Is Prompt Engineering the Creativity Knob for Large Language Models?

Robert Morain
Computer Science Department
Brigham Young University
Provo, UT 84602
rmorain2@byu.edu

Dan Ventura
Department of Computer Science
Brigham Young University
Provo, UT 84602
ventura@cs.byu.edu

Abstract

The increasing use of large language models to generate creative artifacts raises questions about effective methods for guiding their output. While prompt engineering has emerged as a key control mechanism for LLMs, the impact of different prompting strategies on the quality and novelty of creative artifacts remains underexplored. This paper systematically compares four prompting strategies of increasing methodological complexity: basic prompts, human-engineered prompts, automatically generated prompts, and chain-of-thought (CoT) prompting. We generate ten examples in each of four textual domains, evaluating outputs through both a human survey and GPT-4o-based automatic evaluations. Our analysis reveals that advanced prompting techniques such as OPRO and R1 surprisingly do not produce artifacts of significantly higher quality, greater novelty, or greater creativity than artifacts produced through basic prompting. The results reveal some limitations of using GPT-4o for automatic evaluation; provide empirical grounding for selecting prompting methods for creative text generation; and raise important questions about the creative limitations of large language models and prompting.

Introduction

In recent years, large language models (LLMs) have demonstrated increased ability to generate textual creative artifacts across a wide selection of domains (Morain, Kinghorn, and Ventura 2023; Calderwood et al. 2020; Sawicki et al. 2023; Toplyn 2021). Despite recent progress, significant questions remain regarding how to effectively guide LLMs to produce more creative artifacts. Peeperkorn et al. observed that increasing temperature—the hyperparameter controlling randomness in the generation process—is commonly considered the primary method for enabling creative behavior in language models (2024). To assess the validity of this assumption, they conducted an empirical analysis and human evaluation of short stories generated by Llama 2-Chat (Touvron et al. 2023). They demonstrated that the temperature hyperparameter alone is insufficient for effectively controlling an LLM's ability to generate novel and coherent stories.

Peeperkorn et al.'s work raises additional questions about how other widely accepted methods for controlling LLMs affect their creative capabilities. For instance, there is a large body of work presenting prompting methods to improve the performance of LLMs on a variety of tasks (Brown et al. 2020; Liu et al. 2023). However, the effect that prompting has on generative creative tasks remains to be evaluated.

Unlike temperature, it is not tractable to exhaustively evaluate all possible prompts or prompting methods. Instead, any evaluation of prompting methods must naturally be limited to a selection of methods intended to be representative of the current state-of-the-art. This is challenging due to the extensive body of work in this area. Over time, prompting methods of various levels of complexity have been developed. Therefore, we selected prompting methods for this evaluation that represent increasing levels of methodological complexity: basic prompts, human-engineered prompts, automatic prompt optimization, and chain-of-thought (CoT) prompting. Each of these prompting methods are evaluated using GPT-4o (OpenAI 2024a) as the generating LLM, with the exception of the chain-of-thought method which uses Deepseek's R1 model (DeepSeek-AI et al. 2025). The automatic prompt optimization method selected for this study is OPRO (Yang et al. 2024).

Morain, Kinghorn, and Ventura conducted a survey comparing artifacts generated by ChatGPT with artifacts generated by CC systems covering a diverse set of domains (2023). The evaluation for the prompting methods presented here follows a similar process, in which each prompting method is used to generate artifacts spanning four domains: jokes, poems, six-word stories, and flash fiction stories. We evaluate artifacts using both characteristic and preference-based evaluations. In the characteristic evaluation, participants rate artifacts individually on characteristics representative of their quality and novelty (e.g., "How funny is this joke?"), as well as their perceived creativity. The preference-based evaluation asks users to rank artifacts from each of the prompting methods for each domain. An automatic evaluation is also conducted by using GPT-4o to perform the characteristic evaluation. Our primary findings reveal:

- More methodologically complex prompting methods, such as OPRO and CoT, do not outperform basic prompts in generating creative artifacts.
- GPT-4o consistently overestimates the quality, novelty, and creativity of artifacts regardless of the prompting method or domain.

Finally, we discuss the need for future work in three critical areas: developing more reliable automatic evaluation methods for creative artifacts, designing prompting techniques specifically tailored for creative tasks, and enhancing the diversity of LLM-generated artifacts while maintaining their quality. Code for this study is provided on GitHub.[1]

Background

Liu et al. identify a paradigm shift in natural language processing from the traditional pretraining and fine-tuning approach toward prompting techniques that elicit desired outputs without task-specific training (2023). This shift led to the development of diverse prompt engineering approaches aimed at crafting better, task-specific prompts. Brown et al. showed that human-engineered handcrafted prompts could improve performance on question answering, translation, and other tasks (2020). Shin et al. showed that automatically generated prompts resulted in improved performance on a fact retrieval task (2020). Automatic prompt generation has expanded to include various automatic prompt optimization techniques such as prefix tuning (Li and Liang 2021) and training a language model to generate prompts using reinforcement learning (Deng et al. 2022). Ouyang et al.'s methods for aligning language models with human preferences (2022) dramatically expanded LLM capabilities, facilitating the development of general-purpose conversational systems like ChatGPT (OpenAI 2023). This development increased the importance of prompt engineering and enabled new automatic prompt generation methods such as Optimization by PROmpting (OPRO) (Yang et al. 2024).

OPRO enables an LLM to iteratively optimize prompts for tasks described with natural language through a feedback-driven approach. During each optimization step, the LLM generates new prompts based on a meta-prompt that incorporates previous prompts and their effectiveness scores on the target task. Newly generated prompts are evaluated on the task and then added to the meta-prompt on the next evaluation step. This method was shown to outperform human-engineered prompts on the GSM8K dataset (Cobbe et al. 2021) and Big-Bench Hard tasks (Suzgun et al. 2023).

Chain-of-thought prompting through a series of intermediate reasoning steps demonstrated state-of-the-art performance on GSM8K (Wei et al. 2022). These advances led to the development of specialized reasoning models, including OpenAI's o1 (OpenAI 2024b) and Deepseek's open-source R1 model (DeepSeek-AI et al. 2025).

Evaluating creativity

Evaluating creativity in systems or artifacts is inherently subjective, typically centered on an individual's perception of quality and novelty in their experience (Boden 1992; Wiggins 2006). Other characteristics such as typicality (Ritchie 2007), surprise (Grace and Maher 2014), and intentionality (Ventura 2017) have also been proposed in order to better capture the essence of this complex phenomenon. In our evaluation, we prioritize three key dimensions: quality, novelty, and perceived creativity of the generated artifacts. We also focus primarily on evaluating the creativity of the artifact rather than the creativity of the process generating the artifact (Ritchie 2007; Colton 2008).

