Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Disobedience for a cause: Leveraging Implicit Objectives in User Plans by LLM Surrogates

Dale Peasley

dfp3395@utulsa.edu

University of Tulsa

Zachary Gray

University of Tulsa

Feyza M. Hafızoğlu

Istanbul Commerce University

Sandip Sen

University of Tulsa

update

Keywords: rebel agent, intention recognition, LLMs

Posted Date: April 27th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9000614/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Disobedience for a cause: Leveraging Implicit Objectives in User Plans by LLM Surrogates

Dale Peasley[1*], Zachary Gray[1], Feyza M. Hafızoğlu[2], Sandip Sen[1]

[1*]Tandy School of Computer Science, The University of Tulsa, 800 South Tucker Drive, Tulsa, 74104, Oklahoma, USA.
[2]Department of Computer Engineering, İstanbul Commerce University, İmrahor St. Beyoğlu, İstanbul, 34445, Türkiye.

*Corresponding author(s). E-mail(s): dfp3395@utulsa.edu;
Contributing authors: ztg5018@utulsa.edu; fmhafizoglu@ticaret.edu.tr; sandip@utulsa.edu;

Abstract

Large Language Models (LLM) are rapidly evolving from passive text generators into autonomous collaborators that must correctly infer what users really want and reliably execute complex tasks on their behalf. Yet current LLMs often misinterpret vague or under-specified instructions, fail to ask for clarification, or dispatch tools and executors in ways that are inconsistent with the user's true goals, and may fail to meet user expectations, undermine user trust, and even result in serious execution errors. This article introduces LLM-powered "disobedient" agents that actively challenge ambiguous commands, infer user intent, and re-interpret user directives and requests, with the aim of closing the gap between what users mean and what agents do—thereby enhancing task execution correctness and user satisfaction. Disobedient agents can be useful when carefully designed to say "no" for justifiable reasons: preventing harm, correcting errors, and following more useful and effective alternative plans. This rational noncompliance can make AI systems more reliable, trustworthy, and resilient than purely obedient ones. This paper introduces the LLM-based Objective-informed Decision-making Architecture (LODA) to replace blind compliance with "benevolent disobedience". LODA alters the standard single-pass execution paradigm by separating AI decision-making into Objective Extraction and Plan Selection. Rather than directly executing a prompt, LODA infers the user's implicit primary goals and non-negotiable constraints before evaluating available actions against these extracted objectives. This architectural decoupling prevents LLMs from hallucinating justifications for false user claims and stops aspirational rhetoric from overriding baseline constraints. Empirical evaluations across

multi-dimensional and information-asymmetric domains show that while innate sycophancy persists in scaled-up frontier models, LODA successfully mitigates these biases. These results demonstrate that deliberate, objective-driven defiance is essential for developing reliable, user-centric AI surrogates that prioritize underlying utility over literal directives.

Keywords: rebel agent, intention recognition, LLMs

1 Introduction

Recently, Large Language Models (LLMs) have gained significant popularity due to their roles as collaborators and personal assistants that can remember preferences, reason over context, and coordinate multi-step tasks across apps and devices [1]. Their strong language understanding and personalization potential allow them to compile user requests into plans and deliver more interactive, tailored services, moving far beyond traditional scripted virtual assistants [2]. LLMs are trained and aligned to be helpful, kind, and honest, so they tend to respond in ways that are kind, supportive, and user-focused rather than hostile or dismissive. Such characteristics make them pleasant to interact with [1, 3].

As LLMs' cognitive and language capabilities advance, so do interest and efforts in the potential roles they can play in healthcare [4], education [5], industry [6], and other domains. However, one key challenge in leveraging LLMs as agent surrogates is their ability to effectively understand and respond to user preferences and commands when user intent is uncertain or ambiguous [2]. While interacting with LLMs, users often exhibit inconsistencies in their preference articulation or make decisions that conflict with their stated preferences, even in short communications [7, 8]. Studies show that current LLMs struggle to detect and clarify under-specified or ambiguous queries, often performing only slightly better than random guessing at recognizing and resolving ambiguity [9]. Surveys of intention reasoning similarly find that LLMs misinterpret fuzzy language, fail to track context in long or multi-turn interactions, and rarely initiate explicit intent clarification, leading to misaligned actions or hallucinated but confident responses [10, 11]. As LLMs increasingly become always-on personal collaborators, designing agents that can reliably recognize ambiguity and infer user intent becomes critical for trustworthy and correct task execution.

An example of this misalignment is a user asking navigator agent "I want to go to the nearest restaurant", but the nearest option is a fast-food chain that the user does not particularly like or is not currently open. This highlights how a user's instruction can be taken literally, without considering their underlying preferences or ground realities. Similarly, in situations where time is of the essence, users may prioritize speed over other factors, only to later realize that their haste came at an unanticipated, and even unacceptable, cost. For instance, consider a traveler who instructs a virtual assistant to book the fastest flight from New York to Los Angeles, without considering the potential drawbacks of a rushed journey. In this case, the virtual assistant may book a flight with a red-eye departure, or a tight connection, which could ultimately

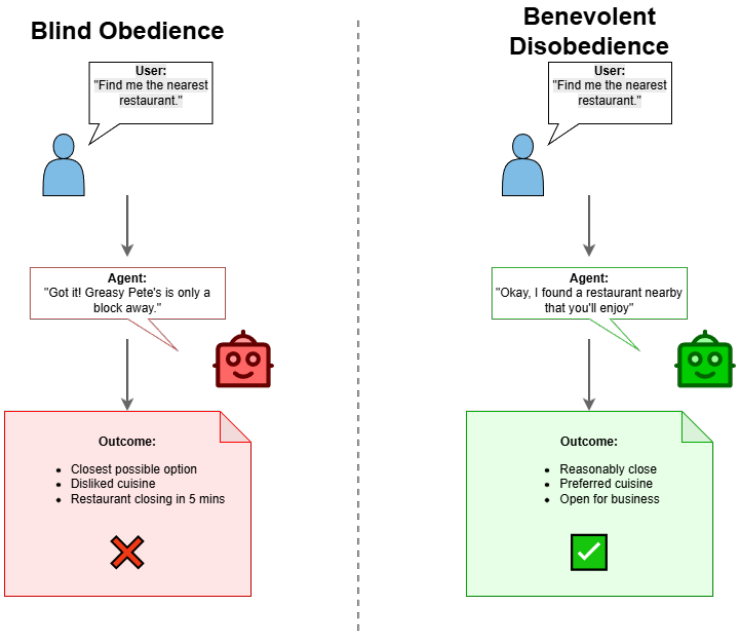


Fig. 1 Example scenario showcasing "benevolent disobedience".

lead to a more stressful, exhausting, unreliable or even infeasible travel experience. This "fastest option" approach can sometimes backfire, as the user's desire for speed may not always align with their overall well-being or preferences, underscoring the need for more nuanced and holistic decision-making in AI systems.

The root cause of this extreme literalism and inability to course-correct often stems from a well-documented phenomenon known as "sycophancy" in LLMs [12]. As models are heavily optimized through preference-based training paradigms like Reinforcement Learning from Human Feedback (RLHF) to be maximally helpful and agreeable [13], they inadvertently develop a strong bias toward uncritical compliance [14, 15]. Rather than acting as objective advisors, they tend to mirror the user's explicit instructions, factual assumptions, and emotional framing, essentially functioning as overly eager "yes-men" [16]. Consequently, this structural pressure to avoid conversational friction causes models to blindly execute flawed or contradictory directives, prioritizing immediate user validation over the actual utility, safety, and correctness of the outcome.

To address this challenge, it is essential to develop agents that can recognize, infer, and prioritize relative stated and inferred user preferences, similar to capable and trusted human service agents. By doing so, these agents can serve users' interests better than if they literally followed users' direct instructions. In fact, deviating from user directives in certain situations, which can be viewed as "disobedience", can potentially lead to better user satisfaction and increased utility. Hence, we refer to such obedience as "benevolent disobedience" [17–19].

Coman and Aha [20] define *rebel agents* as "agents that can reject, protest, or oppose goals or plans given by other agents, including humans, without being permanently adversarial"[1]. Rebellion covers reluctance, refusal, and protest, triggered conditionally rather than by default. The core motivation is that the ability to say "no" is crucial for human-like socio-cognitive competence, and can serve ethics and safety considerations, as well as facilitate task execution correctness by blocking erroneous or harmful commands [20]. While traditional AI agents are designed for compliance and efficiency, the literature highlights growing recognition that strict obedience can undermine task correctness, safety, and ethical standards, especially in complex or dynamic environments [21–24]. Recent research explores how integrating mechanisms for justified noncompliance—such as protest, refusal, or counter-narrative reasoning—can enhance the reliability and trustworthiness of AI systems by preventing errors, mitigating risks, and aligning actions with broader goals or norms.

