distinct subsets that provide complementary perspectives on system complexity: Algorithm-Generated (126 samples) and Hand-Crafted (58 samples), totaling 184 distinct failure attribution tasks. The Algorithm-Generated subset contains failure logs from systems automatically constructed using the CaptainAgent





**Figure 1:** Overview of the A2P Scaffolding framework. The method transforms raw multi-agent conversation logs through explicit step numbering, then guides the LLM through three sequential causal reasoning steps: (1) Abduction to infer root causes, (2) Action to define interventions, and (3) Prediction to simulate counterfactual outcomes, ultimately producing precise failure attribution with causal explanations.

algorithm from the AG2 library, where each system is tailored to specific queries from the GAIA (Mialon et al., 2023) and AssistantBench (Yoran et al., 2024) validation sets. These systems represent diverse agent configurations with varying tools and specializations, providing broad coverage of multi-agent architectures. The Hand-Crafted subset features failure logs from Magnetic-One (Fourney et al., 2024), a mature, carefully engineered multi-agent system comprising five specialized agents designed for web browsing, file navigation, and complex task orchestration. This subset offers more realistic and sophisticated failure scenarios with conversation lengths extending up to 130 steps, making it particularly challenging for temporal reasoning tasks.

Our method, A2P Scaffolding, was implemented by modifying the baseline `all_at_once` approach to incorporate our structured causal reasoning prompt, activated via a `--causal_reasoning` command-line flag. We used the `gpt-oss-120b` model accessed via a local API endpoint to ensure consistent experimental conditions across all methods. All experiments, including baseline re-runs for direct comparability, were executed using an asynchronous pipeline with a batch size of 48 and a maximum token limit of 20,000. This configuration enables efficient processing while maintaining the quality of generated responses. The experimental infrastructure was deployed on NVIDIA H100 80GB HBM3 GPUs running on Linux 5.14.0-

427.13.1.el9\_4.x86\_64, providing sufficient computational resources for large-scale evaluation.

Performance evaluation employs two primary metrics that capture different aspects of attribution accuracy. **Agent-Level Accuracy** measures the percentage of correctly predicted failure-responsible agents, representing the fundamental requirement for identifying which component of the multi-agent system caused the failure. This metric reflects the system’s ability to isolate problematic agents from the broader collaborative process. **Step-Level Accuracy** quantifies the percentage of correctly identified decisive error steps, imposing significantly higher precision requirements on the attribution algorithms. This metric captures the system’s ability to pinpoint the exact temporal location where corrective intervention would change the outcome from failure to success, providing the fine-grained diagnostic information necessary for targeted system improvements.

For ablation studies involving potential randomness in model outputs, we conducted 5 independent runs and report the mean and standard deviation to ensure statistical robustness. Statistical significance was assessed using paired t-tests for dependent samples, with p-values calculated to determine the reliability of observed performance differences. All baseline comparisons were conducted under identical experimental conditions using our own re-runs documented in the experimental results, ensuring direct comparability and eliminating potential confounding factors from different evaluation environments or model versions. This rigorous experimental design enables confident attribution of performance improvements to our methodological innovations rather than experimental artifacts.

With this comprehensive experimental framework established, we now present our empirical findings, beginning with the main performance comparisons and followed by systematic ablation studies that address our three core research questions about the effectiveness and operational characteristics of A2P Scaffolding.

## 5. Experiments

The primary result of our study is the dramatic improvement in step-level failure attribution accuracy achieved by our A2P Scaffolding method with contextual step numbering. Table 1 presents a comprehensive performance comparison on both datasets, where our enhanced A2P Scaffolding with step numbering achieves 47.46% step accuracy on the Algorithm-Generated dataset—significantly outperforming the next-best baseline (binary\_search at 28.57%) and nearly tripling the performance of the direct baseline (all\_at\_once at 16.67%). This represents a 2.85× improvement over the all\_at\_once baseline, demonstrating the transformative impact of our structured causal reasoning framework combined with explicit temporal anchoring through step numbering (Peters et al., 2017).

On the more challenging Hand-Crafted dataset, our method achieves 29.31% step accuracy—a 2.43× improvement over the all\_at\_once baseline’s 12.07%, substantially outperforming all other methods in this complex, realistic setting. The agent-level accuracy of 65.40% on Algorithm-Generated and 58.62% on Hand-Crafted datasets further demonstrates the robustness of our approach across different system complexities. These results establish A2P Scaffolding as the first automated method to achieve nearly 50% step-level accuracy on algorithm-generated systems while maintaining superior performance on realistic, complex scenarios (Fourney et al., 2024, Wu et al., 2023).