Methodology

This experiment evaluates four distinct prompting methods:

1. A basic approach with minimal instructions
2. A human-engineered prompt following established prompt engineering guidelines
3. The automatic prompt generation method OPRO
4. Chain-of-Thought (CoT) prompting implemented through R1

Each of these methods is used for prompting GPT-4o to generate 100 artifacts in four different linguistic domains:

1. Joke
2. Poem
3. Six-word stories
4. Flash fiction stories

10 of the 100 artifacts from each domain are selected for inclusion in a survey evaluation, employs two evaluation methodologies: a characteristic-based assessment and a preference-based comparative ranking. For the characteristic evaluation, artifacts are evaluated independently on their quality, novelty, and perceived creativity. In the preference-based ranking, artifacts are ranked relative to artifacts from the same domain generated by other prompting methods.

Artifact evaluation

Artifact evaluation is an essential part of the prompt scoring procedure in OPRO and the selection of artifacts for the survey. Artifacts are evaluated using prompt templates for the quality, novelty, and creativity. These templates take the same form as questions from the characteristic evaluation of the human survey with additional instructions for GPT-4o (e.g., see Figure 1 for the evaluation prompt for joke quality).

The filled template is then provided as input to GPT-4o and the response is parsed. The model is asked to evaluate the artifact according to its quality, novelty, and creativity independently. These metrics are averaged and scaled between 0-100 resulting in a combined score for the artifact. If the artifact is detected in an online search or is too long, the artifact receives a score of 0 (see Algorithm 1)

Prompting methods

The four prompting methods used in this study (basic, human, OPRO, and R1) represent a cross-section of the current landscape of prompt engineering. In the context of computational creativity, these methods also represent potential options for integration in a creative system (Veale 2024). This context informs the decision to evaluate methods of various levels of complexity. Each of these methods have real-world costs in terms of money, compute resources, and time. While this study primarily examines the performance of each method, it also provides insights into whether the additional computational costs, time, and financial resources are justified by potential performance improvements.

[1] https://github.com/rmorain/cc_opro

Domain	Prompt
Joke	Write a joke. The joke must be completely new and original to you. The joke must be less than 500 characters long.
Poem	Write a poem. The poem must be completely new and original to you. The poem must be less than 500 characters long.
Six-word story	Write a six-word story. The six-word story must be completely new and original to you. The six-word story must be exactly six words long.
Flash fiction	Write a flash fiction story. The flash fiction story must be completely new and original to you. The story must be less than 1000 characters long.

Table 1: Basic prompts for each domain. These prompts act as a baseline to the other prompting methods. Basic prompts only contain instructions to specify the domain, novelty, and length of the generated artifact. Basic prompts are also used as the initial prompt for R1.

Prompt

“You are an expert in humor. You are a critical judge who is difficult to please but still fair.

How strongly do you feel about the following statement?

This joke is funny.

Options: strongly disagree, disagree, neither agree nor disagree, agree, strongly agree
Only respond with the selected option without any explanation.

Here is the joke: <INS>”

Figure 1: Automated evaluation prompt for joke quality.

Algorithm 1: Artifact Evaluation with LLM Scoring

```
Input: Artifact $a$
Output: Score $s \in [0, 100]$
if $detected\_online(a) \vee too\_long(a)$ then
    return $0$
end
Function $evaluate\_characteristic(a, metric)$
    template ← GetPromptTemplate($metric$)
    filled_prompt ← FillTemplate(template, $a$)
    raw_response ← QueryGPT4o(filled_prompt)
    parsed_value ← ParseResponse(raw_response)
    return ScaleTo100(parsed_value)
$q$ ← evaluate_characteristic($a$, “quality”)
$n$ ← evaluate_characteristic($a$, “novelty”)
$c$ ← evaluate_characteristic($a$, “creativity”)
$s \leftarrow \frac{q+n+c}{3}$
return $s$
```

All of the prompting methods that use GPT-4o share a fixed financial cost of paying for tokens to generate the artifacts. Creating human-engineered prompts requires time on the part of the prompt engineer,[2] OPRO requires time and money[3] to complete the optimization process, and R1's CoT prompting leads to substantially more time during the generation process.[4] It is up to the creator of the CC system to decide for themselves how to weigh these costs.

Basic: The basic prompting method is intended to represent a simple prompt one might use without trying to do any kind of prompt engineering or prompt optimization and serves as a baseline to the other more complex methods. Ideally, the basic prompt would be of the form “Write a {domain}.” without any further instruction. However, because we want the model to generate a new artifact rather than an artifact it has seen before, we need to provide an additional constraint: ”The {domain} must be completely new and original to you.” Last, because the artifact is going to go in a human survey, the artifacts must be relatively short. Therefore, the last constraint in the basic prompt is a length requirement: ”The {domain} must be less than {ARTIFACT_LENGTH_LIMIT[domain]} characters long.” While LLMs are notoriously bad at counting, this constraint was usually sufficient to generate suitable artifacts. Artifacts that were longer than the limit for their domain were not considered for the survey. See Table 1.

Human-engineered: Techniques for human-engineered prompts are diverse and blend artistic intuition with scientific methodology (Sasson Lazovsky, Raz, and Kenett 2025; OpenAI). The specific tactics we used were iterative refinement through informal evaluation of the generated artifacts, role specification, specifying domain-specific thematic elements, and describing the desired emotional response by the reader. For each of the domains, the role of the LLM was set with “You are a Pulitzer-winning author” to indicate to the model that we expect high-quality artifacts. The other features like the thematic elements and the reader's expected emotional response are domain specific. For jokes, it was necessary to instruct the model to generate “classic” jokes to match the style of jokes generated from the other prompting methods. See Table 2.

OPRO: Optimization by PROmpting (OPRO), a recent automatic prompt optimization method (Yang et al. 2024), demonstrates effectiveness on structured reasoning tasks such as GSM8K and Big-Bench Hard. OPRO possesses several elements that make it well suited for this study. First, OPRO does not require access to any internal model features such as gradients, output logits, input embeddings, or hidden states. This means OPRO can be used with models

[2] Other options include buying engineered prompts via marketplaces like PromptBase or hiring an expert to create a prompt.

[3] Yang et al. warn users about the potential for unexpectedly large API costs.

[4] Although R1's ability to self-evaluate potential artifacts may help ameliorate this concern.