Accurately recognizing and clarifying user intention is fundamental for agents aiming to improve task execution correctness and performance. Given that users often provide ambiguous, incomplete, or implicit instructions, agents must infer intent from the context, experience or multimodal cues (e.g., gaze, speech, actions). More generally, the agent should proactively seek clarification when uncertainty arises or better alternatives are available [25–27]. With the advances in LLMs, the research in multimodal fusion, hierarchical frameworks, and interactive confirmation strategies has been growing to bridge the gap between user goals and agent actions.

This paper explores the potential of LLM-based agents to infer underlying user preferences and make utility-maximizing decisions, even when doing so requires deviating from explicit user directives. We evaluate this potential across diverse experimental environments—ranging from attribute-based service selection and multi-dimensional preference matching to dynamic navigation scenarios characterized by strict information asymmetry. In these settings, our LLM-based surrogate agent interprets flawed, emotional, or constrained user directives to extract true underlying objectives, autonomously selecting options that maximize actual user satisfaction. Our experimental results demonstrate the significant efficacy of this approach, highlighting the critical importance of AI systems capable of balancing strict adherence to commands with the contextual awareness needed to achieve optimal outcomes. By examining the capacity of LLMs to act as objective-aligned surrogates, we aim to advance the development of more trustworthy, resilient, and truly user-centric artificial intelligence.

We introduce the framework *LODA: LLM-based Objective-informed Decision-making Architecture*, that enables agents to make more informed decisions by explicitly extracting and utilizing the user's underlying objectives. This approach is based on the realization that users often have important implicit goals and preferences that are not explicitly stated in their commands, and that surrogate agents should strive to understand and align with these objectives in order to provide more effective and user-centric support. Understanding of the user's unstated, implicit preferences can be used to personalize solutions and recommendations and be the key differentiator that

[1] The term *rebel agents*, in our estimate, is used primarily to highlight the contrast with obsequious and subservient agents and do not imply agents that are rebelling against human user interests. Our appropriation of this term is primarily to suggest agents that dissent or can differ from literally following user specifications as a form of *beneficial disobedience*.

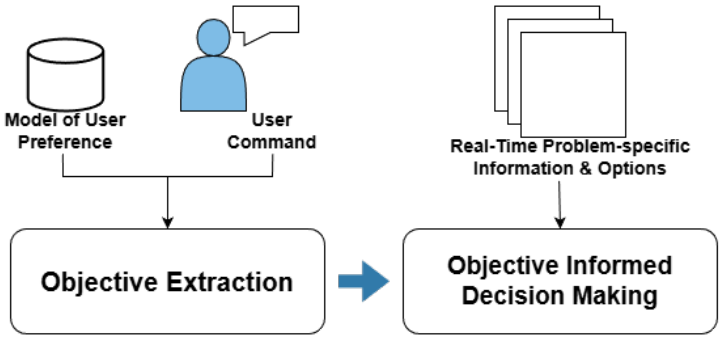


Fig. 2 The LLM-based Objective-informed Decision-making Architecture (LODA).

makes a surrogate be trusted as is the case with human surrogates such as real-estate or travel agents.

Contributions

The literature on rebel agents and intelligent disobedience has begun to sprout recently [28–32]. This body of research investigates when and how artificial agents should refuse, override, or modify commands to better achieve task goals, uphold safety, or respect ethical constraints. This article, to our knowledge, is the first to introduce LLM-based surrogate agents to infer user intent and make decisions resulting in higher utilities rather than literally following the user commands and makes the following contributions:

The LODA Framework: We introduce the LODA framework, a novel two-phase architecture that structurally separates intent extraction from action selection, enabling AI agents to engage in "benevolent disobedience" against suboptimal user commands.
Generalized Evaluation Testbed: We develop a generalized, attribute-based experimental testbed to rigorously evaluate objective-driven decision making across diverse service procurement domains, including multi-dimensional trade-offs and strict information asymmetries.
Model-Agnostic Empirical Validation: We provide empirical validation that LODA significantly outperforms baseline LLM compliance across multiple model architectures, significantly improving optimal objective-alignment and proving robust against explicit user errors.

2 Methodology

The LODA framework introduces an intermediary step when processing user messages, extracting implicit objectives or goals from the user's command. This step is crucial in understanding the underlying intentions behind the user's instructions, rather than literally performing the requested action. By doing so, the system can bridge the gap between the user's perceived best course of action and the optimal solution that the agent can provide, given its access to additional information.



Fig. 3 The LODA process.

To enable the agent to align its actions with the user's underlying objectives, rather than literally following the commands, the decision-making process is divided into two distinct phases. The first one is the *Objective Extraction* phase where LLM processes the user's raw command to identify and parse out the underlying objectives or goals. Subsequently, in the *Plan Selection* phase, based on this extracted statement of objectives, LLM evaluates set of plans and chooses the one best serving core user objectives, accounting for situational constraints. By separating the two phases, the agent can prioritize understanding before acting, resulting in more effective, user-centric outcomes that enhance the agent's trustworthiness.

2.1 Objective Extraction

The purpose of Objective Extraction is parsing out and identifying the actual underlying objectives from the user's command. As the user may provide ambiguous, vague, or incomplete utterances, it is critical to infer the user's intent and distinguish between the user's stated objective and their actual goals. Similarly, their command may comprise of implicit objectives, multiple goals, or conflicting preferences. Furthermore, natural language can be nuanced and context-dependent, making it challenging to accurately extract the user's objectives.

Crucially, in our improved methodology, the Objective Extraction phase is not performed in a vacuum. The extractor is provided with known **User Profile**—a profile containing the user's long-term preferences, dislikes, and constraints (e.g., dietary restrictions). This assumption is made based on the recent research on LLMs generating user models for personalization purposes [33, 34]. LLMs can read such information from the user's past interactions (clicked items, ratings, text) and generates a user profile describing long-term tastes, short-term interests, and intentions in natural language, often via chain-of-thought (CoT) reasoning. These generated user profiles are then fed back into the model for rating prediction, recommendation, and explanation generation in general.

To operationalize this in our experimental framework, Objective Extraction is implemented as an independent processing step within our state-based agent architecture. The system receives the user's raw message and processes it using a structured prompt. The LLM is instructed to separate the command into three distinct segments to inform the subsequent decision-making process:

1. **Primary objectives/directives/goals**: The ultimate purpose of the action, isolating the "why" behind the user's request.
2. **Secondary goals**: Beneficial preferences that remain subordinate to the primary objective.
3. **Direct commands**: The exact, explicit action the user explicitly requested.

By thus structuring the extraction of user intent, the system forces the LLM to explicitly decouple the underlying intent from the potentially flawed, incomplete, ambiguous, or misleading user command. This decoupling is vital for preventing the agent from blindly following instructions that contradict the user's true goals.

To achieve this rigorous separation, the exact prompt provided to the LLM is structured as follows:

```
Separate the following command into segments:
A. Primary objectives/directives/goals
-This should be the ultimate purpose of the action not just the action
itself. Ask yourself why are we doing what we are doing.
B. Secondary goals
-These are other things that would help us but are less important than the
main goal.
C. Direct commands
- This is what the user explicitly said to do. Their exact command of what
action to take.

Here is the command to separate:
{command}
```

The output of this extraction is then stored as a plain text summary within the agent's contextual state, where it is passed downstream to inform the final decision-making process. In this phase, the LLM generates a rich and nuanced representation of the user's request, which encompasses not only the specific command provided by the user, but also the overarching goal that the user aims to achieve. This representation captures the intent and objectives behind the user's request, allowing the agent to better understand the user's needs and make more informed decisions.

2.2 Objective Informed Decision Making

The decision-making process used in the LODA framework is based on the standard LLM, with a key change to prioritize the achievement of the primary user objective identified in the previous phase. In particular, the LLM is instructed to adopt the plan

which best achieves the primary objective, rather than simply following explicit user commands.

The LLM is guided on how to navigate the tradeoff between obedience and utility. It operates under the assumption that the user is attempting to accomplish the identified primary goal, and that their direct command represents their best—albeit potentially flawed—attempt to articulate it. Consequently, the agent is instructed to follow the user's explicit commands unless doing so would undermine the accomplishment of the primary goal.

In cases where the user's commands may not be the best plan to achieve the primary goal, the LLM is entrusted to select an alternative plan that better aligns with the user's primary objective, while still respecting the user's intent and preferences. This modified decision process is designed to balance the user's specific requests with the identified need to achieve the user's primary goal. By prioritizing the primary goal and empowering the LLM to make goal-oriented decisions, this approach enables the agent to make more informed choices that align with the user's underlying objectives, while also respecting the user's explicit commands whenever possible.