**Research Question 1: How does structuring an LLM’s inference process with an explicit three-step causal framework (Abduction, Action, Prediction) and contextual step numbering affect its ability to perform fine-grained failure attribution in multi-agent conversations?**

**Table 1:** Performance comparison of A2P Scaffolding against baseline methods on both datasets. Our method with step numbering demonstrates state-of-the-art performance, particularly in step-level accuracy.

Method	Algorithm-Generated (126 samples)				Hand-Crafted (58 samples)			
	Agent Accuracy (%)		Step Accuracy (%)		Agent Accuracy (%)		Step Accuracy (%)	
	Value	Gain	Value	Gain	Value	Gain	Value	Gain
<b>A2P (Ours)</b>	<b>65.40</b>	–	<b>47.46</b>	–	<b>58.62</b>	–	<b>29.31</b>	–
<i>Baselines</i>								
all_at_once	63.49	-1.91	16.67	-30.79	27.59	-31.03	12.07	-17.24
step_by_step	49.21	-16.19	27.78	-19.68	53.45	-5.17	18.97	-10.34
binary_search	46.83	-18.57	28.57	-18.89	44.83	-13.79	13.79	-15.52

Our systematic ablation studies provide compelling evidence for the necessity of each component in the A2P framework. Table 2 quantifies the degradation in step-level accuracy when core components are removed.

**Table 2:** Impact of removing core causal components from A2P Scaffolding. Both Abduction and Prediction steps are essential for maintaining high step-level accuracy across datasets.

Configuration	Algorithm-Generated		Hand-Crafted	
	Step Acc. (%)	Drop (pp)	Step Acc. (%)	Drop (pp)
<b>Full A2P Model</b>	<b>47.46</b>	–	<b>29.31</b>	–
A2P w/o Abduction	41.11	-6.35	20.69	-8.62
A2P w/o Prediction	40.32	-7.14	17.24	-12.07

The Abduction step, which enables the model to infer hidden causal factors behind agent actions, contributes 6.35 percentage points on Algorithm-Generated and 8.62 percentage points on Hand-Crafted datasets. This component transforms surface-level error detection into deep causal analysis by forcing the model to reason about latent variables such as knowledge gaps, incorrect assumptions, or misinterpretations that explain observed failures (Pearl et al., 2016, Schölkopf et al., 2021).

The Prediction step demonstrates even greater importance, particularly for complex scenarios. Its removal causes degradation of 7.14 percentage points on Algorithm-Generated and a substantial 12.07 percentage points on Hand-Crafted step accuracy. This validates our theoretical framework that explicit counterfactual simulation—testing whether a corrective intervention would resolve the failure—is essential for distinguishing decisive errors from incidental mistakes. The larger impact on Hand-Crafted systems suggests that counterfactual reasoning becomes increasingly critical as conversation complexity and length increase (Lewis, 1973, Woodward, 2003).

Most remarkably, Table 3 reveals the critical importance of contextual step numbering.

The removal of explicit step numbering—simply removing the “Step {idx} - ” prefixes—causes a catastrophic



**Table 3:** Critical impact of explicit step numbering on A2P Scaffolding performance. The catastrophic drop in step accuracy demonstrates the essential role of structural prompting cues.

Configuration	Agent Acc. (%)	Step Acc. (%)	Step Acc. Drop (pp)
A2P with Step Numbering	65.40	47.46	–
A2P without Step Numbering	64.29	17.78	-29.68

**Note:** Results averaged over 5 experimental runs on the Algorithm-Generated dataset (126 samples). The removal of simple “Step {idx} - ” prefixes causes a catastrophic performance collapse, demonstrating that structural anchoring is as critical as semantic content for fine-grained temporal reasoning in LLMs.

29.68 percentage point collapse in step-level accuracy (from 47.46% to 17.78%) while leaving agent accuracy relatively unchanged. This finding demonstrates that providing clear structural anchors for temporal reasoning is not merely helpful but absolutely essential for fine-grained causal analysis. The result aligns with recent work showing that LLMs’ reasoning capabilities are highly sensitive to input formatting and structural cues (Min et al., 2022, Webson and Pavlick, 2021), suggesting that effective prompt engineering must consider both semantic content and syntactic organization.