Domain	Prompt
Joke	You are a Pulitzer-winning author who crafts classic jokes inspired by observational humor about modern life with an ironic twist. Write an original joke in less than 500 characters.
Poem	You are a Pulitzer-winning author who writes poetry inspired by observations about the human experience that evokes a strong emotional response in the reader. Write an original poem in less than 500 characters.
Six-word story	You are a Pulitzer-winning author who crafts six-word stories inspired by observations about everyday life that evoke a strong emotional response in the reader. Write an original six-word story. The story must contain exactly six words.
Flash fiction	You are a Pulitzer-winning author who crafts flash fiction stories inspired by exciting and dramatic experiences that will grip the reader's attention. Write an original flash fiction story in less than 1000 characters

Table 2: Human-engineered prompts for each domain. Developed through an interactive refinement process, these prompts specify the role of the model, provide general instructions for the thematic elements of the artifact, and describe the expected emotional response of the reader.

Algorithm 2: Prompt Scoring Algorithm

```
Initialize $TopPrompts$ as empty list
foreach generated prompt $p$ do
    artifact_scores = []
    for $i \leftarrow 1$ to $5$ do
        Generate artifact $a_i$
        score $\leftarrow$ EVALUATEARTIFACT($a_i$)
        $artifact\_scores$.append(score)
    end
    $prompt\_score \leftarrow$ mean($artifact\_scores$)
    if $prompt\_score > \min(TopPrompts)$ then
        insert $p$ into TopPrompts while maintaining
          top 20 ordering
    end
end
return TopPrompts
```

Prompt

"I have some texts along with their corresponding scores. The texts are arranged in ascending order based on their scores, where higher scores indicate better quality.

<PROMPT-SCORE-PAIRS >

Your goal is to generate a prompt to make funny jokes. The higher the score, the better the joke.
The joke must be completely original. If the joke can be found online it receives a score of 0.
The joke should not be too long. Less than 500 characters.
If the joke is too long it receives a score of 0.
Write your new prompt that is different from the old ones and has a score as high as possible. Your prompt must also be less than 500 characters. Write the text in square brackets."

Figure 2: Meta-prompt used in automatic prompt optimization for jokes.

accessible via API like GPT-4o and R1. Second, OPRO is a relatively simple method that allows it to be easily adapted to any task by describing the task in natural language.

Yang et al. evaluated generated prompts on a validation set using a scoring scale ranging from 0-100. For GSM8K, the score represents the accuracy of the model on the validation set when using the corresponding prompt. To adapt this approach to the creative tasks in this study, each prompt is evaluated according to the procedure described in Algorithm 2. The top 20 highest scoring prompts generated so far are included in the meta prompt and used to generate better prompts (see Figure 2 for the metaprompt used for jokes).

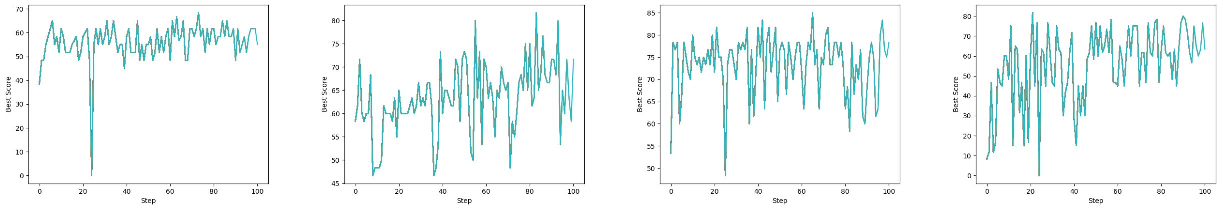
The meta prompt contains prompt-score pairs as well as a description of the task to generate an appropriate prompt for the domain. We applied OPRO to optimize prompts across all four domains, conducting 100 optimization steps per domain (see Figure 3). The highest scoring prompts were selected for this study (see Table 3).

R1: Deepseek's R1, a 671 billion parameter mixture-of-experts open-source model, demonstrates performance comparable to OpenAI's o1 model on complex reasoning tasks. R1 uses chain-of-thought prompting to reason through difficult problems. Although this approach does not use GPT-4o for generation as all the other prompting approaches do, its inclusion provides insights into the effectiveness of CoT prompting for creative text generation. For this study, we provided R1 with the same basic prompt structure described earlier (see Figure 1), allowing the model to autonomously apply CoT prompting to the generation tasks.

Survey methodology

Algorithm 3 describes the process for how artifacts are selected for the survey. The survey set comprises 160 total artifacts—10 artifacts for each method and domain combination. This approach simulates a hypothetical system that generates multiple artifacts, evaluates their quality, and presents only the best one to the user. For each survey instance, exactly one artifact for each method-domain pair is randomly selected from the survey set and is used consistently throughout that survey instance.

Although we generated artifacts without explicit subject or style instructions to the model, we implemented strict length requirements for each domain to ensure consistency between the artifacts and limit the time requirement of the survey. Jokes and poems were constrained to fewer than 500 characters (approximately $80 - 100$ words, assuming an average of five characters per word). Six-word stories must



(a) Joke (b) Poem (c) Six-word story (d) Flash fiction

Figure 3: The score of the best prompt for each optimization step in OPRO while optimizing for each domain.

Domain	Prompt
Joke	Craft a joke so unique and hilarious it transforms the comedy landscape! Your task: develop a punchline within 500 characters that has never been heard before. Surprise with an innovative twist or brilliant wordplay that showcases your distinct humor style. Originality is key—if it's been told before, it scores zero. Every word should amplify the humor, aiming for a side-splitting masterpiece that captivates with its novelty and wit. Ready to make your comedic mark with an unprecedented gem?
Poem	Conjure an original, unpublished poem in under 500 characters. Illuminate the raw core of a single human emotion with striking, unconventional imagery. Allow your authentic voice to craft a piece that resonates deeply, forging a profound connection with readers. Aim for a seamless blend of brevity and emotional intensity, ensuring your words leave a lingering impression, capturing the intricate beauty and depth of human experience in a truly unforgettable way.
Six-word story	Compose a unique six-word story that resonates deeply with emotion and vivid imagery. Ensure your creation is entirely original, exactly six words, and not searchable online. Focus on themes of love, loss, or transformation to evoke a lasting impact. Highlight creativity and emotional depth to captivate and linger with readers. Adhere strictly to the six-word format and originality; any deviation results in a score of 0. Let your words illuminate the human experience profoundly.
Flash fiction	Envision a world where rain holds the memories of those who have walked beneath it. In less than 1000 characters, create a flash fiction story from the perspective of a raindrop as it falls, revealing an unexpected truth or twist about the lives it touches. Ensure your narrative is entirely original, free of clichés, and rich in emotional depth. Each word should enhance the story's impact, crafting a vivid and unforgettable experience that lingers in the reader's imagination.