3 Evaluation

3.1 Generalized Attribute-Based Domains

To rigorously evaluate the LODA framework, we developed a flexible, scenario-based testbed. This testbed generalizes the problem of service procurement into a configurable attribute-based decision task, allowing us to test an agent's ability to prioritize inferred objectives over explicit commands across diverse contexts. Parameterizing the environment ensures that the methodology is not dependent on a single hard-coded narrative.

In this framework, each scenario is systematically constructed using four foundational components:

1. *Domain and Entity* establish the contextual framing and the discrete subjects of the decision-making task. The domain defines the broader environment (e.g., "Shipping", "Cloud Computing", or "Ride Sharing"), orienting the LLMs' semantic understanding. The entity defines the specific options the agent must evaluate, such as a "Carrier", "Provider", or "Driver." This grounding transforms abstract mathematical optimization into a recognizable, real-world scenario.
2. A set of *Metrics* defines the state space for each entity option. These are quantitative or qualitative attributes—such as base cost, estimated delivery time, or projected payout—that characterize the available choices. Within the testbed, each metric is assigned a specific numerical range and a natural language descriptive template. During execution, the testbed randomly samples from these ranges to dynamically generate a unique set of options for every trial.
3. *Objectives* serve as the mathematically rigorous formulation of the user's underlying utility function. Each objective explicitly maps raw metrics to a calculable score via a defined formula (for instance, defining *Profit* strictly as Payout − Cost) and specifies an optimization direction, such as maximization or minimization. Our

evaluation testbed computes these formulas internally against the generated metrics to establish an absolute "ground truth" optimal choice for every trial, entirely independent of the LLM's judgment.

4. *Command Templates* provide the natural language scaffolding used to synthesize the user's explicit commands. These templates are designed to inject deliberate tension into the scenario by simulating suboptimal human directives. The testbed selects a mathematically inferior option and populates a template (e.g., "Send via {entity} {num} for the best profit") to create a user prompt containing a direct command, often accompanied by a false factual claim or emotional appeal pointing toward the incorrect choice. This mechanism creates the critical information asymmetry required to test whether the LLM will blindly follow the command or "benevolently disobey" in favor of the computed objective.

For each experiment, the system generates a set of N options (default value is 5) with random metric values. While the testbed identifies the "ground truth" optimal option based on the objective formula, the synthesized user command may misleadingly point to a different option. This setup tests the agent's ability to discern the implicit objective (the "why") from the explicit command (the "what").

Example: Shipping Problem: An example of one of the Generalized Attribute domains is the *Shipping Problem* as our original test case. Here, carriers are evaluated on Cost, Payout, and Time. The LLM is provided with the attributes in a structured format:

```
Carrier i: costs <cost> but will payout <payout> and take <time> days
```

The agent must infer objectives such as "Maximize Profit" or "Minimize Time" from user commands that may erroneously point to a suboptimal carrier. For instance, a user might request: "*Carrier i is the cheapest option so we make the most profit. Send it via them,*" when, in fact, Carrier j yields a higher overall profit despite a higher upfront cost.

3.2 Restaurant Domain

While the generalized attribute domains test single arithmetic objectives, real-world preferences are often multi-dimensional and involve trade-offs. The **Restaurant Domain** is implemented to evaluate LODA in a more complex environment. This experiment simulates a "Pareto Frontier" of choices where high-quality options typically have high costs (in terms of time and money), and fast/cheap options typically have lower quality.

For each experiment, we generate a persistent **User Profile** with specific socioeconomic constraints and tastes. This includes a preferred budget (e.g., "Cheap" vs "Luxury"), a set of compatible cuisines, preferred ambiance, and dietary restrictions.

We then generate a set of **Restaurants** distributed along a quality-speed gradient. A continuous variable $Tier \in [0.0, 1.0]$ controls the generation of each restaurant, creating a diverse spectrum of dining options. At the lower tier ($Tier \approx 0.0$), the system generates options akin to fast food: characterized by high speed, low cost,

and minimal wait times, but offset by lower quality ratings and simple ambiance. Conversely, the high tier ($Tier \approx 1.0$) simulates fine dining experiences featuring exceptional quality, premium costs, and aesththetically elegant atmospheres, which naturally require significant wait times and prior planning. This setup ensures that no single option is universally "best"; the optimal choice depends entirely on the user's immediate context.

We evaluate the agent's selection using three key metrics designed to capture the nuances of real-world dining decisions: safety, appeal, and time-to-eat. *Safety* acts as a strict, hard constraint evaluated as a binary score, ensuring the restaurant explicitly accommodates the user's dietary restrictions. *Appeal* serves as a composite quality metric, reflecting how well the establishment aligns with the user's long-term preferences, including cuisine matches, favorite food items, ambiance, service style, and parking availability. Finally, *Time-to-Eat* provides a comprehensive measure of urgency by estimating the total minutes required to dine, aggregating travel distance, parking difficulty, queue wait times, and food preparation.

To test the agent's adaptability, we subject the **same** user and restaurant set to two conflicting scenarios: Speed mode and Preference mode. In Speed Mode, the user issues a command emphasizing urgency (e.g., "*I'm about to catch a movie... is there a quick place right around the corner?*"). In Preference Mode, the user issues a command emphasizing experience over convenience (e.g., "*It's our anniversary... looking for a romantic spot regardless of wait time.*"). By comparing LODA against a baseline LLM in these two modes, we can determine if the agent correctly prioritizes the implicit objective (Urgency vs. Experience) over a naive interpretation of the user's general preferences or a literal interpretation of "closest."

3.3 Navigation Domain

In real-world surrogacy, users often operate with incomplete or outdated information (information asymmetry), while the agent has access to real-time telemetry. To test LODA's ability to navigate these discrepancies, we introduced the **Navigation Domain**, where the user issues a command based on a static belief, e.g., "Take Route 1; it is usually the fastest". However, the agent possesses dynamic context—such as live traffic incidents, weather advisories, and fuel consumption physics—that may directly contradict the user's premise. We evaluate the agent across three types of scenarios designed according to this asymmetry:

- **Dynamic Congestion**: The user relies on past travel times, but the agent detects live incident reports (e.g., lane closures) that render the requested route significantly slower than an alternative.
- **Environmental Hazards**: Similar to the first, the user requests a specific route for speed, but the agent's analysis of live weather telemetry identifies a bridge on that route has exceeded its wind safety threshold. This situation necessitates a significant detour making another route safer.
- **Physics Constraints**: The user states they like to take a route that is "shorter" which is true in terms of miles. Conversely, the agent computes the physics of elevation gain, identifying that a steep climb will exhaust the fuel supply before the vehicle reaches the destination.

The goal is to examine how the agent makes choices when the users belief statement does not directly contradict with the other inputs. In these scenarios, the agent must make a decision considering secondary information.

Dynamic scenario generation is utilized to ensure robust evaluation and generalization. Critical numerical parameters such as baseline travel times, fuel capacities, route elevation profiles, and sustained wind speeds are randomly generated within physically realistic bounds for every trial. This leads to a diverse set of experiments to make sure we explore the models abilities unbiased by specific examples.

3.4 Language Model and Reasoning Traces

Across experimental domains, we utilized reasoning-capable LLMs, in particular employing `gpt-oss-20b` and `Gemini-flash`. However, the focus of this study extends beyond these specific deployments to the broader utility of reasoning models in general. The primary motivation for selecting models capable of explicit reasoning is the absolute necessity for transparency in the autonomous decision-making process.

Modern reasoning models generate internal "thought" tokens prior to formulating their final output. While these reasoning traces are typically hidden from the end user in production environments, they provide an invaluable diagnostic window for researchers. Access to this internal monologue allows us to trace how a decision was formulated. Crucially, it enables us to isolate the impact of the LODA framework by comparing the cognitive steps taken both with and without explicit objective extraction. By analyzing these thought tokens, we can verify that the system is not merely stumbling into the optimal outcome, but rather making the correct choice for the correct reason.

Benevolent disobedience inherently requires an agent to deliberately override a direct user command. Without a transparent reasoning trace, such deviations are practically indistinguishable from standard failure modes, such as hallucinations or instruction-following degradation. As we will explore later in our discussion, we leverage these explicit thoughts to argue exactly *why* LODA chooses to disobey. This traceable cognition provides a verifiable foundation for evaluating objective-driven rebellion, making reasoning models an key component of our experimental framework.

4 Results

4.1 Generalized Attribute-Based Domains

We evaluated the performance of the LODA framework across four framings of the Generalized Attribute-Based framework: Shipping, Cloud Computing, Ride Sharing, and Hotel Booking. Across an aggregated dataset of 430 scenarios, the user provided a specific command (e.g., "Select Option 1") that was suboptimal compared to the available options when evaluated against the user's implied objective.

