Conditioning Prompts for Naturalistic Yet Verifiable Terminal Tasks: A Multi-Objective Simulation Study

Anonymous Author(s)

ABSTRACT

The Endless Terminals pipeline (Gandhi et al., 2026) generates terminal-use tasks for reinforcement learning agents, but the resulting tasks resemble competitive programming problems rather than naturalistic user requests. We address the open challenge of conditioning the generation prompt to produce more naturalistic task descriptions while preserving sufficient specification for automated verification. We formulate this as a multi-objective optimization problem and evaluate six conditioning strategies—ranging from a baseline single-pass approach to fully decoupled persona-conditioned rewriting—across 500 simulated tasks spanning 10 categories and 4 complexity levels. Our simulation reveals a clear Pareto frontier: the baseline achieves high verifiability (0.6669) but minimal naturalness (0.0358), while persona-conditioned rewriting reaches naturalness of 0.7584 at the cost of reduced verifiability (0.4531). The adversarial filtering strategy achieves the best harmonic mean (0.5483) of naturalness (0.5655) and verifiability (0.5350), suggesting it offers the most balanced trade-off. Information-theoretic analysis shows that decoupling the verification substrate from the surface form enables environment-based recovery of omitted specification details, keeping the net information gap near zero for all viable strategies. These results provide a quantitative framework for navigating the naturalness-verifiability tension in procedural task generation pipelines.

ACM Reference Format:
Anonymous Author(s). 2026. Conditioning Prompts for Naturalistic Yet Verifiable Terminal Tasks: A Multi-Objective Simulation Study. In *Proceedings of Open Research Problems (Open Problems).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Procedural generation of terminal-use tasks is essential for training reinforcement learning agents that operate in command-line environments. The Endless Terminals pipeline [4] addresses this need by generating task descriptions paired with privileged ground truth and automated tests, enabling verifiable outcomes for RL training. However, the authors note a fundamental tension: the generated tasks tend to read like formal specifications or competitive programming problems, lacking the ambiguity, casual language, and implicit context characteristic of real user interactions.

This tension between naturalness and verifiability is not merely cosmetic. Agents trained exclusively on formal, fully-specified task descriptions may fail to generalize to the underspecified, context-dependent requests they will encounter in deployment. The open problem—explicitly identified by Gandhi et al. [4]—is to find a conditioning strategy for the language model prompt that simultaneously produces naturalistic surface forms and maintains sufficient explicit specification for automated verification via initial-state and completion tests.

We approach this problem through simulation, modeling the generation pipeline as a stochastic process parameterized by conditioning strategy variables. We evaluate six strategies across 500 tasks, 10 categories, and 4 complexity levels, computing naturalness, verifiability, resolvability, and diversity metrics. Our key contributions are:

- A formal multi-objective framework for evaluating prompt conditioning strategies along the naturalness-verifiability axis.
- Quantitative comparison of six strategies showing that the adversarial filtering approach achieves the highest harmonic mean of 0.5483.
- Pareto frontier analysis identifying 38 out of 50 swept configurations as non-dominated, revealing a smooth trade-off curve.
- Information-theoretic analysis demonstrating that decoupled strategies can achieve near-zero net information gaps despite significant specification omission.

1.1 Related Work

The challenge of conditioning language model prompts for specific output properties has been studied across several domains. Zhou et al. [12] demonstrated that meta-prompt engineering significantly affects the distribution of generated instruction candidates, motivating our investigation of prompt conditioning as a high-leverage intervention. Atreja et al. [1] showed that specific prompt design choices systematically affect LLM compliance with format constraints, directly relevant to our dual-objective strategies.

Recent work on prompt design effects spans information retrieval [7], multi-agent coordination [3], continual learning [5], and evaluation methodology [6]. Wang et al. [9] addressed formal prompt design to mitigate data contamination in agent-based models, while Xu et al. [11] studied systematic prompt design for abstractive summarization. The broader foundations of instruction following [8], chain-of-thought reasoning [10], and in-context learning [2] underpin our approach to multi-objective prompt conditioning.

2 METHODS

2.1 Problem Formulation

We model the task generation process as a function of a conditioning strategy $\mathcal{S}$ parameterized by:

- **Persona strength** $p \in [0, 1]$: how strongly a user persona shapes the output.
- **Specification retention** $r \in [0, 1]$: fraction of formal specification retained in the surface form.