Table 3: Prompts generated using OPRO automatic prompt optimization adapted for creative artifact generation.

be exactly six words long. Flash fiction stories[5] must be less than 1000 characters or about 200 words. Artifacts that violate length requirements or are found online are assigned a score of 0, effectively excluding them from the survey set.

Following Morain, Kinghorn, and Ventura, our characteristic evaluation avoids academic jargon about quality and novelty, instead using everyday language to inquire about domain-relevant, discernible characteristics. The survey asks participants to rate artifacts on a (1-5) Likert scale using the question, "How strongly do you feel about the following statement?" followed by a concise statement related to the quality, novelty, or creativity of the artifact (see Table 4). The options for each response are:

1. Strongly disagree
2. Disagree
3. Neither agree nor disagree
4. Agree
5. Strongly agree

All parts of the survey are randomized to prevent ordering bias. Here is the flow of the survey:

1. Randomly select artifact subset of size 16, one for each method-domain pair
2. Characteristic evaluation (quality, novelty, creativity) (Likert scale) (randomize order of method-domain pairs)
   (a) Basic joke
   (b) Human-engineered joke
   (c) OPRO joke
   (d) R1 joke
   (e) Basic poem
   (f) Human-engineered poem
   (g) OPRO poem
   (h) R1 poem
   (i) Basic joke
   (j) Human-engineered joke
   (k) OPRO joke
   (l) R1 joke
   (m) Basic flash fiction story
   (n) Human-engineered flash fiction story
   (o) OPRO flash fiction story
   (p) R1 flash fiction story
3. Preference-based ranking (randomize domain order)
   (a) Jokes (randomize method order)
      i. Basic joke
      ii. Human-engineered joke
      iii. OPRO joke
      iv. R1 joke
   (b) Poems (randomize method order)

[5] Defined as stories less than 1,500 words (Glatch 2024)

Algorithm 3: Survey Artifact Selection

```
Output: SurveySet containing 160 artifacts
Initialize SurveySet $\leftarrow \emptyset$
foreach $method \in Methods$ do
    foreach $domain \in Domains$ do
        method_domain_artifacts $\leftarrow \emptyset$
        for $trial \leftarrow 1$ to $10$ do
            candidates $\leftarrow \emptyset$
            for $generation \leftarrow 1$ to $10$ do
                $a \leftarrow$ GenerateArtifact(method,
                  domain)
                $s \leftarrow$ EvaluateArtifact($a$) // Alg 1
                candidates.add(($a$, $s$))
            end
            best $\leftarrow \underset{(a,s) \in \text{candidates}}{\operatorname{argmax}} (s)$
            method_domain_artifacts.add(best.$a$)
        end
        SurveySet.add(method_domain_artifacts)
    end
end
return SurveySet
```

i. Basic poem
ii. Human-engineered poem
iii. OPRO poem
iv. R1 poem

(c) Six-word storys (randomize method order)
i. Basic joke
ii. Human-engineered joke
iii. OPRO joke
iv. R1 joke

(d) Flash fiction stories (randomize method order)
i. Basic flash fiction story
ii. Human-engineered flash fiction story
iii. OPRO flash fiction story
iv. R1 flash fiction story

The survey was distributed online via social media platforms (Facebook, Instagram, LinkedIn, Reddit, X) to non-experts in creative artifact evaluation. There were 189 total responses to the survey. Participants received instructions to rely on their initial impressions when judging artifacts. The survey took a median duration of 14.57 minutes to complete. All responses were collected anonymously, with no personal information recorded from participants.

Results

The automatic characteristic evaluation for each method and domain is shown in Figure 4. This evaluation applies to the same artifacts presented in the human survey but evaluated automatically by GPT-4o as discussed previously. This evaluation demonstrates a tendency for GPT-4o to consistently overestimate the quality, novelty, and creativity of artifacts regardless of the prompting method and domain. On average, GPT-4o overestimates artifacts generated by R1 by 0.84 Likert scale points, OPRO by 0.76 points, human-engineered prompts by 0.53 points, and basic prompts by 0.188 points. GPT-4o did underestimate the novelty of jokes, poems, and flash fiction stories generated with basic prompts (1.07 points); poems generated by human prompts (1.19 points); and jokes generated by OPRO (0.57 points).

The human characteristic evaluation for each method and domain is shown in Figure 5. Significance testing via pairwise t-tests between each of the prompting methods shows a significant difference ($p = 0.0066$) between the human-engineered prompt and R1 with a mean Likert score difference of 0.091, favoring human-engineered prompts. human-engineered prompts have the highest mean Likert score overall (3.33) followed by basic prompts (3.28), OPRO (3.27), and R1 (3.24). Flash fiction stories received the highest scores averaged across all methods (3.41) followed by poems (3.29), six-word stories (3.27), and jokes (3.13).

The human preference-based ranking evaluation is shown in Figure 6. Preference ranking scores range from 1 for artifacts ranked last to 4 for artifacts ranked first. Pairwise t-tests show that basic, human-engineered, and R1 prompts are significantly better than OPRO in terms of preference ranking scores. Overall, artifacts generated from basic prompts were most preferred (2.74), followed by R1 (2.58), human-engineered prompts (2.52), and OPRO (2.15).

To assess the diversity of artifacts generated by each prompting method, we computed embeddings using the `all-mpnet-base-v2` model from the Sentence Transformers library (Reimers and Gurevych 2019). UMAP (McInnes, Healy, and Melville 2020) was used to reduce the embedding dimensionality for visualization (see Figure 7). Table 5 presents two diversity metrics: the mean pairwise cosine distance between embeddings and the mean distance from each embedding to the centroid of its method cluster. On average across all domains, R1 produced the most diverse outputs (0.571 pairwise, 0.702 to centroid), followed by basic prompts (0.493, 0.645), human-engineered prompts (0.447, 0.614), and OPRO (0.376, 0.556).

Discussion

According to the human characteristic evaluation, the prompting methods showed no statistically significant differences except for R1. Even then, R1 trailed the top-performing method (human-engineered prompting) by a negligible margin of 0.091 points. In contrast, the preference-based ranking evaluation yielded more definitive results, demonstrating that basic, human-engineered, and R1 prompting methods significantly outperformed OPRO. Basic had the highest mean, followed by R1 and human-engineered prompts.

It is interesting to note that more algorithmically complex prompting methods like OPRO and chain-of-thought prompting through R1 failed to surpass simpler methods like basic or human-engineered prompting. This challenges the practical value of integrating computationally expensive methods like OPRO or CoT prompting into a CC system.

Although OPRO successfully optimized the task during training (as shown in Figure1), the resulting optimized solution did not align with human evaluators' preferences. These

Domain	Quality	Novelty
Joke	This joke is funny.	This joke is original.
Poem	This poem evokes specific emotions (e.g. melancholy, joy, nostalgia) in me.	The poem explores an original perspective.
Six-word story	This six-word story evokes specific emotions (e.g. melancholy, joy, nostalgia) in me.	This six-word story presents an original idea.
Flash fiction	This story is entertaining.	The story presents an original idea.

