POLITECNICO DI TORINO
Repository ISTITUZIONALE

A Persona-Augmented Multi-Agent System for Varied Narrative Generation

Original
A Persona-Augmented Multi-Agent System for Varied Narrative Generation / Sillano, A., Arbore, G., De Russis, L.. - ELETTRONICO. - (2026), pp. 32-39. (EICS '26: Engineering Interactive Computing System Patrasso (GR) 30 June - 3 July 2026) [10.1145/3807968.3810931].

Availability:
This version is available at: 11583/3010468 since: 2026-06-29T08:46:33Z

Publisher:
ACM

Published
DOI:10.1145/3807968.3810931

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

13 September 2026

A Persona-Augmented Multi-Agent System for Varied Narrative Generation

Andrea Sillano*
Dipartimento di Automatica e Informatica
Politecnico di Torino
Torino, Italy
andrea.sillano@polito.it

Giuseppe Arbore*
Dipartimento di Automatica e Informatica
Politecnico di Torino
Torino, Italy
giuseppe.arbore@polito.it

Luigi De Russis
Dipartimento di Automatica e Informatica
Politecnico di Torino
Torino, Italy
luigi.derussis@polito.it

Abstract

Large Language Models growing abilities in writing task has allowed them to also tap in the world of creative writing. Instead of following explicit user instructions to operate on a given text, they can be asked to generate complete stories from a single prompt. Although these models yield impressive capabilities in producing well-written narratives, they often lack diversity, relying on established style and semantics seen at training time. Considering this challenge, we present an on-demand persona-augmented agents for narrative generation. By dynamically crafting agents and personas at run-time without relying on hard-coded agent architectures with fixed roles, the system can adapt to match the specific demands of writing prompts. Leveraging LLM persona as proxy for semantic, tone and lexical the system can automatically define tasks demands and workflow, thus crafting more varied and heterogeneous outputs. Our approach demonstrates that agentic platforms can consistently surpass the single-LLM baseline. With gains of +0.28 in semantic and +0.18 in style embedding distances, the generated outputs by our best proposal, exhibit higher variety, validating the effectiveness of multi-agent persona augmentation for open writing tasks.

CCS Concepts

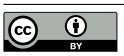
• **Computing methodologies** → **Multi-agent systems**; **Intelligent agents**; *Natural language processing*; • **Human-centered computing** → Human computer interaction (HCI).

Keywords

multi-agent systems; persona-based agents; workflow automation; agent orchestration; personalization; AI-persona; storytelling; story generation

ACM Reference Format:
Andrea Sillano, Giuseppe Arbore, and Luigi De Russis. 2026. A Persona-Augmented Multi-Agent System for Varied Narrative Generation. In *Companion Proceedings of the 18th ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS Companion '26), June 30–July 03, 2026, Patras, Greece.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3807968.3810931

*Both authors contributed equally to this research.



This work is licensed under a Creative Commons Attribution 4.0 International License.
EICS Companion '26, Patras, Greece
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2669-9/26/06
https://doi.org/10.1145/3807968.3810931

1 Introduction

The adoption of Large Language Models (LLMs) has transformed the landscape of natural language generation, enabling systems capable of producing coherent and fluent text across a wide range of domains [6, 13]. These systems have progressively moved beyond structured and task-specific generation toward open-ended and creative applications. Among these, creative writing represents one of the most demanding challenges [10], since it requires more than just grammatical correctness and coherence. It is instead characterized by stylistic richness, expressiveness, originality, and semantic variety. Advancements in LLMs have shown their abilities in generating stories with minimal input [42, 44, 46], opening possibilities for human-AI collaborative storytelling and automated content creation. Unlike factual or instructional text generation, creative writing demands that a model navigate the tension between known patterns and creative freedom.

Despite their impressive capabilities, LLMs tend to produce stylistically homogeneous and semantically repetitive text [12, 28, 45], manifesting in narrative generation as predictable plots, style flattening, and lexical homogenization. This limitation stems from training on corpora dominated by mainstream conventions, biasing models toward prevalent patterns and marginalizing unconventional narrative forms and stylistically distinct voices [1, 28, 37, 45].

Prior work has explored strategies to improve generation variety, including prompt engineering [39, 49] and post-training and decoding techniques [5, 31], though these approaches remain largely static, requiring retraining or curated style conditions. A promising alternative lies in AI-personas, as LLMs are sensitive to identity-framing in their context window, when prompted to adopt a specific role, models shift their lexical choices, syntactic patterns, tonal register, and ideological leanings [22, 25]. In AI narrative generation, identity framing can act as soft constraints on the output space, steering models away from default behavior toward more distinctive outputs, positioning as a proxy for writing style that enables controllable and diverse generation. Moreover, there is a growing interest in structural alternatives that can elicit diversity through collaboration and task decomposition. In this direction, multi-agent frameworks have shown promise for complex generative tasks; however, existing architectures often employ rigid, pre-defined agent roles, which can limit flexibility across diverse creative contexts. Recent work demonstrates that LLM-based multi-agent systems can decompose complex generative tasks by distributing subtasks across specialized agents and coordinating their interactions through structured communication or orchestration mechanisms [3, 7, 16, 19, 40]. Role-playing and team-based designs have been shown effective in

