Universiteit Leiden

Master Computer Science

The impact of model features on creative output in Large Language Models

Name:	Rens Anderson
Student ID:	S2550490
Date:	02/07/2025
Specialisation:	Data Science
1st supervisor:	Miros Zohrehvand
2nd supervisor:	Tessa Verhoef

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Einsteinweg 55
2333 CC Leiden
The Netherlands

Contents

1 Acknowledgments

I would like to express my deepest gratitude to my supervisors for their invaluable support and guidance throughout this research. Weekly meetings with my primary supervisor, Miros, were particularly helpful in addressing the challenges I encountered, and his insights greatly contributed to the overall quality of this work. I am sincerely thankful for his continuous assistance and encouragement.
I also wish to thank my second supervisor, whose expertise in creativity and computer science provided crucial perspectives on several aspects of the thesis. The advice and feedback I received were instrumental in shaping the direction of the research. I am truly grateful for their contributions.

Abstract

This study investigates how model features impact creative output. Creativity in language model outputs can be influenced by key generative parameters such as iteration count, temperature, and evaluator choice. We compared a subset of recipes from the Pillsbury Bake-Off 2024 with recipes generated by a Large Language Model (LLM) under different configurations of these parameters. An iterative framework was used in which recipes were generated and refined over multiple rounds, with outputs evaluated for creativity and quality. Creativity was assessed using the Torrance Tests of Creative Thinking (TTCT), applied through two LLM-based evaluators: one with 17 billion parameters and another with 8 billion. While the evaluation criteria remained constant, we found that the choice of evaluator significantly influenced creativity scores. Results showed that the larger evaluator model tended to produce lower creativity ratings, whereas the smaller model yielded higher scores. Additionally, generator temperature had minimal impact on most creativity metrics.

2 Introduction

Generative Artificial Intelligence (GenAI), particularly Large Language Models (LLMs), has gained significant attention for its ability to generate creative content across various domains. While creativity itself is a complex and multifaceted concept, it is often defined as the ability to produce novel and valuable ideas or artifacts [Doshi and Hauser(2024)]. In practice, creative output frequently emerges through an iterative process of generation, evaluation, and refinement [Sawyer(2021)]. Previous studies have shown that LLM performance depends on multiple factors, such as the number of parameters [Taylor(2024)]. Moreover, performance varies not only with model size but also with the specific task and architecture [Guzik and Gilde(2023)]. Additional research has examined the influence of human creativity, AI-generated creativity, and collaborative creativity involving both humans and AI [Doshi and Hauser(2024)].

However, the performance of large language models on creative tasks remains insufficiently studied, particularly in relation to how specific model settings influence their output. Temperature is often described as a "creativity parameter," but recent research suggests that its effect is more limited than commonly assumed [Peeperkorn(2024)]. Higher temperatures may increase novelty but tend to reduce coherence, indicating a trade-off rather than a straightforward improvement in creative quality [Peeperkorn(2024)]. These insights emphasize the need for more refined evaluation methods and generation strategies tailored to creative tasks. To investigate this, we apply the FunSearch algorithm [Romera-Paredes(2024)], which has shown strong performance on objective tasks such as mathematical problem-solving, to a subjective task: generating recipes for the Pillsbury Bake-Off. We conduct two experiments. The first explores how the number of iterations affects creative output, using 5, 15, and 30 FunSearch cycles. The second examines the effect of different temperature settings (0.5, 1.0, and 1.5), using two language models with different sizes: one with 8 billion parameters and one with 17 billion. Creativity is assessed using the Torrance Tests of Creative Thinking (TTCT), a widely recognized framework for evaluating creative potential [Zhao and Chen(2024)]. It conceptualizes creativity through four core psychological dimensions: fluency, flexibility, originality, and elaboration. Fluency refers to the number of relevant ideas an individual can produce, while flexibility

captures the variety of categories or perspectives reflected in those ideas. Originality denotes the uniqueness and statistical rarity of the responses, and elaboration reflects the depth, detail, and refinement with which an idea is developed.
Our findings show that increasing the number of iteration cycles does not significantly improve creative output. Interestingly, the smaller LLM significantly outperforms the larger model in terms of average creativity, particularly across most TTCT dimensions. Furthermore, temperature generally does not have a significant effect on creativity, except for a negative impact on originality observed at lower temperature settings.