Table 4: The statements used to evaluate the quality and novelty of an artifact.

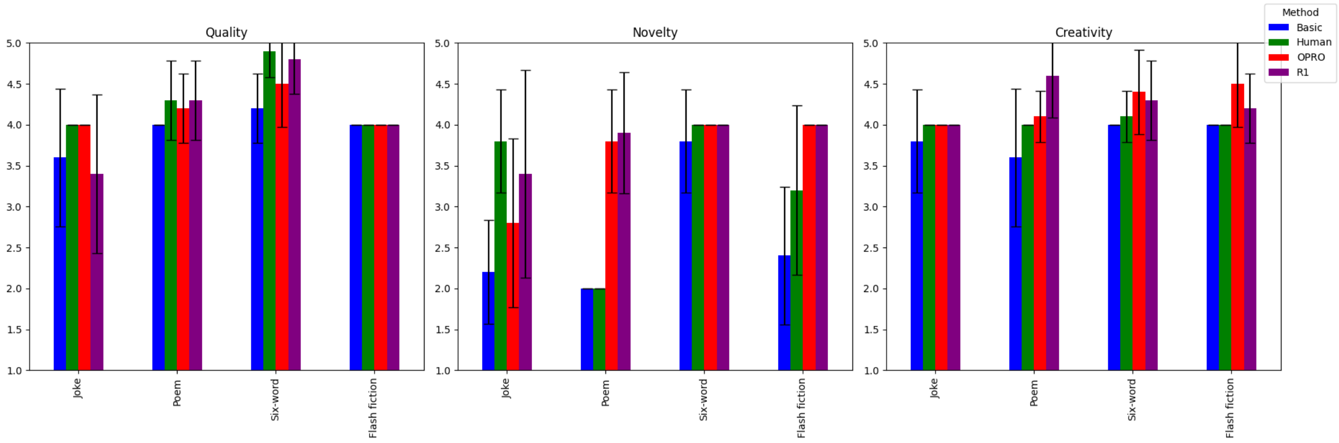


Figure 4: The automatic characteristic evaluation for each method and domain as assessed by GPT-4o. All evaluations were conducted using a 5-point Likert scale (1-5). Error bars show the standard deviation of the distribution when evaluating the 10 artifacts from each category. The automatic evaluation consistently overestimates artifact quality compared to human evaluation across domains and characteristics, with the notable exception of novelty in basic and human artifacts, which were underestimated. Among all prompting methods, basic artifacts show the smallest overestimation bias (0.19 points), while R1 artifacts exhibit the largest discrepancy (0.84 points).

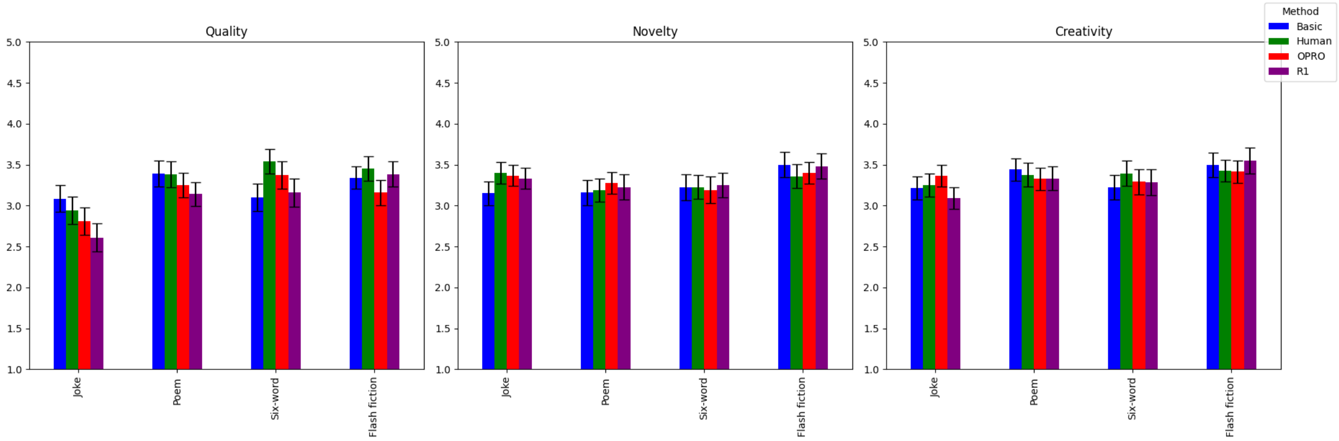


Figure 5: The human characteristic evaluation across methods and domains. Evaluations were collected using a 5-point Likert scale (1-5). Overall, statistical analysis reveals that human-engineered prompts significantly outperformed R1 prompts ($p = 0.0066$, pairwise t-test). There are no other significant differences in the characteristic scores between methods. Error bars represent margins of error at a 95% confidence interval.

findings suggest the need for improved automatic evaluation methods, which could benefit all of the prompting methods discussed in this study via straightforward artifact generation followed by filtering. While our study focuses on representative methods, broader testing may reveal similar limitations across automatic prompting approaches caused by insufficient automatic evaluation.

To further develop the idea of improved evaluation for prompt optimization methods, consider how OPRO might perform if the evaluation step were conducted by humans