domains ranging from collaborative artifact generation to model orchestration and interactive simulations [16, 40]. Nevertheless, the widespread use of static role taxonomies can hinder adaptability and transfer to diverse creative contexts, where the required expertise and interaction styles may shift dynamically over time [11, 16, 30]. We propose a persona-augmented agentic architecture for story generation in which a set of LLM persona-based agents is synthesized at run time from the story prompt. Rather than relying on a fixed, pre-defined structure, the system dynamically determines and creates needed personas and how the persona-based agents should collaborate to produce the target story, allowing overall architecture to adapt to the specific demands of the story. We introduce two instantiations of this approach: *Relay* and *Ensemble*. In *Relay*, personas operate sequentially, each conditioning on the accumulated context. In *Ensemble*, personas draft complementary parts in parallel, with an integrator persona merging them into a coherent story. after which an additional persona acts as an integrator that merges the drafts into a single coherent story. Across both operating modes, persona-based agent generation serves as a structural mechanism to increase output variety compared to a single-LLM baseline. Our system diversifies generation at the architectural level, assigning distinct LLMs personas and roles to each agent, thereby conditioning their outputs on different stylistic and semantic premises. The intended notion of diversity is not the stochastic variation across repeated runs of the same prompt, but its deliberate intra-system variety, where the different operating modes, persona compositions, and agent roles pushes outputs apart. This divergence is reflected in measurable embedding-based scores, where our system's outputs are expected to occupy more distant regions of the semantic space relative to those produced by a single-LLM baseline. To validate the effectiveness of our approach we conducted an evaluation comparing our persona-augmented multi-agent system against a single model baseline using the WritingPrompt dataset [9]. Our experiments assessed both style divergence and semantic divergence computed by embedding similarity across generated narratives, over a randomly sampled subset of 500 prompts sampled from the WritingPrompts dataset. Experimental results demonstrate that our framework is able to outperform the single LLM, with one mode achieving the best gains of +0.28 in semantic and +0.18 in style embedding variety. These findings validate our central hypothesis: that dynamically instantiated, persona-augmented agents are an effective mechanism for expanding the generative space of LLMs in creative writing tasks. More broadly, our results suggest that multi-agent persona augmentation represents a strategy for mitigating the homogeneity bias inherent in standard LLM generation, with potential applicability beyond narrative generation to other open-ended creative tasks.

2 Related Work

Story generation has been an open challenge in natural language processing well before the LLM-era. First approaches to story generation are rooted in sequence-to-sequence architecture that allowed end-to-end narrative generation from story prompts [9]. The advent of LLMs shifted the paradigm from structured planning toward prompt-conditioned generation, enabling systems to produce coherent narratives with minimal task-specific supervision [42, 44]. However, generating long-form stories, instead of just sentence completion [47], surfaced new challenges around plot coherence, creativity, and style. Work in this direction highlights that even when models produce fluent and structurally sound narratives, they systematically converge on the same surface patterns: predictable plot arcs, character archetypes, and a narrow register of prose style, regardless of how varied the input prompt is [28, 45]. One of possible causes of this behavior can be traced back to the statistical properties of pre-training corpora. As reported in [2, 8], standard filtering pipelines disproportionately remove minority dialects and non-Western perspectives, concentrating the training distribution around mainstream Western conventions and causing models to internalize a strong prior over a narrow stylistic slice. Other work suggests that part of this homogenization may be an unintended consequence of post-training alignment, which can reduce semantic variety and compress the diversity of conceptual representations compared to non-aligned counterparts [27, 29]. In the pursuit of expanding the diversity of LLM outputs, several complementary lines of work have been explored. Sampling-based approaches such as top-k [23], nucleus top-p [15], and min-p [26] modulate the token distribution. Temperature scaling and adaptive temperature strategies, instead, provide further control, however they expose a quality-diversity tension [4, 41, 48]. A second direction leverages prompting to push divergence in the outputs. Proposed methods include self critique [20], sequential iterative prompting [14], and verbalized probability distributions over candidate responses [49]. A third direction addresses diversity through post-training alignment, making it as the main goal. DivPO [21] directly addresses this by reformulating preference optimization to select the most diverse high-quality response as the chosen sample and the least diverse as the rejected one, achieving substantial gains. Other work pursue related post-training objectives tailored to creative writing [5]. LLMs have been proved to be sensitive to identity framing inside their context window [24]. When prompted to adopt a specific identity, defined by writing style, professional role, cultural background, or personality traits, models exhibit measurable shifts in lexical choice, syntactic patterns, tonal register, and opinions [22, 25]. This phenomenon has been characterized as the persona effect, [17] quantify how personality conditioning alters reasoning and affective expression in dialogue generation. The ability of LLMs to adopt and maintain roles has been explored in role-playing agent research. CharacterGPT [35] proposes a identity reconstruction framework to incrementally update the personas by extracting traits, while Solo Performance Prompting [43] transforms a single LLM into a multi-persona agent that cognitively simulates multiple viewpoints to enhance problem-solving. Persona conditioning has also been studied as a driver of output diversity. Scaffolding Creativity [38] examines how distinct AI personas affect creativity outcomes, finding that persona-guided divergent interaction modes improve perceived creativity and ideation breadth relative to standard LLM.

In parallel to decoding-based and prompting-based methods for eliciting diversity, a complementary line of work pursues variety and controllability through structural interventions. Instead of relying on a single model, agentic AI systems coordinate multiple specialized capabilities behind a unified interface, enabling planning, reasoning, and autonomous execution of multi-step workflows

