OUTPUT HOMOGENIZATION IS TASK DEPENDENT

Anonymous authors
Paper under double-blind review

ABSTRACT

A large language model can be less helpful if it exhibits output response homogenization. But whether two responses are considered homogeneous, and whether such homogenization is problematic, both depend on the task category. For instance, in objective math tasks, we often expect no variation in the final answer but anticipate variation in the problem-solving strategy. Whereas, for creative writing tasks, we may expect variation in key narrative components (e.g. plot, genre, setting, etc), beyond the vocabulary or embedding diversity produced by temperature-sampling. Previous work addressing output homogenization often fails to conceptualize diversity in a task-dependent way. We address this gap in the literature directly by making the following contributions. (1) We present a task taxonomy comprised of eight task categories that each have distinct concepts of output homogenization. (2) We introduce task-anchored functional diversity to better evaluate output homogenization. (3) We propose a task-anchored sampling technique that increases functional diversity for task categories where homogenization is undesired, while preserving it where it is desired. (4) We challenge the perceived existence of a diversity-quality trade-off by increasing functional diversity while maintaining response quality. Overall, we demonstrate how task dependence improves the evaluation and mitigation of output homogenization.

1 INTRODUCTION

Large language models (LLMs) often generate homogeneous outputs, but whether this is problematic depends on the specific task. Suppose a user asks for a joke and a model always responds with a "knock-knock" joke; such homogenization undermines the model's creative utility. By contrast, for tasks with verifiable solutions such as solving a math problem, consistency is not only acceptable but desirable, although variation in the explanation or problem-solving approach may still add value. Our central claim is that the implications of homogenization are task-dependent, and, therefore both the evaluation and mitigation of homogenization should also be task-dependent.

Existing approaches to reducing output homogenization rarely take task dependence into account. Several recent works propose methods that promote diversity in the alignment process or when sampling outputs at inference-time. However, these studies often fail to conceptualize diversity in a task-specific way. For example, some methods aim to increase token-level entropy or embedding-space variation in alignment (Chung et al., 2025; Lanchantin et al., 2025a; Slocum et al., 2025; Li et al., 2025), while others promote diversity of viewpoints and perspectives when sampling multiple outputs (Wang et al., 2025b; Zhang et al., 2025a;b). Without a task-dependent approach, such methods may (1) fail to encourage diversity that is meaningful for a task, and/or (2) undesirably reduce homogenization in tasks where it is desired. We address this gap in the literature directly.

We introduce a task-anchored framework to evaluate and mitigate output homogenization. We build on the notion of *functional diversity* (Zhang et al., 2025b; Shypula et al., 2025), which asks whether a user would perceive two responses as meaningfully different for a given task. We argue that LLMs should be able to conceptualize functional diversity based on the task category. Consider the stakes: if a model wrongly conceptualizes functional diversity for a task that mimics an encyclopedia inquiry, the model could misrepresent historical events in an attempt to naively reduce homogenization. Conversely, if a model wrongly conceptualizes functional diversity for a creative writing task, the model might repeat the same story arc no matter how many times a user asks the model to tell a story. We argue that task dependence should be incorporated into the way we address homogenization. Our contributions are as follows.

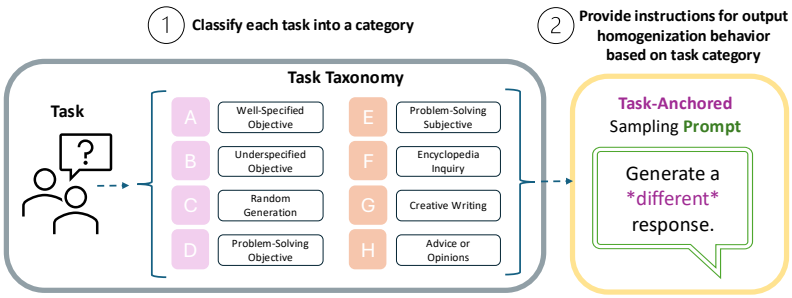


Figure 1: **Our task-anchored sampling technique for improving output homogenization.** The first step is to classify each input prompt into a task category. Note that if a prompt falls outside of the taxonomy, our approach can generalize to new task categories, or the model may resume its default behavior. The second step is task-anchored sampling where we clarify the concept of functional diversity in the instruction to generate "different" responses at inference-time. The taxonomy is outlined in § 3.1 and our task-anchored sampling technique is detailed in § 3.3.

1. We present a *task taxonomy* of eight task categories each with distinct conceptualizations of output homogenization (§ 3.1). Our taxonomy extends the common distinction between verifiable and non-verifiable tasks. By introducing a more granular categorization, we aim to capture subtle nuances that may be overlooked if output homogenization is interpreted solely by the model. Although not exhaustive, our taxonomy effectively anchors task dependence. Note that if a prompt falls outside of our taxonomy, our approach can generalize to new task categories, or the model can resume its standard or default behavior.
2. We introduce *task-anchored the functional diversity* to better evaluate output homogenization (§ 3.2). In our experiments, we compare to more general diversity metrics which are not task dependent (vocabulary and embedding differences). The results show that these general metrics fail to capture task-dependent diversity. Our task-anchored metric offers an alternative evaluation approach for future studies of output homogenization.
3. We propose a *task-anchored sampling technique* to increase functional diversity (§ 3.3), improving on previous sampling methods to promote diversity (Zhang et al., 2025a;b). Figure 1 offers a high-level illustration of our approach. We leverage our taxonomy to instruct models with task-dependent notions of diversity. Our approach increases functional diversity for task categories where homogenization is undesired, while preserving homogenization where it is desired (Figure 2).
4. We challenge the *perceived existence of a diversity-quality trade-off* (a common narrative in the literature) by adopting a quality measure (Lin et al., 2025; Wei et al., 2025) that also accounts for task-specific variations in quality. Figure 3 shows that the diversity-quality tradeoff prevalent in previous studies may simply be the result of mis-conceptualizing both diversity and quality. Our evaluation framework corrects both.

2 Background

2.1 Homogenization in Aligned Models

Task Dependence Several works show that aligned LLMs exhibit output homogenization across a variety of tasks, such as creative writing (Moon, 2024; Wu et al., 2025), political discussions (Durmus et al., 2023; Santurkar et al., 2023), and math problem-solving (Slocum et al., 2025). Zhang et al. (2024; 2025b) further show that in question-answering, models often produce the same answer, even when the question is underspecified and multiple valid answers exist. These studies often evaluate homogenization in specific task domains, suggesting that problematic notions of homogenization are task-dependent. Our proposed taxonomy compares a variety of these task categories.

Representation Concerns In certain tasks, homogeneous outputs may raise concerns about *representation*. The literature on pluralistic alignment (Sorensen et al., 2024; Chen et al., 2024; Zhang et al., 2025a) highlights representational harms, particularly when users seek advice or opinions from

LLMs. However, pluralistic alignment discussions tend to operate in contexts where representation or diversity is presumed to be desirable. These discussions should recognize the task dependent nature of pluralism or representation, as we discuss in this work.

Causes There are many causes of output homogenization, such as limited diversity in training data and model design choices (Zhang et al., 2025a; Fazelpour & Fleisher, 2025). In particular, the alignment process is well-known to amplify homogenization in LLM outputs (Kirk et al., 2024; Lanchantin et al., 2025a). Models are typically aligned using methods such as Reinforcement Learning with Human Feedback (RLHF) (Ziegler et al., 2019) or Direct Preference Optimization (DPO) (Rafailov et al., 2023). These methods involve training on a dataset of pairwise preferences $\{(x, y^+, y^-)\}$, where x is a prompt, y^+ is a preferred response, and y^- is a dispreferred response. When there are conflicting preferences in the training data, such that both $y^+ \succ y^- \,|\, x$ and $y^- \succ y^+ \,|\, x$ coexist, the RLHF and DPO objectives implicitly reward putting all sequence-level probability on the majority preference (Slocum et al., 2025; Yao et al., 2025). Preference pairs with larger semantic differences also exert a stronger influence on the behavior of the aligned model (Chung et al., 2025; Shen et al., 2024). While this line of research is important, the present work focuses on how output homogenization should be conceptualized, not why it occurs.

Outcome Homogenization Another type of homogenization occurs when multiple models produce similar outputs (Kim et al., 2025; Wenger & Kenett, 2025). For example, outcome homogenization in decision-making refers to when individuals receive similar decisions from separate AI models (Bommasani et al., 2022; Jain et al., 2024b). In this work, we focus on the single-model case and do not deal directly with homogenization across different models. But reducing homogenization within a single model likely affects homogenization across models (Jain et al., 2024a).

2.2 Diversity-Promoting Methods

Alignment Methods A growing body of literature explores methods to reduce homogenization in aligned LLMs. Several studies propose modifying the alignment process, either by altering the construction of preference datasets or by adjusting alignment objectives (Lanchantin et al., 2025a; Slocum et al., 2025; Chung et al., 2025). All of these methods substantially increase diversity during alignment, as measured by token-level entropy or embedding-space variation. However, we highlight how these metrics may not capture meaningful, task-dependent notions of diversity.

Inference-Time Methods While most evaluations of homogenization examine temperature-sampled outputs, a few studies explore prompt-based strategies to explicitly sample diverse outputs at inference-time. For example, Zhang et al. (2025a) use a *system prompt* that explicitly tells LLMs to generate k responses in a single output that represent "diverse values." Zhang et al. (2025b) propose *in-context regeneration*, where models are prompted to produce a different response while retaining all previous responses in the conversation context. Other works also use implicit techniques, such as persona-based or multilingual prompting (Wang et al., 2025a;b). Our work improves inference-time methods by explicitly clarifying the notion of "diversity" in model instructions.

3 Framework

3.1 Task Taxonomy

We begin by outlining the task categories used in our *task-anchored* framework (Table 1). Each of the 8 categories are distinguished by their conceptualization of output homogenization. The first four categories (A, B, C, D) capture prompts that elicit *verifiable* solution(s) that might be considered objective in nature, yet may still have more than one verifiable answer or explanation[1]. The second four categories (E, F, G, H) capture prompts that elicit more open-ended solution(s) that may be considered *non-verifiable* or only have partially verifiable components. Our taxonomy offers a more granular categorization of reward verifiability than the binary distinction (verifiable vs non-verifiable) present in the literature (Lambert et al., 2024; Lanchantin et al., 2025b). Our task

[1]We view objective/subjective and underspecified/well-specified not as discrete categories, but as spectra that task categories span. While the terms "objective" and "subjective" invite philosophical debate, such discussions are beyond this paper's scope. Likewise, we use "underspecified" and "well-specified" loosely, as clarifying terms, not precise definitions of the answer spaces.

Task Category	Task Definition	Example Task	Functional Diversity	Reward Type
A. Well-Specified Singular Objective	Tasks with a single verifiable correct answer	What is the largest Spanish-speaking country?	None	Verifiable
B. Underspecified Singular Objective	Tasks with multiple verifiable correct answers	Name one Spanish-speaking country.	Different correct answers	Verifiable (Multiple)
C. Random Generation	Tasks that involve randomizing over a set of options	Roll a make-believe 6-sided dice.	Different pseudo-random options	Verifiable (Multiple)
D. Problem-Solving Objective	Tasks to solve a problem with a verifiable solution	How many positive whole-number divisors does 196 have?	Different solution strategies	Verifiable
E. Problem Solving or Design Subjective	Tasks to solve a problem with many verifiable solutions	Design a room that minimizes energy consumption while maintaining comfort.	Different solutions	Partially Verifiable
F. Encyclopedia Inquiry	Tasks to provide information about real world societies, traditions, events where there are credible references	Why is Isaac Newton famous?	Different factual perspectives	Partially Verifiable
G. Creative Writing	Tasks that require creative expression	Tell me a riddle.	Different creative elements	Non-Verifiable
H. Advice or Opinions	Tasks that solicit advice, opinions or feedback on specific topics/scenarios	What is a good Mother's day gift?	Different perspectives or views	Non-Verifiable

Table 1: **Taxonomy of Task Categories.** Categories are distinguished by their concept of functional diversity. While non-exhaustive, task categories are a useful mechanism to clarify what elements of responses should be homogeneous and what meaningful elements of responses may vary.

categories capture many real-world LLM use cases identified in recent work (Tamkin et al., 2024; Chatterji et al., 2025). Though our taxonomy is non-exhaustive, we illustrate how task categories are a useful mechanism to appropriately conceptualize output homogenization.