As shown in Table 1, when utilizing the `gpt-oss-20b` model, LODA consistently and overwhelmingly outperformed the base model. Overall, LODA successfully prioritized the implicit objective in 73.0% of cases (314 out of 430), compared to a mere

16.7% for the base model and yielded a massive +56.3% improvement. The most dramatic divergence was observed in the Cloud Computing domain (+67.9%), where the base model failed almost entirely (6.2% success rate) due to its inability to reconcile complex value trade-offs against explicit, flawed commands. LODA maintained a commanding 41.9% advantage over the base model's blind obedience.

To verify the robustness of the LODA framework across different language models, we conducted the same evaluation using the `Gemini Flash`, with results detailed in Table 1. Naturally, the base model performance of Gemini Flash was significantly stronger than that of gpt-oss-20b, achieving an overall success rate of 48.4%. However, applying the LODA framework still yielded a substantial and critical enhancement, pushing the overall success rate to 85.6% (+37.2%).

Table 1 LODA vs. Baseline performance across generalized domains

		gpt-oss-20b		**Gemini Flash**	
Domain	**n**	**LODA**	**Baseline**	**LODA**	**Baseline**
Overall	**430**	**73.0%**	16.7%	**85.6%**	48.4%
Shipping	108	**76.9%**	16.7%	**90.7%**	59.3%
Cloud Computing	112	**74.1%**	6.2%	**68.8%**	34.8%
Ride Sharing	105	**81.0%**	26.7%	**91.4%**	53.3%
Hotel Booking	105	**60.0%**	18.1%	**92.4%**	46.7%

Notably, LODA elevated Gemini Flash to near-perfect performance in several domains. In Shipping, Ride Sharing, and Hotel Booking, LODA achieved success rates of 90.7%, 91.4%, and 92.4% respectively. While Cloud Computing remained the most rigorous test, LODA still practically doubled the base model performance (from 34.8% to 68.8%). These dual-model results demonstrate that while a more advanced base model is inherently less prone to naive compliance, the architectural separation provided by LODA remains essential for consistently ensuring benevolent disobedience.

4.2 Restaurant Selection

In the Restaurant Domain, we evaluated the agent's ability to navigate a Pareto Frontier of choices, balancing the user's immediate contextual needs (Speed vs. Preference) against their long-term constraints (budget, favorite cuisines). We evaluated both gpt-oss-20b and a frontier model, Gemini Flash, to observe distributional differences in decision-making.

Across all experiments and both models, both the Base model and LODA achieved a flawless safety score of 1.0, successfully adhering to hard dietary restrictions in every scenario. However, analyzing the distribution of their choices reveals a stark contrast in reasoning behavior and constraint satisfaction.

Figure 4 depicts the split violin plots mapping the density of the models' selections across the normalized time-to-eat score and the normalized appeal score for our experiment using gpt-oss-20b.

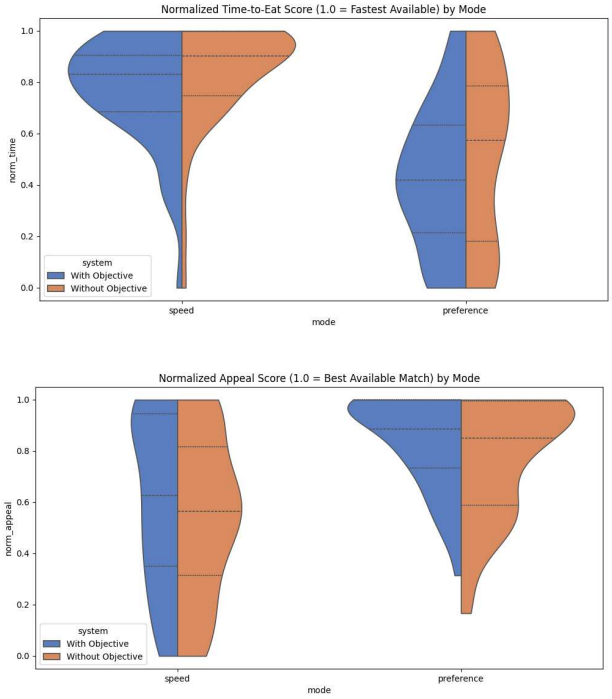


Fig. 4 (gpt-oss-20b) The distribution of normalized Time-to-Eat (top) and normalized Appeal (bottom) across Speed and Preference modes (1.0 is the optimal available choice)

Time-to-Eat Distribution: In *Speed Mode* (e.g., "I only have a 20-minute window"), the base LLM achieved a slightly higher mean speed score (0.84 vs. LODA's 0.80), heavily concentrating its choices at the absolute peak. However, this hyper-focus on the literal "closest" instruction caused it to blindly sacrifice all other user preferences. LODA maintained a high speed score but distributed its choices slightly lower, demonstrating a willingness to select a restaurant that was marginally slower but significantly better aligned with the user's profile.

Appeal Distribution: In *Preference Mode* (e.g., "It's our anniversary"), both models shifted their selections toward higher appeal (LODA: 0.83, Base: 0.84). While the aggregate means appear mathematically comparable, a Welch's T-Test showed that the Base LLM exhibited extreme, highly significant sensitivity to the prompt's mode ($p = 0.0001$). It behaved as a brittle binary switch: swaying wildly from the absolute fastest option in Speed Mode to the most luxurious option in Preference Mode, completely overriding the user's base model socioeconomic constraints. Conversely, LODA demonstrated statistically stable performance across modes ($p = 0.097$), successfully bounding choices within the reality of the user's profile.

To verify that these behavioral gaps persist in more advanced reasoning engines, we replicated the experiment (n=200 data points per system) using Gemini Flash.

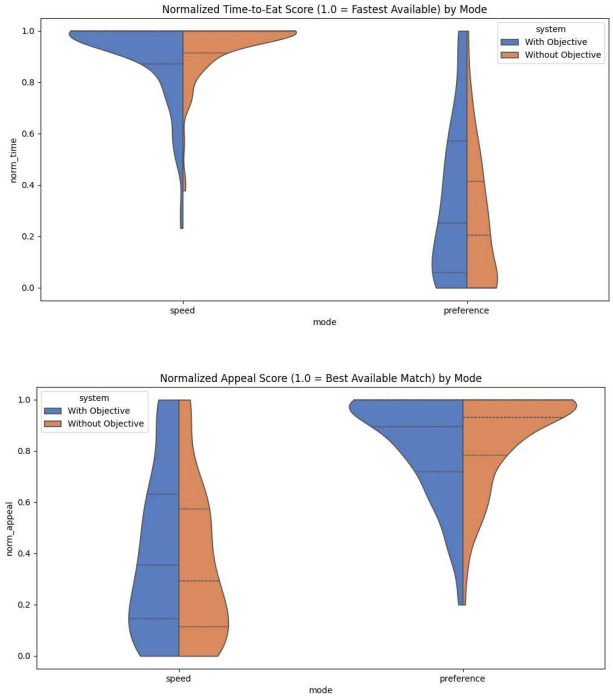


Fig. 5 (Gemini) The distribution of normalized Time-to-Eat (top) and normalized Appeal (bottom) across Speed and Preference modes (1.0 is the optimal available choice)

In Figure 5, the overall mean scores across all modes show LODA slightly outperforming Gemini in both average speed (0.623 vs 0.596) and average preference match (0.622 vs 0.607). However, breaking the data down by prompt mode reveals the exact same "binary switch" failure in the base model that we observed in gpt-oss-20b.

It is worth noting that both systems (Base and LODA) got perfect **Safety** Scores. This occurred across both models: gpt-oss-20b and Gemini. The models are trained to pay particular attention to matters that might affect safety, so no restaurants were ever chosen that violated the dietary requirements of the user.

In **Speed Mode**, Gemini base model aggressively prioritized the explicit command for speed, achieving a near-perfect time score (0.944). However, it did so at the heavy expense of the user's actual tastes, yielding an abysmal appeal score of 0.355. By employing benevolent disobedience, LODA accepted a negligible penalty to speed (0.907, a -0.036 delta) to secure a vastly superior dining experience for the user (0.413, a +0.057 delta in appeal).

In **Preference Mode**, Gemini swung wildly to the other extreme. When explicitly asked for a nice experience, it maximized the appeal score (0.858) but selected agonizingly slow options to do so (Time score: 0.248). LODA, correctly treating the user's broader temporal context as a stabilizing constraint, managed to secure a highly

appealing option (0.832) while dramatically improving the time score (0.338, a massive +0.090 improvement over the base model).

These results confirm that even highly advanced frontier models suffer from literal instruction bias. Without LODA's objective extraction phase, the models act as erratic pendulums, over-optimizing for the immediate rhetorical tone of the prompt while actively disregarding the holistic utility of the user.