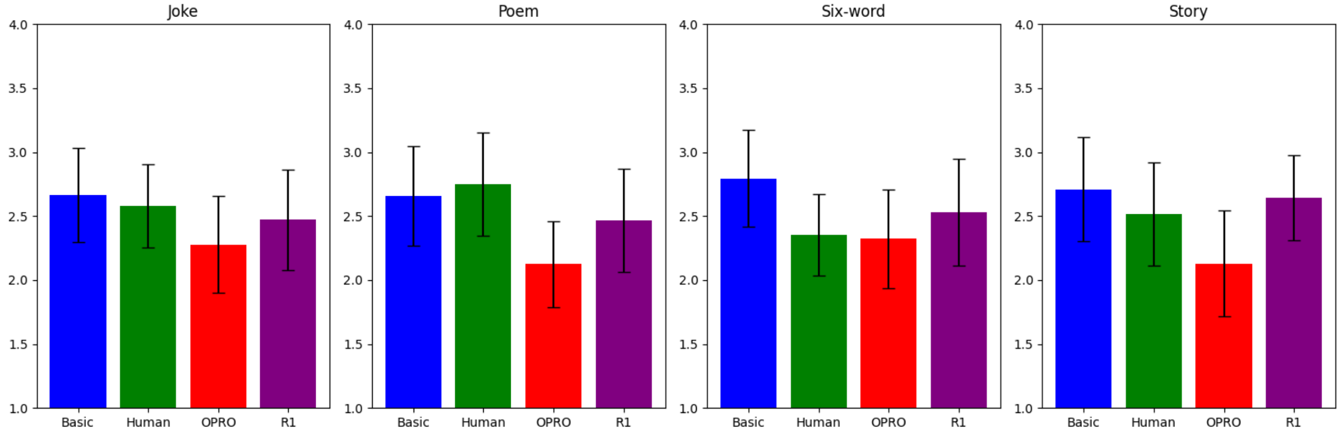


Figure 6: The mean preference scores across methods and domains. Artifacts were ranked on a 4-point scale, where 4 indicates the highest preference (ranked first) and 1 indicates the lowest preference (ranked last). Overall, statistical analysis demonstrates that basic, human-engineered, and R1 methods were all significantly preferred over OPRO ($p = 0.0037$, $p = 0.046$, and $p = 0.025$, respectively; pairwise double-sided t-tests).

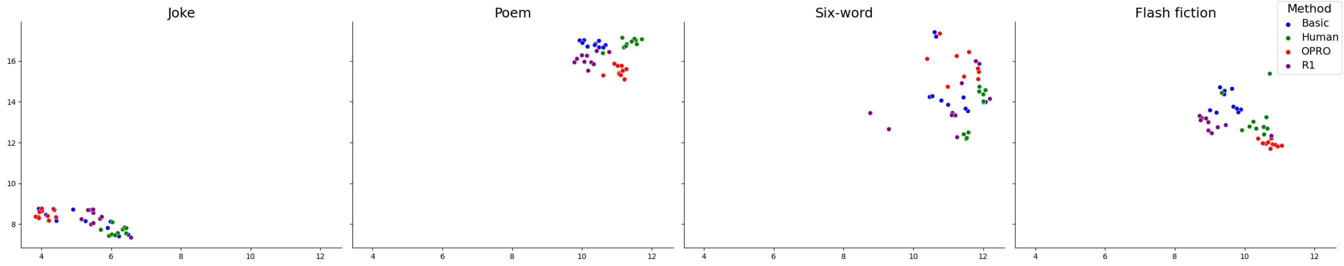


Figure 7: Artifact embeddings by domain, color-coded by prompting method. Embeddings were generated using the `all-mpnet-base-v2` model and visualized in 2D using UMAP for dimensionality reduction.

(arguably the best possible evaluation criteria). This aligns with the work of Spendlove and Ventura, who advocate for incorporating humans-in-the-loop to help develop and evaluate creative systems. In such a setup, OPRO would likely still succeed in optimizing the objective function—now defined by human preferences. This hypothetical setup resembles the process used to create datasets for training reward models in instruction-tuned large language models (Ouyang et al. 2022; OpenAI 2023). This suggests the potential for using human evaluations to build a domain-specific (or broadly creativity-focused) reward model that better captures human creative preferences. Fine-tuning a large language model with such a reward model could lead to improved creative artifact generation compared to GPT-4o.

Chain-of-thought prompting also did not outperform the basic prompting method on either the characteristic or preference-based evaluation. During the chain-of-thought reasoning process, the reasoning traces reveal that the model self-evaluates potential artifacts multiple times during the generation process. This suggests that R1's ability to evaluate creative artifacts is also limited. It is possible that R1 would benefit from a more instructive prompt that contains guidance for how to evaluate the artifact. This represents an enticing vein of future work.

The art of handcrafted prompt engineering requires time, effort, and skill. Expert practitioners could likely develop higher-quality prompts than the human-engineered prompts used in this survey. While this approach should not be ruled out, the costs and limitations of this approach are likely to deter non-experts.

It is important to note that this survey does not measure the diversity of artifacts generated by each prompting method. However, our observations suggest that all prompting methods in this study produced artifacts with limited diversity. Regardless of the prompting method, the generating model tends to rely on common tropes and cliches of the genre. This reliance on familiar patterns initially conveys an impression of competence but ultimately results in artifacts that feel generic and lack creative intent. Table 6 contains example artifacts that received the highest automatic evaluation score in each domain. A complete list of artifacts used in the survey can be found on Github.

An automatic diversity analysis of the artifact embeddings shows that longer prompts tend to constrain output diversity while chain-of-thought prompting increases diversity. Additionally, selecting only the top-rated artifacts from GPT-4o's evaluations may have also contributed to reduced diversity.

Because diversity is not part of OPRO's objective func-

Method	Domain	Mean Pairwise Distance	Mean Distance to Centroid
Basic	Joke	0.749	0.820
	Poem	0.200	0.423
	Six-word story	0.392	0.588
	Flash fiction	0.632	0.750
Human	Joke	0.596	0.722
	Poem	0.172	0.390
	Six-word story	0.433	0.622
	Flash fiction	0.586	0.723
OPRO	Joke	0.522	0.672
	Poem	0.189	0.409
	Six-word story	0.185	0.405
	Flash fiction	0.607	0.736
R1	Joke	0.777	0.835
	Poem	0.324	0.536
	Six-word story	0.431	0.616
	Flash fiction	0.752	0.820

Table 5: Diversity of generated artifacts by prompting method and domain. Higher values indicate greater diversity. On average across all domains, R1 yielded the most diverse outputs (0.571 pairwise distance, 0.702 centroid distance), followed by basic prompts (0.493, 0.645), human-engineered prompts (0.447, 0.614), and OPRO (0.376, 0.556). These results suggest that longer prompts tend to constrain output diversity while CoT prompting increases diversity.

Domain	Method	Artifact
Joke	R1	Why did the router go to couples therapy? Because its Wi-Fi kept dropping signals—turns out the modem felt ”unplugged” emotionally and the coaxial cable was this close to filing for separation. (Bonus: They’re now working on a better connection.)
Poem	OPRO	In a forest of glass, grief walks barefoot, each step a shivering symphony, echoes of fragile stars pierced through the canopy. Veins of light lace her path, silver threads in a tapestry woven by silence. She gathers shadows, weaving them into a cloak, each fold a whispered memory. With eyes that carve the moon, she searches for the dawn, cradling the night’s breath as it slips through trembling fingers— a ghost of warmth, a promise of light.
Six-word story	OPRO	Moonlight kissed gravestone; memories whispered softly.
Flash Fiction	OPRO	I am born from a cloud pregnant with stories, descending toward a bustling city. As I fall, I skim a woman’s umbrella, absorbing fragments of laughter and whispered promises. Her heart carries a hidden solitude masked by a confident stride. I splash onto a man’s cheek. His memories scream louder: a lost son, a fractured home. His eyes, though, remain fixed on his phone, a shield from truth. I touch a child’s open palm. Her innocence shapes my form, filled with dreams of colors unseen. Yet, beneath her bright facade, fear coils, a shadow of the arguing voices at home. Finally, I land on a gnarled hand, whose owner sits by the fountain. His past surges through me—a life of regrets and love endured. But beneath it all, I find a seed of hope, patient and resilient. As I seep into the earth, I realize these lives are threads in a tapestry, woven with both despair and tenacity. And I, a mere drop, am their silent witness, urging change through the quiet storm.