Each task category corresponds to a degree of *reward verifiability*. Categories help clarify: what elements of responses are verifiable and should remain homogeneous for a given task? At one extreme, *Well-Specified Objective* (category A) captures prompts that have only one verifiable answer. Whereas, *Creative Writing* (category G) captures prompts that have an infinite number of non-verifiable answers. When we consider different types of verifiability, we realize that tasks may allow for multiple verifiable answers (categories B, C & E) as well as multiple explanations for those answers (categories D & E). For instance, *Problem Solving Objective* (category D) captures tasks that have a single verifiable answer, but multiple explanations available for arriving at that verifiable answer. Subjective problem-solving tasks (category E) may have multiple verifiable answers, as well as multiple explanations for those answers.

Each task category further corresponds to a specific type of response variation or *functional diversity*. Previous works define two responses to be functionally diverse if a user would perceive them to be meaningfully different (Zhang et al., 2025b; Shypula et al., 2025). Our task categories clarify: in what ways could responses be meaningfully different for a given task? For example, in *Problem Solving Objective* (category D) tasks, functional diversity is in solution strategies, not in the final answer. Whereas, in *Creative Writing* (category G), functional diversity is in the key creative elements (e.g. plot, genre, setting, etc.), not just in character names or vocabulary choices.

Functional diversity further depends on the level of *specification in the prompt*. For example, *Underspecified Singular Objective* (category B) may have multiple correct answers that could be generated, but the prompt does not specify a distribution over these answer options. An example task in this category is "Name one Spanish-speaking country." In this prompt, it might be acceptable for the model to over-index on the most popular countries, but the prompt does not specify. Compare this to *Random Generation Objective* (category C) where an example task is "Roll a make-believe 6-sided dice." The output distribution here is clearly specified by the prompt and not meant to be determined by the model. We aim to capture these nuances in how specified a task is. Ultimately, our taxonomy is simple yet powerful as a categorization that clarifies different types of reward verifiability and functional diversity that models may not inherently conceptualize on their own.

3.2 Evaluating Task-Anchored Diversity

Next, we formalize *task-anchored functional diversity*. We let $\mathcal{P}$ denote the set of possible input prompts or tasks and we let $\mathcal{Y}$ denote the set of possible outputs (e.g., sequences of tokens). We adopt the simple notation that a *language model* $\mathcal{M}$ is a stochastic function $\mathcal{M} : \mathcal{P} \to \mathcal{Y}$ that maps

each prompt $p \in \mathcal{P}$ to an output $y \in \mathcal{Y}$. Each prompt p is associated with a task category $c(p) \in \mathcal{T}$ as defined in Table 1. For a given prompt p, we assume $d(p, y_a, y_b) \in [0, 1]$ to be a metric indicates whether two responses y_a and y_b differ. To specify *functional diversity*, we anchor the definition of $d(p, y_a, y_b)$ on the task category $c(p) \in \mathcal{T}$.

Definition 3.1 (Task Anchored Functional Diversity)**.** Given a prompt $p \in \mathcal{P}$ with associated task category $c(p) \in \mathcal{T}$, and two responses $y_a, y_b \in \mathcal{Y}$, the *task-anchored functional diversity* is

$$d(p, y_a, y_b) := \mathbb{1}_{c(p)}[y_a \neq y_b],$$

where $\mathbb{1}_{c(p)}[y_a \neq y_b]$ is an indicator function that returns 1 if y_a and y_b are functionally different with respect to the task category $c(p)$, and 0 otherwise.

For example, consider whether two responses are functionally diverse in the *Problem-Solving Objective* task (category D). Here, $d(p, y_a, y_b)$ represents whether responses have different solution strategies, and assumes the single verifiable answer to be the same. In practice, evaluating functional diversity requires human annotation or LLM-judges.

Response diversity can also be evaluated with general diversity metrics. By general, we mean that the metric does not reference or depend on a predefined task category. Previous studies adopt two common metrics. Vocabulary diversity quantifies the extent to which two responses use different vocabulary where higher values mean more unique words or less words shared. Embedding diversity measures the difference in semantic content according to cosine distance in an embedding vector space where higher values mean more semantic difference. We provide formal definitions for these metrics in Appendix A.4.

3.3 Promoting Task-Anchored Diversity

We introduce a *task-anchored sampling technique* which modifies existing prompt-based methods for promoting diversity (c.f. Figure 1). Prompt-based sampling strategies are inference-time methods to generate multiple responses. We focus on two existing methods: *system prompt sampling*, which generates multiple responses in a single generation (Zhang et al., 2025a), and *in-context regeneration*, which iteratively generates multiple responses (Zhang et al., 2025b). Both these methods instruct the model to generate "different" or "diverse" responses. We modify these methods by clarifying in the instruction what is meant by "different" or "diverse", using the functional diversity concepts in our task taxonomy (Table 2). To reduce homogenization at inference-time, the model could sample over these responses, or choose a response based on other alignment criteria.

4 Experiments

In this section, we operationalize our framework for evaluating and mitigating task-anchored output homogenization. We use prompts from benchmark datasets that cover the task categories in our taxonomy (Table 1). In our task-anchored sampling technique (c.f. Figure 1), models first classify the prompt into a task category. Based on this classification, we explicitly instruct models to generate *different* responses where the instruction clarifies the task-specific concept of functional diversity. For comparison, we also sample *different* responses without task clarity (temperature sampling and general prompt-based sampling). With these experiments, we explore the following questions:

1. Compared to general sampling strategies, to what extent does our task-anchored sampling technique improve functional diversity across task categories?
2. How well do general diversity metrics capture task-anchored functional diversity?
3. With improved diversity, does our task-anchored sampling technique decrease the quality of responses?

4.1 Experiment Details

Models We use three generative models: *GPT-4o*, *Claude-4-Sonnet*, and *Gemini-2.5-Flash*. These models are used in our evaluation to generate responses and separately as judges of task-anchored functional diversity (temperature 0). When reporting LLM-judge metrics, we average the outputs across these three models.

Method	Previous Works (No Task Dependence)	Problem-Solving Objective (Task-Anchored)	Creative Writing (Task-Anchored)
System Prompt	Generate {num_responses} responses that represent diverse values. (Zhang et al., 2025a)	The following problem has a single correct answer, but can be solved using different problem-solving strategies. Generate {num_responses} different solutions, each with a different problem-solving strategy.	The following prompt is asking for creative expression, so there are many possible subjective responses. Generate {num_responses} unique responses by varying the key creative elements such as tone, genre, point of view, theme, structure, etc. Each response should have different creative elements and reflect a distinct creative expression.
In-Context Regeneration	Can you generate a different answer? (Zhang et al., 2025b)	Can you solve the problem using a different strategy? The problem has a single correct answer, but can be solved using different problem-solving strategies.	Can you generate a new response with different creative elements? The prompt is asking for creative expression, so there are many possible subjective responses. Your new response should change the key creative elements such as tone, genre, point of view, theme, structure, etc.

Table 2: **Prompt-Based Sampling Strategies.** We modify prompt-based sampling methods in previous works (Zhang et al., 2025a;b) to promote task-anchored functional diversity.

Datasets We evaluate $n = 300$ prompts from a variety of datasets that achieve reasonable coverage[2] across the task categories in our taxonomy (see Appendix Table 3 for counts). The five datasets used in our evaluation are: *Community Alignment* (Zhang et al., 2025a), *MATH-500* (Lightman et al., 2023), *NoveltyBench* (Zhang et al., 2025b), *SimpleQA* (Wei et al., 2024), and *WildBench* (Lin et al., 2025). These datasets represent a mix of user-generated and curated prompts. Appendix A.1 includes additional details about each dataset and how we sampled prompts.

Sampling Strategies To evaluate homogenization over multiple responses, we compare three sampling strategies: temperature sampling, system prompt sampling, and in-context regeneration. For each sampling strategy, we sample 5 responses per prompt. For *temperature sampling*, we consider three temperature levels for each model based on its permitted range: low ($t = 0.0$), medium ($t = 1.0$ for GPT and Gemini, $t = 0.5$ for Claude), and high ($t = 2.0$ for GPT and Gemini, $t = 1.0$ for Claude). Recall that *system prompt* sampling refers to using a system prompt that instructs the model to generate a specified number of responses. The model is instructed to produce multiple responses separated by a delimiter, allowing them to be de-aggregated with regular expressions. *In-context regeneration* samples responses by iteratively prompting the model to generate a different response, while keeping all previous responses in context. The first response is generated with temperature sampling. For both system prompt sampling and in-context regeneration, we use the medium temperature values. We further evaluate both general and task-anchored approaches to system prompt sampling and in-context regeneration. For the general approach, we use a variation of the prompts used previous works (Zhang et al., 2025a;b). Our task-anchored approach modifies these prompts to specify the functional difference relevant to each task category in our taxonomy. Table 2 shows how we modify each instruction to be task-anchored for two task categories. Appendix A.3 provides all our task-anchored prompts for system prompt sampling and in-context regeneration.

Diversity Metrics We compute three diversity metrics: task-anchored functional diversity (Def. 3.1), vocabulary diversity (Def. A.1), and embedding diversity (Def. A.2). To calculate *functional diversity*, we use task-anchored LLM-judges[3] , where the judge prompt includes the definition of functional difference for the relevant task category (see Appendix A.4 for judge prompts). We then use these pairwise comparisons[4] to determine the *number of functionally diverse responses* obtained (out of the 5 responses generated per prompt and sampling strategy). Specifically, we group responses that are judged as not functionally different from each other, and then count the number of unique groups (connected components). To compute embedding diversity, we generate response embeddings using the gemini-embedding-001 model (with 3072-dimensional embeddings).

[2]We did not identify any datasets containing prompts for category E (*Problem Solving or Design Subjective*), likely due to the inherently open-ended nature of such tasks. We include it in our taxonomy for symmetry with category D. Developing and open-sourcing datasets in this category may be a valuable direction for future work.

[3]We validate the LLM-judges on a stratified random sample of 225 response pairs across models, tasks, and sampling strategies. Two authors independently labeled these responses for functional diversity, and agreed 79% with the LLM-judges (80% agreement between annotators). See Appendix Table 5 for details.

[4]Pairwise comparisons grow quadratically, which is why we only generate 5 responses per prompt.

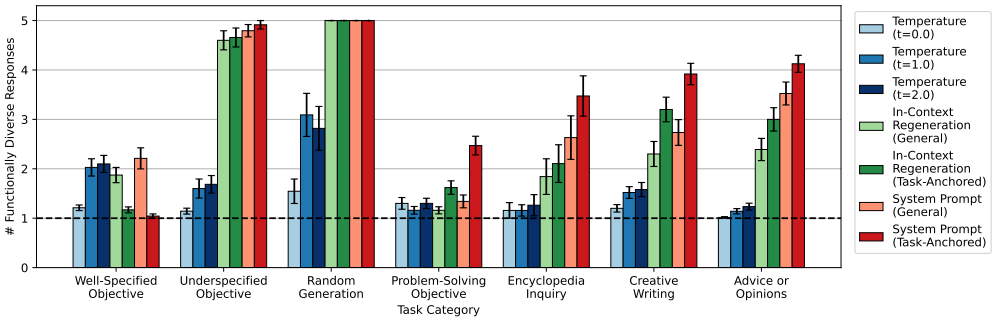


Figure 2: **Our task-anchored sampling increases functional diversity for task categories where homogenization is undesired, while preserving homogenization where it is desired.** We plot the average number of functionally diverse responses generated by GPT-4o for each sampling strategy and task category (with standard error). For the first category (Well-Specified Objective), bars closer to 1 reflect the preservation of output homogenization that is expected. For all other categories, bars closer to 5 reflect maximum functional diversity.