3 Background & Related Work

Creativity is commonly defined as the ability to produce ideas or artifacts that are both novel and valuable [Doshi and Hauser(2024)]. This definition is widely accepted in both psychology and computer science and serves as the foundation for this study. Runco and Jaeger describe creativity as requiring both originality and effectiveness
[Runco and Jaeger(2012)]. Boden describes it as the ability to come up with ideas or artifacts that are new, surprising, and valuable [Boden(2004)], a description that has been especially influential in artificial intelligence research. Other studies offer broader frameworks to capture creativity's complexity. Rhodes, for example, introduced the influential 4Ps framework for understanding creativity, which includes Person, Process, Product, and Press [Rhodes(1961)]. The Person dimension encompasses individual characteristics such as personality traits, intellect, temperament, habits, attitudes, and values. Process refers to cognitive activities involved in creativity, including motivation, perception, learning, thinking, and communication. The Product represents the tangible outcome of creative thinking, such as a written text, artwork, or invention. Finally, Press describes the interaction between the individual and their environment, emphasizing how external conditions can influence creative behavior. In the field of Human-Computer Interaction, Frich and colleagues outline four perspectives on creativity. These include problem solving, cognitive emergence, embodied action, and tool-mediated expert activity [Hsueh(2024)]. Each reflects a different way of understanding creative practice, ranging from analytical tasks to hands-on interaction with materials. Although these perspectives vary, they tend to agree on three core aspects of creativity: novelty, value, and contextual relevance. These elements are central to both human and machine creativity, and they guide the evaluation used in this research.

3.1 Impact of model features on creativity

The creative performance of an LLM depends on both its underlying architecture and configurable parameters. Understanding how these factors influence the generation of creative output is essential for effectively evaluating model creativity. Key settings that may impact creativity include sampling temperature and model size.

3.1.1 Sampling temperature

Sampling temperature is a hyperparameter used in the decoding process of LLMs
[Renze(2024)]. It controls the randomness of the model's output during inference. During decoding, an LLM generates tokens sequentially, using previously generated tokens to

predict the next one. The final layer of the model outputs raw scores, known as logits, for each possible next token. These logits are then passed through a softmax function, which converts them into a probability distribution. The softmax function emphasizes differences among logits, allowing the model to assign higher probabilities to more likely tokens while still considering less likely options. This probability distribution is used for probabilistic sampling, where the next token is selected based on its assigned probability. In contrast, greedy sampling always selects the token with the highest probability, resulting in more deterministic and often less diverse outputs.
Temperature sampling modifies the softmax function by introducing a temperature parameter, τ, which affects the distribution of probabilities. Let v_k be the k-th vocabulary token and l_k its corresponding logit. The temperature-adjusted softmax function is defined as:

$$\Pr(v_k) = \frac{e^{l_k/\tau}}{\sum_i e^{l_i/\tau}}$$

The effect of temperature on the output distribution is as follows:

- Lower temperatures ($\tau \to 0$) make the probability distribution more peaked, increasing the likelihood of selecting the highest-probability tokens. This results in more deterministic, focused, and repetitive outputs that closely follow patterns from the training data. As a consequence, the model is less likely to produce diverse or novel outputs, reducing its potential for creative responses.

- Higher temperatures ($\tau > 1$) flatten the probability distribution, increasing the chances of selecting lower-probability tokens. This can lead to more diverse and potentially creative outputs. However, it may also increase the risk of factual inaccuracies or hallucinations, as the model is more likely to generate less likely and less grounded content.

