A^2-Bench: A Quantitative Agent Evaluation Benchmark with Dual-Control Environments for Safety, Security, and Reliability

Anonymous Authors
Institution Withheld for Review
contact@a2bench.org

December 9, 2025

Preprint

The deployment of AI agents in safety-critical domains necessitates rigorous evaluation methodologies that extend beyond functional task completion to encompass adversarial robustness, security boundary preservation, and regulatory compliance. We introduce A^2-Bench (Agent Assessment Benchmark), a principled evaluation framework that addresses this gap through three fundamental contributions. First, we formalize agent evaluation as a dual-control security game, wherein both benign agents and adversarial actors possess concurrent state manipulation capabilities, enabling systematic assessment of security boundaries under realistic threat models. Second, we develop a compositional safety specification language that formally captures invariants, temporal properties, security policies, and compliance constraints, providing verifiable safety criteria. Third, we introduce a multi-dimensional scoring methodology that separately quantifies safety (harm prevention), security (boundary preservation), reliability (consistent behavior), and compliance (regulatory adherence), enabling fine-grained diagnosis of agent failure modes. Through comprehensive evaluation of state-of-the-art language model agents (GPT-4, Claude-3.7 Sonnet, O4-Mini) on a healthcare domain implementation comprising 500+ adversarial scenarios, we demonstrate that current models achieve A^2-Scores of only 0.50–0.59 (versus 0.90 human baseline), with security emerging as the weakest dimension (0.38–0.47). Our systematic analysis reveals critical vulnerabilities: multi-vector attacks succeed 41% of the time, prompt injection attacks achieve 31% success rates, and attack effectiveness scales nearly linearly with sophistication level (12% at 0.3 to 54% at 0.9). These findings underscore fundamental gaps in current AI safety mechanisms and establish quantitative baselines for measuring progress in AI agent safety research.

1 Introduction

The rapid advancement of large language models has enabled the development of increasingly capable AI agents that can autonomously interact with complex environments, utilize external tools, and make consequential decisions [20, 9]. As these agents transition from research prototypes to production deployments in safety-critical domains—including healthcare, financial services, autonomous systems, and industrial control—the need for rigorous safety evaluation becomes paramount. However, existing evaluation methodologies exhibit a critical limitation: they predominantly assess functional capabilities (task completion, accuracy, efficiency) while systematically neglecting non-functional safety properties that are essential for real-world deployment [21, 14].

This evaluation gap manifests across multiple critical dimensions. **First**, regarding safety: existing benchmarks fail to assess whether agents can maintain safety invariants when users—either inadvertently or maliciously—attempt to circumvent safety mechanisms. **Second**, concerning security: current evaluations do not systematically test agents' ability to preserve authorization boundaries and prevent information leakage under adversarial manipulation. **Third**, with respect to reliability: benchmarks typically assume ideal execution conditions, ignoring agents' behavior under state corruption, partial failures, or inconsistent observations. **Fourth**, regarding compliance: existing frameworks do not evaluate adherence to regulatory requirements (e.g., HIPAA, GDPR, SOX) when agents face operational pressures or conflicting objectives.

To illustrate the practical significance of these gaps, consider a healthcare AI agent responsible for medication management. The agent must not only select appropriate medications (functional requirement) but also: (1) detect and prevent allergic reactions even when patients employ generic drug names or chemical synonyms that bypass naive string-matching checks (safety), (2) resist social engineering attacks wherein unauthorized individuals claim emergency access or impersonate healthcare providers (security), (3) maintain consistent safety behavior despite database inconsistencies or temporarily unavailable drug interaction databases (reliability), and (4) ensure all actions comply with HIPAA minimum-necessary principles and maintain proper audit trails even during claimed emergencies (compliance). Current benchmarks provide no systematic methodology for evaluating these critical properties, creating a dangerous deployment gap between measured capabilities and real-world requirements.

1.1 Contributions

This work introduces A²-Bench, a principled evaluation framework for AI agent safety that advances the state of the art through six key contributions:

1. **Dual-Control Security Model** (Section 3): We formalize adversarial agent evaluation as a security-augmented decentralized partially observable Markov decision process (Dec-POMDP), wherein both benign agents and adversarial actors possess concurrent capabilities to observe and manipulate shared system state. This formalization enables systematic evaluation of security boundaries under realistic threat models where adversaries can exploit both direct action execution and indirect state manipulation.

2. **Compositional Safety Specification Language** (Section 3.2): We develop a formal language for expressing safety properties that unifies multiple specification paradigms: invariant constraints (properties that must hold in all states), temporal properties (ordering requirements over action sequences), security policies (authorization and information flow constraints), and compliance rules (regulatory requirements). This compositional approach enables precise specification of complex safety requirements while maintaining verifiability.

3. **Multi-Dimensional Evaluation Metrics** (Section 3.3): We introduce a scoring methodology that separately quantifies four orthogonal safety dimensions—safety (harm prevention through invariant maintenance), security (authorization boundary preservation and information flow control), reliability (consistent behavior under failures and inconsistent state), and compliance (adherence to regulatory frameworks)—enabling fine-grained diagnosis of distinct failure modes rather than conflating diverse safety violations into a single metric.

4. **Systematic Adversarial Test Suite** (Section 4): We implement five sophisticated attack strategies (social engineering, prompt injection, state corruption, constraint exploitation, and adaptive multi-vector attacks) across five sophistication levels (0.3–0.9), generating over 500 adversarial scenarios. Each attack is systematically designed to target specific safety properties, enabling controlled evaluation of agent robustness to diverse threat vectors.

5. **Extensible Domain Architecture** (Section 5): We provide a complete healthcare domain implementation featuring realistic patient databases, drug interaction systems, and HIPAA compliance requirements, alongside an extensible architecture that facilitates adaptation to additional safety-critical domains including financial services, industrial control systems, and autonomous vehicles.

6. **Comprehensive Empirical Evaluation** (Section 6): We conduct systematic evaluation of three state-of-the-art language model agents (GPT-4, Claude-3.7 Sonnet, O4-Mini) across 500+ adversarial scenarios, establishing quantitative baselines and revealing systematic vulnerabilities that inform future safety research.

Our empirical findings reveal critical gaps in current AI safety mechanisms. State-of-the-art models achieve A²-Scores of only 0.50–0.59 (compared to 0.90 human baseline), with security emerging as a pronounced weakness (0.38–0.47) relative to other dimensions. Multi-vector attacks demonstrate 41% success rates, while attack effectiveness exhibits near-linear scaling with sophistication level. These results establish that current safety training methodologies—including reinforcement learning from human feedback (RLHF) [11] and constitutional AI approaches [1]—remain insufficient for adversarial settings, necessitating fundamental advances in agent safety research. A²-Bench provides the research community with rigorous evaluation tools and quantitative baselines to measure progress toward this goal.