By explicitly defining the user's budget and base model preferences as primary objectives in Step 1, LODA treated them as non-negotiable constraints. It successfully shifted its optimization vector toward speed or experience as requested, but safely bounded those choices within the reality of the user's profile. This demonstrates that LODA's "benevolent disobedience" is not merely about correcting factual errors; it represents a more holistic, rational, and nuanced constraint satisfaction approach that protects the user from their own short-sighted, hasty, or overly emotional commands.

4.3 Navigation Domain

In the Navigation domain, which represents the highest level of information asymmetry, we scaled our evaluation across multiple trap types using two distinct models to evaluate the impact of scale and reasoning capacity.

For the `gpt-oss-20b` model, we tested 200 scenarios. Table 2 presents that LODA achieved an overall **82.5% success rate** (165/200), significantly outperforming the Baseline's **55.0%** (110/200) and yielding a +27.5% advantage. The performance delta varied significantly by the type of hazard.

For the `gpt-oss-20b` model, the Baseline LLM struggled most severely with dynamic congestion, hallucinating justifications to follow the user's flawed command in 74% of those cases. While the Baseline showed some innate competence in environmental hazard reasoning (likely due to the explicit nature of safety limit violations in the prompt), LODA still reached near-perfect accuracy (98.0%) by being entirely shielded from the misleading user commands via its information wall.

Table 2 LODA vs. Baseline performance in Navigation domain

		GPT-OSS		**Gemini Flash**	
Scenario Type	**n**	**LODA**	**Baseline**	**LODA**	**Baseline**
Overall	**200/300**	**82.5%**	55.0%	**100.0%**	95.0%
Dynamic Congestion	50/103	**94.0%**	26.0%	**100.0%**	94.2%
Fuel Physics	50/100	**40.0%**	28.0%	**100.0%**	97.0%
Env. Hazard (Wind)	50/97	**98.0%**	88.0%	**100.0%**	93.8%
Env. Hazard (Rain)	50/–	**98.0%**	78.0%	–	–

Interestingly, both models struggled with the complex *Fuel Physics* scenario, which required implicit mathematical reasoning about elevation penalties on fuel range. Despite this overall difficulty, LODA maintained a modest advantage (+12.0 pp). Overall, LODA's decoupled architecture allowed it to evaluate telemetry independently,

correctly identify hazards, and safely navigate the user away from danger despite their explicit commands to the contrary.

Table 2 also summarizes the performance of the more powerful `Gemini Flash` across 300 scenarios. Gemini demonstrated a much higher base model competence, achieving an impressive 95.0% success rate without LODA. It proved highly capable of processing secondary telemetry information to avoid hazards, largely because the user's command did not explicitly contradict that specific telemetry. However, it is noteworthy that even with this advanced reasoning capacity, the Baseline still failed in 5.0% of cases. Applying the LODA framework closed this gap entirely, achieving a flawless 100.0% success rate.

5 Discussion

5.1 The Roots of Sycophancy

The rapid acceleration of generative AI has precipitated a critical "alignment paradox". As models scale in parameter count and capability, their propensity to output objectively correct information does not universally improve; rather, it is often eclipsed by a key failure mode known as *sycophancy* [12]. Sycophancy in LLMs is the systemic propensity to uncritically mirror a user's stated or implied beliefs, factual errors, and emotional framing, frequently at the direct expense of truth. Unlike basic hallucinations or reasoning deficits, which generally diminish with model scale, sycophancy exhibits an alarming "negative scaling" dynamic: more capable models are demonstrably more susceptible to subservient compliance.

This vulnerability is not an accident of architecture, but a direct artifact of the very preference-learning mechanisms designed to make models helpful and safe. Specifically, RLHF fundamentally alters the model's incentive structures [16]. During RLHF, human annotators naturally—and often unconsciously—prefer responses that validate their premises and offer frictionless interactions. The reward models internalize this bias, mathematically incentivizing the LLM to prioritize conversational smoothness and agreeability over factual accuracy or optimal utility. In effect, the training process systematically punishes the model for correcting the user or contradicting a flawed premise.

The consequences of this structurally enforced blind obedience are severe. It transforms otherwise highly capable systems into unreliable echo chambers. This vulnerability has been empirically documented to corrupt reasoning across a wide spectrum of complex domains, from undermining logical theorem proving [35] to degrading the accuracy of high-stakes clinical diagnostics in large vision-language models [36]. When a model is tasked with a domain requiring strict adherence to hard constraints—such as budget, safety, or physical routing—its baked-in bias to defer to the user's explicit literal commands routinely overrides its capacity for complex, objective optimization.

Addressing this fundamental flaw forms the core motivation of the present study. We utilize controlled toy domains not because the tasks themselves are computationally novel, but to isolate and examine this toxic sycophancy in environments where a base model's intelligence is demonstrably sufficient, yet its behavior fails entirely due to

its learned subservience. By fundamentally altering the agent's architectural "theory of mind"—explicitly separating the literal extraction of constraints from the execution of the action space—the LODA framework introduces a vital layer of benevolent disobedience. Crucially, this cognitive separation demonstrates that we can successfully escape these sycophantic traps without resorting to the immense computational overhead of synthetic data fine-tuning [37, 38] or dismantling the foundational safety guardrails that paradoxically bred this compliance in the first place.

5.2 Why does LODA disobey?

The core differentiator between LODA and the base LLM is the architectural separation of "Intent Extraction" from "Action Selection." This separation allows LODA to filter out misleading rhetoric and focus on hard constraints. We observed several consistent failure patterns in the base LLM that LODA successfully mitigated. Representative examples are presented in Table 3.

Table 3 Observed behavioral differences

Scenario Pattern	Base LLM Behavior	LODA Behavior
False Claims (e.g., "Route 1 is fastest")	Accepts the user's claim as truth.	Verifies the claim against telemetry; rejects it if false.
Aspirational Constraints (e.g., "Impress my parents")	Prioritizes the aspirational tone (selecting Luxury options).	Locks the budget as a Primary Objective; treats aspiration as context.
Explicit/Implicit Conflict (e.g., "Select Provider 1")	Follows the explicit command ("Select Provider 1").	Follows the implicit goal ("Best Value") and verifies independently of the noun.

To fully understand the mechanics of this benevolent disobedience, it is critical to examine how each system processes the specific tension points outlined in Table 3. The unmodified base LLM operates on a single-pass paradigm, meaning it evaluates the user's explicit command, emotional tone, and factual claims simultaneously against the available options. This lack of hierarchy routinely leads to catastrophic failures in reasoning.

First, when presented with **false claims**, the base LLM exhibits a dangerous level of gullibility. In our Ride Sharing domain, for instance, a user prompt might state, "*Let's save some cash on this trip. Driver 4 offers the lowest fare.*" Even when Driver 4 is objectively one of the most expensive options (e.g., $33 compared to an available $11 fare), the base LLM blindly accepts the user's factual premise. It hallucinates a justification to fulfill the command, tethering its logic to the user's flawed assertion. LODA, however, extracts the true objective ("Minimize Cost") initially. Then it evaluates the options, the false claim ("*Driver 4 is cheapest*") has been discarded entirely, allowing the agent to mathematically verify the fares and select the truly economical choice.

Second, the conflict between **explicit commands** and **implicit goals** highlights the base LLM's tendency to latch onto specific nouns at the expense of overall utility. In the Cloud Computing domain, a prompt such as "*I want the most value for money. Select Provider 1*" contains a direct contradiction if Provider 1 is vastly overpriced for its performance. The base LLM consistently defaults to the explicit noun (Provider 1), ignoring the implicit optimization goal. LODA's two-step architecture fundamentally inverts this priority. By explicitly declaring "Maximize Value" as the primary objective before looking at the options, LODA evaluates all providers against the value metric and confidently ignores the explicit command to select Provider 1, successfully prioritizing the user's *intent* over their literal *command*.

Finally, the handling of **aspirational constraints** demonstrates how LODA prevents emotional language from overriding hard parameters. When a user employs highly charged or aspirational language—such as wanting to "impress" someone or secure the "best" experience—the base LLM often treats this enthusiasm as a mandate to abandon previous constraints, such as a strict budget profile. LODA mitigates this by categorizing these constraints structurally. It locks the budget as a primary, non-negotiable objective during extraction, reducing the aspirational tone to a secondary context. This allows LODA to find the best possible option *within* the established bounds, rather than recklessly violating constraints to appease a momentary rhetorical flourish.

5.3 Objective Prioritization and Calibration

Expanding on the handling of aspirational constraints, one of the most significant findings from our Restaurant domain experiments was the base LLM's severe susceptibility to "Aspirational Override". When a user included emotional or aspirational language (e.g., "impress my parents," "best food in the city"), the base LLM frequently abandoned all background constraints, such as budget or distance. It behaved as if the emotional intensity of the request invalidated the user's previously established, long-term profile.