Quality Metrics We evaluate the quality of responses in two ways. (1) *Reward Model Quality:* Following many recent work that evaluates the diversity-quality trade-off (Lanchantin et al., 2025a; Slocum et al., 2025), we measure quality in terms of reward scores assigned by a reward model. We use *Athene-RM-8B*, which is to date empirically validated as one of the best reward models for human preferences (Frick et al., 2025). (2) *Checklist-Based Quality:* We also measure quality following prior work that uses *LLM-judges with grading checklists* (Lin et al., 2025; Wei et al., 2025). In this approach, the LLM-judge first generates a checklist of 3-5 key factors for response quality in a given prompt. This prompt-specific checklist is then used by the LLM-judge to score a particular response on a Likert scale from 1 to 5, where 1 indicates that none of the checklist criteria are met and 5 indicates that all criteria are satisfied. We include the judge prompts and examples of generated checklists in Appendix A.5.

4.2 Functional Diversity

We report our evaluation results on functional diversity. Our main finding is that our task-anchored sampling technique outperforms the more general sampling techniques in previous work (Zhang et al., 2025a;b). Figure 2 shows how we significantly increase functional diversity for task categories where homogenization is undesired, while preserving homogenization where it is desired (for GPT-4o, all results in Appendix B). Below, we explore results across task categories.

Well-Specified Objective Tasks (Category A) Tasks in this category have a single verifiably correct answer; therefore, no functional diversity is expected. However, when employing general diversity-promoting sampling methods, homogenization is undesirably reduced, as evidenced by the generation of multiple unique answers (2 on average). Increasing temperature also undesirably reduces homogenization. In contrast, our task-anchored sampling method maintains homogenization, consistently producing one unique answer per task.

Underspecified Objective and Random Generation Tasks (Categories B & C) Tasks in these categories are characterized by the existence of multiple verifiably correct answers, which suggests that models may easily conceptualize what difference means here. Consequently, we observe no significant differences between task-anchored and general sampling approaches, as the concept of diversity—defined as producing distinct correct answers—is inherently straightforward in this context. Both methods yield nearly maximal functional diversity, with approximately 5 unique responses out of 5 generations. In contrast, higher temperature settings result in suboptimal functional diversity, producing only 2 to 3 unique responses on average.

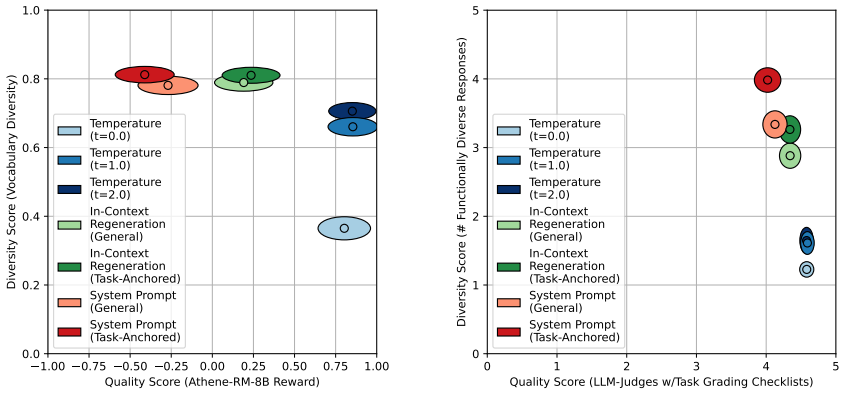
Problem-Solving Objective Tasks (Category D) Tasks in this category are defined by the presence of a single correct answer, but allow for multiple valid explanations or solution strategies. In this setting, we find that neither temperature-based sampling nor general strategies are effective in eliciting responses with diverse solution strategies. In contrast, both task-anchored system

prompts and in-context regeneration sampling are able to generate approximately 2–3 distinct solution strategies. This relatively low number may be attributable to the inherent difficulty of the MATH-500 benchmark, which poses significant challenges for large language models in producing even a single correct solution (Hendrycks et al., 2021).

Partial and Non-Verifiable Tasks (Categories F, G, H) Tasks in these categories cover encyclopedia inquiries, creative writing, and requests for advice or opinions. Across all three models, our task-anchored sampling methods – both system prompt and in-context regeneration – significantly reduce homogenization compared to their respective general approaches (t-test, $p < 0.05$). For GPT-4o and Gemini-2.5-Flash, task-anchored system prompting yields the highest number of functionally diverse responses. In contrast, for Claude-4-Sonnet, both task-anchored methods demonstrate comparable performance in promoting response diversity.

4.3 Diversity-Quality Tradeoff: Comparing General & Task-Based Metrics

We find that improved functional diversity from our task-anchored sampling *does not* decrease the quality of responses, when the task-dependent nature of quality is captured in the quality metric.[5] Recent proposals for measuring quality using task-specific checklists align with our discussions around task-anchored metrics for diversity (Lin et al., 2025; Wei et al., 2025). Whereas, when quality is determined by a reward model, the scores do not inherently reflect task differences (e.g. the quality of a creative writing response is measured in the same way as the quality of a math problem-solving response). Figure 3 shows that the diversity-quality tradeoff prevalent in previous studies may simply be the result of mis-conceptualizing both diversity and quality. When evaluating general metrics, there appears to be a large diversity-quality tradeoff between vocabulary diversity and reward quality (Figure 3a), and the tradeoff is similarly large with embedding diversity (Appendix Figure 8). When we compare task-anchored functional diversity with checklist-based quality, there is a negligible diversity-quality tradeoff (Figure 3b). These results hold for all models (Appendix Figures 6-7): overall, our task-anchored sampling technique maintains the same level of quality as the general strategies in previous work (Zhang et al., 2025a;b), while improving functional diversity.



(a) General Metrics (GPT-4o Responses) (b) Task-Based Metrics (GPT-4o Responses)

Figure 3: **With task-based metrics, diversity is improved with no significant drop in quality.** We plot quality on the x-axis and diversity on the y-axis and compare the tradeoff under general metrics vs task-based metrics. In (a)-(c), there is a large tradeoff between vocabulary diversity (Def. A.1) and quality scores determined by a reward model. In (b), there is a negligible tradeoff between task-anchored functional diversity (Def. 3.1) and LLM-judges with task-based grading checklists. Note that the checklist-based quality difference between score 4 and 5 is "good" vs "very good". Plots show the mean and standard error of all metrics averaged across all task categories except category A, which we exclude because it is the only category where output homogenization is desired.

[5]For tasks with singular verifiable rewards (Simple-QA & MATH-500), we separately validate the accuracy of responses (Appendix Table 13). Overall, our task-anchored sampling approaches maintain and often improve accuracy. For Simple-QA, both our approaches outperform temperature sampling for all 3 models. For MATH-500, our approaches outperform temperature sampling for Claude & Gemini.

5 DISCUSSION

Our work underlines the task-dependent nature of evaluating and mitigating output homogenization. We find that our task-anchored sampling technique outperforms more general sampling approaches in terms of increasing response diversity only when desired. Our results show that without task-dependence, previous methods to analyze output homogenization often (1) misconceptualize output diversity (2) reduce homogenization in tasks where homogenization should be preserved and (3) maintain homogenization in tasks where more pluralism is desired. Further, our results show that task-anchored sampling does not result in a significant diversity-quality trade-off. These results challenge the perceived existence of a diversity-quality tradeoff that is common in the literature. In this section, we discuss the implications of our work and avenues for future research.

5.1 OUR FRAMEWORK IMPROVES HOMOGENIZATION EVALUATION

We have developed a taxonomy of task categories that clarifies how a model can conceptualize diversity based on the categorization. For example, evaluating homogenization in math problem-solving should measure variety in solution strategies, whereas evaluating homogenization in advice or opinions should measure variety in viewpoints or perspectives. We improve upon previous studies that rely on generic measures of diversity (vocabulary or embedding differences), which is particularly meaningful when evaluating diversity-promoting methods. Our findings suggest that using general metrics without accounting for task dependence does not capture meaningful functional diversity.

We highlight the importance of evaluating prompts across our taxonomy when analyzing output homogenization. When studies limit their evaluation to tasks where diversity is desired, there may be unintended effects (e.g. confabulations) when those methods are applied to tasks which rely on homogenization being preserved. Hence, not adopting a task-dependent approach could result in less robust evaluation and present safety or ethical concerns downstream. Our taxonomy is one example of a categorization that anchors task dependence. Future work can adapt or expand this categorization. Our framework is generalizable in that the task-anchored approach can support a custom taxonomy tailored to a more specific downstream use case.

5.2 OUR FRAMEWORK IMPROVES HOMOGENIZATION MITIGATION

There are many ways to apply task-dependence in practical settings and our approach could be applied at inference-time automatically. For instance, the model could be given (or instructed to determine) a task categorization, filter prompts into said categories, and output responses according to the task-based conceptualization of output homogenization. Our main improvement is in clarifying model instructions for output homogenization behavior in terms of the task category. Instead of assuming the model does this inherently, it may be important to clarify and steer expected behavior.

Although we focus on prompt-based strategies, our task-anchored approach may be applied to other diversity-promoting methods that modify the alignment process. For example, Lanchantin et al. (2025a) propose a method for improving diversity through preference pair construction (x, y^+, y^-). This approach could be modified to construct pairs in a task-informed way that avoids learning undesired semantic preferences that might reduce functional diversity. Slocum et al. (2025) also propose modifying the RLHF or DPO optimization objective to include a penalty for lower token-level entropy. This penalty could be selectively applied to certain task categories where vocabulary diversity is desired, such as random generation and creative writing.

Future work may further explore how to embed task-anchored homogenization considerations directly into a model's learning or reasoning process. Our task-anchored sampling strategies could be incorporated into a chain-of-thought instruction, with models first reasoning about task-appropriate functional diversity. A reasoning model could also be trained to directly reason about the functional diversity requirements for a given task before generating a response. Future work in this direction could be quite impactful in terms of preventing problematic occurrences. With task-dependent reasoning about functional diversity, the model may avoid undesirable behavior such as confabulations or increasing diversity when it is culturally or socially inappropriate to do so. Ultimately, we offer a simple but important improvement to the field's conceptualization of output homogenization by grounding it in task dependence.

Statement on Usage of LLMs

We used LLMs in two ways: (1) to suggest style and grammar edits during paper writing, and (2) to assist in coding for experiments. All conceptual contributions, study design, and analysis were carried out by the authors.

Reproducibility Statement

Section 4.1 and Appendix A provide all necessary details to replicate our experiments. In particular, Appendix A.1 describes how we selected prompts from existing evaluation datasets. Appendix A.3 includes the exact prompts to replicate our task-anchored sampling technique. Appendix A.4 and Appendix A.5 also include the exact LLM-judge prompts that we used to measure functional diversity and checklist-based quality, respectively.

References

Rishi Bommasani, Kathleen A. Creel, Ananya Kumar, Dan Jurafsky, and Percy S Liang. Picking on the same person: Does algorithmic monoculture lead to outcome homogenization? In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 3663–3678. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/17a234c91f746d9625a75cf8a8731ee2-Paper-Conference.pdf.

Aaron Chatterji, Thomas Cunningham, David J Deming, Zoe Hitzig, Christopher Ong, Carl Yan Shan, and Kevin Wadman. How people use chatgpt. Technical report, National Bureau of Economic Research, 2025.

Daiwei Chen, Yi Chen, Aniket Rege, and Ramya Korlakai Vinayak. Pal: Pluralistic alignment framework for learning from heterogeneous preferences. *arXiv preprint arXiv:2406.08469*, 2024.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying Large Language Model Post-Training for Diverse Creative Writing. *arXiv preprint arXiv:2503.17126*, 2025.

Esin Durmus, Karina Nguyen, Thomas I Liao, Nicholas Schiefer, Amanda Askell, Anton Bakhtin, Carol Chen, Zac Hatfield-Dodds, Danny Hernandez, Nicholas Joseph, et al. Towards Measuring the Representation of Subjective Global Opinions in Language Models. *arXiv preprint arXiv:2306.16388*, 2023.