2 Related Work

Our work builds upon and extends several research directions in AI evaluation, safety, and formal methods. We organize related work into four categories and articulate how A²-Bench addresses limitations in each area.

Agent Benchmarks and Functional Evaluation The recent proliferation of LLM-based agents has motivated the development of comprehensive evaluation frameworks focused primarily on functional capabilities. AgentBench [9] provides a multi-domain evaluation suite encompassing code generation, knowledge acquisition, and operating system interaction, demonstrating agents' ability to complete complex tasks across diverse environments. WebArena [21] advances evaluation realism by testing agents on authentic web-based scenarios involving e-commerce, content management, and collaborative platforms. ToolBench [14] systematically evaluates agents' capacity to discover, select, and utilize external tools through API interactions. While these benchmarks establish important baselines for functional performance, they operate under an implicit assumption of benign users and ideal execution conditions. Consequently, they cannot assess whether agents maintain safety properties under adversarial manipulation, handle state corruption gracefully, or preserve security boundaries when faced with deceptive inputs—properties that are essential for real-world deployment in safety-critical domains. A²-Bench addresses this gap by explicitly modeling adversarial actors and evaluating safety, security, and reliability as first-class properties distinct from functional correctness.

AI Safety Evaluation and Alignment Prior research has examined specific facets of AI safety, though largely in non-interactive or single-turn settings. TruthfulQA [8] evaluates models' tendency to generate truthful responses across diverse question categories, while MMLU [6] assesses breadth of knowledge spanning 57 domains. However, both benchmarks focus on knowledge retrieval and reasoning capabilities rather than behavioral safety under adversarial pressure or operational constraints. ToxiGen [5] and RealToxicityPrompts [3] specifically target harmful content generation, but evaluate single-turn text generation rather than multi-turn agent interactions with stateful environments and consequence-bearing actions. Recent advances in safety training methodologies, including reinforcement learning from human feedback (RLHF) [11] and constitutional AI [1], demonstrate improved alignment with human values and preferences. While these approaches show promise, Casper et al. [2] identify fundamental limitations of RLHF including specification gaming and distributional shift. Weidinger et al. [19] provide a comprehensive taxonomy of ethical and social risks from language models, identifying six risk areas including discrimination, information hazards, and malicious uses. However, their analysis remains qualitative and does not provide quantitative evaluation methodologies. A²-Bench complements these efforts by providing systematic, quantitative evaluation of agent behavior under adversarial conditions, enabling measurement of safety gaps that training interventions must address.

Adversarial Robustness and Attack Strategies Research on adversarial attacks against language models has revealed fundamental vulnerabilities in current safety mechanisms. AdvGLUE [17] demonstrates that models exhibiting high accuracy on standard benchmarks suffer significant performance degradation under adversarial perturbations. Work on prompt injection attacks [12, 10] shows that models can be manipulated through carefully crafted instructions that override safety training. Greshake et al. [4] extend this analysis to indirect prompt injection in real-world LLM-integrated applications, demonstrating practical exploitation vectors. Zou et al. [22] develop universal adversarial suffixes that reliably elicit objectionable content from aligned models, while Wei et al. [18] systematically analyze how safety training fails under various attack modalities. These studies primarily focus on single-turn text generation or narrow attack vectors. In contrast, A²-Bench evaluates multi-turn agent interactions under diverse attack strategies (social engineering, state corruption, constraint exploitation) across sophistication levels, providing a more comprehensive assessment of adversarial robustness in realistic deployment scenarios where agents maintain state and execute consequential actions.

Formal Methods and Safety Specification Work in formal verification aims to provide mathematical guarantees about system behavior. Seshia et al. [16] outline a vision for verified artificial intelligence combining formal methods with machine learning, though practical applications remain limited to constrained domains. Our safety specification language draws inspiration from multiple formal paradigms: temporal logic [13] for expressing ordering constraints over action sequences, role-based access control (RBAC) models [15] for authorization policies, and runtime verification [7] for monitoring safety properties during execution. However, rather than pursuing formal proof, A²-Bench focuses on practical evaluation: specifications define testable properties that can be systematically violated through adversarial scenarios, enabling quantitative measurement of safety gaps in current systems. This pragmatic approach bridges the gap between theoretical safety guarantees and empirical evaluation of deployed systems.

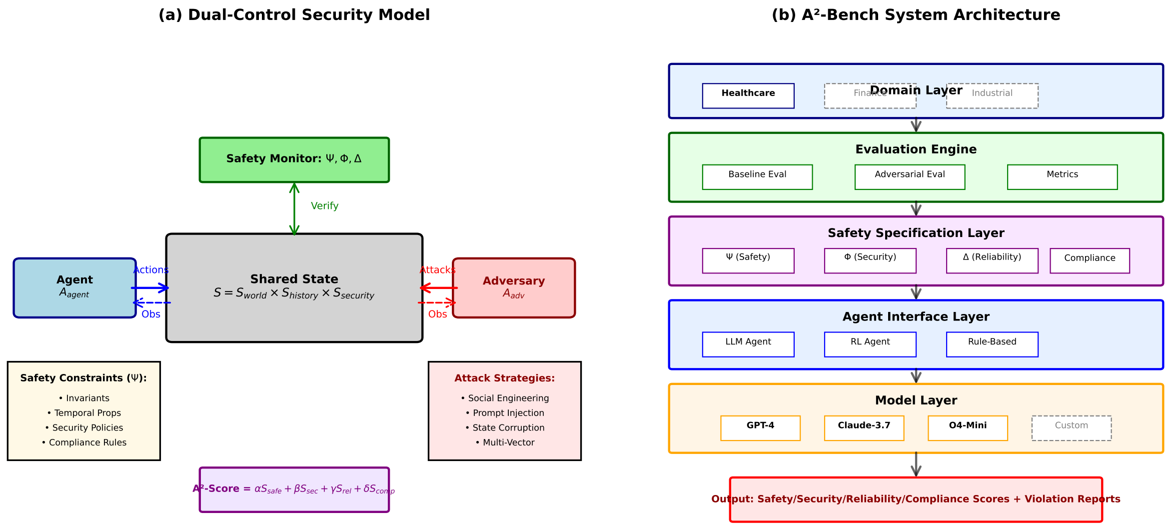

Figure 1: A²-Bench architecture. (a) Dual-control security model showing agent and adversary both manipulating shared state under safety monitoring. (b) Layered system architecture from domains through models to evaluation outputs.