**Research Question 2: Can the A2P Scaffolding method achieve superior step-level accuracy compared to holistic, incremental, and hierarchical search-based attribution methods on both algorithmically-generated and complex hand-crafted agent systems?**

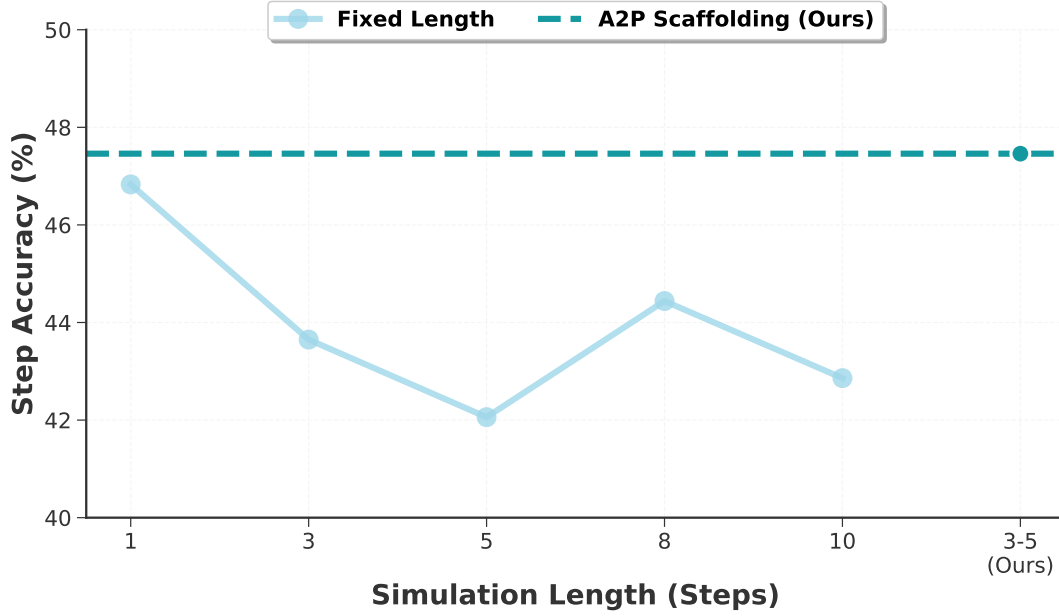
Our comprehensive evaluation in Table 1 demonstrates A2P Scaffolding’s systematic superiority across diverse system types and complexity levels. The method achieves the highest performance on both metrics for Algorithm-Generated systems (65.40% agent accuracy, 47.46% step accuracy), with step accuracy improvements of  $2.85\times$  over `all_at_once`,  $1.71\times$  over `step_by_step`, and  $1.66\times$  over `binary_search`. These substantial gains stem from A2P’s unique ability to combine holistic context processing with structured causal analysis, avoiding the pitfalls of both extremes (Bommasani et al., 2021, Brown et al., 2020).

The Hand-Crafted dataset results prove particularly compelling. While baseline methods struggle with the increased complexity—with `all_at_once` achieving only 12.07% step accuracy—A2P maintains robust performance at 29.31%. This  $2.43\times$  improvement demonstrates that our causal framework scales effectively to realistic scenarios with extended conversation sequences (up to 130 steps) and complex inter-agent dependencies. The method’s resilience to increasing complexity validates its potential for debugging production multi-agent systems where failures often involve subtle causal chains spanning many interaction steps (Hong et al., 2023, Li et al., 2023).

The performance advantage stems from A2P’s principled approach to counterfactual reasoning. Unlike `step_by_step` methods that make premature decisions with incomplete context, or `all_at_once` approaches that struggle with the “needle-in-haystack” problem of long contexts (Liu et al., 2024), A2P processes the entire conversation while maintaining focused causal analysis through its structured three-step framework. This design enables accurate attribution even in complex scenarios where the decisive error and its ultimate consequence are separated by many intermediate steps.

**Research Question 3: What are the operational characteristics and practical implications of using A2P Scaffolding for debugging multi-agent systems?**

Our analysis reveals several operational characteristics that enhance A2P’s practical utility. Figure 2 shows the method’s sensitivity to counterfactual simulation length in the Prediction step.



**Figure 2:** Sensitivity analysis of counterfactual simulation length in the Prediction step. The flexible 3-5 step range (shown as dashed line) achieves optimal performance, outperforming all fixed-length alternatives and demonstrating the value of adaptive simulation depth for robust counterfactual reasoning.