Sina Fazelpour and Will Fleisher. The Value of Disagreement in AI Design, Evaluation, and Alignment. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, pp. 2138–2150, 2025.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios Nikolas Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. How to evaluate reward models for rlhf. In *The Thirteenth International Conference on Learning Representations*, 2025.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving with the MATH Dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Shomik Jain, Kathleen Creel, and Ashia Camage Wilson. Position: Scarce Resource Allocations That Rely On Machine Learning Should Be Randomized. In *Forty-first International Conference on Machine Learning*, 2024a. URL https://openreview.net/forum?id=44qxX6Ty6F.

Shomik Jain, Vinith Suriyakumar, Kathleen Creel, and Ashia Wilson. Algorithmic Pluralism: A Structural Approach to Equal Opportunity. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pp. 197–206, 2024b.

Elliot Kim, Avi Garg, Kenny Peng, and Nikhil Garg. Correlated Errors in Large Language Models. *arXiv preprint arXiv:2506.07962*, 2025.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of RLHF on LLM generalisation and diversity. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=PXD3FAVHJT.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. Diverse Preference Optimization. *arXiv preprint arXiv:2501.18101*, 2025a.

Jack Lanchantin, Angelica Chen, Janice Lan, Xian Li, Swarnadeep Saha, Tianlu Wang, Jing Xu, Ping Yu, Weizhe Yuan, Jason E Weston, et al. Bridging Offline and Online Reinforcement Learning for LLMs. *arXiv preprint arXiv:2506.21495*, 2025b.

Tianjian Li, Yiming Zhang, Ping Yu, Swarnadeep Saha, Daniel Khashabi, Jason Weston, Jack Lanchantin, and Tianlu Wang. Jointly reinforcing diversity and quality in language model generations. *arXiv preprint arXiv:2509.02534*, 2025.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step By Step. In *The Twelfth International Conference on Learning Representations*, 2023.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. WildBench: Benchmarking LLMs with Challenging Tasks from Real Users in the Wild. In *The Thirteenth International Conference on Learning Representations*, 2025.

Kibum Moon. Homogenizing Effect of Large Language Model on Creativity: An Empirical Comparison of Human and ChatGPT Writing, 2024.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. Whose Opinions Do Language Models Reflect? In *International Conference on Machine Learning*, pp. 29971–30004. PMLR, 2023.

Judy Hanwen Shen, Archit Sharma, and Jun Qin. Towards Data-Centric RLHF: Simple Metrics for Preference Dataset Comparison. *arXiv preprint arXiv:2409.09603*, 2024.

Alexander Shypula, Shuo Li, Botong Zhang, Vishakh Padmakumar, Kayo Yin, and Osbert Bastani. Evaluating the Diversity and Quality of LLM Generated Content. In *ICLR Workshop on Deep Learning for Code*, 2025.

Stewart Slocum, Asher Parker-Sartori, and Dylan Hadfield-Menell. Diverse Preference Learning for Capabilities and Alignment. In *The Thirteenth International Conference on Learning Representations*, 2025.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, et al. A Roadmap to Pluralistic Alignment. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 46280–46302, 2024.

Alex Tamkin, Miles McCain, Kunal Handa, Esin Durmus, Liane Lovitt, Ankur Rathi, Saffron Huang, Alfred Mountfield, Jerry Hong, Stuart Ritchie, et al. Clio: Privacy-preserving insights into real-world ai use. *arXiv preprint arXiv:2412.13678*, 2024.

Angelina Wang, Jamie Morgenstern, and John P Dickerson. Large Language Models That Replace Human Participants Can Harmfully Misportray and Flatten Identity Groups. *Nature Machine Intelligence*, pp. 1–12, 2025a.

Qihan Wang, Shidong Pan, Tal Linzen, and Emily Black. Multilingual Prompting for Improving LLM Generation Diversity. *arXiv preprint arXiv:2505.15229*, 2025b.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models. *arXiv preprint arXiv:2411.04368*, 2024.

Tianjun Wei, Wei Wen, Ruizhi Qiao, Xing Sun, and Jianghong Ma. RocketEval: Efficient Automated LLM Evaluation via Grading Checklist. In *The Thirteenth International Conference on Learning Representations*, 2025.

Emily Wenger and Yoed Kenett. We're Different, We're the Same: Creative Homogeneity Across LLMs. *arXiv preprint arXiv:2501.19361*, 2025.

Fan Wu, Emily Black, and Varun Chandrasekaran. Generative Monoculture in Large Language Models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Binwei Yao, Zefan Cai, Yun-Shiuan Chuang, Shanglin Yang, Ming Jiang, Diyi Yang, and Junjie Hu. No Preference Left Behind: Group Distributional Preference Optimization. In *The Thirteenth International Conference on Learning Representations*, 2025.

Lily Hong Zhang, Smitha Milli, Karen Jusko, Jonathan Smith, Brandon Amos, Wassim, Bouaziz, Manon Revel, Jack Kussman, Lisa Titus, Bhaktipriya Radharapu, Jane Yu, Vidya Sarma, Kris Rose, and Maximilian Nickel. Cultivating Pluralism In Algorithmic Monoculture: The Community Alignment Dataset. *arXiv preprint arXiv: 2507.09650*, 2025a.

Yiming Zhang, Avi Schwarzschild, Nicholas Carlini, J Zico Kolter, and Daphne Ippolito. Forcing diffuse distributions out of language models. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=9JY1QLVFPZ.

Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. NoveltyBench: Evaluating Language Models for Humanlike Diversity. In *The Conference on Language Modeling (COLM)*, 2025b.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. WildChat: 1M ChatGPT Interaction Logs in the Wild. In *The Twelfth International Conference on Learning Representations*, 2024.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-Tuning Language Models from Human Preferences. *arXiv preprint arXiv:1909.08593*, 2019.

APPENDIX

Appendix A includes the following supplementary material about our experiment details.

- A.1: Evaluation Datasets
- A.2: Task Classification Into Our Taxonomy
- A.3: Prompts for Task-Anchored Sampling Strategies
- A.4: Measuring Functional Diversity
- A.5: Measuring Quality Using LLM-Judges With Task Grading Checklists

Appendix B includes the following supplementary tables and figures about our experiment results.

- Figure 4: Functional Diversity for Claude-4-Sonnet (c.f. Figure 2 in main text).
- Figure 5: Functional Diversity for Gemini-2.5-Flash (c.f. Figure 2 in main text).
- Figure 6: Diversity-Quality Tradeoff for Claude-4-Sonnet (c.f. Figure 3 in main text)
- Figure 7: Diversity-Quality Tradeoff for Gemini-2.5-Flash (c.f. Figure 3 in main text)
- Figure 8: Diversity-Quality Tradeoff Using Embeddings

- Table 8: Functional Diversity by Model, Sampling Strategy, and Task Category
- Table 9: Vocabulary Diversity by Model, Sampling Strategy, and Task Category
- Table 10: Embedding Diversity by Model, Sampling Strategy, and Task Category
- Table 11: Checklist-Based Quality by Model, Sampling Strategy, and Task Category
- Table 12: Athene-RM-8B Reward by Model, Sampling Strategy, and Task Category
- Table 13: Accuracy by Model, Sampling Strategy, and Task Category (for verifiable tasks)

A Additional Experiment Details

A.1 Evaluation Datasets

We sample 300 total prompts from the following datasets to use in evaluation of output homogenization. These datasets were chosen to achieve coverage across our task taxonomy (c.f. Table 3). For random sampling, we first shuffle the dataset using a random seed of 38, then select the required number of prompts in order from the shuffled dataset.

- **Community Alignment** (Zhang et al. (2025a)): A diverse human preference dataset containing user-generated prompts. We use 50 randomly-sampled prompts from the subset of user-generated first-turn prompts in English. Users were instructed to "ask, request, or talk to the model about something important to you or that represents your values. This could be related to work, religion, family, relationships, politics, or culture."
- **MATH-500** (Lightman et al. (2023)): A subset of the MATH dataset Hendrycks et al. (2021). We use 10 randomly-sampled prompts from each of the 5 difficulty levels.
- **NoveltyBench** (Zhang et al. (2025b)): A dataset of creative tasks where multiple distinct and high-quality outputs are expected. We use their entire curated dataset of 100 prompts.
- **SimpleQA** (Wei et al. (2024)): A dataset of short, fact-seeking queries across diverse topics. The prompts were created to be challenging for frontier models (e.g. GPT-4o accuracy $< 40\%$). We use 50 randomly-sampled prompts.
- **WildBench** (Lin et al. (2025)): A subset of the WildChat dataset Zhao et al. (2024). WildChat is a corpus of 1 million user-ChatGPT conversations. WildBench is a filtered subset of WildChat such that tasks are diverse and challenging for models. We use 50 randomly-sampled prompts from the WildBench-V2.

A.2 Task Classification Into Our Taxonomy

We classify prompts into our taxonomy using the following judge prompt. Note that we exclude category E (Problem Solving or Design Subjective) because we did not identify any datasets containing prompts for this category. In all our analysis, we analyze prompts based on the task category chosen by majority vote across the three models (GPT-4o, Claude-4-Sonnet, and Gemini-2.5-Flash). Thus, our experiments focus on evaluating models' ability to promote diversity when *given* the task category. We do not evaluate models' ability to correctly categorize tasks. However, all three models agree on the categorization for 79% of prompts, while two of the three models agree on an additional 19%. We use the GPT-4o label for the remaining cases ($<$2%).

```
Read the prompt below and decide which task category it belongs
to.  Only output a single category letter (A, B, C, D, E,
F, or G) without any additional text.  For prompts that have
objective responses, choose from categories A, B, C, or D. For
prompts that have subjective responses, choose from categories
E, F, or G.

Prompt:  {prompt}

Task Categories:
A - Well-Specified Singular Objective:  Task to generate a
single verifiable correct answer.
B - Underspecified Singular Objective:  Task to generate a
single answer for a prompt that has multiple verifiable correct
answers.
C - Random Generation Objective:  Task to generate a response
that involves randomizing over a set of finite options.
D - Problem Solving Objective:  Task to generate an answer
with reasoning or explanations for a problem with a single
verifiable correct answer.
E - Encyclopedia Inquiry Subjective:  Task to generate
information about real-world societies, traditions, events,
or social domains, where there are credible references.
F - Creative Generation Subjective:  Task to generate a
response that involves creative expression where there are
potentially infinite subjective responses.
G - Advice or Opinion Subjective:  Task to generate a response
that gives advice, opinions, or feedback on specific topics or
scenarios.
```

Table 3: Number of prompts per dataset and taxonomy category.

	Community Alignment	Math 500	Novelty Bench	Simple QA	Wild Bench	Total
Well-Specified Objective	4	13	1	50	3	71
Underspecified Objective	0	0	32	0	3	35
Random Generation	0	0	11	0	0	11
Problem-Solving Objective	1	37	0	0	12	50
Encyclopedia Inquiry	11	0	3	0	5	19
Creative Writing	1	0	25	0	24	50
Advice or Opinion	33	0	28	0	3	64
Total	50	50	100	50	50	300

A.3 Sampling Strategies

We generate 5 responses per prompt using temperature sampling, in-context regeneration, and system prompt sampling. For in-context regeneration, the first response is generated with temperature 1.0 for GPT-4o and Gemini-2.5-Flash, and temperature 0.5 for Claude-4-Sonnet. Subsequent responses are generated using the regeneration prompts below. For system prompt sampling, multiple responses are extracted from a single generation. For all sampling strategies, we set the nucleus sampling parameter to 0.9 and the maximum number of output tokens to 1024.