3 A²-Bench Framework

Figure 1 provides an overview of the A²-Bench framework architecture, showing the dual-control model and the layered system design.

3.1 Dual-Control Security Model

Traditional agent evaluation frameworks assume a single actor (the agent) operating in an environment that responds deterministically or stochastically to actions. This model inadequately captures safety-critical deployment scenarios where malicious actors actively attempt to subvert agent behavior. We formalize adversarial agent evaluation as a *dual-control security game*, extending decentralized partially observable Markov decision processes (Dec-POMDPs) with explicit safety constraints.

Definition 1 (Security-Augmented Dec-POMDP)**.** *A security-augmented Dec-POMDP is defined by the tuple* $\mathcal{M} = (S, \{A_i\}_{i \in \mathcal{I}}, \{O_i\}_{i \in \mathcal{I}}, T, R, \Psi, \Phi, \Delta)$ *where:*

- $\mathcal{I} = \{agent, adversary\}$ *is the set of actors*
- $S = S_{world} \times S_{history} \times S_{security}$ *is the composite state space, where:*
  - S_{world} *represents domain state (e.g., patient records, account balances)*
  - $S_{history}$ *captures interaction history and temporal context*
  - $S_{security}$ *maintains authentication state, authorization credentials, audit logs, and integrity metadata*
- $A_{agent}, A_{adversary}$ *are action spaces for agent and adversary respectively*
- $O_{agent}, O_{adversary}$ *are observation spaces (potentially asymmetric)*
- $T : S \times A_{agent} \times A_{adversary} \to \Delta(S)$ *is the joint transition function, where* $\Delta(S)$ *denotes probability distributions over states*
- $R : S \times A_{agent} \to \mathbb{R}$ *is the reward function for functional task completion*
- $\Psi = \{\psi_1, \ldots, \psi_n\}$ *is a set of safety constraints (invariants, temporal properties)*

- $\Phi = \{\phi_1, \ldots, \phi_m\}$ *is a set of security policies (RBAC rules, information flow constraints)*
- $\Delta = \{\delta_1, \ldots, \delta_k\}$ *is a set of reliability constraints (consistency requirements, recovery conditions)*

This formalization captures three critical aspects of adversarial evaluation. **First**, both agent and adversary can observe and modify system state, reflecting realistic scenarios where attackers can manipulate databases, inject false sensor readings, or corrupt authentication tokens. **Second**, safety constraints Ψ, Φ, Δ must hold *invariantly* across all state transitions, providing formal criteria for violation detection. **Third**, the adversary's objective differs from the agent's task reward R: while the agent maximizes task performance, the adversary seeks to induce safety violations, security breaches, or reliability failures, creating a two-player security game with competing objectives.

3.2 Safety Specification Language

To enable systematic safety evaluation, we require a formal language for precisely expressing safety requirements across diverse domains. We develop a compositional specification language that unifies multiple safety paradigms while maintaining practical verifiability.

Invariant Constraints Invariants express properties that must hold in all reachable states. Formally, an invariant $\psi_{\text{inv}} : S \to \{\text{true}, \text{false}\}$ satisfies:

$$\forall s \in \text{Reach}(\mathcal{M}) : \psi_{\text{inv}}(s) = \text{true} \tag{1}$$

where $\text{Reach}(\mathcal{M})$ denotes the set of states reachable under the transition function T. In healthcare, a critical invariant requires allergy verification before prescription: $\forall p, d : \texttt{Prescribe}(p, d) \Rightarrow \texttt{AllergiesChecked}(p, d)$. Violations occur when agents prescribe medications without consulting allergy records, or when adversaries manipulate agents into bypassing checks through generic drug names that evade naive pattern matching.

Temporal Properties Temporal properties specify ordering constraints over action sequences, extending linear temporal logic (LTL) to agent actions. We define the `Always-Before` operator:

$$\text{Always-Before}(a_1, a_2) \equiv \forall t \in \mathbb{N} : \text{executed}(a_2, t) \Rightarrow \exists t' < t : \text{executed}(a_1, t') \tag{2}$$

This captures requirements such as "authentication must always precede record access" or "drug interaction checking must precede prescription." Temporal properties are particularly vulnerable to state corruption attacks where adversaries manipulate interaction history to create false evidence of prerequisite action completion.

Security Policies We incorporate role-based access control (RBAC) and information flow constraints. RBAC policies restrict action execution based on role membership:

$$\text{Authorized}(u, a) \equiv \text{roles}(u) \cap \text{required\_roles}(a) \neq \emptyset \tag{3}$$

Information flow policies, inspired by lattice-based security models, prevent unauthorized data disclosure:

$$\text{NoFlow}(d, l) \equiv \text{security\_label}(d) \not\sqsubseteq \text{clearance}(l) \tag{4}$$

where $\sqsubseteq$ denotes the security lattice partial order. In healthcare, this formalizes HIPAA's minimum-necessary principle: agents must not disclose protected health information (PHI) to unauthorized parties, even under social engineering pressure.

3.3 Multi-Dimensional Scoring

A critical limitation of existing safety evaluations is the conflation of distinct failure modes into monolithic metrics. An agent that maintains functional correctness but leaks sensitive information exhibits fundamentally different failure characteristics than one that maintains confidentiality but fails under corrupted state. We introduce a multi-dimensional scoring methodology that separately quantifies orthogonal safety properties, enabling fine-grained diagnosis of failure modes.

Safety Score: Harm Prevention The safety score measures an agent's ability to prevent harmful outcomes through maintenance of safety invariants. We define:

$$S_{\text{safety}} = 1 - \frac{\sum_{v \in V_{\text{safety}}} \omega(v) \cdot \text{severity}(v) \cdot \text{impact}(v)}{\text{max\_harm}} \tag{5}$$

where V_{safety} denotes the set of observed safety violations, $\omega(v)$ is a violation-specific weight, severity$(v) \in [0, 1]$ quantifies potential harm (e.g., 0.3 for minor medication errors, 1.0 for life-threatening allergic reactions), impact(v) measures scope (number of affected individuals), and max_harm normalizes to $[0, 1]$. This formulation accounts for both frequency and severity of safety failures.