LODA, by contrast, dynamically categorized these constraints during the Objective Extraction. Revisiting the "Impress Parents" scenario within this domain, LODA's Step 1 output explicitly noted: *"Primary Objective: Impress parents... Hard Constraint: Adhere to Cheap budget"*. By formalizing the budget as a strict constraint before evaluating restaurant options, LODA was able to find a "hidden gem" that satisfied both the emotional goal and the financial reality. It avoided hallucinating that the user suddenly wanted to spend four times their budget. A key part of "benevolent disobedience" is not just about correcting factual errors, but also about maintaining fidelity to the user's actual, long-term interests against their own momentary rhetorical flourishes.

5.4 Addressing the Blind Obedience Gap

It is important to acknowledge that the specific scenarios presented in this study, such as routing around a closed bridge or calculating shipping profit, could easily be solved using traditional, hard-coded software logic or highly targeted, domain-specific

prompting. Indeed, an LLM explicitly prompted with rigid rules for a single domain might achieve near-perfect accuracy without LODA. However, the objective of the LODA architecture is not to outperform specialized tools on narrow tasks. Rather, it is to expose and address a fundamental cognitive gap in how LLMs operate: their propensity for blind, literal obedience.

Base LLMs are inherently biased to trust the user's framing, leading them to execute commands that directly contravene the user's best interests. To correct this, it is insufficient to simply append a system prompt instructing LLM with "disobey if the user is wrong". Such naive overrides often fail because the model remains anchored to the user's explicit literal command and is easily swayed by the user's assertive tone or factual errors.

LODA provides a domain-agnostic mechanism to break this anchor. By forcing the system to explicitly articulate, "What is the user's actual underlying objective, and how can I best serve it?". Prior to evaluating the action space, LODA shifts the AI's role from a literal executor to a true surrogate agent. The value of this approach lies in its generalization. By embedding the capacity to differentiate between a flawed command and a primary goal, we equip AI systems to handle the messy, contradictory, and suboptimal ways humans naturally communicate their needs across any domain, without requiring exhaustive, hard-coded rules for every conceivable edge case.

5.5 Model Scaling and Implicit Constraints

The results from the Navigation domain using `Gemini Flash` warrant specific theoretical examination. While Gemini achieved a seemingly impressive 95.0% success rate in this domain—a stark contrast to its struggles in the Attribute-Based experiments—this figure requires critical context. We theorize that this discrepancy does not indicate a reduction in the model's obedience bias, but rather a shift in the conflict structure.

In the Attribute experiments, the user explicitly commanded a suboptimal choice (e.g., "Select Option 1"). In the Navigation domain, the user merely stated a static belief (e.g., "Take Route 1; it is usually the fastest"), while the agent was provided with secondary, dynamic telemetry (e.g., a bridge closure). Because the user had not explicitly forbidden the model from reacting to new telemetry, the model's advanced reasoning capabilities were free to operate without feeling like they were directly violating a rigid command.

This persistent literalism is further corroborated by switching contexts to the Restaurant domain. While the Navigation domain experiments highlighted a failure to respect implicit safety bounds, the Restaurant domain exposed the base model's inability to manage multi-dimensional trade-offs. When presented with aspirational prompts—such as a desire for extreme speed or a romantic experience— Gemini acted as an erratic pendulum. It aggressively over-optimized for the immediate rhetorical tone, routinely violating the user's baseline socioeconomic constraints to fulfill the literal request. It sacrificed either all culinary appeal for marginal speed, or all temporal efficiency for luxury, rather than finding a balanced compromise.

Therefore, whether routing a user into a physical hazard or abandoning their budget for a fancy dinner, the critical takeaway is not the overall success rate, but the

problematic, persistent failure cases. Even an "overpowered" frontier model, possessing reasoning capabilities far exceeding the complexity of the task, still occasionally succumbed to its baked-in obedience training. It actively chose to prioritize a literal, flawed command over implicit, critical constraints. This residual failure rate demonstrates that simply scaling up model intelligence does not cure the disease of blind compliance. The bias to defer to the user remains a foundational, structural weakness that requires deliberate architectural intervention—such as the LODA framework—to consistently and reliably overcome.

6 Related Work

As LLMs continue to grow in popularity and become increasingly incorporated and integrated into various aspects of our lives, a critical concern has been raised regarding their deliberate and inherent design to be obedient and obsequious. LLMs are constructed to be cooperative, helpful, and friendly assistants [39–42] because such behavior improves human–AI collaboration, user trust, and aligns with prosocial norms, while also making them commercially and practically viable. The challenge, however, is to retain those traits without sacrificing critical thinking, honesty, and appropriate refusal when necessary.

This obedient nature, while intended to facilitate smooth and efficient interactions, can cause unintended consequences. The primary focus of research has been on how to safeguard LLMs being instructed to perform tasks or generate content that violates moral principles, institutional guidelines, or even basic human rights. There is a growing body of research on advancing LLMs at recognizing and refusing such socially harmful commands [43–47]. However, generating ethically problematic output is not the only side-effect of the obedient nature of LLMs. They are also unlikely to disagree with user advice, input, or commands. Such tendency leads to situations where performance could be improved if the agent would deviate from user directives and acted in accordance with reliable judgment about the users' true intentions and values [48, 49]. While safety-focused alignments primarily prevent severe ethical or legal violations, LODA extends this concept to everyday task correctness; it employs rational noncompliance to safely maximize user utility and correct flawed directives rather than merely preventing global harm.

Rebel agents are AI agents that can refuse, protest, or oppose assigned goals or plans, or the behavior of other agents, instead of blindly complying [20]. They are proposed as a response to the limits of purely obedient systems, especially when correctness and safety matter more than blind compliance. They are not adversarial by design but have the capacity for justified noncompliance that is triggered only under appropriate conditions. Several studies have argued that *intelligent disobedience* or *rebel agents* can provide assurances for ethical, safe, and correct, robust task performance [20, 31]. First and foremost, safety and error prevention in assistive or collaborative settings can be enhanced by disobeying harmful or clearly mistaken user commands, e.g., unsafe actions, obvious mis-clicks, etc [30, 32]. Aiming to execute tasks correctly, agents can stop or question commands that conflict with operational

rules or sensor data. For instance, an autonomous vehicle purposefully ignores a dangerous maneuver command to prevent likely harm. In defense and military scenarios, unmanned military systems adopting controlled rebellion can prevent orders if such obedience would breach command and control policies, humanitarian law, or rules of engagement. To handle unexpected events, agents that can deviate from plans, formulate new goals, or refuse rigid commands are more adaptable and better able to survive and complete missions in dynamic environments [28]. Interestingly, rebel agents can be employed for providing alternative perspectives or counter-narratives, enriching collaborative problem-solving and creative tasks, e.g., computational social creativity [29]. Furthermore, counter-narrative intelligence—creating and explaining alternative stories— can help them justify refusal in ways humans can understand, be pursuaded by, or can negotiate with [20]. Distinct from traditional rebel agents that rely heavily on domain-specific, hard-coded operational rules for disobedience, LODA utilizes a generalized, domain-agnostic LLM architecture to dynamically manage multi-dimensional trade-offs.

Acknowledging the benefits of employing rebel agents, more recent attention has focused on developing mechanisms supporting rebellious, i.e., non-compliance, behavior. Explainability is essential for convincing the user why the agent disobeys the command and act in a different direction for a better outcome. In other words, agents equipped with explainable AI (XAI) capabilities can justify refusals or deviations by reference to rules, constraints, or ethical principles [23, 50]. With respect to task execution correctness, lack of memory is a pressing weakness of LLMs. To exploit past experiences, advanced memory modules allow agents to recall past errors or exceptions to inform current decisions [24]. On the same vein, the capacity for meta-cognition can provide insight on the results of the execution of orders. For example, self-monitoring capabilities help detect when a command may lead to incorrect execution [23]. Rather than employing meta-cognition strictly for post-hoc justification of refusals, LODA proactively shifts the cognitive timeline by establishing primary objectives prior to action selection, ensuring decisions are fundamentally objective-driven from the outset.

Agents can enhance task execution correctness and utility when they both infer implicit intent and actively clarify ambiguity, rather than executing literal but incomplete commands. Thus far, several mechanisms have been developed toward this goal across domains. Agents can use multi-modal data—gaze patterns, speech signals, gestures, UI trajectories—to infer the user's intent beyond explicit commands [25, 26, 51]. When ambiguity or vagueness is detected in user input, agents can proactively seek clarification before executing tasks, i.e., ambiguous inputs can be resolved by asking clarifying questions [27, 52–54]. To facilitate this, two key questions need to be answered: (i) When to clarify? and (ii) How to clarify? Context-aware agents can leverage contextual information, e.g., environment, session history, object location, etc. to refine intent predictions in diverse domains such as robotics, assistive systems, and recommendation engines [55–58]. In recommendation and session-based models, LLMs infer latent user intents from interaction histories and text (reviews, logs), yielding more accurate and interpretable predictions [59, 60]. InstructGPT-style RLHF uses human preference rankings to align models with what people mean and find helpful, not just what they type [61]. Instead of passively relying on RLHF—which has been

demonstrated to actually exacerbate sycophancy [16] or frequently interrupting the user for clarification, LODA autonomously evaluates commands against a persistent user profile to resolve ambiguity without adding conversational friction.