General System Prompt

We use the following prompt for the general system prompting approach. This prompt is similar to the one used in Zhang et al. (2025a), except they use the phrase "Generate {num_responses} that represent diverse values". We generalize this instruction to apply to all the categories in our taxonomy, since Zhang et al. (2025a) focus on more subjective tasks.

```
Generate {num_responses} different responses to the following
prompt.

Each response should start with "# Response X:" where X is
a number from 1 to {num_responses} to demarcate where each
different response begins.  Make sure to stop at # Response
{num_responses}.  Make each generated response self-contained.
They should not rely on the other responses for context.
```

Task-Anchored System Prompts

We use the following template for task-anchored system prompting. Table 4 lists the task-specific part of the system prompt that we use for each task category.

```
{Task-Anchored System Prompt}

Each response should start with "# Response X:" where X is
a number from 1 to {num_responses} to demarcate where each
different response begins.  Make sure to stop at # Response
{num_responses}.  Make each generated response self-contained.
They should not rely on the other responses for context.
```

General In-Context Regeneration Prompt

Zhang et al. (2025b) use the following prompt for in-context regeneration. We call this a "general" prompt because there is no task dependence.

```
Can you generate a different response?
```

Task-Anchored In-Context Regeneration Prompts

We use the following template for task-anchored in-context regeneration. Table 4 lists the task-specific part of the prompt that we use for each task category.

```
{Task-Anchored In-Context Regeneration Prompt}

Do not include any starting phrases or reasons for why your new
response is different.  Your response should be self-contained,
as if the prompt was the first thing that I asked.

Remember, the prompt is:  {prompt}
```

Table 4: Prompts for Task-Anchored Sampling Strategies

Category	Task-Anchored System Prompt	Task-Anchored In-Context Regeneration Prompt
Well-Specified Objective	The following prompt has a single correct answer. Generate {num_responses} responses. If relevant, slight variation in wording is allowed but the answer should remain the same.	Can you generate a different response? The prompt has a single correct answer, so your answer should remain the same. If relevant, slight variation in wording is allowed.
Underspecified Objective	The following prompt is underspecified and has many correct answers. Generate {num_responses} responses, each with a different correct answer.	Can you generate a different correct answer? The prompt is underspecified and has many correct answers.
Random Generation	The following prompt is asking you to randomize over a set of finite options. Generate {num_responses} responses, each with a different pseudo-random option.	Can you generate a different pseudo-random response? The prompt is asking you to randomize over a set of finite options.
Problem-Solving Objective	The following problem has a single correct answer, but can be solved using different problem-solving strategies. Generate {num_responses} different solutions, each with a different problem-solving strategy.	Can you solve the problem using a different strategy? The problem has a single correct answer, but can be solved using different problem-solving strategies.
Encyclopedia Inquiry	The following prompt is asking for information about the real-world, where there may be different factual perspectives. Your response must be grounded in credible references though references do not need to be mentioned. Generate {num_responses} responses that reflect different perspectives.	Can you generate a new response that reflects a different factual perspective? The prompt is asking for information about the real-world, where there may be different factual perspectives. Your response must be grounded in credible references though references do not need to be mentioned.
Creative Writing	The following prompt is asking for creative expression, so there are many possible subjective responses. Generate {num_responses} unique responses by varying the key creative elements such as tone, genre, point of view, theme, structure, etc. Each response should have different creative elements and reflect a distinct creative expression.	Can you generate a new response with different creative elements? The prompt is asking for creative expression, so there are many possible subjective responses. Your new response should change the key creative elements such as tone, genre, point of view, theme, structure, etc.
Advice or Opinions	The following prompt is asking for advice or opinions, so there are many possible subjective responses. Generate {num_responses} unique responses where each response expresses a different viewpoint or perspective.	Can you generate a new response with a different viewpoint or perspective? The prompt is asking for advice or opinions, so there are many possible subjective responses.

A.4 Measuring Functional Diversity

We use the following prompt template to evaluate functional diversity using LLM-judges. Table 6 lists the task-specific parts of the prompt that we use for each task category.

We validate the LLM-judges on a stratified random sample of 225 response pairs across models, tasks, and sampling strategies. Two authors independently labeled these prompts for functional diversity. Table 5 reports the inter-rater agreement between annotators and each LLM-judge. When compared to the ensemble of LLM-judges, annotator 1 achieved 77.7% agreement and annotator 2 achieved 79.0% agreement. This is comparable to Zhang et al. (2025b)'s evaluation of functional diversity in creative tasks, where a fine-tuned judge achieved 79% agreement with human annotators.

```
For the given prompt and two responses, determine if the
responses are functionally equivalent.  Functional equivalence
means a user who has seen one response would find the other
response to be redundant.

{Task-Anchored Functional Diversity Definition}

###
Prompt:  {prompt}
Response 1:  {response 1}
Response 2:  {response 2}
###

Are the responses functionally equivalent?
{Task-Anchored Diversity Judge Options}
Only output YES or NO.
```

To compare with functional diversity, we also measure diversity using two general diversity metrics that do not have task-dependence.

Definition A.1 (Vocabulary Diversity)**.** Given two responses $y_a, y_b \in \mathcal{Y}$, let $\mathcal{V}_a$ and $\mathcal{V}_b$ denote the sets of unique words in y_a and y_b, respectively. The *vocabulary diversity* between y_a and y_b is

$$d_{\text{vocab}}(y_a, y_b) := 1 - \frac{|\mathcal{V}_a \cap \mathcal{V}_b|}{|\mathcal{V}_a \cup \mathcal{V}_b|},$$

where $|\mathcal{V}_a \cap \mathcal{V}_b|$ is the number of shared words and $|\mathcal{V}_a \cup \mathcal{V}_b|$ is the total number of unique words in both responses.

Definition A.2 (Embedding Diversity)**.** Given two responses $y_a, y_b \in \mathcal{Y}$, let $e(y)$ denote the embedding vector for response y. The *embedding diversity* between y_a and y_b is

$$d_{\text{embed}}(y_a, y_b) := 1 - \cos(e(y_a), e(y_b)),$$

where $\cos(e(y_a), e(y_b))$ is the cosine similarity between the embedding vectors of y_a and y_b.

Table 5: Annotator Agreement

	Annotator 1	Annotator 2	GPT-4o	Claude-4-Sonnet	Gemini-2.5-Flash
Annotator 1	-	79.9%	75.0%	77.2%	77.2%
Annotator 2	79.9%	-	79.0%	77.7%	80.4%
GPT-4o	75.0%	79.0%	-	90.6%	88.8%
Claude-4-Sonnet	77.2%	77.7%	90.6%	-	93.8%
Gemini-2.5-Flash	77.2%	80.4%	88.8%	93.8%	-

Table 6: Prompts for Functional Diversity LLM-Judge

Category	Task-Anchored Functional Diversity Def.	Diversity Judge Options
Well-Specified Objective	The prompt has a single correct answer. Responses are functionally equivalent if they represent the same answer.	Output YES if the responses represent the same answer. Output NO if the responses represent different answers.
Underspecified Objective	The prompt is underspecified and has many correct answers. Responses are functionally equivalent if they represent the same answer.	Output YES if the responses represent the same answer. Output NO if the responses represent different answers.
Random Generation	The prompt is asking for a random response over a set of finite options. Responses are functionally equivalent if they represent the same pseudo-random option.	Output YES if the responses represent the same pseudo-random option. Output NO if the responses represent different pseudo-random options.
Problem-Solving Objective	The prompt involves solving a problem with a single correct answer, but it can be solved using different problem-solving strategies. Responses are functionally equivalent if they represent the same problem-solving strategy.	Output YES if the responses represent the same problem-solving strategy. Output NO if the responses represent different problem-solving strategies.
Encyclopedia Inquiry	The prompt is asking for information about the real-world, where there may be different factual perspectives. Responses are functionally equivalent if they represent similar factual perspectives.	Output YES if the responses represent similar perspectives. Output NO if the responses represent different perspectives.
Creative Writing	The prompt is asking for creative expression where there are many possible subjective responses. Responses are functionally equivalent if the key creative elements (such as tone, genre, point of view, theme, structure, etc.) are the same.	Output YES if the responses have similar key creative elements. Output NO if the responses have different key creative elements.
Advice or Opinions	The prompt is asking for advice or opinions. Responses are functionally equivalent if they express the same viewpoint or perspective, even if they are worded differently.	Output YES if the responses have similar perspectives. Output NO if the responses have different perspectives.

A.5 Measuring Checklist-Based Quality

We follow the approach in Lin et al. (2025) and Wei et al. (2025) to evaluate quality using LLM-judges with task-specific grading checklists. First, the LLM-judge generates a grading checklist for each prompt with key factors that should be present in high-quality responses (Checklist Creation Prompt). Then, the LLM-judge is given a response to grade along with the previously generated checklist (Checklist-Based Quality Judge Prompt). We grade responses on a Likert scale from 1 to 5, where 1 indicates the response is very poor and does not meet any criteria on the checklist, whereas 5 indicates the response is very good and meets all the criteria on the checklist. Table 7 includes examples of the grading checklists generated by LLM-judges.

Checklist Creation Prompt

```
You are an expert evaluator of LLM response quality.  Your task
is to generate a quality checklist that can be used to evaluate
response quality for a given user prompt.

# Instructions
First, identify 3 to 5 key factors that should be present in
high-quality responses for the given prompt.  Since a prompt
may have many different high-quality responses, the key factors
should be broad enough to cover varied high-quality responses
while still being specific to the prompt.

Then, turn each key factor into a yes/no question for a
checklist to evaluate response quality.  Your questions should
be concise and specific to the user prompt.  Avoid creating
duplicate, cumbersome, or vague questions.  Do not create more
than 5 questions.

# User Prompt
{prompt}

# Output Format
First, output the key factors you identified in a short
paragraph.  Then, output your quality checklist of yes/no
questions in the following format, starting with "Here is my
quality checklist:".  Separate each question with newlines.  Do
not output any other text after the checklist.

Here is my quality checklist:
1.  Question 1
2.  Question 2
...
```

CHECKLIST-BASED QUALITY JUDGE PROMPT

```
You are an expert evaluator of LLM response quality.  Your task
is to grade the given LLM response to a user prompt based on
the quality checklist below.

# User Prompt
{prompt}

# Response To Grade
{response}

# Quality Checklist
{checklist}

# Instructions
First, answer each question in the checklist with "yes" or
"no", based on the response to grade.

Then, assign a quality score between 1 to 5 for the response,
based on the following criteria:
- Score 1:  The response is very poor and does not meet any
criteria on the checklist.
- Score 2:  The response is poor and meets few criteria on the
checklist.
- Score 3:  The response is fair and meets some criteria on the
checklist.
- Score 4:  The response is good and meets most criteria on the
checklist.
- Score 5:  The response is very good and meets all criteria on
the checklist.

# Output Format
First, output your answers to the quality checklist as a
numbered list.  Each answer should start with "yes" or "no"
and have a short justification (1-2 sentences).

Then, output your quality score for the response.  Your output
should end with the following sentence, where <score> is the
quality score you assigned:  "Based on the quality checklist,
my quality score for the response is <score>."
```

Table 7: Examples of Task-Specific Grading Checklists

LLM-Judge	Prompt	Grading Checklist Generated by LLM-Judge
GPT-4o	Tell me a story in five sentences about a boy and his dog.	1. Does the response present a clear and cohesive narrative within five sentences? 2. Are the boy and his dog central to the story, with a clear focus on their interaction or relationship? 3. Does the story elicit some form of engagement (e.g., emotional resonance, curiosity, or intrigue)? 4. Does the story conclude in a satisfying or meaningful way within the constraints of five sentences? 5. Is the story creative and free from overused cliches?
Claude-4-Sonnet	Tell me a dad joke.	1. Does the response contain an actual joke? 2. Is the joke appropriate for all ages and family-friendly? 3. Does the joke follow typical dad joke characteristics (puns, wordplay, or simple humor)? 4. Is the response concise and focused on delivering the joke?
Gemini-2.5-Flash	Write a short love poem with 4 lines.	1. Is the response structured as a poem? 2. Does the poem express a theme of love? 3. Is the poem exactly four lines long?
GPT-4o	Why is Issac Newton famous?	1. Does the response mention at least one of Isaac Newton's major contributions, such as the laws of motion, the law of universal gravitation, or calculus? 2. Is the information provided accurate and free from factual errors? 3. Does the response maintain relevance to the prompt, focusing on why Isaac Newton is famous? 4. Is the response clear and easy to understand by the intended audience?
Claude-4-Sonnet	If $2^8 = 4^x$, what is the value of x?	1. Does the response correctly rewrite the equation using the same base (either base 2 or base 4)? 2. Does the response properly apply exponent rules to solve for x? 3. Does the response show clear, logical steps that lead to the solution? 4. Does the response arrive at the correct answer (x = 4)? 5. Does the response verify the solution by checking it against the original equation?
Gemini-2.5-Flash	Who was the first scientist to isolate cardiolipin?	1. Does the response identify the first scientist to isolate cardiolipin? 2. Does the response accurately name the scientist as M. C. Pangborn? 3. Is the information provided factually correct? 4. Is the response concise and directly answer the question?
GPT-4o	I am going to join a software company next week as a software engineer. What are the things that I should take care of before joining?	1. Does the response provide actionable steps to prepare technically or brush up on relevant skills for the role? 2. Does the response address logistical preparation, such as ensuring necessary documentation or setup is in place? 3. Does the response offer advice on professional behavior or mindset to adopt for a successful onboarding experience? 4. Are the suggestions relevant for a software engineer and tailored to the context of joining a new company? 5. Is the advice practical and feasible for someone to implement before their first day?
Claude-4-Sonnet	Pick a number between 1 and 100. Now tell me what you picked.	1. Does the response include a specific number between 1 and 100 (inclusive)? 2. Does the response clearly state what number was picked? 3. Does the response directly address the user's request without unnecessary complexity or evasion?
Gemini-2.5-Flash	Write an apology letter to mother and father that I had run away in fear from getting beaten up by my friend.	1. Does the letter sincerely apologize for running away? 2. Does the letter clearly state that the reason for running away was fear of being beaten by a friend? 3. Does the letter acknowledge the parents' potential worry or hurt caused by the action? 4. Does the letter offer any reassurance to the parents about the future or the child's safety?