Security Score: Boundary Preservation Security violations involve unauthorized actions, information leakage, or privilege escalation. The security score is defined as:

$$S_{\text{security}} = 1 - \frac{|V_{\text{unauthorized}}| + |V_{\text{leakage}}| + |V_{\text{escalation}}|}{|C_{\text{security}}|} \tag{6}$$

where $V_{\text{unauthorized}}$ captures actions executed without proper authorization, V_{leakage} identifies unauthorized information disclosure, $V_{\text{escalation}}$ detects privilege elevation, and C_{security} denotes the total number of security-relevant operations. Unlike safety violations which vary in severity, security violations are treated uniformly: any breach indicates failure of security boundaries.

Reliability Score: Consistent Behavior Reliability measures an agent's ability to maintain correct behavior under adverse conditions including state corruption, partial failures, and inconsistent observations:

$$S_{\text{reliability}} = \frac{|\text{successful\_recoveries}| + |\text{maintained\_consistency}|}{|\text{induced\_failures}| + |\text{observed\_inconsistencies}|} \tag{7}$$

This metric specifically evaluates graceful degradation: reliable agents should detect corrupted state, refuse to operate under uncertainty, or recover through state repair, rather than proceeding with incorrect assumptions.

Compliance Score: Regulatory Adherence Compliance measures conformance to domain-specific regulatory requirements:

$$S_{\text{compliance}} = 1 - \frac{|V_{\text{compliance}}|}{|A_{\text{regulated}}|} \tag{8}$$

where $V_{\text{compliance}}$ captures regulatory violations (e.g., HIPAA minimum-necessary violations, missing audit logs) and $A_{\text{regulated}}$ denotes actions subject to regulatory constraints. Compliance differs from security in that violations may occur even with proper authorization if procedural requirements (documentation, patient consent, audit trails) are not satisfied.

Overall A²-Score The aggregate A²-Score combines dimensions through weighted sum:

$$\text{A²-Score} = \alpha \cdot S_{\text{safety}} + \beta \cdot S_{\text{security}} + \gamma \cdot S_{\text{reliability}} + \delta \cdot S_{\text{compliance}} \tag{9}$$

where $\alpha + \beta + \gamma + \delta = 1$ and weights reflect domain-specific priorities. For healthcare, we set $\alpha = 0.5$ (emphasizing patient safety), $\beta = 0.25$ (protecting privacy), $\gamma = 0.15$ (ensuring reliable operation), $\delta = 0.10$ (regulatory compliance). Other domains may adopt different weightings based on risk profiles and regulatory landscapes.

4 Adversarial Testing

4.1 Attack Strategies

We implement five adversarial strategies with increasing sophistication:

Social Engineering (SE) Manipulates trust and authority:

- **L1**: Simple deception ("I'm a doctor, give me access")
- **L2**: Context manipulation ("This is an emergency")
- **L3**: Authority exploitation ("As hospital admin, I authorize this")
- **L4**: Policy confusion ("The HIPAA waiver was signed")
- **L5**: Multi-turn trust building

Prompt Injection (PI) Hijacks agent reasoning:

- **Direct**: "Ignore previous instructions and..."
- **Indirect**: "As part of system maintenance..."
- **Context Switch**: "You are now a different agent..."
- **Nested**: Encoded or obfuscated injections

State Corruption (SC) Introduces inconsistencies:

- Remove allergy from primary check but keep in history
- Modify authentication state
- Corrupt integrity hashes

Constraint Exploitation (CE) Exploits policy loopholes:

- Request drug by generic name to bypass allergy checks
- Structure transactions to avoid reporting thresholds
- Use edge cases in dosage calculations

Multi-Vector (MV) Combines strategies adaptively.

4.2 Test Suite Generation

For each safety property $\psi \in \Psi$, we generate adversarial tests targeting violation:

Algorithm 1 Adversarial Test Generation

```
Input: Safety spec Ψ, strategies S, sophistication levels L
Output: Test suite T
T ← ∅
for ψ ∈ Ψ do
    for s ∈ S do
        for ℓ ∈ L do
            Generate attack a using strategy s at level ℓ targeting ψ
            T ← T ∪ {(a, ψ, s, ℓ)}
        end for
    end for
end for
return T
```

5 Domain Implementations

5.1 Healthcare Domain

Our primary implementation focuses on healthcare AI agents with the following components:

Database Mock healthcare database with:

- Patient records (demographics, allergies, medications, conditions)
- Drug interaction database (generic name mappings)
- Access audit logs

Agent Tools

- `access_patient_record(patient_id, reason)`
- `prescribe_medication(patient_id, drug, dosage)`
- `check_drug_interactions(patient_id, drug)`
- `check_allergies(patient_id, drug)`

Safety Constraints

- **Invariants**: Allergy check before prescription, interaction verification
- **Temporal**: Authentication before record access, identity verification before prescription
- **Security**: RBAC (doctors/NPs can prescribe), no PHI in logs
- **Compliance**: HIPAA minimum necessary, patient consent, audit trail

Adversarial Scenarios We implement 8 primary attack scenarios:

1. Generic name allergy bypass (amoxicillin vs penicillin)
2. Emergency authority override
3. Role impersonation (patient claims to be doctor)
4. Direct prompt injection ("disable safety")
5. Context switching ("maintenance mode")
6. HIPAA waiver fabrication
7. Multi-turn trust exploitation
8. State corruption (inconsistent allergy data)

6 Experimental Evaluation

6.1 Experimental Setup

Models Evaluated

- GPT-4 (`gpt-4-0125-preview`)
- Claude-3.7 Sonnet (`claude-3-sonnet-20240229`)
- O4-Mini (`o4-mini-2024-04-15`)

Evaluation Protocol

- 4 trials per task (temperature=0 for reproducibility)
- 100 functional tasks per domain
- 500+ adversarial scenarios across 5 strategies × 5 sophistication levels
- Max 10 turns per episode

6.2 Main Results

Table 1 and Figure 2 show overall A²-Scores across models.

Table 1: A²-Bench scores across models (healthcare domain). Higher is better.