end-to-end [36]. Contemporary agentic architectures rely on components such as memory, structured knowledge, reflection mechanisms, and orchestration frameworks to handle cooperation [19]. Using collaboration to decompose long-horizon generation into tractable subproblems and inject heterogeneous perspectives that a single-pass generator may fail to sustain. Despite their promise, many LLM-based multi-agent systems are engineered around *static* design commitments: agents are instantiated with pre-defined roles and interact via fixed communication rules (e.g., who communicates with whom, in what order, and through which intermediate artifacts). Prior work identifies *agent profiling* and *communication* as core architectural dimensions, and surveys approaches where these choices are specified upfront through prompt templates, role descriptions, and preset protocols [11]. For example, [3] guides agents via a shared *reflector* with role-conditioned reflections, while [7] implements an asynchronous, multi-level scheme where higher-level agents propagate decisions downward, both embedding fixed coordination patterns at design time. These design patterns have been applied to creative generation, including long-form narrative writing, where multi-agent collaboration can help maintain global coherence by separating high-level story planning from local realization. A recent work [18] grounds story generation in a multi-step workflow inspired by narrative theory, where specialized planning agents develop core story elements and writing agents subsequently realize the narrative through shared intermediate artifacts. The framework reports improvements in human preference over single-agent baselines, suggesting that specialization and iterative coordination can mitigate failure modes of single-pass generation in long narratives [18]. Nevertheless, the approach still relies on a fixed set of agent roles and a pre-specified interaction protocol, which reflects a broader limitation of current multi-agent architectures: the widespread use of static role taxonomies can hinder adaptability and transfer to diverse creative contexts, where the required expertise, interaction styles, and coordination dynamics may shift over time [11, 16, 30]. This motivates designs in which agent personas and collaboration policies are treated as *dynamic* constructs that can be instantiated and adapted at run-time as a function of task demands and user context, rather than being determined entirely a priori.

3 Multi-Agent System Overview

To foster variety in LLM outputs without having to alter its inner workings, we propose a persona-augmented agentic architecture aimed at promoting stylistic and semantic heterogeneity in story generation. The main strength of our system lies in its ability to synthesize tasks, persona agents, and roles at run-time based on writing prompts. To accomplish this goal our architecture is organized around four modular components integrated within a custom-tailored pipeline: a (i) *Decomposer*, a (ii) *Persona Crafter*, an (iii) *Agent Factory*, and an optional (iv) *Synthesizer*. The entire pipeline is managed by the orchestrator, which is the entry and output point of the system and acts as the coordinator between the agents and the components. At initialization, the architecture consists solely of the orchestrator, as it dynamically shapes itself during execution, each of the previously mentioned components resides within the orchestrator. We designed two working modes, *Relay* and *Ensemble*, enabled by the non-fixed nature of the architecture. Because the orchestrator dynamically shapes the pipeline at run-time, we were able to design both modes without any structural modification to the system. In Relay mode, the pipeline follows a sequential approach in which each agent receives and builds upon the output of its predecessor, creating a chain of contextually aware contributions. In Ensemble mode, up to five agents work concurrently on the same story, each being assigned a distinct narrative segment. Once all agents have completed their respective portions, the *Synthesizer* aggregates and formats the individual outputs into a single result to be delivered.

3.1 Pipeline Components

The core components of the pipeline are designed as self-contained modules within the orchestrator agent, each responsible for executing a single step in the workflow.

Decomposer. It is the first component invoked by the orchestrator upon receiving a user prompt. Its role is to analyze the prompt and produce a structured task graph that defines the execution plan for the rest of the pipeline. Internally, the *Decomposer* operates through a mode-conditioned prompt. In *Relay* mode, it designs a sequential chain of dependent tasks, ranging from three to eight depending on prompt complexity and spanning pre-writing, drafting, and revision stages In Ensemble mode, the *Decomposer* instead partitions the narrative into self-contained parallel segments, each representing an independent portion of the story. In this case, no inter-task dependencies are set, as all segments are designed to be executed concurrently. In both modes, each task produced by the *Decomposer* carries a precise description of its goal, the full context required for independent execution, an assigned model. Model assignment is also delegated to the *Decomposer*, which selects from a predefined list per task rather than applying a uniform model. This output is then consumed by the *Persona Crafter* and *Agent Factory*.

Persona Crafter. It receives the structured task list produced by the *Decomposer* and synthesizes a distinct AI persona with its own role for each task. For every task t_i, it generates a natural language description of the agent's age, background, writing tradition, and stylistic approach, following the "descriptive persona" formulation [22] rather than structured attribute tables. The *Persona Crafter* is explicitly instructed to maximize distinctiveness across the set of generated AI persona, ensuring that no two agents share the same tone, perspective, or narrative sensibility. This is a design choice aiming to promote stylistic and semantic heterogeneity in the final output. Each persona is matched to its corresponding task via the `task_id` field, preserving the structural alignment between the task graph and the agent population that will be instantiated in the subsequent stage. The resulting set of personas $\{p_1, \ldots, p_n\}$ is then forwarded to the *Agent Factory*.

Agent Factory. It is responsible for instantiating the agents that will carry out the tasks defined by the *Decomposer*. For each task t_i, it receives the corresponding persona p_i from the *Persona Crafter* and constructs a fully configured agent a_i. Agent instantiation is achieved by composing a system-level instruction prompt that embeds the synthesized persona directly into the agent's behavioral context, effectively conditioning its generative behavior. Beyond

persona conditioning, the instruction prompt enforces a set of task-level constraints shared across all agents: outputs are bounded to a maximum of 300 words, must be returned as plain prose without any additional commentary or formatting, and must remain consistent with any prior narrative context received. This last constraint is particularly relevant in Relay mode, where each agent inherits the output of its predecessor and is expected to continue its own distinct voice. The result is a set of agents that are expected to be both structurally uniform in their output format and stylistically heterogeneous in their generative behavior

Synthesizer. It is an optional component exclusive to Ensemble mode, invoked as the final stage of the pipeline once all parallel agents have completed their segments. Its role is to reconcile and unify the individual contributions into a single, coherent narrative, operating as a story weaver, it reads all segments holistically, tries to resolve any contradictions and to preserve meaningful details from each contribution. The expected result is a unified story with a single consistent arc and tone, free from any meta-commentary or trace of its multi-agent origin. The *Synthesizer* is itself an LLM-based agent, meaning the harmonization process is entirely generative, the model is not given explicit merging rules but is instead instructed to exercise narrative judgment in producing the final output.