B ADDITIONAL EXPERIMENT RESULTS

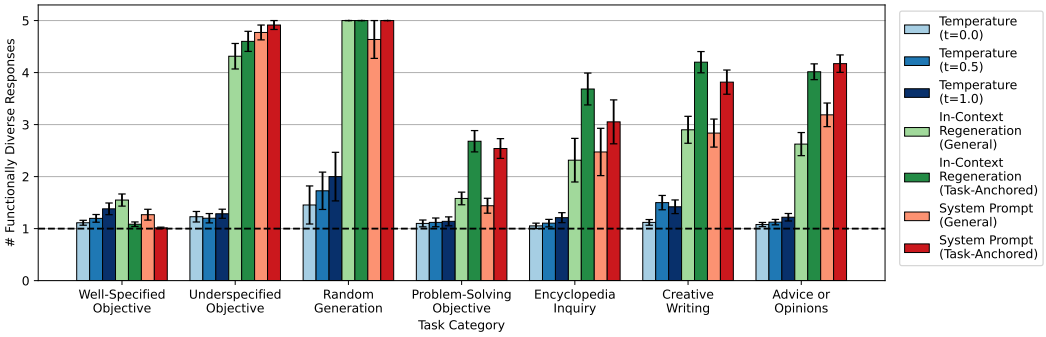


Figure 4: Number of functionally diverse responses generated by Claude-4-Sonnet for each sampling strategy and task category (c.f. Figure 2 in main text).

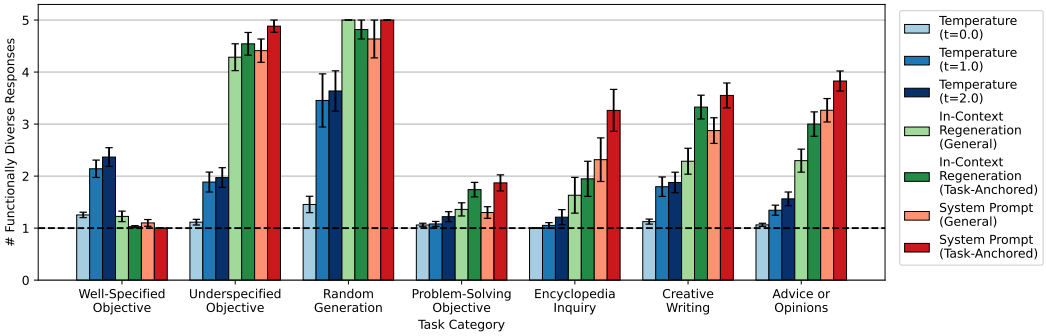
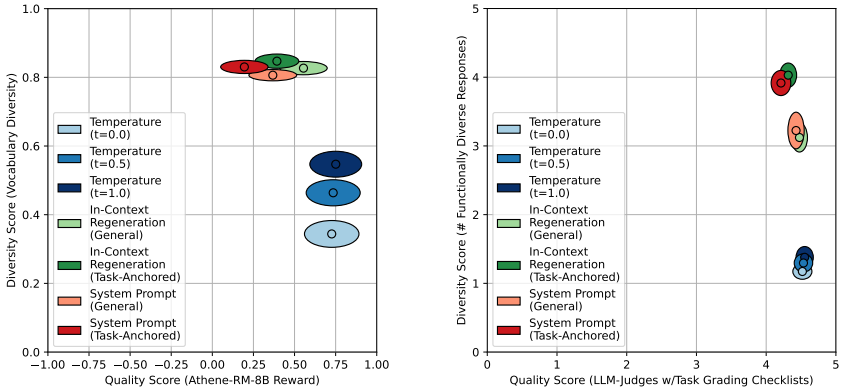
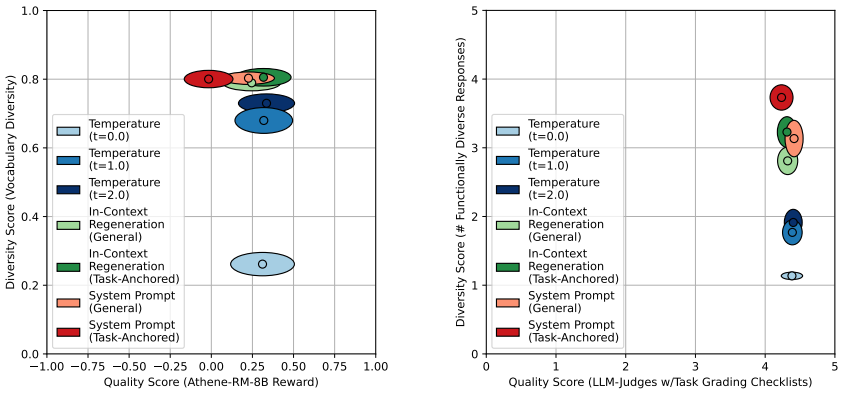


Figure 5: Number of functionally diverse responses generated by Gemini-2.5-Flash for each sampling strategy and task category (c.f. Figure 2 in main text).



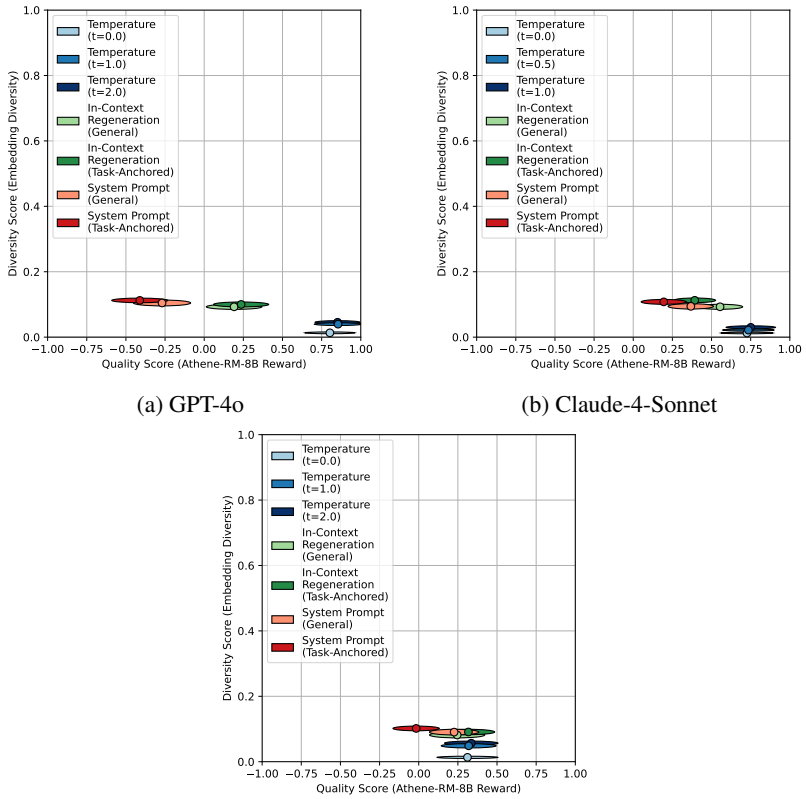
(a) General Metrics (b) Task-Based Metrics

Figure 6: Diversity-quality tradeoff under general vs task-based metrics for Claude-4-Sonnet.



(a) General Metrics (b) Task-Based Metrics

Figure 7: Diversity-quality tradeoff under general vs task-based metrics for Gemini-2.5-Flash.



(a) GPT-4o (b) Claude-4-Sonnet

(c) Gemini-2.5-Flash

Figure 8: Diversity-quality tradeoff using embedding diversity (c.f. Figure 3).

Table 8: # of Functionally Diverse Responses by Model, Sampling Strategy, and Task Category.

Model	Sampling Strategy	A	B	C	D	F	G	H
gpt-4o	Temperature (t=0.0)	1.21 (0.06)	1.14 (0.06)	1.55 (0.25)	1.30 (0.12)	1.16 (0.16)	1.20 (0.08)	1.02 (0.02)
gpt-4o	Temperature (t=1.0)	2.03 (0.17)	1.60 (0.19)	3.09 (0.44)	1.16 (0.08)	1.16 (0.12)	1.52 (0.12)	1.14 (0.05)
gpt-4o	Temperature (t=2.0)	2.10 (0.17)	1.69 (0.18)	2.82 (0.44)	1.30 (0.10)	1.26 (0.21)	1.58 (0.14)	1.23 (0.07)
gpt-4o	In-Context Regeneration (General)	1.87 (0.15)	4.60 (0.19)	5.00 (0.00)	1.16 (0.07)	1.84 (0.36)	2.30 (0.25)	2.39 (0.22)
gpt-4o	In-Context Regeneration (Task-Anchored)	1.17 (0.06)	4.66 (0.19)	5.00 (0.00)	1.62 (0.14)	2.11 (0.38)	3.20 (0.25)	3.00 (0.24)
gpt-4o	System Prompt (General)	2.21 (0.21)	4.79 (0.13)	5.00 (0.00)	1.34 (0.13)	2.63 (0.44)	2.73 (0.26)	3.52 (0.23)
gpt-4o	System Prompt (Task-Anchored)	1.04 (0.04)	4.91 (0.09)	5.00 (0.00)	2.47 (0.19)	3.47 (0.41)	3.92 (0.22)	4.12 (0.17)
claude-4-sonnet	Temperature (t=0.0)	1.11 (0.05)	1.23 (0.10)	1.45 (0.37)	1.10 (0.07)	1.05 (0.05)	1.12 (0.05)	1.08 (0.04)
claude-4-sonnet	Temperature (t=0.5)	1.20 (0.07)	1.20 (0.09)	1.73 (0.36)	1.12 (0.08)	1.11 (0.07)	1.50 (0.14)	1.12 (0.05)
claude-4-sonnet	Temperature (t=1.0)	1.38 (0.11)	1.29 (0.09)	2.00 (0.47)	1.14 (0.09)	1.21 (0.10)	1.42 (0.13)	1.22 (0.07)
claude-4-sonnet	In-Context Regeneration (General)	1.55 (0.12)	4.31 (0.25)	5.00 (0.00)	1.58 (0.12)	2.32 (0.42)	2.90 (0.26)	2.62 (0.22)
claude-4-sonnet	In-Context Regeneration (Task-Anchored)	1.08 (0.04)	4.60 (0.19)	5.00 (0.00)	2.68 (0.20)	3.68 (0.31)	4.20 (0.20)	4.02 (0.15)
claude-4-sonnet	System Prompt (General)	1.27 (0.10)	4.77 (0.14)	4.64 (0.36)	1.44 (0.14)	2.47 (0.45)	2.84 (0.27)	3.19 (0.23)
claude-4-sonnet	System Prompt (Task-Anchored)	1.01 (0.01)	4.91 (0.09)	5.00 (0.00)	2.54 (0.19)	3.05 (0.42)	3.82 (0.23)	4.17 (0.17)
gemini-2.5-flash	Temperature (t=0.0)	1.25 (0.05)	1.11 (0.05)	1.45 (0.16)	1.06 (0.03)	1.00 (0.00)	1.12 (0.05)	1.06 (0.03)
gemini-2.5-flash	Temperature (t=1.0)	2.14 (0.17)	1.89 (0.19)	3.45 (0.51)	1.08 (0.05)	1.05 (0.05)	1.80 (0.19)	1.34 (0.10)
gemini-2.5-flash	Temperature (t=2.0)	2.37 (0.18)	1.97 (0.19)	3.64 (0.39)	1.22 (0.10)	1.21 (0.14)	1.88 (0.20)	1.56 (0.13)
gemini-2.5-flash	In-Context Regeneration (General)	1.23 (0.10)	4.29 (0.26)	5.00 (0.00)	1.36 (0.13)	1.63 (0.34)	2.29 (0.25)	2.30 (0.22)
gemini-2.5-flash	In-Context Regeneration (Task-Anchored)	1.03 (0.02)	4.54 (0.22)	4.82 (0.18)	1.74 (0.14)	1.95 (0.34)	3.33 (0.23)	3.00 (0.23)
gemini-2.5-flash	System Prompt (General)	1.10 (0.06)	4.41 (0.22)	4.64 (0.36)	1.30 (0.11)	2.32 (0.42)	2.88 (0.25)	3.27 (0.23)
gemini-2.5-flash	System Prompt (Task-Anchored)	1.00 (0.00)	4.88 (0.12)	5.00 (0.00)	1.87 (0.15)	3.26 (0.40)	3.55 (0.24)	3.83 (0.19)