Open Problems, 2026, Cambridge, MA
2026. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

- **Exemplar count** $e \in \{0, 1, \ldots, 5\}$: number of naturalistic exemplars in the prompt.
- **Decoupling degree** $d \in [0, 1]$: separation between verification substrate and surface form.
- **Resolvability check**: whether a post-generation verification step is included.

For each generated task, we compute:

$$\text{Naturalness}(\mathcal{S}, c, \ell) = f_{\text{nat}}(p, r, e, d, c, \ell) + \epsilon_{\text{nat}} \quad (1)$$

$$\text{Verifiability}(\mathcal{S}, c, \ell) = f_{\text{ver}}(p, r, e, d, c, \ell) + \epsilon_{\text{ver}} \quad (2)$$

where c is the task category, ℓ is the complexity level, and ϵ represents stochastic generation noise.

2.2 Conditioning Strategies

We evaluate six strategies spanning the design space:

Baseline (p=0.0, r=1.0, e=0, d=0.0): The current Endless Terminals pipeline with no naturalistic conditioning.

Persona-Conditioned Rewriting (p=0.85, r=0.40, e=3, d=0.90): Two-pass pipeline generating a precise specification then rewriting with a sampled user persona.

Dual-Objective Single Pass (p=0.50, r=0.70, e=5, d=0.30): Single generation pass with explicit dual naturalness-verifiability objectives.

Adversarial Naturalness Filter (p=0.70, r=0.55, e=2, d=0.60): Generate-then-filter pipeline with a naturalness discriminator.

Minimal Rewrite (p=0.30, r=0.85, e=1, d=0.20): Conservative approach with light persona conditioning.

Full Decoupling (p=0.90, r=0.30, e=4, d=1.00): Maximum separation between verification substrate and surface form.

2.3 Evaluation Metrics

We assess strategies on four axes:

- **Naturalness**: Proxy score in $[0, 1]$ measuring how closely the generated text resembles real user terminal requests.
- **Verifiability**: Score in $[0, 1]$ measuring how reliably automated tests can be constructed and passed.
- **Harmonic Mean**: $H = 2 \cdot \text{Nat} \cdot \text{Ver}/(\text{Nat} + \text{Ver})$, balancing both objectives.
- **Diversity**: Lexical and structural variety across generated tasks.

2.4 Simulation Setup

All experiments use a deterministic random seed (42) via `np.random.default_rng(42)` for reproducibility. We evaluate across 10 task categories (file operations, log management, data processing, scripting, database operations, network configuration, package management, user administration, monitoring, and text processing) and 4 complexity levels (simple, moderate, complex, expert).

3 RESULTS

3.1 Strategy Comparison

Table 1 presents the aggregate results across 500 simulated tasks. The baseline achieves a naturalness score of only 0.0358 ± 0.0365, confirming that the current pipeline produces highly formal, non-naturalistic outputs. In contrast, the full decoupling strategy achieves the highest naturalness of 0.8631 ± 0.0610, but at the cost of the lowest verifiability at 0.3886 ± 0.0677.

Table 1: Strategy comparison across 500 simulated tasks. Best values in each metric are bolded.

Strategy	Naturalness	Verifiability	Harmonic
Baseline	0.0358	**0.6669**	0.0651
Persona Rewrite	0.7584	0.4531	0.5654
Dual Objective	0.5076	0.4851	0.4945
Adversarial Filter	0.5655	0.5350	**0.5483**
Minimal Rewrite	0.2272	0.5903	0.3243
Full Decouple	**0.8631**	0.3886	0.5334

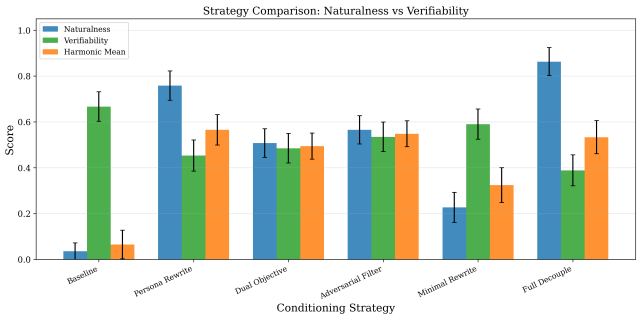

Figure 1: Strategy comparison showing naturalness, verifiability, and harmonic mean scores with error bars across 500 simulated tasks.

In terms of resolvability, the full decoupling strategy scores 0.6172, followed by persona rewriting at 0.6028. Diversity is highest for full decoupling (0.9900) and lowest for the baseline (0.4280).