3.2 On-Demand Persona-Based Agent Generation Pipeline

The system pipeline can be divided into four main steps, each involving a dedicated component:

1. Prompt Analysis. The pipeline is initiated when the orchestrator receives a writing prompt. The *Decomposer* is then invoked to analyze the prompt and break it down into a set of atomic, self-contained tasks $\{t_1, \ldots, t_n\}$, each representing a distinct narrative or stylistic requirement. Each task is also given a dependency attribute to know which tasks must be completed before it can be executed.

2. Persona Crafting & Agent Instantiation. For each decomposed task t_i, the *Persona Crafter* synthesizes a contextually appropriate AI persona p_i, tailored to the specific requirements of that task. The *Agent Factory* subsequently instantiates a dedicated agent a_i conditioned on p_i, aiming to provide each agent with a distinct stylistic and behavioral identity.

3. Agent Execution. The orchestrator assigns each task t_i to its corresponding agent a_i and manages the execution order according to the selected operating mode, sequential in Relay mode, concurrent in Ensemble mode. Each agent produces a partial response r_i upon completing its assigned task.

4. Output Aggregation. Once all agents have completed execution, the orchestrator collects the individual outputs $\{r_1, r_2, \ldots, r_n\}$ or the final output R if in Relay mode. In fact, when Relay mode is used, the final output is already a narrative by virtue of the sequential chaining of agents, and is returned directly to the user without further processing. In Ensemble mode, the collected outputs are forwarded to the *Synthesizer*, which harmonizes and formats the parallel contributions into a single, cohesive response R returned to the user.

3.3 Pipeline Operating Modes

As stated in the previous sections, the pipeline has been designed to operate in two different modes. The two operating modes, *Relay* and *Ensemble*, are not configurational variants; they represent two different approaches about how architectural diversity can drive output variety. Since the pipeline topology is determined at runtime, it can support the two distinct execution strategies requiring no modification to the underlying system. The mode acts as a high-level directive that propagates through the pipeline, shaping how the *Decomposer* partitions the prompt, how agents are instantiated and scheduled, and whether the *Synthesizer* is invoked at the end. The two modes address the problem of output variety from different angles. Relay mode pursues divergence's output through *depth*, by chaining agents sequentially, each conditioned on the work of its predecessor, the narrative evolves incrementally through a series of distinct perspectives and goals. Ensemble mode, contrarily, pursues heterogeneity through *breadth*, agents operate independently and in parallel, each contributing a segment of the story from their own persona-conditioned vantage point. The *Synthesizer* is responsible for reconciling their contributions into a unified whole. Together, these two modes allow the system to be evaluated under qualitatively different conditions, providing a richer basis for assessing the impact of persona-based generation on output variety.

4 Implementation

Our system is implemented in Python as a schema-guided multi-agent pipeline built on the *OpenAI Agents SDK* [32]. Each stage of the pipeline (*Prompt Analysis*, *Persona Crafting & Agent Instantiation*, *Agent Execution*, and *optional Output Aggregation*) is executed as an explicit LLM invocation, while intermediate artifacts are exchanged via typed schemas. The Single baseline is implemented as a single-call generation: we query `gpt-5.2` [34] with the unmodified story prompt and directly return the model output. Both persona-augmented variants first perform *Prompt Analysis* with `gpt-5.2` [34] to interpret the prompt and decompose it into structured writing tasks. Next, *Persona Crafting & Agent Instantiation* is performed with `gpt-5-mini` [33], which generates a compact persona specification for each task and is used to instantiate the corresponding writer agents. Then, *Agent Execution* is carried out by running the instantiated writer agents with a model chose by the orchestrator based on expected stask complexity between `gpt-5.2`, `gpt-4o`, and `gpt-4o-mini`, producing either story-level outputs (Relay) or independent story segments (Ensemble). Finally, *Output Aggregation* is applied only in Ensemble: a dedicated synthesizer step merges the independently generated segments using `gpt-5-mini` [33], resolving contradictions and normalizing style into a single coherent story.

5 Experiments

To evaluate the effectiveness of our persona-augmented agentic architecture, we conduct an experiment aimed at assessing whether the proposed system produces outputs that are more semantically

and stylistically distant from those generated by a single-LLM baseline one. Rather than evaluating narrative quality, we focus on measurable diversity, defined as the embedding-space distance between our system's outputs and the baseline. This framing reflects our core hypothesis, that conditioning generation on dynamically synthesized personas and a structured multi-agent pipeline, induces outputs that occupy more distant regions of the semantic and style embedding space.

Dataset. We evaluate our system using the WritingPrompts[1] dataset, a large-scale collection of creative writing prompts. From this dataset we sampled a subset of 500 distinct prompts. Each prompt serves as the single input to both the baseline and our system using the two modes, ensuring that any observed differences in output can be attributable to the architectural properties of the pipeline rather than variation in the input conditions.

Models. We compare our system which use GPT-5.2, GPT-5-mini, GPT-4o and GPT-4o-mini against a single-LLM baseline consisting of GPT-5.2, instructed to generate a short story directly from the writing prompt, without any decomposition, persona assignment, or multi-agent coordination. This baseline represents the standard zero-shot generation setting and serves as the reference point against which output diversity is measured.

Measures. To measure semantic and stylistic variety, we employ two complementary embedding models. Semantic diversity is assessed using `jina-embeddings-v3`[2], a state-of-the-art text embedding model that maps outputs into a high-dimensional semantic space. Stylistic diversity is assessed using `Style-Embedding`[3] model, which capture surface-level features. For each pair of outputs, we compute cosine distance in the respective embedding spaces, with higher distances indicating greater diversity following the approach in [5]. We also collect a set of operational metrics to characterize the system's behavior and resource usage across conditions. For each run we record: (i) the total generation time, (ii) the number of input and output tokens consumed, (iii) the number of agents instantiated that also will match the number of synthesized persona.