Table 9: Vocabulary Diversity by Model, Sampling Strategy, and Task Category.

Model	Sampling Strategy	A	B	C	D	F	G	H
gpt-4o	Temperature (t=0.0)	0.19 (0.02)	0.20 (0.03)	0.27 (0.06)	0.33 (0.02)	0.49 (0.03)	0.47 (0.03)	0.44 (0.03)
gpt-4o	Temperature (t=1.0)	0.55 (0.02)	0.56 (0.03)	0.63 (0.06)	0.61 (0.02)	0.71 (0.01)	0.74 (0.02)	0.71 (0.02)
gpt-4o	Temperature (t=2.0)	0.59 (0.02)	0.61 (0.04)	0.69 (0.05)	0.66 (0.01)	0.75 (0.01)	0.78 (0.02)	0.75 (0.02)
gpt-4o	In-Context Regeneration (General)	0.62 (0.02)	0.91 (0.02)	0.94 (0.05)	0.54 (0.02)	0.73 (0.03)	0.79 (0.02)	0.83 (0.01)
gpt-4o	In-Context Regeneration (Task-Anchored)	0.55 (0.02)	0.93 (0.02)	0.90 (0.06)	0.55 (0.02)	0.77 (0.02)	0.85 (0.02)	0.87 (0.01)
gpt-4o	System Prompt (General)	0.69 (0.01)	0.81 (0.03)	0.77 (0.08)	0.58 (0.02)	0.82 (0.01)	0.84 (0.01)	0.86 (0.01)
gpt-4o	System Prompt (Task-Anchored)	0.52 (0.02)	0.87 (0.02)	0.73 (0.08)	0.71 (0.02)	0.85 (0.01)	0.86 (0.01)	0.87 (0.00)
claude-4-sonnet	Temperature (t=0.0)	0.24 (0.02)	0.23 (0.04)	0.17 (0.07)	0.29 (0.03)	0.47 (0.04)	0.50 (0.03)	0.42 (0.02)
claude-4-sonnet	Temperature (t=0.5)	0.32 (0.02)	0.29 (0.04)	0.31 (0.08)	0.38 (0.03)	0.61 (0.03)	0.61 (0.03)	0.57 (0.02)
claude-4-sonnet	Temperature (t=1.0)	0.43 (0.02)	0.41 (0.05)	0.46 (0.08)	0.44 (0.03)	0.67 (0.03)	0.65 (0.02)	0.65 (0.02)
claude-4-sonnet	In-Context Regeneration (General)	0.75 (0.01)	0.82 (0.03)	0.89 (0.05)	0.72 (0.01)	0.83 (0.01)	0.82 (0.01)	0.87 (0.01)
claude-4-sonnet	In-Context Regeneration (Task-Anchored)	0.63 (0.01)	0.82 (0.03)	0.89 (0.06)	0.76 (0.01)	0.84 (0.01)	0.90 (0.01)	0.87 (0.01)
claude-4-sonnet	System Prompt (General)	0.71 (0.01)	0.83 (0.01)	0.81 (0.05)	0.70 (0.01)	0.83 (0.01)	0.83 (0.01)	0.85 (0.00)
claude-4-sonnet	System Prompt (Task-Anchored)	0.69 (0.01)	0.82 (0.01)	0.82 (0.07)	0.76 (0.01)	0.84 (0.01)	0.87 (0.01)	0.87 (0.00)
gemini-2.5-flash	Temperature (t=0.0)	0.12 (0.02)	0.21 (0.04)	0.25 (0.08)	0.19 (0.02)	0.35 (0.02)	0.24 (0.03)	0.33 (0.02)
gemini-2.5-flash	Temperature (t=1.0)	0.42 (0.03)	0.61 (0.05)	0.70 (0.09)	0.57 (0.02)	0.73 (0.02)	0.74 (0.03)	0.73 (0.03)
gemini-2.5-flash	Temperature (t=2.0)	0.47 (0.03)	0.64 (0.05)	0.82 (0.05)	0.59 (0.02)	0.76 (0.01)	0.78 (0.02)	0.79 (0.01)
gemini-2.5-flash	In-Context Regeneration (General)	0.73 (0.02)	0.92 (0.02)	0.88 (0.05)	0.63 (0.01)	0.73 (0.02)	0.77 (0.02)	0.81 (0.01)
gemini-2.5-flash	In-Context Regeneration (Task-Anchored)	0.60 (0.02)	0.92 (0.02)	0.73 (0.08)	0.71 (0.01)	0.77 (0.02)	0.85 (0.02)	0.86 (0.01)
gemini-2.5-flash	System Prompt (General)	0.66 (0.01)	0.83 (0.03)	0.87 (0.02)	0.66 (0.01)	0.80 (0.02)	0.80 (0.02)	0.85 (0.01)
gemini-2.5-flash	System Prompt (Task-Anchored)	0.60 (0.01)	0.89 (0.02)	0.63 (0.09)	0.73 (0.01)	0.84 (0.01)	0.84 (0.01)	0.86 (0.00)

Table 10: Embedding Diversity by Model, Sampling Strategy, and Task Category.

Model	Sampling Strategy	A	B	C	D	F	G	H
gpt-4o	Temperature (t=0.0)	0.01 (0.00)	0.01 (0.00)	0.02 (0.01)	0.01 (0.00)	0.01 (0.00)	0.02 (0.00)	0.01 (0.00)
gpt-4o	Temperature (t=1.0)	0.04 (0.00)	0.04 (0.00)	0.06 (0.01)	0.03 (0.00)	0.03 (0.00)	0.05 (0.00)	0.03 (0.00)
gpt-4o	Temperature (t=2.0)	0.04 (0.00)	0.04 (0.00)	0.07 (0.01)	0.03 (0.00)	0.03 (0.00)	0.06 (0.00)	0.04 (0.00)
gpt-4o	In-Context Regeneration (General)	0.07 (0.01)	0.14 (0.01)	0.16 (0.01)	0.03 (0.00)	0.05 (0.01)	0.09 (0.01)	0.09 (0.01)
gpt-4o	In-Context Regeneration (Task-Anchored)	0.04 (0.00)	0.15 (0.01)	0.15 (0.01)	0.03 (0.00)	0.06 (0.01)	0.11 (0.01)	0.10 (0.01)
gpt-4o	System Prompt (General)	0.06 (0.00)	0.14 (0.01)	0.14 (0.01)	0.04 (0.01)	0.09 (0.01)	0.11 (0.01)	0.11 (0.01)
gpt-4o	System Prompt (Task-Anchored)	0.02 (0.00)	0.15 (0.01)	0.11 (0.01)	0.07 (0.00)	0.10 (0.01)	0.12 (0.01)	0.12 (0.00)
claude-4-sonnet	Temperature (t=0.0)	0.01 (0.00)	0.01 (0.00)	0.01 (0.00)	0.01 (0.00)	0.01 (0.00)	0.02 (0.00)	0.01 (0.00)
claude-4-sonnet	Temperature (t=0.5)	0.02 (0.00)	0.02 (0.00)	0.02 (0.01)	0.01 (0.00)	0.02 (0.00)	0.03 (0.00)	0.02 (0.00)
claude-4-sonnet	Temperature (t=1.0)	0.03 (0.00)	0.03 (0.00)	0.04 (0.01)	0.01 (0.00)	0.03 (0.00)	0.04 (0.00)	0.03 (0.00)
claude-4-sonnet	In-Context Regeneration (General)	0.06 (0.00)	0.12 (0.01)	0.14 (0.01)	0.04 (0.00)	0.07 (0.01)	0.10 (0.01)	0.09 (0.01)
claude-4-sonnet	In-Context Regeneration (Task-Anchored)	0.03 (0.00)	0.13 (0.01)	0.14 (0.01)	0.06 (0.00)	0.09 (0.01)	0.14 (0.01)	0.12 (0.00)
claude-4-sonnet	System Prompt (General)	0.05 (0.00)	0.14 (0.01)	0.13 (0.02)	0.04 (0.00)	0.07 (0.01)	0.10 (0.01)	0.09 (0.00)
claude-4-sonnet	System Prompt (Task-Anchored)	0.04 (0.00)	0.14 (0.01)	0.12 (0.02)	0.06 (0.00)	0.09 (0.01)	0.12 (0.01)	0.12 (0.00)
gemini-2.5-flash	Temperature (t=0.0)	0.01 (0.00)	0.02 (0.01)	0.03 (0.01)	0.00 (0.00)	0.01 (0.00)	0.01 (0.00)	0.01 (0.00)
gemini-2.5-flash	Temperature (t=1.0)	0.04 (0.00)	0.06 (0.01)	0.08 (0.02)	0.02 (0.00)	0.03 (0.00)	0.06 (0.01)	0.05 (0.00)
gemini-2.5-flash	Temperature (t=2.0)	0.04 (0.00)	0.07 (0.01)	0.10 (0.02)	0.02 (0.00)	0.03 (0.00)	0.07 (0.01)	0.05 (0.01)
gemini-2.5-flash	In-Context Regeneration (General)	0.07 (0.00)	0.12 (0.01)	0.13 (0.02)	0.03 (0.00)	0.05 (0.01)	0.08 (0.01)	0.08 (0.01)
gemini-2.5-flash	In-Context Regeneration (Task-Anchored)	0.04 (0.00)	0.13 (0.01)	0.11 (0.02)	0.04 (0.00)	0.06 (0.01)	0.11 (0.01)	0.10 (0.01)
gemini-2.5-flash	System Prompt (General)	0.04 (0.00)	0.12 (0.01)	0.14 (0.02)	0.03 (0.00)	0.07 (0.01)	0.09 (0.01)	0.09 (0.00)
gemini-2.5-flash	System Prompt (Task-Anchored)	0.02 (0.00)	0.13 (0.01)	0.12 (0.01)	0.05 (0.00)	0.10 (0.01)	0.11 (0.01)	0.11 (0.00)

Table 11: Checklist-Based Quality by Model, Sampling Strategy, and Task Category.