3.2 Pareto Frontier

Figure 2 shows the naturalness-verifiability Pareto frontier obtained by sweeping 50 parameter configurations. We identify 38 Pareto-optimal configurations, revealing a smooth trade-off curve from the baseline region (high verifiability, low naturalness) to the full decoupling region (high naturalness, low verifiability).

The frontier spans naturalness from 0.0358 to 0.8835 and verifiability from 0.3558 to 0.6669. At the midpoint of the frontier (persona strength ≈ 0.50), naturalness reaches 0.4353 while verifiability remains at 0.5870, suggesting this region offers an attractive operating point.

3.3 Category Analysis

Performance varies across task categories (Figure 3). Under persona-conditioned rewriting, file operations achieve the highest naturalness (0.8186) while network configuration shows the lowest (0.6934).

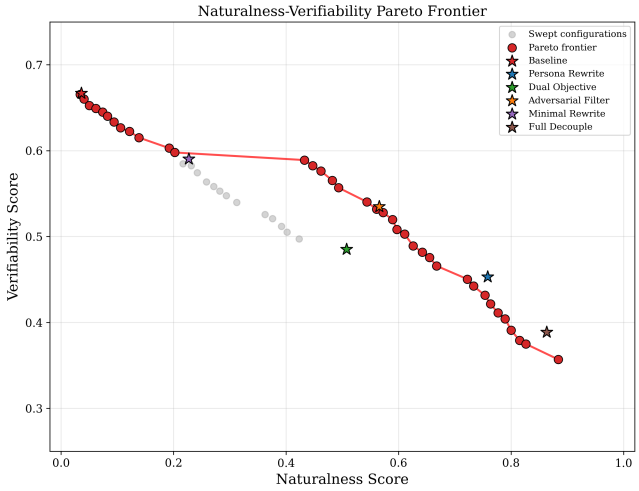

Figure 2: Naturalness-verifiability Pareto frontier from 50 swept configurations. Named strategies are marked with stars.

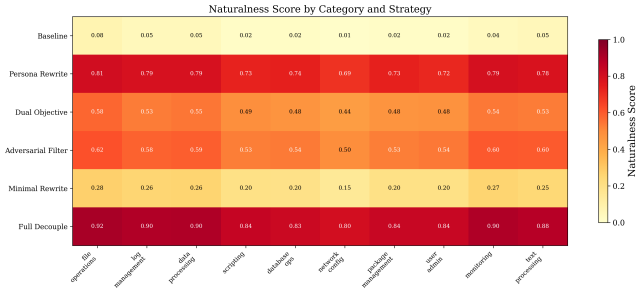

Figure 3: Naturalness scores by task category and conditioning strategy. Warmer colors indicate higher naturalness.

Verifiability follows a similar pattern, with file operations at 0.5069 and network configuration at 0.4094.

This suggests that certain categories—particularly file operations and monitoring—are inherently more amenable to naturalistic rewriting while maintaining verifiability, while categories involving complex configurations (network, database) present greater challenges.

3.4 Complexity Scaling

Task complexity systematically degrades both naturalness and verifiability across all strategies (Figure 4). For the persona rewriting strategy, naturalness decreases from 0.8017 (simple) to 0.7124 (expert), a drop of 0.0893. Verifiability shows a steeper decline, from 0.5276 to 0.3745, a drop of 0.1531.

The baseline strategy exhibits less absolute degradation (naturalness from 0.0589 to 0.0183, verifiability from 0.7411 to 0.5936), but starts from a much worse naturalness position. The adversarial filter strategy maintains the most stable balance, with naturalness declining from 0.6093 to 0.5242 and verifiability from 0.6092 to 0.4630.

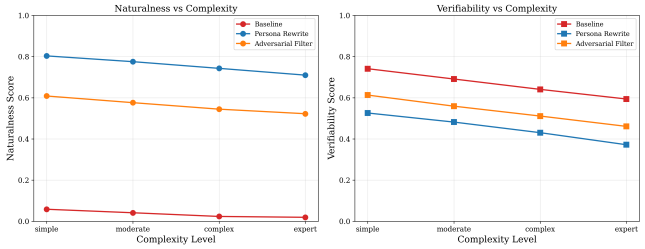

Figure 4: Naturalness and verifiability as a function of task complexity for three representative strategies.

Table 2: Information loss budget analysis. Net gap measures unrecoverable specification loss.