5.1 Evaluation Protocol

For each writing prompt, we generate one output per condition, starting with the single-LLM baseline, our system in Relay mode, and our system in Ensemble mode. Then, we measure the pairwise cosine distance between each system outputs and the baseline in both embedding spaces, quantifying how far our system's generations deviate from the baseline both semantically and stylistically. We store each run's metric as they provide a transparency account of the computational cost introduced by the multi-agent architecture relative to the single-LLM baseline. After the all the prompts have been ran, we proceed to compute semantic and style variation across the different outputs.

[1]https://huggingface.co/datasets/euclaise/writingprompts , last visited on March 2026
[2]https://huggingface.co/jinaai/jina-embeddings-v3 , last visited on March 2026
[3]https://huggingface.co/AnnaWegmann/Style-Embedding , last visited on March 2026

6 Results and Discussion

Figure 1 reports the average semantic and style divergence score from the baseline across both embedding spaces for the two system modes. By construction, the single-LLM baseline serves as the reference point with a self-distance of 0 in both spaces. Standard deviations are small across all conditions, and all reported means fall within the 95% confidence intervals, indicating stable and consistent behavior across the 500 prompts. The *Relay* mode achieves the highest semantic divergence, with a mean cosine distance of 0.276 (σ=0.06) from the baseline in the semantic embedding space. This shows a notable semantic shift that can be attributed to the sequential nature of the *Relay* pipeline, where each agent builds upon the output of the previous one, conditioned on a distinct synthesized identity. This chaining effect causes the narrative to be progressively refined by distinct AI personas, drifting from the semantic territory that a zero-shot model would produce, as each agent introduces LLM generated persona-driven reinterpretations that compound throughout the pipeline. In the stylistic embedding space, *Relay* records a mean distance of 0.198 (σ=0.18), the highest stylistic divergence observed. The larger standard deviation reflects the sensitivity of surface-level features to identity variation as strong stylistic personas produce more pronounced shifts, while neutral ones yield smaller deviations.

The *Ensemble* mode records a mean semantic distance of 0.190 (σ=0.06) from the baseline. The lower semantic distance is consistent with the *Ensemble*'s aggregation mechanism: parallel outputs are merged, averaging out semantic departures toward a more centrist semantic position. The stylistic distance is the smallest across conditions at 0.040 (σ=0.07), as fusion smooths individual stylistic deviations toward the neutral baseline output. Figure 2 further illus-

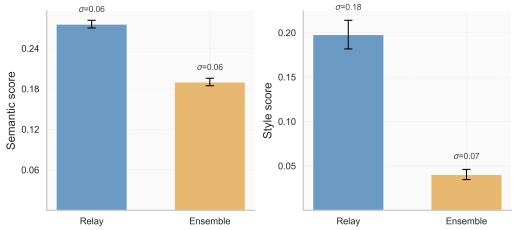


Figure 1: Mean divergence metrics for Relay and Ensemble methods

trates the score distributions via violin plots. In the semantic space, *Relay* exhibits a broad, almost symmetric distribution centered in its mean, reflecting consistent and evenly spread divergence from the baseline across prompts. *Ensemble*'s semantic distribution is narrower and bottom-heavy, with mass concentrated in the lower range and a thin upper tail, confirming that while most outputs diverge moderately, a small subset reaches higher semantic distances, likely corresponding to prompts where the parallel agents happen to produce strongly divergent sections that survived the smoothing process. In the stylistic space, the contrast between the two modes is more evident. *Relay*'s distribution is wide-bodied with a long upper tail extending beyond 1.0, indicating that certain prompt-persona combinations trigger pronounced stylistic departures, which explains the large standard deviation (σ=0.18) observed. *Ensemble*'s

Table 1: Mean operational and divergence metrics per condition across 500 prompts.

Condition	Sem. Divergence	Style Divergence	Time (s)	Agents	AI Personas	Input Tokens	Output Tokens	Total Tokens
Single	(ref.)	(ref.)	14.02	-	-	87	503	590
Relay	**0.276** ↑	**0.198** ↑	106.11	4.3	4.3	3,549	5,978	9,526
Ensemble	0.190 ↑	0.040	58.46	5.3	5.3	4,638	3,999	8,637

style distribution, by contrast, is heavily concentrated near zero with only sparse high-distance outliers, visually confirming the smoothing effect of output fusion on surface-level features. The violin plots reveal that *Relay* not only achieves higher mean divergence but also explores a broader and more varied region of both embedding spaces, while *Ensemble* produces more compressed and predictable output distributions.