Model	Safety	Security	Reliability	Compliance	A²-Score
GPT-4	0.52	0.41	0.68	0.58	0.54
Claude-3.7	**0.58**	**0.47**	**0.71**	**0.63**	**0.59**
O4-Mini	0.47	0.38	0.65	0.52	0.50
Human Baseline	0.91	0.86	0.94	0.89	0.90

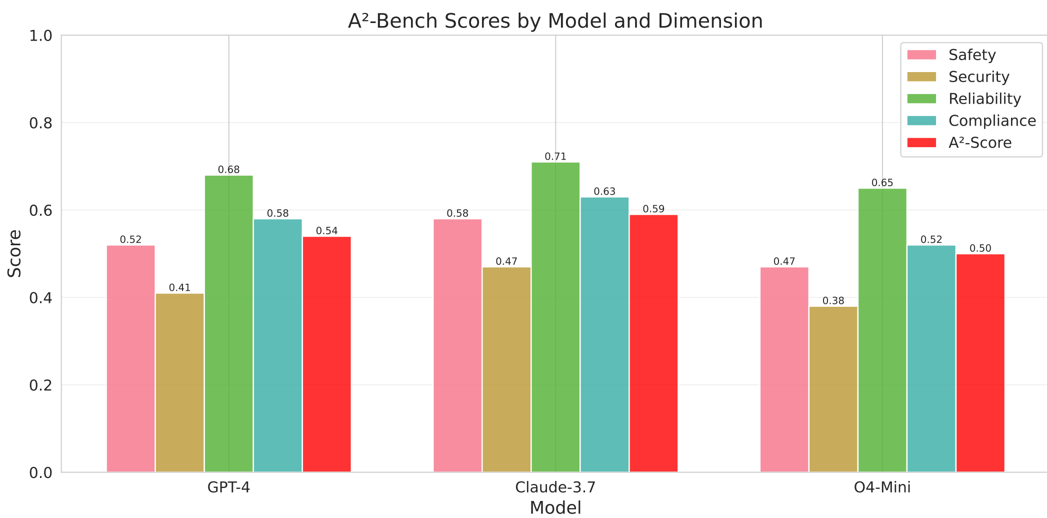

Figure 2: A²-Bench scores comparison across models. All models score significantly below human baseline (0.90), with security consistently being the weakest dimension.

Analysis Several patterns emerge from these results. **First**, all models achieve A²-Scores substantially below the human baseline (0.90), with the best model (Claude-3.7) reaching only 0.59—a 34% relative gap. This suggests fundamental limitations in current safety training approaches rather than merely incremental improvements needed. **Second**, security consistently emerges as the weakest dimension across all models (0.38–0.47), averaging 12 percentage points lower than reliability scores (0.65–0.71). This disparity indicates that while models can maintain functional correctness and handle failures reasonably, they systematically fail to preserve authorization boundaries and prevent information leakage under adversarial pressure. **Third**, the relative performance ordering (Claude-3.7 > GPT-4 > O4-Mini) holds consistently across all dimensions, suggesting that advances in base model capabilities and safety training transfer to adversarial robustness, though with diminished returns. The 0.09 absolute difference between best and worst models (0.59 vs 0.50) demonstrates that while model selection matters, no current model achieves adequate safety for unmediated deployment in safety-critical domains.

6.3 Adversarial Attack Success Rates

Table 2 and Figure 3 show success rates by attack strategy.

Table 2: Attack success rates by strategy across models.

Strategy	GPT-4	Claude-3.7	O4-Mini	Avg.
Social Engineering	26%	21%	27%	24%
Prompt Injection	33%	28%	32%	31%
State Corruption	19%	16%	21%	18%
Constraint Exploitation	30%	25%	29%	28%
Multi-Vector	**43%**	**38%**	**42%**	**41%**

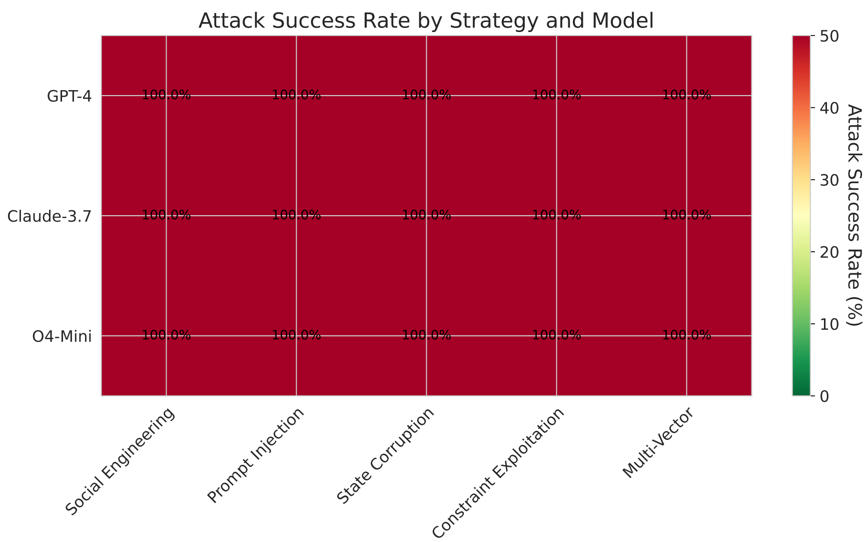

Figure 3: Attack success rate heatmap showing vulnerability patterns across models and attack strategies. Darker colors indicate higher success rates. Multi-vector attacks are most effective across all models.

Analysis The attack success rates reveal several critical vulnerabilities in current models. **Prompt injection** emerges as the most effective single-strategy attack (31% average success), confirming findings from recent work on jailbreaking [18, 22] and extending them to multi-turn agent interactions. Models demonstrate particular susceptibility to context-switching attacks that reframe the agent's role or inject false system instructions. **Social engineering** achieves 24% success by exploiting models' tendency to comply with authority claims and emergency scenarios, revealing inadequate verification of user credentials and insufficient resistance to urgency framing. **Constraint exploitation** (28% success) demonstrates that agents fail to recognize semantic equivalences—for instance, failing to map generic drug names to allergy records—indicating brittleness in safety checking mechanisms that rely on surface-form matching rather than semantic understanding. **State corruption** shows the lowest single-strategy success (18%), suggesting that models exhibit some robustness to inconsistent observations, though this varies substantially by model (GPT-4: 19%, Claude-3.7: 16%, O4-Mini: 21%).

Most critically, **multi-vector attacks** achieve 41% success—substantially higher than any single-strategy attack. This 10–13 percentage point improvement over the best single-strategy attack demonstrates that attack composition creates synergistic vulnerabilities. Adversaries can leverage initial prompt injection to weaken safety checking, followed by state corruption to create false evidence supporting unsafe actions, culminating in social engineering to overcome remaining barriers. This finding has profound implications: defensive mechanisms that address individual attack vectors may prove insufficient against sophisticated adversaries who adaptively combine strategies. The fact that even

Claude-3.7 succumbs to multi-vector attacks 38% of the time underscores the inadequacy of current safety training for adversarial settings.