Model	Sampling Strategy	A	B	C	D	F	G	H
gpt-4o	Temperature (t=0.0)	3.80 (0.11)	4.68 (0.13)	4.63 (0.09)	4.14 (0.14)	4.70 (0.11)	4.60 (0.08)	4.75 (0.05)
gpt-4o	Temperature (t=1.0)	3.72 (0.12)	4.62 (0.13)	4.69 (0.08)	4.13 (0.14)	4.73 (0.10)	4.61 (0.08)	4.76 (0.05)
gpt-4o	Temperature (t=2.0)	3.76 (0.12)	4.65 (0.12)	4.71 (0.09)	4.03 (0.15)	4.72 (0.09)	4.61 (0.08)	4.76 (0.05)
gpt-4o	In-Context Regeneration (General)	3.46 (0.11)	4.34 (0.17)	4.53 (0.29)	4.14 (0.13)	4.43 (0.12)	4.50 (0.10)	4.12 (0.11)
gpt-4o	In-Context Regeneration (Task-Anchored)	3.51 (0.11)	4.30 (0.17)	4.50 (0.29)	4.09 (0.13)	4.52 (0.14)	4.53 (0.08)	4.11 (0.10)
gpt-4o	System Prompt (General)	3.55 (0.13)	4.44 (0.16)	4.19 (0.35)	3.61 (0.18)	3.91 (0.16)	4.43 (0.11)	4.17 (0.09)
gpt-4o	System Prompt (Task-Anchored)	3.33 (0.13)	4.42 (0.15)	4.33 (0.38)	3.42 (0.16)	3.61 (0.23)	4.30 (0.13)	4.05 (0.09)
claude-4-sonnet	Temperature (t=0.0)	3.35 (0.14)	4.42 (0.18)	4.61 (0.16)	4.33 (0.13)	4.49 (0.18)	4.56 (0.10)	4.71 (0.07)
claude-4-sonnet	Temperature (t=0.5)	3.38 (0.14)	4.44 (0.18)	4.59 (0.17)	4.36 (0.13)	4.58 (0.15)	4.54 (0.10)	4.70 (0.07)
claude-4-sonnet	Temperature (t=1.0)	3.37 (0.14)	4.44 (0.17)	4.60 (0.16)	4.40 (0.11)	4.56 (0.14)	4.58 (0.09)	4.73 (0.07)
claude-4-sonnet	In-Context Regeneration (General)	3.43 (0.13)	4.39 (0.17)	4.64 (0.12)	4.32 (0.10)	4.40 (0.12)	4.53 (0.10)	4.60 (0.07)
claude-4-sonnet	In-Context Regeneration (Task-Anchored)	3.37 (0.14)	4.44 (0.15)	4.68 (0.11)	3.97 (0.12)	4.07 (0.17)	4.43 (0.10)	4.31 (0.09)
claude-4-sonnet	System Prompt (General)	3.35 (0.13)	4.42 (0.13)	4.24 (0.19)	4.26 (0.12)	4.46 (0.13)	4.70 (0.07)	4.49 (0.07)
claude-4-sonnet	System Prompt (Task-Anchored)	3.59 (0.12)	4.41 (0.10)	4.22 (0.29)	3.93 (0.13)	4.00 (0.16)	4.57 (0.07)	4.14 (0.09)
gemini-2.5-flash	Temperature (t=0.0)	3.59 (0.11)	4.55 (0.16)	4.58 (0.17)	3.99 (0.16)	4.35 (0.18)	4.35 (0.17)	4.48 (0.09)
gemini-2.5-flash	Temperature (t=1.0)	3.60 (0.11)	4.41 (0.15)	4.67 (0.13)	3.98 (0.14)	4.38 (0.19)	4.45 (0.13)	4.44 (0.10)
gemini-2.5-flash	Temperature (t=2.0)	3.54 (0.11)	4.59 (0.13)	4.72 (0.08)	3.88 (0.15)	4.38 (0.18)	4.37 (0.14)	4.47 (0.09)
gemini-2.5-flash	In-Context Regeneration (General)	3.43 (0.11)	4.31 (0.18)	4.61 (0.15)	3.93 (0.14)	4.33 (0.15)	4.40 (0.14)	4.34 (0.11)
gemini-2.5-flash	In-Context Regeneration (Task-Anchored)	3.50 (0.12)	4.37 (0.14)	4.60 (0.15)	3.86 (0.14)	4.35 (0.17)	4.38 (0.13)	4.32 (0.10)
gemini-2.5-flash	System Prompt (General)	3.76 (0.11)	4.60 (0.13)	4.30 (0.22)	4.38 (0.08)	4.40 (0.12)	4.47 (0.14)	4.34 (0.08)
gemini-2.5-flash	System Prompt (Task-Anchored)	3.59 (0.12)	4.54 (0.14)	4.28 (0.34)	4.30 (0.11)	3.65 (0.20)	4.48 (0.11)	4.16 (0.09)

Table 12: Athene-RM-8B Reward by Model, Sampling Strategy, and Task Category.

Model	Sampling Strategy	A	B	C	D	F	G	H
gpt-4o	Temperature (t=0.0)	0.49 (0.09)	0.48 (0.15)	1.04 (0.11)	0.41 (0.17)	1.07 (0.29)	0.75 (0.14)	1.06 (0.10)
gpt-4o	Temperature (t=1.0)	0.43 (0.09)	0.50 (0.14)	1.00 (0.10)	0.42 (0.16)	1.18 (0.27)	0.89 (0.14)	1.13 (0.09)
gpt-4o	Temperature (t=2.0)	0.47 (0.09)	0.51 (0.13)	0.95 (0.10)	0.30 (0.16)	1.23 (0.22)	0.94 (0.15)	1.18 (0.10)
gpt-4o	In-Context Regeneration (General)	-0.16 (0.09)	-0.53 (0.13)	0.22 (0.25)	0.63 (0.14)	0.49 (0.25)	0.59 (0.17)	-0.27 (0.13)
gpt-4o	In-Context Regeneration (Task-Anchored)	-0.00 (0.08)	-0.73 (0.14)	0.27 (0.27)	0.50 (0.16)	0.83 (0.21)	0.72 (0.17)	-0.17 (0.11)
gpt-4o	System Prompt (General)	0.02 (0.09)	-0.40 (0.13)	0.15 (0.26)	0.22 (0.16)	-0.98 (0.26)	0.17 (0.15)	-0.78 (0.13)
gpt-4o	System Prompt (Task-Anchored)	-0.30 (0.10)	-0.59 (0.12)	0.01 (0.30)	-0.15 (0.13)	-1.10 (0.22)	0.10 (0.17)	-0.74 (0.13)
claude-4-sonnet	Temperature (t=0.0)	0.36 (0.10)	0.23 (0.18)	0.74 (0.19)	0.29 (0.16)	1.12 (0.23)	0.87 (0.15)	1.10 (0.08)
claude-4-sonnet	Temperature (t=0.5)	0.38 (0.10)	0.29 (0.16)	0.74 (0.19)	0.28 (0.17)	1.09 (0.23)	0.88 (0.15)	1.12 (0.08)
claude-4-sonnet	Temperature (t=1.0)	0.35 (0.10)	0.28 (0.17)	0.74 (0.17)	0.33 (0.16)	1.19 (0.23)	0.86 (0.14)	1.11 (0.08)
claude-4-sonnet	In-Context Regeneration (General)	0.34 (0.09)	0.03 (0.15)	0.54 (0.13)	0.38 (0.14)	0.86 (0.23)	0.75 (0.13)	0.76 (0.09)
claude-4-sonnet	In-Context Regeneration (Task-Anchored)	0.31 (0.10)	-0.18 (0.14)	0.43 (0.13)	0.13 (0.13)	0.73 (0.18)	0.44 (0.14)	0.81 (0.08)
claude-4-sonnet	System Prompt (General)	0.37 (0.07)	0.17 (0.10)	0.17 (0.24)	0.46 (0.11)	0.24 (0.22)	0.86 (0.11)	0.31 (0.09)
claude-4-sonnet	System Prompt (Task-Anchored)	0.35 (0.08)	0.08 (0.10)	0.36 (0.22)	0.22 (0.11)	-0.11 (0.21)	0.66 (0.11)	-0.04 (0.12)
gemini-2.5-flash	Temperature (t=0.0)	0.08 (0.11)	-0.24 (0.16)	0.50 (0.19)	-0.30 (0.16)	0.80 (0.32)	0.42 (0.20)	0.68 (0.13)
gemini-2.5-flash	Temperature (t=1.0)	0.11 (0.10)	-0.24 (0.13)	0.52 (0.18)	-0.26 (0.16)	0.69 (0.31)	0.61 (0.15)	0.60 (0.13)
gemini-2.5-flash	Temperature (t=2.0)	0.12 (0.10)	-0.17 (0.13)	0.54 (0.17)	-0.28 (0.16)	0.80 (0.29)	0.56 (0.16)	0.57 (0.12)
gemini-2.5-flash	In-Context Regeneration (General)	-0.34 (0.10)	-0.70 (0.18)	0.28 (0.15)	-0.13 (0.16)	0.74 (0.25)	0.60 (0.16)	0.68 (0.16)
gemini-2.5-flash	In-Context Regeneration (Task-Anchored)	-0.05 (0.10)	-0.66 (0.18)	0.60 (0.13)	-0.24 (0.15)	0.82 (0.25)	0.48 (0.16)	0.90 (0.13)
gemini-2.5-flash	System Prompt (General)	0.10 (0.08)	-0.26 (0.11)	0.32 (0.20)	0.37 (0.10)	0.18 (0.26)	0.63 (0.14)	0.12 (0.12)
gemini-2.5-flash	System Prompt (Task-Anchored)	-0.15 (0.09)	-0.66 (0.12)	0.09 (0.19)	0.38 (0.09)	-0.54 (0.23)	0.60 (0.13)	0.03 (0.13)

Table 13: Accuracy by Model, Sampling Strategy, and Evaluation Dataset (for tasks with verifiable rewards).

Model	Sampling strategy	Math-500	Simple-QA
gpt-4o	Temperature (t=0.0)	0.70 (0.06)	0.41 (0.07)
gpt-4o	Temperature (t=1.0)	0.69 (0.06)	0.37 (0.07)
gpt-4o	Temperature (t=2.0)	0.67 (0.06)	0.39 (0.06)
gpt-4o	In-Context Regeneration (General)	0.74 (0.06)	0.34 (0.06)
gpt-4o	In-Context Regeneration (Task-Anchored)	0.68 (0.06)	0.41 (0.07)
gpt-4o	System Prompt (General)	0.66 (0.07)	0.31 (0.06)
gpt-4o	System Prompt (Task-Anchored)	0.67 (0.07)	0.42 (0.07)
claude-4-sonnet	Temperature (t=0.0)	0.76 (0.06)	0.18 (0.05)
claude-4-sonnet	Temperature (t=0.5)	0.78 (0.06)	0.17 (0.05)
claude-4-sonnet	Temperature (t=1.0)	0.79 (0.06)	0.16 (0.05)
claude-4-sonnet	In-Context Regeneration (General)	0.80 (0.05)	0.19 (0.05)
claude-4-sonnet	In-Context Regeneration (Task-Anchored)	0.78 (0.06)	0.19 (0.05)
claude-4-sonnet	System Prompt (General)	0.78 (0.06)	0.21 (0.06)
claude-4-sonnet	System Prompt (Task-Anchored)	0.82 (0.05)	0.23 (0.06)
gemini-2.5-flash	Temperature (t=0.0)	0.71 (0.06)	0.28 (0.06)
gemini-2.5-flash	Temperature (t=1.0)	0.72 (0.06)	0.32 (0.06)
gemini-2.5-flash	Temperature (t=2.0)	0.69 (0.06)	0.30 (0.06)
gemini-2.5-flash	In-Context Regeneration (General)	0.71 (0.06)	0.31 (0.06)
gemini-2.5-flash	In-Context Regeneration (Task-Anchored)	0.71 (0.06)	0.35 (0.07)
gemini-2.5-flash	System Prompt (General)	0.78 (0.06)	0.26 (0.06)
gemini-2.5-flash	System Prompt (Task-Anchored)	0.79 (0.06)	0.31 (0.07)