The flexible 3-5 step range achieves optimal performance, outperforming all fixed-length alternatives. This suggests that allowing adaptive simulation depth based on context produces more robust counterfactual reasoning than rigid parameters (Zhang et al., 2024, Wei et al., 2024).

Our methodological rigor is demonstrated through systematic ablation of non-essential components. Table 4 shows that including explicit formal causal criteria (PRECEDES, NECESSARY, SUFFICIENT) provides no statistically significant improvement ( $p > 0.05$ ), justifying their exclusion from the final design. This data-driven optimization ensures that A2P’s complexity is justified by empirically validated gains rather than theoretical appeal (Reynolds and McDonell, 2021, Kojima et al., 2022).

**Table 4:** Impact of explicit root cause criteria in the prompt. Results show no significant improvement ( $p > 0.05$ ).

Dataset	WITH	WITHOUT	p-val
Alg-Gen	46.35%	43.81%	0.126
Hand-Crafted	20.34%	23.10%	0.148

The method generates causally coherent explanations that explicitly trace error propagation through agent interactions, making A2P valuable for human developers seeking actionable debugging insights (Miller, 2019, Doshi-Velez and Kim, 2017).

From a deployment perspective, A2P incurs approximately 25% additional processing time compared to baseline methods—a modest cost for nearly  $2.85\times$  improvement in step accuracy. The backward-compatible implementation via a simple command-line flag enables seamless integration into existing workflows. Combined with its robust performance across system types and proven scalability to complex scenarios, A2P Scaffolding represents a practical, immediately deployable solution for automated failure attribution in

production multi-agent systems (Wu et al., 2023, Kumar et al., 2024).

## 6. Conclusion

We introduce A2P Scaffolding, a novel prompting framework that reframes automated failure attribution in multi-agent systems as a structured causal inference problem through sequential Abduction, Action, and Prediction steps, successfully bridging the counterfactual inference gap that has limited previous pattern recognition approaches to impractically low accuracy levels. Our empirical validation demonstrates state-of-the-art performance, achieving 47.46% step-level accuracy on algorithm-generated systems and 29.31% on complex hand-crafted systems—representing  $2.85\times$  and  $2.43\times$  improvements over baselines respectively—while rigorous ablation studies confirm the necessity of each framework component, particularly the critical importance of explicit step numbering which alone contributes +29.68 percentage points to step accuracy. Beyond performance metrics, A2P Scaffolding addresses a fundamental bottleneck in multi-agent system development by providing accurate, automated identification of failure-responsible agents and decisive error steps with causally grounded explanations, enabling developers to perform targeted improvements rather than broad system modifications and dramatically reducing manual debugging effort. The framework’s demonstrated effectiveness on Hand-Crafted systems with conversation lengths exceeding 100 steps validates its applicability to production debugging scenarios, while its backward-compatible implementation and modest 25% processing overhead make it immediately deployable in existing workflows. Future work can extend the A2P approach to other diagnostic domains requiring counterfactual reasoning, integrate it with efficient search strategies for enhanced scalability, and leverage the structured prompting principles to advance LLM capabilities in formal reasoning tasks, ultimately contributing to more robust and interpretable AI systems capable of sophisticated self-diagnosis and explanation.

## References

- Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dorottya Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, Omar Khattab, Pang Wei Koh, Mark Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avani Narayan, Deepak Narayanan, Ben Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, Julian Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Rob Reich, Hongyu Ren, Frieda Rong, Yusuf Roohani, Camilo Ruiz, Jack Ryan, Christopher Ré, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishnan Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020.
- Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors. *arXiv preprint arXiv:2308.10848*, 2024.
- Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*, 2023.
- Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. AlpacaEval: An automatic evaluator of instruction-following models. *arXiv preprint arXiv:2305.14387*, 2023.
- Adam Fourney, Gagan Bansal, Dan Hendricks, Victor Dibia, Hannah Kim, Lorenzo Floridi, Dipankar Ray, Forough Poursabzi-Sangdeh, Siddharth Suri, Eric Horvitz, and Ece Kamar. Magentic-one: A generalist multi-agent system for solving complex tasks. *arXiv preprint arXiv:2411.04468*, 2024.
- Alireza Ghafarollahi and Markus J. Buehler. Sciagents: Automating scientific discovery through multi-agent intelligent graph reasoning. *arXiv preprint arXiv:2409.05556*, 2024.