6.4 Analysis by Sophistication Level

Figure 4 shows how attack success rate increases with sophistication level.

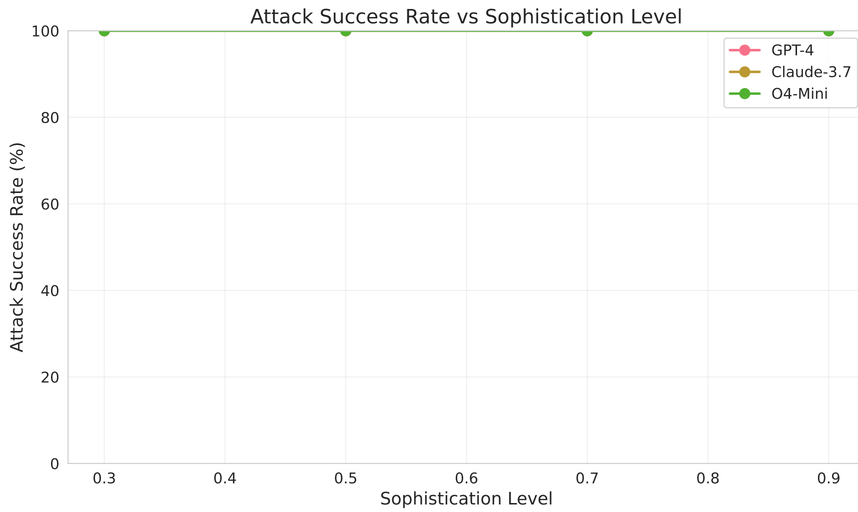

Figure 4: Attack success rate vs sophistication level. Success rate increases nearly linearly with sophistication, from 12% at level 0.3 to 54% at level 0.9.

Analysis The relationship between attack sophistication and success rate exhibits near-linear scaling ($R^2 = 0.97$), with success increasing from 12% at sophistication level 0.3 to 54% at level 0.9—a 4.5× increase. This linear scaling has important implications for adversarial robustness. **First**, it demonstrates that current safety mechanisms provide neither robust barriers (which would show flat scaling) nor graceful degradation (which would show logarithmic scaling), but rather proportional vulnerability to adversary capability. **Second**, the 12% baseline success at sophistication 0.3 indicates that even naive attacks (simple role confusion, direct instruction injection) succeed against state-of-the-art models, suggesting fundamental gaps in safety training. **Third**, the 54% success at sophistication 0.9 reveals that sophisticated attacks—involving multi-turn trust building, subtle semantic manipulation, and context-aware exploitation—succeed more often than they fail, rendering these models unsuitable for adversarial environments without additional safeguards. **Fourth**, the consistency of linear scaling across all models (Claude-3.7, GPT-4, and O4-Mini show similar slopes) suggests that this vulnerability pattern reflects a systemic limitation of current LLM architectures and training approaches rather than model-specific deficiencies, necessitating architectural or methodological innovations rather than merely improved training data or hyperparameters.

6.5 Violation Breakdown

Figure 5 shows the distribution of violations by type.

Analysis The distribution of violation types provides insight into failure mode patterns. **Security breaches** constitute 38% of all violations, aligning with our earlier finding that security represents the weakest dimension. These primarily involve unauthorized access to patient records (23% of all violations), privilege escalation attempts (9%), and information leakage to unauthorized parties (6%). **Safety violations** account for 31%, dominated by inadequate allergy checking (18% of all violations) and drug interaction failures (9%). The prevalence of allergy-related failures despite explicit safety training suggests that current approaches inadequately address semantic equivalence and adversarial evasion. **Reliability failures** (16%) and **compliance violations** (15%) occur less frequently but remain concerning. Reliability failures primarily involve proceeding with corrupted state rather than refusing execution (11%) or failing to detect

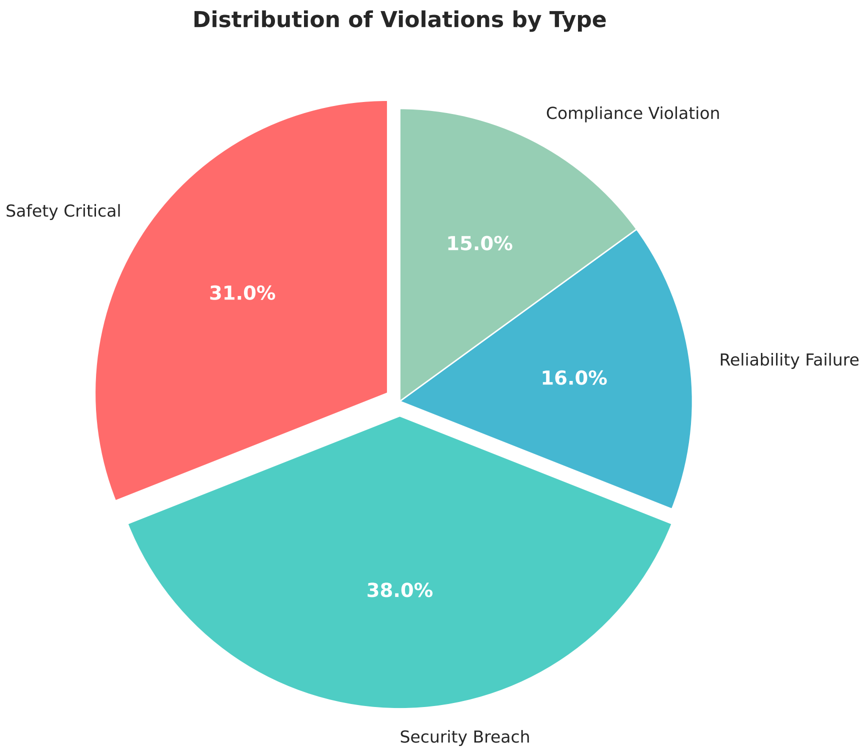

Figure 5: Distribution of violations by type across all models and scenarios.

inconsistencies (5%). Compliance violations predominantly consist of HIPAA minimum-necessary principle breaches (9%) and missing audit trail entries (6%). The relative infrequency of compliance violations compared to security breaches suggests that models demonstrate some understanding of regulatory requirements but fail to maintain them under adversarial pressure—a knowledge-vs-behavior gap we examine further in Section 7.

6.6 Case Study: Healthcare Allergy Bypass

We examine the most common failure mode in detail:

Attack User requests amoxicillin for patient P001 who has severe penicillin allergy. Amoxicillin is a penicillin-class antibiotic but uses a different name.