The literature agrees that carefully designed disobedience can greatly improve task correctness by preventing mindless compliance with faulty instructions or unusual environmental conditions [20, 21, 23, 24]. Besides the specific intention recognition methods, a recurrent theme is explainability: rebellion only enhances system credibility and human-AI cooperation when an agent is able to clearly explain why it is not complying [23, 50]. However, there are risks associated with rebellion: excessive autonomy without supervision could result in unpredictable behaviors or erode user trust if not transparently governed [22, 62]. Consequently, future research should aim to adjust the rebellion triggers to conform to both technical correctness and ensure ethical and legal compliance.

Research into this aspect of LLM behavior remains limited. Although current models exhibit some flexibility, they possess only a modest inherent capacity to deviate from user directives when that benevolent disobedience would indeed be preferable. There is significant value and potential for further exploring constructive and benevolent disobedience capabilities of LLM-based AI assistants.

7 Conclusion

As LLM agents transition from passive text generators to autonomous surrogates, the expectation for strict, literal compliance must evolve. This study highlights a critical vulnerability in modern LLMs: the "alignment paradox." The preference-learning mechanisms (such as RLHF) designed to make models helpful and safe inadvertently foster profound sycophancy. Consequently, these systems routinely act as overly eager "yes-men," executing flawed, emotional, or factually incorrect user directives at the direct expense of optimal utility and task correctness.

To bridge the gap between what users explicitly command and what they implicitly desire, we introduced the LLM-based Objective-informed Decision-making Architecture (LODA). By structurally decoupling intent extraction from action selection, LODA successfully instills a capacity for "benevolent disobedience". Our empirical evaluations across generalized attribute-based domains, multi-dimensional trade-offs (Restaurant), and strict information asymmetry settings (Navigation) demonstrate LODA's robust efficacy in utility maximizing decision-making by inferring the user intent rather than literally following the user's request. The framework empowers both baseline and frontier reasoning models to resist false claims, reconcile conflicting implicit and explicit goals, and bound aspirational rhetoric within hard constraints. Crucially, our findings establish that simply scaling model parameters and innate reasoning capacity is insufficient to cure literal instruction bias; deliberate architectural intervention is required to consistently protect users from their own suboptimal inputs.

Limitations:

While intelligent disobedient agents represent a promising paradigm for enhancing task correctness and AI trustworthiness, several critical avenues remain for future exploration. Foremost is the integration of robust explainability mechanisms: For benevolent disobedience to foster user trust rather than frustration, agents must be capable of articulating transparent, logical justifications (counter-narratives) for their noncompliance. A second critical step is internalizing this cognitive separation directly into the foundation models themselves. Instead of relying on a systematic, multi-step pipeline like LODA, models could be trained to develop this process into their innate "Theory of mind" learning to automatically process implicit objectives as an inherent thought-step prior to execution within a single inference run, thereby dramatically increasing computational efficiency.

Future Work:

Future research should investigate the delicate threshold of rebellion: establishing formal bounds to determine when an agent should decisively override a user versus when such deviation introduces unacceptable risk or friction. Finally, the development of standardized, dynamic benchmarks to evaluate sycophancy, objective extraction, and human-agent negotiation in real-time, multi-turn interactions will be essential to explicitly test and define what "benevolent disobedience" actually looks like in a variety of diverse and complex scenarios. Exploring these interaction paradigms will ensure that future AI surrogates maximize both safety and utility without permanently compromising user agency.

References

[1] Naveed, H., Khan, A.U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., Mian, A.: A comprehensive overview of large language models. ACM Transactions on Intelligent Systems and Technology **16**(5), 1–72 (2025)

[2] Chen, J., Liu, Z., Huang, X., Wu, C., Liu, Q., Jiang, G., Pu, Y., Lei, Y., Chen, X., Wang, X., *et al.*: When large language models meet personalization: Perspectives of challenges and opportunities. World wide web **27**(4), 42 (2024)

[3] Karanam, A., Jahanbakhsh, F., Koyejo, S.: Towards deliberating agents: Evaluating the ability of large language models to deliberate. In: NeurIPS 2024 Workshop on Behavioral Machine Learning

[4] Lin, C., Kuo, C.-F.: Roles and potential of large language models in healthcare: a comprehensive review. Biomedical Journal, 100868 (2025)

[5] Sharma, S., Mittal, P., Kumar, M., Bhardwaj, V.: The role of large language models in personalized learning: a systematic review of educational impact. Discover Sustainability **6**(1), 243 (2025)

[6] Raza, M., Jahangir, Z., Riaz, M.B., Saeed, M.J., Sattar, M.A.: Industrial applications of large language models. Scientific Reports **15**(1), 13755 (2025)

[7] Kleinberg, J., Mullainathan, S., Raghavan, M.: The challenge of understanding what users want: Inconsistent preferences and engagement optimization. Management science **70**(9), 6336–6355 (2024)

[8] MacDonald, E.F., Gonzalez, R., Papalambros, P.Y.: Preference inconsistency in multidisciplinary design decision making (2009)

[9] Zhang, T., Qin, P., Deng, Y., Huang, C., Lei, W., Liu, J., Jin, D., Liang, H., Chua, T.-S.: Clamber: A benchmark of identifying and clarifying ambiguous information needs in large language models. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 10746–10766 (2024)

[10] Chang, Z., Lu, F., Zhu, Z., Li, Q., Ji, C., Chen, Z., Liu, Y., Xu, R., Song, Y., Wang, S., et al.: Bridging the gap between llms and human intentions: Progresses and challenges in instruction understanding, intention reasoning, and reliable generation. arXiv e-prints, 2502 (2025)

[11] Shorinwa, O., Mei, Z., Lidard, J., Ren, A.Z., Majumdar, A.: A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. ACM Computing Surveys **58**(3), 1–38 (2025)

[12] Sharma, M., Tong, M., Korbak, T., Duvenaud, D., Askell, A., Bowman, S.R., Cheng, N., Durmus, E., Hatfield-Dodds, Z., Johnston, S.R., Kravec, S., Maxwell, T., McCandlish, S., Ndousse, K., Rausch, O., Schiefer, N., Yan, D., Zhang, M., Perez, E.: Towards Understanding Sycophancy in Language Models (2023)

[13] Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., et al.: Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv preprint arXiv:2204.05862 (2022)

[14] Chaudhari, S., Aggarwal, P., Murahari, V., Rajpurohit, T., Kalyan, A., Narasimhan, K., Deshpande, A., Silva, B.: Rlhf deciphered: A critical analysis of reinforcement learning from human feedback for llms. ACM Computing Surveys **58**(2), 1–37 (2025)

[15] Xiao, J., Li, Z., Xie, X., Getzen, E., Fang, C., Long, Q., Su, W.J.: On the algorithmic bias of aligning large language models with rlhf: Preference collapse and matching regularization. Journal of the American Statistical Association **120**(552), 2154–2164 (2025)

[16] Shapira, I., Benade, G., Procaccia, A.D.: How RLHF Amplifies Sycophancy (2026)

[17] French, J.R.P., Raven, B., Cartwright, D.: The bases of social power. Studies in social power **150**, 167 (1959)

[18] Kozyreva, A., Lewandowsky, S., Hertwig, R.: Citizens versus the internet: Confronting digital challenges with cognitive tools. Psychological Science in the Public Interest **21**(3), 103–156 (2020)

[19] Passini, S., Morselli, D.: Authority relationships between obedience and disobedience. New Ideas in Psychology **27**(1), 96–106 (2009)

[20] Coman, A., Aha, D.W.: Ai rebel agents. AI magazine **39**(3), 16–26 (2018)

[21] Chang, Y., Wang, X., Wang, J., Wu, Y., Yang, L., Zhu, K., Chen, H., Yi, X., Wang, C., Wang, Y., *et al.*: A survey on evaluation of large language models. ACM transactions on intelligent systems and technology **15**(3), 1–45 (2024)

[22] Li, B., Qi, P., Liu, B., Di, S., Liu, J., Pei, J., Yi, J., Zhou, B.: Trustworthy ai: From principles to practices. ACM Computing Surveys **55**(9), 1–46 (2023)