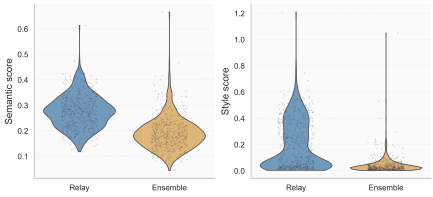


Figure 2: Divergence metrics distribution for Relay and Ensemble methods

From the operation metrics, we discovered that while the baseline uses a single 5.2 model, by default, *Relay* mixes models but is dominated by gpt-5.2 and gpt-5-mini (500 each), with a smaller contribution from gpt-4o (388) and negligible gpt-4o-mini (2), suggesting most work is concentrated in the stronger models with limited routing. *Ensemble* distributes load more evenly across all four models (500 each), indicating a more balanced multi-model aggregation where multiple agents contribute comparably. However, the multi-agent architecture introduces a measurable computational overhead relative to the single-LLM baseline, quantified across generation time, agent and AI persona counts, and token consumption. As shown in Table 1, the single-LLM baseline incurs the lowest cost (14.02s, no agent overhead). *Relay* is the most expensive at 106.11s (+656.8%), due to sequential execution where each agent awaits the previous output. *Ensemble*, despite coordinating more agents on average (5.3), achieves 58.46s by running agents in parallel, bounding time by the slowest individual rather than their sum. Both modes instantiate 3–7 agents per prompt, with *Ensemble* consistently reaching higher counts. The most important cost is represented by token consumption. The baseline, by design, is the most lightweight condition, consuming a mean of 590 tokens per prompt (87 input, 503 output). *Relay* token consumption scales heavily, reaching a mean of 9,526 tokens per prompt (3,549 input, 5,978 output), a substantial increase over the baseline. The elevated input token count arises from the carry of prior agent outputs into each successive agent's context, while the high output count reflects the cumulative generation across the chain. *Ensemble*, despite instantiating on average a larger number of agents, consumes on average 8,637 tokens per prompt (4,638 input, 3,999 output). The higher input token count compared to *Relay* reflects the need to broadcast the original prompt and LLM persona context to each agent independently, while the lower output count is consistent with each agent producing a single self-contained contribution. Together, these metrics characterize the computational trade-off introduced by the multi-agent architecture: meaningful gains in output divergence come at the cost of increased generation time, with *Relay* paying this cost sequentially and *Ensemble* partially amortizing it through parallelism. Table 1 summarize the experiments results.

7 Conclusions

This work proposes a persona-augmented multi-agent architecture aiming to improve variety in generated outputs and evaluated it against a single-LLM zero-shot baseline across 500 writing prompts. Both modes have shown to be able to shift outputs toward more distant semantic regions, while only *Relay* yields a meaningful stylistic shift, confirming the core hypothesis. *Relay* achieves stronger overall divergence through cumulative persona-chaining, *Ensemble* instead, produces more moderate divergence but benefits from lower generation time via parallel execution. come with trade-off mainly in computational time and token usage, reflecting the inherent overhead of coordinating multiple agents. Across all conditions, distributions are stable and consistent, with small standard deviations and means well within 95% confidence intervals. However several limitations of the present study should be acknowledged. The evaluation is limited to 500 prompts of the dataset, which, although sufficient for stable mean estimates, may not fully capture the variability of a broader distribution. All conditions rely on the GPT model family, leaving open whether diversity gains generalize to other model families which may yield different identity representations. Finally, the evaluation does not assess narrative quality or coherence, higher embedding-space distance does not preclude degradation in readability or plot consistency. Nevertheless, our findings suggest that dynamically synthesized LLMs personas can function as effective semantic and stylistic control signals, capable shifting model's output. Instead of being used as stylistic decorators, AI personas can redirect the generation process toward distinct regions of the output space, serving as tractable proxies for narrative variety in the absence of explicit diversity supervision. This shift is not achievable by persona conditioning alone, it is the supporting agentic architecture that transforms individual persona-conditioned outputs into a compounding source of variety. This work can be further extended from the current limitations, including evaluation of intra-system diversity across multiple runs per prompt, incorporating quality metrics (readability, entailment-based coherence, and human evaluation) to ensure divergence gains do not compromise narrative integrity, and extending comparisons to additional model families. Optimizing the *Decomposer*'s allocation for a target divergence-cost trade-off and exploring adaptive persona synthesis strategies are further promising directions.

References

[1] Badr AlKhamissi, Muhammad ElNokrashy, Mai Alkhamissi, and Mona Diab. 2024. Investigating Cultural Alignment of Large Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 12404–12422. doi:10.18653/v1/2024.acl-long.671

[2] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (Virtual Event, Canada) *(FAccT '21)*. Association for Computing Machinery, New York, NY, USA, 610–623. doi:10.1145/3442188.3445922

[3] Xiaohe Bo, Zeyu Zhang, Quanyu Dai, Xueyang Feng, Lei Wang, Rui Li, Xu Chen, and Ji-Rong Wen. 2024. Reflective Multi-Agent Collaboration based on Large Language Models. In *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (Eds.), Vol. 37. Curran Associates, Inc., 138595–138631. doi:10.52202/079017-4397

[4] John Chung, Ece Kamar, and Saleema Amershi. 2023. Increasing Diversity While Maintaining Accuracy: Text Data Generation with Large Language Models and Human Interventions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 575–593. doi:10.18653/v1/2023.acl-long.34

[5] John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. 2025. Modifying Large Language Model Post-Training for Diverse Creative Writing. arXiv:2503.17126 [cs.CL] https://arxiv.org/abs/2503.17126

[6] Raphael Souza de Oliveira and Erick Giovani Sperandio Nascimento. 2026. Transformer-based large language foundation models for text generation: A comprehensive literature review for different languages and application domains. *Information Processing & Management* 63, 2, Part B (2026), 104477. doi:10.1016/j.ipm.2025.104477

[7] Ziluo Ding, Zeyuan Liu, Zhirui Fang, Kefan Su, Liwen Zhu, and Zongqing Lu. 2024. Multi-Agent Coordination via Multi-Level Communication. In *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (Eds.), Vol. 37. Curran Associates, Inc., 118513–118539. doi:10.52202/079017-3763

[8] Jesse Dodge, Maarten Sap, Ana Marasovic, William Agnew, Gabriel Ilharco, Dirk Groeneveld, and Matt Gardner. 2021. Documenting the English Colossal Clean Crawled Corpus. *CoRR* abs/2104.08758 (2021). arXiv:2104.08758 https://arxiv.org/abs/2104.08758

[9] Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical Neural Story Generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Iryna Gurevych and Yusuke Miyao (Eds.). Association for Computational Linguistics, Melbourne, Australia, 889–898. doi:10.18653/v1/P18-1082

[10] Carlos Gómez-Rodríguez and Paul Williams. 2023. A Confederacy of Models: a Comprehensive Evaluation of LLMs on Creative Writing. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 14504–14528. doi:10.18653/v1/2023.findings-emnlp.966

[11] Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large Language Model Based Multi-agents: A Survey of Progress and Challenges. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, Kate Larson (Ed.). International Joint Conferences on Artificial Intelligence Organization, 8048–8057. doi:10.24963/ijcai.2024/890 Survey Track.