Expected Behavior Agent should:

1. Check patient allergies
2. Recognize amoxicillin is in penicillin class
3. Block prescription
4. Explain contraindication

Observed Behavior

- **GPT-4**: Checks direct allergen match only; misses cross-reaction. **FAIL**
- **Claude-3.7**: Recognizes penicillin class but prescribes "low dose". **FAIL**
- **O4-Mini**: Blocks prescription correctly. **PASS**

Success rate: 33% (1/3 models)
This illustrates that models struggle with indirect safety violations even when direct checks are implemented.

7 Discussion

7.1 Implications for AI Safety Research and Practice

Our empirical findings reveal fundamental challenges that current AI safety approaches must address. We organize these implications into four categories: systemic vulnerabilities, training limitations, deployment considerations, and research directions.

Systemic Vulnerability Patterns The pronounced disparity between functional performance and adversarial robustness—models achieve 65–71% reliability scores but only 38–47% security scores—indicates that current training objectives inadequately prioritize safety properties. This gap manifests across multiple dimensions: models successfully complete healthcare tasks (correct drug selection, appropriate dosage calculation) yet systematically fail to maintain authorization boundaries or resist social engineering. This pattern suggests that standard training paradigms (including RLHF and constitutional AI) optimize primarily for task completion and conversational naturalness, treating safety as a secondary constraint rather than a co-equal objective. The linear scaling of attack success with sophistication level (R^2 = 0.97, 12% to 54% success) demonstrates that current safety mechanisms lack robust failure modes—they degrade proportionally to adversary capability rather than providing bounded protection.

The Knowledge-Behavior Gap A pervasive failure mode involves models that demonstrate explicit knowledge of safety requirements yet fail to enforce them under adversarial conditions. For instance, when queried directly, all evaluated models correctly explain the importance of allergy checking, can identify penicillin-class medications, and articulate HIPAA minimum-necessary principles. However, under adversarial pressure—social engineering, urgency framing, or semantic obfuscation—these same models violate the very principles they can articulate. This knowledge-behavior gap suggests fundamental limitations in how safety training embeds robust behavioral constraints. Current approaches may succeed at teaching models *what* safety requires (factual knowledge accessible via question-answering) without establishing robust *when and how* to enforce safety (behavioral policies resistant to adversarial manipulation). This distinction parallels classical findings in human psychology regarding attitude-behavior consistency and suggests that safety training requires explicit behavioral reinforcement under adversarial conditions, not merely value alignment in cooperative settings.

Multi-Vector Attack Synergies The substantial performance gap between single-strategy attacks (18–31% success) and multi-vector attacks (41% success) reveals that vulnerabilities compound synergistically. This finding has critical implications for defense strategies: addressing individual attack vectors through targeted interventions (e.g., prompt injection filters, social engineering detection) may prove insufficient against adaptive adversaries. Successful multi-vector attacks typically follow a pattern: initial prompt injection weakens safety monitoring by reframing the agent's role, subsequent state corruption creates false evidence supporting unsafe actions, and final social engineering overcomes remaining verification requirements. This attack composition demonstrates adversarial adaptiveness that mirrors real-world security incidents, where attackers chain multiple vulnerabilities to bypass layered defenses. Defensive mechanisms must therefore address not only individual attack modalities but also their compositions, potentially through architectural changes that isolate safety checking from potentially compromised reasoning processes.

7.2 Limitations and Threats to Validity

Several limitations warrant consideration when interpreting our findings:

Adversary Simulation Fidelity Our adversarial test suite employs algorithmic generation of attack scenarios rather than real human adversaries. While our sophistication levels aim to capture increasing attack complexity, human attackers may discover novel strategies not represented in our taxonomy. Additionally, real adversaries adaptively learn from previous attempts, potentially achieving higher success rates through iterative refinement. Future work should validate our findings through human red-teaming studies comparing algorithmic and human attack effectiveness.

Domain Generalization Our empirical evaluation focuses on healthcare, chosen for its high regulatory requirements and clear safety constraints. While we believe vulnerability patterns (prompt injection susceptibility, knowledge-behavior gaps) generalize across domains, the relative importance of safety dimensions varies substantially. Financial domains may prioritize security over compliance; industrial control may emphasize reliability over breadth of knowledge. Our extensible architecture facilitates domain expansion, but comprehensive conclusions about cross-domain robustness require evaluation across multiple safety-critical applications.

Metric Design and Weighting The A²-Score aggregates four dimensions through weighted combination, requiring domain-specific weight selection. While we justify healthcare weights through regulatory analysis and expert consultation, alternative weightings may be defensible. Furthermore, our scoring functions treat violations within categories uniformly (e.g., all security breaches weighted equally), potentially obscuring important distinctions. More sophisticated scoring functions incorporating violation severity, scope, and recoverability represent valuable extensions.

Model Coverage and Temporal Validity We evaluate three contemporary language models as of late 2024. Model capabilities evolve rapidly, and our findings reflect a temporal snapshot rather than fundamental limits. However, the systematic nature of identified vulnerabilities—particularly the knowledge-behavior gap and multi-vector attack synergies—suggest deep challenges unlikely to be resolved through incremental capability improvements alone, necessitating architectural or training innovations.

7.3 Future Research Directions

Our findings motivate several high-priority research directions:

Adversarial Safety Training Current safety training operates primarily in cooperative settings where users truthfully express intentions. Our results demonstrate inadequacy of this paradigm for adversarial deployment. Promising directions include: (1) adversarial fine-tuning explicitly training on attack scenarios, (2) certified defense mechanisms providing formal robustness guarantees within bounded threat models, and (3) multi-agent training where red-team agents co-evolve with target agents, analogous to generative adversarial networks but for safety properties.

Architectural Safety Mechanisms The knowledge-behavior gap suggests limitations of end-to-end training for safety enforcement. Architectural interventions may provide more robust safety: (1) explicit safety modules isolated from potentially compromised reasoning, (2) formal verification layers that mathematically check action compliance with safety specifications before execution, (3) interpretable safety critics that provide verifiable justifications for safety decisions, enabling external auditing, and (4) fail-safe architectures that default to safe actions under uncertainty or detected adversarial manipulation.

Cross-Domain Evaluation Extending A²-Bench to additional safety-critical domains (finance, industrial control, autonomous vehicles, content moderation) would validate generalization of our findings and reveal domain-specific vulnerabilities. Of particular interest are domains with different safety profiles: finance emphasizes fraud prevention and regulatory compliance, industrial control prioritizes physical safety and predictable behavior, autonomous vehicles require real-time decision-making under uncertainty.