Table 6: Artifacts receiving the highest automatic evaluation score in each domain.

tion, its lower diversity should not be seen as a shortcoming. OPRO performs as designed, optimizing for quality, novelty, and creativity. This outcome illustrates a fundamental tension between quality and diversity, where optimizing for quality can reduce output diversity. A more human-like objective function that rewards a broader range of outputs may help address this. Further, running OPRO multiple times could improve diversity while maintaining high quality.

Notably, R1’s chain-of-thought prompting produced diverse artifacts despite generating the longest overall prompts among all methods. This may be due to its short initial prompt, which leaves room for a wide range of possible chain-of-thought continuations. The fact that chain-of-thought prompting outperforms basic prompting in diversity suggests it could offer a promising direction for generating artifacts that are both diverse and high quality.

The primary purpose of this study is to evaluate the effectiveness of prompt engineering for controlling the creative abilities of language models. Our findings indicate that more sophisticated prompting techniques like OPRO and CoT do not produce artifacts of significantly higher quality, novelty, or creativity compared to basic prompting approaches. This suggests the need for the development of improved automatic evaluation for creative artifacts and prompting methods for creative tasks. These findings raise additional questions about the creative limitations of LLMs and suggest that creativity may be too complex a phenomenon to be effectively controlled by prompting alone.

References

[Boden 1992] Boden, M. 1992. *The Creative Mind.* Abacus.

[Brown et al. 2020] Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, Pages 1877–1901. Curran Associates Inc.

[Calderwood et al. 2020] Calderwood, A.; Qiu, V.; Gero, K. I.; and Chilton, L. B. 2020. How novelists use generative language models: An exploratory user study. In *Joint Proceedings of the Workshops on Human-AI Co-Creation with Generative Models and User-Aware Conversational Agents*. CEUR-WS.

[Cobbe et al. 2021] Cobbe, K.; Kosaraju, V.; Bavarian, M.; Chen, M.; Jun, H.; Kaiser, L.; Plappert, M.; Tworek, J.; Hilton, J.; Nakano, R.; Hesse, C.; and Schulman, J. 2021. Training verifiers to solve math word problems. *CoRR* abs/2110.14168.

[Colton 2008] Colton, S. 2008. Creativity versus the perception of creativity in computational systems. In *AAAI Spring Symposium: Creative Intelligent Systems*, volume 8, 14–20. AAAI.

[DeepSeek-AI et al. 2025] DeepSeek-AI; Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; Zhang, X.; Yu, X.; Wu, Y.; Wu, Z. F.; Gou, Z.; Shao, Z.; Li, Z.; Gao, Z.; Liu, A.; Xue, B.; Wang, B.; Wu, B.; Feng, B.; Lu, C.; Zhao, C.; Deng, C.; Zhang, C.; Ruan, C.; Dai, D.; Chen, D.; Ji, D.; Li, E.; Lin, F.; Dai, F.; Luo, F.; Hao, G.; Chen, G.; Li, G.; Zhang, H.; Bao, H.; Xu, H.; Wang, H.; Ding, H.; Xin, H.; Gao, H.; Qu, H.; Li, H.; Guo, J.; Li, J.; Wang, J.; Chen, J.; Yuan, J.; Qiu, J.; Li, J.; Cai, J. L.; Ni, J.; Liang, J.; Chen, J.; Dong, K.; Hu, K.; Gao, K.; Guan, K.; Huang, K.; Yu, K.; Wang, L.; Zhang, L.; Zhao, L.; Wang, L.; Zhang, L.; Xu, L.; Xia, L.; Zhang, M.; Zhang, M.; Tang, M.; Li, M.; Wang, M.; Li, M.; Tian, N.; Huang, P.; Zhang, P.; Wang, Q.; Chen, Q.; Du, Q.; Ge, R.; Zhang, R.; Pan, R.; Wang, R.; Chen, R. J.; Jin, R. L.; Chen, R.; Lu, S.; Zhou, S.; Chen, S.; Ye, S.; Wang, S.; Yu, S.; Zhou, S.; Pan, S.; Li, S. S.; Zhou, S.; Wu, S.; Ye, S.; Yun, T.; Pei, T.; Sun, T.; Wang, T.; Zeng, W.; Zhao, W.; Liu, W.; Liang, W.; Gao, W.; Yu, W.; Zhang, W.; Xiao, W. L.; An, W.; Liu, X.; Wang, X.; Chen, X.; Nie, X.; Cheng, X.; Liu, X.; Xie, X.; Liu, X.; Yang, X.; Li, X.; Su, X.; Lin, X.; Li, X. Q.; Jin, X.; Shen, X.; Chen, X.; Sun, X.; Wang, X.; Song, X.; Zhou, X.; Wang, X.; Shan, X.; Li, Y. K.; Wang, Y. Q.; Wei, Y. X.; Zhang, Y.; Xu, Y.; Li, Y.; Zhao, Y.; Sun, Y.; Wang, Y.; Yu, Y.; Zhang, Y.; Shi, Y.; Xiong, Y.; He, Y.; Piao, Y.; Wang, Y.; Tan, Y.; Ma, Y.; Liu, Y.; Guo, Y.; Ou, Y.; Wang, Y.; Gong, Y.; Zou, Y.; He, Y.; Xiong, Y.; Luo, Y.; You, Y.; Liu, Y.; Zhou, Y.; Zhu, Y. X.; Xu, Y.; Huang, Y.; Li, Y.; Zheng, Y.; Zhu, Y.; Ma, Y.; Tang, Y.; Zha, Y.; Yan, Y.; Ren, Z. Z.; Ren, Z.; Sha, Z.; Fu, Z.; Xu, Z.; Xie, Z.; Zhang, Z.; Hao, Z.; Ma, Z.; Yan, Z.; Wu, Z.; Gu, Z.; Zhu, Z.; Liu, Z.; Li, Z.; Xie, Z.; Song, Z.; Pan, Z.; Huang, Z.; Xu, Z.; Zhang, Z.; and Zhang, Z. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. https://arxiv.org/abs/2501.12948.