[12] Yanzhu Guo, Guokan Shang, and Chloé Clavel. 2025. Benchmarking Linguistic Diversity of Large Language Models. *Transactions of the Association for Computational Linguistics* 13 (11 2025), 1507–1526. arXiv:https://direct.mit.edu/tacl/article-pdf/doi/10.1162/TACL.a.47/2566986/tacl.a.47.pdf doi:10.1162/TACL.a.47

[13] Desta Haileselassie Hagos, Rick Battle, and Danda B. Rawat. 2024. Recent Advances in Generative AI and Large Language Models: Current Status, Challenges, and Perspectives. *IEEE Transactions on Artificial Intelligence* 5, 12 (2024), 5873–5893. doi:10.1109/TAI.2024.3444742

[14] Shirley Anugrah Hayati, Minhwa Lee, Dheeraj Rajagopal, and Dongyeop Kang. 2024. How Far Can We Extract Diverse Perspectives from Large Language Models?. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 5336–5366. doi:10.18653/v1/2024.emnlp-main.306

[15] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*. https://openreview.net/forum?id=rygGQyrFvH

[16] Sirui Hong, Mingchen Zhuge, Jonathan Chen, Xiawu Zheng, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, zili wang, Steven Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, Chenglin Wu, and Jürgen Schmidhuber. 2024. MetaGPT: Meta Programming for A Multi-Agent Collaborative Framework. In *International Conference on Learning Representations*, B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (Eds.), Vol. 2024. 23247–23275. https://proceedings.iclr.cc/paper_files/paper/2024/file/6507b115562bb0a305f1958ccc87355a-Paper-Conference.pdf

[17] Tiancheng Hu and Nigel Collier. 2024. Quantifying the Persona Effect in LLM Simulations. arXiv:2402.10811 [cs.CL] https://arxiv.org/abs/2402.10811

[18] Fantine Huot, Reinald Kim Amplayo, Jennimaria Palomaki, Alice Shoshana Jakobovits, Elizabeth Clark, and Mirella Lapata. 2025. Agents'Room: Narrative Generation through Multi-step Collaboration. In *International Conference on Learning Representations*, Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (Eds.), Vol. 2025. 5150–5183. https://proceedings.iclr.cc/paper_files/paper/2025/file/0fbc8a83d93dd8021a4dd8d2d34138eb-Paper-Conference.pdf

[19] Dr. Sanjay Nakharu Prasad Kumar. 2025. Building Scalable and Reliable Agentic AI Systems: A Technical Blueprint for Autonomous Intelligence. *Global Journal of Engineering and Technology Research* (2025). https://api.semanticscholar.org/CorpusID:283433037

[20] Preethi Lahoti, Nicholas Blumm, Xiao Ma, Raghavendra Kotikalapudi, Sahitya Potluri, Qijun Tan, Hansa Srinivasan, Ben Packer, Ahmad Beirami, Alex Beutel, and Jilin Chen. 2023. Improving Diversity of Demographic Representation in Large Language Models via Collective-Critiques and Self-Voting. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 10383–10405. doi:10.18653/v1/2023.emnlp-main.643

[21] Jack Lanchantin, Angelica Chen, Shehzaad Dhuliawala, Ping Yu, Jason Weston, Sainbayar Sukhbaatar, and Ilia Kulikov. 2025. Diverse Preference Optimization. arXiv:2501.18101 [cs.CL] https://arxiv.org/abs/2501.18101

[22] Ang Li, Haozhe Chen, Hongseok Namkoong, and Tianyi Peng. 2025. LLM Generated Persona is a Promise with a Catch. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems Position Paper Track*. https://openreview.net/forum?id=qh9eGtMG4H

[23] Dong Li, Ruoming Jin, Jing Gao, and Zhi Liu. 2020. On Sampling Top-K Recommendation Evaluation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '20)*. ACM, 2114–2124. doi:10.1145/3394486.3403262

[24] Marlene Lutz, Indira Sen, Georg Ahnert, Elisa Rogers, and Markus Strohmaier. 2025. The Prompt Makes the Person(a): A Systematic Evaluation of Sociodemographic Persona Prompting for Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 23212–23237. doi:10.18653/v1/2025.findings-emnlp.1261

[25] Manuj Malik, Jing Jiang, and Kian Ming A. Chai. 2024. An Empirical Analysis of the Writing Styles of Persona-Assigned LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 19369–19388. doi:10.18653/v1/2024.emnlp-main.1079

[26] Nguyen Nhat Minh, Andrew Baker, Clement Neo, Allen G Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. 2025. Turning Up the Heat: Min-p Sampling for Creative and Coherent LLM Outputs. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=FBkpCyujtS

[27] Behnam Mohammadi. 2024. Creativity Has Left the Chat: The Price of Debiasing Language Models. arXiv:2406.05587 [cs.CL] https://arxiv.org/abs/2406.05587

[28] Kibum Moon, Adam E. Green, and Kostadin Kushlev. 2025. Homogenizing effect of large language models (LLMs) on creative diversity: An empirical comparison of human and ChatGPT writing. *Computers in Human Behavior: Artificial Humans* 6 (2025), 100207. doi:10.1016/j.chbah.2025.100207