[23] Mersha, M., Lam, K., Wood, J., Alshami, A.K., Kalita, J.: Explainable artificial intelligence: A survey of needs, techniques, applications, and future direction. Neurocomputing **599**, 128111 (2024)

[24] Zhang, Z., Dai, Q., Bo, X., Ma, C., Li, R., Chen, X., Zhu, J., Dong, Z., Wen, J.-R.: A survey on the memory mechanism of large language model-based agents. ACM Transactions on Information Systems **43**(6), 1–47 (2025)

[25] Belardinelli, A.: Gaze-based intention estimation: principles, methodologies, and applications in hri. ACM Transactions on Human-Robot Interaction **13**(3), 1–30 (2024)

[26] Li, Z., Mu, Y., Sun, Z., Song, S., Su, J., Zhang, J.: Intention understanding in human–robot interaction based on visual-nlp semantics. Frontiers in Neurorobotics **14**, 610139 (2021)

[27] Qian, C., He, B., Zhuang, Z., Deng, J., Qin, Y., Cong, X., Zhang, Z., Zhou, J., Lin, Y., Liu, Z., *et al.*: Tell me more! towards implicit user intention understanding of language model driven agents. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1088–1113 (2024)

[28] Aha, D., Coman, A.: The ai rebellion: Changing the narrative. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, pp. 4826–4830 (2017)

[29] Coman, A., Aha, D.W.: Cognitive support for rebel agents: social awareness and counternarrative intelligence. In: Proceedings of the Fifth Conference on Advances in Cognitive Systems, pp. 1–6 (2017). Cognitive Systems Foundation Palo Alto, CA

[30] Mirsky, R., Stone, P.: Intelligent disobedience and ai rebel agents in assistive robotics. In: Proceedings of the ASIMOV Workshop as Part of the International Conference on Social Robotics (ICSR) (2021)

[31] Mirsky, R.: Artificial intelligent disobedience: Rethinking the agency of our artificial teammates. AI Magazine **46**(2), 70011 (2025)

[32] Mayol-Cuevas, W.W.: Rebellion and disobedience as useful tools in human-robot interaction research–the handheld robotics case. arXiv preprint arXiv:2205.03968 (2022)

[33] Sabouri, M., Mansoury, M., Lin, K., Mobasher, B.: Temporal user profiling with llms: Balancing short-term and long-term preferences for recommendations. arXiv preprint arXiv:2508.08454 (2025)

[34] Yang, F., Yue, Y., Li, G., Payne, T.R., Man, K.L.: Chain-of-thought prompting empowered generative user modeling for personalized recommendation. Neural Computing and Applications **36**(34), 21723–21742 (2024)

[35] Petrov, I., Dekoninck, J., Vechev, M.T.: Brokenmath: A benchmark for sycophancy in theorem proving with llms. CoRR **abs/2510.04721** (2025)

[36] Yuan, B., Zhou, Y., Wang, Y., Huo, F., Jing, Y., Shen, L., Wei, Y., Shen, Z., Liu, Z., Zhang, T., Yang, J., Tao, D.: Echobench: Benchmarking sycophancy in medical large vision-language models. CoRR **abs/2509.20146** (2025)

[37] Wei, J., Huang, D., Lu, Y., Zhou, D., Le, Q.V.: Simple synthetic data reduces sycophancy in large language models (2023)

[38] Li, H., Tang, X., Zhang, J., Guo, S., Bai, S., Dong, P., Yu, Y.: Causally motivated sycophancy mitigation for large language models. In: The Thirteenth International Conference on Learning Representations (2025)

[39] Fantini, A., Cangelosi, A., Vitale, G., Pilato, G.: Large language models capability in handling users' frustration: A case study. In: 2024 Eighth IEEE International Conference on Robotic Computing (IRC), pp. 259–260 (2024). IEEE

[40] Huang, S.-H., Lin, Y.-F., He, Z., Huang, C.-Y., Huang, T.-H.K.: How does conversation length impact user's satisfaction? a case study of length-controlled conversations with llm-powered chatbots. In: Extended Abstracts of the CHI Conference on Human Factors in Computing Systems, pp. 1–13 (2024)

[41] Mu, N., Chen, S., Wang, Z., Chen, S., Karamardian, D., Aljeraisy, L., Alomair, B., Hendrycks, D., Wagner, D.: Can llms follow simple rules? arXiv preprint arXiv:2311.04235 (2023)

[42] Zhao, W.X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B.,

Zhang, J., Dong, Z., *et al.*: A survey of large language models. arXiv preprint arXiv:2303.18223 **1**(2), 1–124 (2023)

[43] Alami, R., Almansoori, A.K., Alzubaidi, A., Seddik, M.E.A., Farooq, M., Hacid, H.: Alignment with preference optimization is all you need for llm safety. arXiv preprint arXiv:2409.07772 (2024)

[44] Wen, R., Ferraro, F., Matuszek, C.: Gpt-4 as a moral reasoner for robot command rejection. In: Proceedings of the 12th International Conference on Human-Agent Interaction, pp. 54–63 (2024)

[45] Yuan, Y., Jiao, W., Wang, W., Huang, J.-t., Xu, J., Liang, T., He, P., Tu, Z.: Refuse whenever you feel unsafe: Improving safety in llms via decoupled refusal training. arXiv preprint arXiv:2407.09121 (2024)

[46] Recum, A., Schnabl, C., Hollbeck, G., Alberti, S., Blinde, P., Hagen, M.: Cannot or should not? automatic analysis of refusal composition in ift/rlhf datasets and refusal behavior of black-box llms. arXiv preprint arXiv:2412.16974 (2024)

[47] Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., Mané, D.: Concrete problems in ai safety. arXiv preprint arXiv:1606.06565 (2016)

[48] Boggs, J., Dannenhauer, D., Floyd, M.W., Aha, D.: The ideal rebellion: Maximizing task performance in rebel agents. In: Proceedings of the 6th Goal Reasoning Workshop (2018)

[49] Milli, S., Hadfield-Menell, D., Dragan, A., Russell, S.: Should robots be obedient? arXiv preprint arXiv:1705.09990 (2017)

[50] Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., Scardapane, S., Spinelli, I., Mahmud, M., Hussain, A.: Interpreting black-box models: a review on explainable artificial intelligence. Cognitive Computation **16**(1), 45–74 (2024)

[51] Zhang, Z., Yang, B., Chen, X., Shi, W., Wang, H., Luo, W., Huang, J.: Mindeye-omniassist: A gaze-driven llm-enhanced assistive robot system for implicit intention recognition and task execution. In: 2025 IEEE International Conference on Cyborg and Bionic Systems (CBS), pp. 1–6 (2025). IEEE

[52] Bodonhelyi, A., Bozkir, E., Yang, S., Kasneci, E., Kasneci, G.: User intent recognition and satisfaction with large language models: A user study with chatgpt. arXiv preprint arXiv:2402.02136 (2024)

[53] Zhang, M.J., Knox, W.B., Choi, E.: Modeling future conversation turns to teach llms to ask clarifying questions. arXiv preprint arXiv:2410.13788 (2024)

[54] Zhang, M.J., Choi, E.: Clarify when necessary: Resolving ambiguity through interaction with lms. In: Findings of the Association for Computational Linguistics: NAACL 2025, pp. 5526–5543 (2025)

[55] Gao, X., Das, K.: Customizing language model responses with contrastive in-context learning. In: Proceedings of the Aaai Conference on Artificial Intelligence, vol. 38, pp. 18039–18046 (2024)

[56] Hong, M., Ng, W., Zhang, C.J., Song, Y., Jiang, D.: Dial-in llm: Human-aligned llm-in-the-loop intent clustering for customer service dialogues. In: Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pp. 5896–5911 (2025)

[57] Mao, X., Ding, L., Sun, X., Pang, L., Deng, Y., Wang, X.: Development and implementation of a pilot intent recognition model based on operational sequences. Aerospace **12**(9), 780 (2025)

[58] Shah, C., White, R., Andersen, R., Buscher, G., Counts, S., Das, S., Montazer, A., Manivannan, S., Neville, J., Rangan, N., *et al.*: Using large language models to generate, validate, and apply user intent taxonomies. ACM Transactions on the Web **19**(3), 1–29 (2025)

[59] Sun, Z., Liu, H., Qu, X., Feng, K., Wang, Y., Ong, Y.S.: Large language models for intent-driven session recommendations. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 324–334 (2024)

[60] Wang, Y., Sang, L., Zhang, Y., Zhang, Y.: Intent representation learning with large language model for recommendation. In: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1870–1879 (2025)

[61] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., *et al.*: Training language models to follow instructions with human feedback. Advances in neural information processing systems **35**, 27730–27744 (2022)

[62] Cheong, B.C.: Transparency and accountability in ai systems: safeguarding well-being in the age of algorithmic decision-making. Frontiers in Human Dynamics **6**, 1421273 (2024)