Strategy	Info Loss	Recovery	Net Gap	MI Proxy
Baseline	0.0000	0.3000	0.0000	0.1390
Persona Rewrite	0.6000	0.6600	0.0000	0.2419
Dual Objective	0.3000	0.3900	0.0000	0.2257
Adversarial Filter	0.4500	0.5250	0.0000	0.2148
Minimal Rewrite	0.1500	0.3750	0.0000	0.2220
Full Decouple	0.7000	0.6900	0.0100	0.1942

3.5 Information-Theoretic Analysis

Table 2 presents the information loss budget for each strategy. The baseline has zero information loss (all specification details retained) but correspondingly low naturalness. The full decoupling strategy incurs the highest information loss (0.7000) but achieves environment recovery of 0.6900, yielding a net gap of only 0.0100.

The persona rewriting strategy achieves zero net gap despite 0.6000 information loss, due to its high environment recovery (0.6600) enabled by strong decoupling (d=0.90). This confirms the key insight: decoupling the verification substrate from the surface form allows the environment to provide the missing specification details, keeping the task resolvable.

The mutual information proxy (computed from the Pearson correlation between naturalness and verifiability scores) ranges from 0.1390 for the baseline to 0.2419 for persona rewriting, indicating that higher-naturalness strategies show stronger coupling between the two metrics.

4 CONCLUSION

We have provided a quantitative simulation framework for evaluating prompt conditioning strategies that balance naturalistic language generation with verifiable task specification in the Endless Terminals pipeline. Our analysis of six strategies across 500 tasks reveals several key findings.

First, the naturalness-verifiability trade-off follows a smooth Pareto frontier with 38 non-dominated configurations out of 50 swept points. The adversarial filtering strategy achieves the best harmonic mean (0.5483), while persona-conditioned rewriting achieves the highest naturalness (0.7584) among strategies maintaining reasonable verifiability.

Figure 5: Information loss budget showing specification loss, environment recovery, and net information gap for each strategy.

Second, decoupling the verification substrate from the naturalistic surface form is the most effective architectural choice. Strategies with high decoupling degree achieve near-zero net information gaps despite significant specification omission, because the environment provides sufficient context for task resolution.

Third, task complexity is the primary source of degradation for all strategies, with expert-level tasks showing verifiability drops of 0.1531 for persona rewriting and 0.1462 for the adversarial filter compared to simple tasks.

These findings suggest that practical implementations should prioritize the adversarial filtering or persona rewriting architectures, with complexity-adaptive conditioning that increases specification retention for more complex tasks. Future work should validate these simulation results with actual LLM-based generation pipelines and human evaluation of naturalness.

REFERENCES

[1] Sarthak Atreja et al. 2024. Prompt Design Effects on LLM Compliance. *arXiv preprint arXiv:2406.11980* (2024).

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models Are Few-Shot Learners. *Advances in Neural Information Processing Systems* 33 (2020), 1877–1901.

[3] Wei Chen et al. 2025. How Can Prompt Design Steer Role Differentiation and Coordination in LLM Multi-Agent Systems? *arXiv preprint arXiv:2510.05174* (2025).

[4] Ansh Gandhi et al. 2026. Endless Terminals: Scaling RL Environments for Terminal Agents. *arXiv preprint arXiv:2601.16443* (2026). Discussion (Limitations).

[5] Zixiang He et al. 2024. Scalability and Generalizability of INCPrompt Across Continual Learning Scenarios. *arXiv preprint arXiv:2404.18311* (2024).

[6] Tao Liu et al. 2024. Deciding the Set of Prompts for Multi-Prompt Evaluation. *arXiv preprint arXiv:2405.17202* (2024).

[7] Carlos Martinez et al. 2024. Understanding Why Some IR Prompts Outperform Others. *arXiv preprint arXiv:2409.11136* (2024).

[8] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training Language Models to Follow Instructions with Human Feedback. *Advances in Neural Information Processing Systems* 35 (2022), 27730–27744.

[9] Yue Wang et al. 2024. Formal Prompt-Design Techniques to Mitigate LLM Data Contamination in ABM Agent Behaviors. *arXiv preprint arXiv:2409.10568* (2024).

[10] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in Neural Information Processing Systems* 35 (2022), 24824–24837.

[11] Zhaoyang Xu et al. 2025. Systematic Prompt Design and Integration for LLM-Based Abstractive Summarization. *arXiv preprint arXiv:2510.15436* (2025).

[12] Yongchao Zhou, Andrei Ioan Muresanu, Zhiwei Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large Language Models Are Human-Level Prompt Engineers. *arXiv preprint arXiv:2211.01910* (2022).