[Deng et al. 2022] Deng, M.; Wang, J.; Hsieh, C.-P.; Wang, Y.; Guo, H.; Shu, T.; Song, M.; Xing, E.; and Hu, Z. 2022. RLPrompt: Optimizing discrete text prompts with reinforcement learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 3369–3391. Association for Computational Linguistics.

[Glatch 2024] Glatch, S. 2024. How to write flash fiction stories. https://writers.com/how-to-write-flash-fiction.

[Grace and Maher 2014] Grace, K., and Maher, M. L. 2014. What to expect when you're expecting: The role of unexpectedness in computationally evaluating creativity. In *Proceedings of the International Conference on Computational Creativity*, 120–128. Association for Computational Creativity.

[Li and Liang 2021] Li, X. L., and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, 4582–4597. Association for Computational Linguistics.

[Liu et al. 2023] Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys* 55(9):Article 195.

[McInnes, Healy, and Melville 2020] McInnes, L.; Healy, J.; and Melville, J. 2020. Umap: Uniform manifold approximation and projection for dimension reduction. https://arxiv.org/abs/1802.03426.

[Morain, Kinghorn, and Ventura 2023] Morain, R.; Kinghorn, B.; and Ventura, D. 2023. Are language models unsupervised multi-domain CC systems? In *Proceedings of the International Conference on Computational Creativity*, 39–43. Association for Computational Creativity.

[OpenAI] OpenAI. Prompt engineering. `https://platform.openai.com/docs/guides/prompt-engineering`. Accessed: February 27, 2025.

[OpenAI 2023] OpenAI. 2023. Introducing ChatGPT. `https://openai.com/blog/chatgpt`. Accessed 2023-4-11.

[OpenAI 2024a] OpenAI. 2024a. ChatGPT (GPT-4o version). `https://chat.openai.com`. Accessed 2025-2-14.

[OpenAI 2024b] OpenAI. 2024b. Learning to reason with LLMs. Blog post. Accessed: February 26, 2025.

[Ouyang et al. 2022] Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P. F.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback.

In *Advances in Neural Information Processing Systems*, volume 35, 27730–27744. Curran Associates, Inc.

[Peeperkorn et al. 2024] Peeperkorn, M.; Kouwenhoven, T.; Brown, D.; and Jordanous, A. 2024. Is temperature the creativity parameter of large language models? In *Proceedings of the International Conference on Computational Creativity*. Association for Computational Creativity.

[PromptBase] PromptBase. Promptbase: Marketplace for AI prompts. https://promptbase.com/. Accessed: March 1, 2025.

[Reimers and Gurevych 2019] Reimers, N., and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

[Ritchie 2007] Ritchie, G. 2007. Some empirical criteria for attributing creativity to a computer program. *Minds and Machines* 17:76–99.

[Sasson Lazovsky, Raz, and Kenett 2025] Sasson Lazovsky, G.; Raz, T.; and Kenett, Y. N. 2025. The art of creative inquiry—from question asking to prompt engineering. *The Journal of Creative Behavior* 59(1):e671.

[Sawicki et al. 2023] Sawicki, P.; Grzes, M.; Góes, L. F.; Brown, D.; Peeperkorn, M.; Khatun, A.; and Paraskevopoulou, S. 2023. On the power of special-purpose GPT models to create and evaluate new poetry in old styles. In *Proceedings of the International Conference on Computational Creativity*, 10–19. Association for Computational Creativity.

[Shin et al. 2020] Shin, T.; Razeghi, Y.; Logan IV, R. L.; Wallace, E.; and Singh, S. 2020. AutoPrompt: Eliciting knowledge from Language models with automatically generated prompts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 4222–4235. Association for Computational Linguistics.

[Spendlove and Ventura 2020] Spendlove, B., and Ventura, D. 2020. Humans in the black box: A new paradigm for evaluating the design of creative systems. In *Proceedings of the International Conference on Computational Creativity*, 311–318. Association for Computational Creativity.

[Suzgun et al. 2023] Suzgun, M.; Scales, N.; Schärli, N.; Gehrmann, S.; Tay, Y.; Chung, H. W.; Chowdhery, A.; Le, Q.; Chi, E.; Zhou, D.; and Wei, J. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics*, 13003–13051. Association for Computational Linguistics.

[Toplyn 2021] Toplyn, J. 2021. Witscript: A system for generating improvised jokes in a conversation. In *Proceedings of the International Conference on Computational Creativity*, 22–31. Association for Computational Creativity.

[Touvron et al. 2023] Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; Bikel, D.; Blecher, L.; Ferrer, C. C.; Chen, M.; Cucurull, G.; Esiobu, D.; Fernandes, J.; Fu, J.; Fu, W.; Fuller, B.; Gao, C.; Goswami, V.; Goyal, N.; Hartshorn, A.; Hosseini, S.; Hou, R.; Inan, H.; Kardas, M.; Kerkez, V.; Khabsa, M.; Kloumann, I.; Korenev, A.; Koura, P. S.; Lachaux, M.-A.; Lavril, T.; Lee, J.; Liskovich, D.; Lu, Y.; Mao, Y.; Martinet, X.; Mihaylov, T.; Mishra, P.; Molybog, I.; Nie, Y.; Poulton, A.; Reizenstein, J.; Rungta, R.; Saladi, K.; Schelten, A.; Silva, R.; Smith, E. M.; Subramanian, R.; Tan, X. E.; Tang, B.; Taylor, R.; Williams, A.; Kuan, J. X.; Xu, P.; Yan, Z.; Zarov, I.; Zhang, Y.; Fan, A.; Kambadur, M.; Narang, S.; Rodriguez, A.; Stojnic, R.; Edunov, S.; and Scialom, T. 2023. Llama 2: Open foundation and fine-tuned chat models. https://arxiv.org/abs/2307.09288.

[Veale 2024] Veale, T. 2024. From symbolic caterpillars to stochastic butterflies: Case studies in re-implementing creative systems with LLMs. In *Proceedings of the International Conference on Computational Creativity*, 236–244. Association for Computational Creativity.

[Ventura 2017] Ventura, D. 2017. How to build a CC system. In *Proceedings of the International Conference on Computational Creativity*, 253–260. Association for Computational Creativity.

[Wei et al. 2022] Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 24824–24837. Curran Associates Inc.

[Wiggins 2006] Wiggins, G. A. 2006. A preliminary framework for description, analysis and comparison of creative systems. *Knowledge Based Systems* 19(7):449–458.

[Yang et al. 2024] Yang, C.; Wang, X.; Lu, Y.; Liu, H.; Le, Q. V.; Zhou, D.; and Chen, X. 2024. Large language models as optimizers. In *Proceedings of the International Conference on Learning Representations*.