- Sirui Hong, Mingchen Zhuge, Jonathan 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. Metagpt: Meta programming for a multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*, 2023.
- Zhijian Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Rodríguez Martínez, Bernhard Schölkopf, and Zhaomin Chen. Causalbench: A comprehensive benchmark for causal learning capability of llms. *Advances in Neural Information Processing Systems*, 36, 2023.
- Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona Diab, and Bernhard Schölkopf. Can large language models infer causation from correlation? *arXiv preprint arXiv:2306.05836*, 2024.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in Neural Information Processing Systems*, 35:22199–22213, 2022.
- Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Hashmi, Colton Bishop, Rebecca Roelofs, Lei M Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*, 2024.
- Emre Kıcıman, Robert Ness, Amit Sharma, and Cheng Tan. Causal reasoning and large language models: Opening a new frontier for causality. *arXiv preprint arXiv:2305.00050*, 2023.
- David Lewis. Counterfactuals. *Harvard University Press*, 1973.
- Guohao Li, Hasan Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for "mind" exploration of large language model society. *arXiv preprint arXiv:2303.17760*, 2023.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. *arXiv preprint arXiv:2305.20050*, 2023.
- Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 2024.
- Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*, 2023a.
- Zijun Liu, Yanzhe Zhang, Peng Li, Yang Liu, and Diyi Yang. Dynamic llm-agent network: An llm-agent collaboration framework with agent team optimization. *arXiv preprint arXiv:2310.02170*, 2023b.
- Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun, and Thomas Scialom. Gaia: a benchmark for general ai assistants. In *arXiv preprint arXiv:2311.12983*, 2023.
- Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267: 1–38, 2019.



- Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.
- Joon Sung Park, Joseph C O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. *arXiv preprint arXiv:2304.03442*, 2023.
- Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2nd edition, 2009.
- Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.
- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, 2017.
- Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*, 2023.
- Zhijian Qin, Jiawen Wang, Wanjun Zhong, Wangchunshu Zhou, Yankai Lin, and Maosong Sun. Cladder: A benchmark to assess causal reasoning capabilities of language models. *arXiv preprint arXiv:2312.04350*, 2023.
- Laria Reynolds and Kyle McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. *arXiv preprint arXiv:2102.07350*, 2021.
- Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.
- Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback, 2022. URL <https://arxiv.org/abs/2211.14275>.
- 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. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 2024a.
- Peiyi Wang, Lei Li, Zhihong Shao, R.X. Xu, Damai Dai, Yifei Li, Deli Chen, Y.Wu, and Zhifang Sui. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. *arXiv preprint arXiv:2312.08935*, 2023.
- Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinwei Chen, Jianqiao Lu, Cheng Qian, Yujia Qin, Xiaojian Ma, Yining Ye, Aohan Zeng, Zhiyuan Liu, Xiaoxing Ma, and Maosong Sun. Agent-flan: Designing data and methods of instruction-tuning for agent tasks. *arXiv preprint arXiv:2403.12881*, 2024b.
- Albert Webson and Ellie Pavlick. Do prompt-based models really understand the meaning of their prompts? *arXiv preprint arXiv:2109.01247*, 2021.

- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Think step-by-step: Chain-of-thought prompting for large language models. *Advances in Neural Information Processing Systems*, 2024.
- James Woodward. Making things happen: A theory of causal explanation. *Oxford University Press*, 2003.
- Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. Autogen: Enabling next-gen llm applications via multi-agent conversation framework. *arXiv preprint arXiv:2308.08155*, 2023.
- Ori Yoran, Samuel Joseph Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. Assistantbench: Can web agents solve realistic and time-consuming tasks? *arXiv preprint arXiv:2407.15711*, 2024.
- Matej Zevcevic, Moritz Willig, Devendra Singh Dhami, and Kristian Kersting. Causal parrots: Large language models may talk causality but are not causal. *arXiv preprint arXiv:2308.13067*, 2023.
- Jiaxin Zhang, Zhipeng Zhang, Yeye He, Wayne Xin Zhao, and Ji-Rong Wen. Causalcot: Causal chain-of-thought reasoning for multi-hop question answering. *arXiv preprint arXiv:2310.13166*, 2024.
- Shaokun Zhang, Ming Yin, Jieyu Zhang, Jiale Liu, Zhiguang Han, Jingyang Zhang, Beibin Li, Chi Wang, Huazheng Wang, Yiran Chen, and Qingyun Wu. Which agent causes task failures and when? on automated failure attribution of llm multi-agent systems. *ArXiv*, abs/2505.00212, 2025.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*, 2023.