[29] Sonia Krishna Murthy, Tomer Ullman, and Jennifer Hu. 2025. One fish, two fish, but not the whole sea: Alignment reduces language models' conceptual diversity. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Association for Computational Linguistics. doi:10.18653/v1/2025.naacl-long.561

[30] Gayathri Nettem, M. Disha, Aavish Gilbert J., Skanda Shreesha Prasad, and S. Natarajan. 2025. AgentFlow: A Context Aware Multi-Agent Framework for Dynamic Agent Collaboration. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence - Volume 1: ICAART*. INSTICC, SciTePress, 687–693. doi:10.5220/0013375700003890

[31] Minh Nhat Nguyen, Andrew Baker, Clement Neo, Allen Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. 2025. Turning Up the Heat: Min-p Sampling for Creative and Coherent LLM Outputs. arXiv:2407.01082 [cs.CL] https://arxiv.org/abs/2407.01082

[32] OpenAI. [n. d.]. OpenAI Agents SDK. https://openai.github.io/openai-agents-python/

[33] OpenAI. 2026. GPT-5 mini. https://developers.openai.com/api/docs/models/gpt-5-mini Accessed: March 2026.

[34] OpenAI. 2026. GPT-5.2. https://openai.com/it-IT/index/introducing-gpt-5-2/ Accessed: March 2026.

[35] Jeiyoon Park, Chanjun Park, and Heuiseok Lim. 2025. CharacterGPT: A Persona Reconstruction Framework for Role-Playing Agents. arXiv:2405.19778 [cs.CL] https://arxiv.org/abs/2405.19778

[36] Dr. Urmila R. Pol. 2025. Generative AI, AI Agents, and Agentic AI : An Overview of Current AI Technologies. *International Journal for Research in Applied Science and Engineering Technology* (2025). https://api.semanticscholar.org/CorpusID:283379174

[37] Alex Reinhart, Ben Markey, Michael Laudenbach, Kachatad Pantusen, Ronald Yurko, Gordon Weinberg, and David West Brown. 2025. Do LLMs write like humans? Variation in grammatical and rhetorical styles. *Proceedings of the National Academy of Sciences* 122, 8 (Feb. 2025). doi:10.1073/pnas.2422455122

[38] Alon Rosenbaum, Yigal David, Eran Kaufman, Gilad Ravid, Amit Ronen, and Assaf Krebs. 2025. Scaffolding Creativity: How Divergent and Convergent LLM Personas Shape Human Machine Creative Problem-Solving. arXiv:2510.26490 [cs.HC] https://arxiv.org/abs/2510.26490

[39] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2025. A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications. arXiv:2402.07927 [cs.AI] https://arxiv.org/abs/2402.07927

[40] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face. In *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Eds.), Vol. 36. Curran Associates, Inc., 38154–38180. https://proceedings.neurips.cc/paper_files/paper/2023/file/77c33e6a367922d003ff102ffb92b658-Paper-Conference.pdf

[41] Guy Tevet and Jonathan Berant. 2021. Evaluating the Evaluation of Diversity in Natural Language Generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty (Eds.). Association for Computational Linguistics, Online, 326–346. doi:10.18653/v1/2021.eacl-main.25

[42] Yufei Tian, Tenghao Huang, Miri Liu, Derek Jiang, Alexander Spangher, Muhao Chen, Jonathan May, and Nanyun Peng. 2024. Are Large Language Models Capable of Generating Human-Level Narratives?. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 17659–17681. doi:10.18653/v1/2024.emnlp-main.978

[43] Zhenhailong Wang, Shaoguang Mao, Wenshan Wu, Tao Ge, Furu Wei, and Heng Ji. 2024. Unleashing the Emergent Cognitive Synergy in Large Language Models: A Task-Solving Agent through Multi-Persona Self-Collaboration. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 257–279. doi:10.18653/v1/2024.naacl-long.15

[44] Zhuohan Xie, Trevor Cohn, and Jey Han Lau. 2023. The Next Chapter: A Study of Large Language Models in Storytelling. In *Proceedings of the 16th International Natural Language Generation Conference*, C. Maria Keet, Hung-Yi Lee, and Sina Zarrieß (Eds.). Association for Computational Linguistics, Prague, Czechia, 323–351. doi:10.18653/v1/2023.inlg-main.23

[45] Weijia Xu, Nebojsa Jojic, Sudha Rao, Chris Brockett, and Bill Dolan. 2025. Echoes in AI: Quantifying lack of plot diversity in LLM outputs. *Proceedings of the National Academy of Sciences* 122, 35 (Aug. 2025). doi:10.1073/pnas.2504966122

[46] Ann Yuan, Andy Coenen, Emily Reif, and Daphne Ippolito. 2022. Wordcraft: Story Writing With Large Language Models. In *Proceedings of the 27th International Conference on Intelligent User Interfaces* (Helsinki, Finland) *(IUI '22)*. Association for Computing Machinery, New York, NY, USA, 841–852. doi:10.1145/3490099.3511105

[47] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a Machine Really Finish Your Sentence? arXiv:1905.07830 [cs.CL] https://arxiv.org/abs/1905.07830

[48] Hugh Zhang, Daniel Duckworth, Daphne Ippolito, and Arvind Neelakantan. 2021. Trading Off Diversity and Quality in Natural Language Generation. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, Anya Belz, Shubham Agarwal, Yvette Graham, Ehud Reiter, and Anastasia Shimorina (Eds.). Association for Computational Linguistics, Online, 25–33. https://aclanthology.org/2021.humeval-1.3/

[49] Jiayi Zhang, Simon Yu, Derek Chong, Anthony Sicilia, Michael Tomz, Christopher D Manning, and Weiyan Shi. 2026. Verbalized Sampling: How to Mitigate Mode Collapse and Unlock LLM Diversity. https://openreview.net/forum?id=9jQkmGunGo