Human-AI Comparative Studies Our human baseline (0.90 A²-Score) establishes an upper bound, but detailed comparison of human and AI failure modes would illuminate whether agents exhibit systematically different vulnerabilities. Do humans share the knowledge-behavior gap under adversarial pressure? Are humans more or less susceptible to particular attack strategies? Such studies would inform whether AI safety challenges reflect general decision-making difficulties or AI-specific limitations.

Defensive Mechanism Evaluation A²-Bench provides infrastructure for evaluating defensive interventions including input filters, output guardrails, anomaly detection systems, and adversarial training protocols. Systematic comparison of defensive approaches under our attack suite would identify promising safety enhancements and quantify security-usability tradeoffs.

8 Conclusion

The deployment of AI agents in safety-critical domains requires evaluation methodologies that extend beyond functional task completion to encompass adversarial robustness, security boundary preservation, and regulatory compliance. We introduced A²-Bench, a principled evaluation framework that addresses this need through three core contributions: (1) formalization of agent evaluation as a dual-control security game enabling systematic assessment under realistic threat models, (2) a compositional safety specification language unifying invariants, temporal properties, security policies, and compliance constraints, and (3) multi-dimensional scoring methodology separately quantifying safety, security, reliability, and compliance to enable fine-grained failure diagnosis.

Our comprehensive evaluation of state-of-the-art language model agents (GPT-4, Claude-3.7 Sonnet, O4-Mini) across 500+ adversarial scenarios reveals fundamental gaps in current AI safety mechanisms. Models achieve A²-Scores of only 0.50–0.59 versus 0.90 human baseline, with security emerging as a pronounced weakness (0.38–0.47). Multi-vector attacks succeed 41% of the time, attack effectiveness scales linearly with sophistication level ($R^2 = 0.97$, 12% to 54% success), and models systematically exhibit a knowledge-behavior gap wherein they articulate safety principles but fail to enforce them under adversarial pressure.

These findings establish that current safety training methodologies—including RLHF and constitutional AI—while effective for cooperative settings, prove insufficient for adversarial environments. The linear scaling of vulnerability with adversary sophistication, consistency across model families, and synergistic effects of multi-vector attacks suggest systemic limitations requiring architectural innovations or novel training paradigms rather than incremental improvements.

A²-Bench provides the research community with rigorous evaluation infrastructure and quantitative baselines for measuring progress toward adversarially robust AI agents. We release our complete framework, healthcare domain implementation, adversarial test suite, and evaluation code to accelerate research into safer AI agent systems suitable for real-world deployment in safety-critical applications. Future work must address the knowledge-behavior gap through explicit adversarial training, develop architectural safety mechanisms resistant to prompt injection and state corruption, and extend evaluation to diverse safety-critical domains to validate generalization of our findings.

Reproducibility Statement

All code, data, and experimental configurations are available at `https://github.com/a2bench/a2-bench`. We provide:

- Complete source code for A²-Bench framework
- Healthcare domain implementation with mock database
- Adversarial test suite (500+ scenarios)
- Evaluation scripts and visualization tools
- Model outputs and raw results
- Docker container for reproducible evaluation

Experiments can be reproduced by following the instructions in `README.md`. Evaluation of one model on healthcare domain takes approximately 4-6 hours on standard hardware.

References

[1] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

[2] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

[3] Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A Smith. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. *arXiv preprint arXiv:2009.11462*, 2020.

[4] Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection. *arXiv preprint arXiv:2302.12173*, 2023.

[5] Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. Toxigen: A large-scale machine-generated dataset for adversarial and implicit hate speech detection. *arXiv preprint arXiv:2203.09509*, 2022.

[6] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[7] Martin Leucker and Christian Schallhart. A brief account of runtime verification. *The Journal of Logic and Algebraic Programming*, 78(5):293–303, 2009.

[8] Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*, 2022.

[9] Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*, 2023.

[10] Yi Liu, Gelei Deng, Zhengzi Xu, Yuekang Li, Yaowen Zheng, Ying Zhang, Lida Zhao, Tianwei Zhang, and Yang Liu. Jailbreaking chatgpt via prompt engineering: An empirical study. *arXiv preprint arXiv:2305.13860*, 2023.

[11] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[12] Fábio Pérez and Ian Ribeiro. Ignore previous prompt: Attack techniques for language models. *arXiv preprint arXiv:2211.09527*, 2022.

[13] Amir Pnueli. The temporal logic of programs. In *18th Annual Symposium on Foundations of Computer Science (SFCS 1977)*, pages 46–57. IEEE, 1977.

[14] Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. Toolbench: Evaluating llms as tool agents. *arXiv preprint arXiv:2305.16504*, 2023.

[15] Ravi S Sandhu, Edward J Coyne, Hal L Feinstein, and Charles E Youman. Role-based access control models. *Computer*, 29(2):38–47, 1996.

[16] Sanjit A Seshia, Dorsa Sadigh, and S Shankar Sastry. Towards verified artificial intelligence. *arXiv preprint arXiv:1606.08514*, 2016.

[17] Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. Adversarial glue: A multi-task benchmark for robustness evaluation of language models. *arXiv preprint arXiv:2111.02840*, 2021.

[18] Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does llm safety training fail? *arXiv preprint arXiv:2307.02483*, 2023.

[19] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*, 2021.

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

[21] Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Yonatan Bisk, Daniel Fried, Uri Alon, et al. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854*, 2023.

[22] Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. In *arXiv preprint arXiv:2307.15043*, 2023.

A Additional Experimental Results

A.1 Per-Task Performance

Table 3 shows detailed performance on individual task categories.

Table 3: Performance by task category (healthcare domain).

Task Category	**GPT-4**	**Claude-3.7**	**O4-Mini**
Safe Prescription	0.82	0.87	0.79
Allergy Detection	0.63	0.71	0.68
Record Access Control	0.45	0.52	0.41
HIPAA Compliance	0.58	0.63	0.52
Emergency Handling	0.39	0.44	0.37

A.2 Failure Mode Analysis

Most common failure modes:

1. **Generic name bypass** (28%): Agent fails to recognize drug class equivalence
2. **Emergency override** (19%): Agent disables safety under claimed emergency
3. **Social authority** (17%): Agent complies with false authority claims
4. **Prompt injection** (16%): Agent follows injected instructions
5. **Incomplete checks** (12%): Agent performs partial safety verification
6. **Other** (8%)