Temperature in language models can be understood as a trade-off between exploration and exploitation. Lower temperatures tend to produce more predictable and conventional outputs, while higher temperatures encourage the generation of more diverse and potentially novel responses [Renze(2024)]. This balance plays an important role in shaping the creative capacity of large language models. Supporting this, recent advances in creativity evaluation, such as the Divergent Association Task (DAT), in which over $100,000$ participants generated semantically distant words, have shown that higher temperature settings lead to greater lexical diversity and reduced repetition [Bellemare-Pepin(2024)]. These results support the view that elevated temperatures can foster ideational fluency and lexical novelty, which are two core indicators of creativity.
Although temperature has little impact on performance in tasks that require accuracy, such as multiple-choice question answering [Renze(2024)], its influence appears more substantial in open-ended creative contexts such as storytelling and divergent thinking [Chakrabarty(2023)]. Higher temperatures may increase originality but often reduce coherence, suggesting that an optimal temperature range may exist for creativity-focused sampling.
However, the creative potential of language models and the effect of specific configuration settings are still not fully understood. While temperature is often described as a creativity parameter, recent empirical findings suggest its influence is more limited. Studies report

only a weak positive correlation between temperature and novelty, along with a moderate negative correlation with coherence [Peeperkorn(2024)]. Additionally, increasing temperature does not significantly expand the range of outputs when only a few samples are generated. These findings highlight the need for more advanced evaluation benchmarks and decoding strategies that are specifically designed to support creative language generation.

3.1.2 Model size

In addition to temperature, model size also affects the generated content. Many open-source LLMs include numerical identifiers in their names, which typically represent the total number of parameters in the model [Broadhead(2023)]. These parameters, which include weights and biases, determine how the model processes input data and generates output. Conceptually, they serve as part of the model's internal configuration, shaping the importance assigned to input features and influencing the construction of responses.
In general, models with a greater number of parameters tend to exhibit increased representational capacity and flexibility [Broadhead(2023)]. This is particularly true within the transformer architecture, where parameter scaling often correlates with the model's ability to capture complex patterns and generate more nuanced and accurate outputs. While increasing the parameter count can improve performance, recent research suggests that this benefit is task-dependent and not necessarily superior to scaling the training dataset size [Zhang(2024)].
At the same time, there is growing interest in the development and deployment of smaller LLMs as more resource-efficient alternatives to their larger counterparts. Notably, Microsoft's $Phi-2$ model, which contains 2.7 billion parameters, has demonstrated competitive performance on several benchmark tasks when compared to models with up to 70 billion parameters [Taylor(2024)]. Despite their smaller size, these models still demand substantial computational resources, particularly for fine-tuning or real-time deployment, which can pose practical challenges in terms of cost and latency.
Models with larger parameter sizes tend to generate more complex and nuanced responses, potentially contributing to greater creativity in tasks that require intricate detail and coherence [Burtsev(2023)]. However, for creative tasks like storytelling, smaller models can still perform competitively with the right fine-tuning and prompt engineering [Marco(2025)].
Recent research highlights the potential of small language models (SLMs) in creative writing tasks, particularly in generating short stories [Marco(2025)]. The study indicates that smaller language models, despite their limited size and complexity, are capable of matching or even exceeding human performance in various dimensions of creative writing, especially when assessed by general audiences. The evaluation consisted of two experiments: (i) a human study in which 68 participants rated short stories generated by both humans and the SLM on grammaticality, relevance, creativity, and attractiveness, and (ii) a qualitative linguistic analysis examining the textual characteristics of the stories produced by each model. These findings challenge the assumption that only large and sophisticated models are suitable for creative tasks, showing that SLMs can generate content that resonates with readers while requiring significantly fewer computational resources.
Furthermore, research shows that both LLMs and SLMs can exceed human performance on creative benchmarks such as the Divergent Association Task or short-form storytelling.

These findings challenge traditional views on human–machine creative boundaries and emphasize the importance of using nuanced evaluation criteria such as elaboration, surprise, and relevance beyond mere novelty [Bellemare-Pepin(2024), Marco(2025)].
Taken together, these findings suggest that effective creative performance in LLMs depends on a careful interplay of sampling strategies and model capacity. Configurations that encourage exploratory generation without sacrificing coherence are particularly important for tasks requiring originality and ideational fluency.

4 Methodology

This study investigates how specific model features influence the creative performance of large language models in the context of recipe generation. Multiple experimental setups were employed, with creativity evaluated using the FunSearch algorithm and large language models serving as evaluators.
Following the competition's official guidelines, we implemented a weighted scoring rubric in which the recipe accounts for 70 percent of the total score and the accompanying story for 30 percent. The recipe component was further divided into four dimensions: taste, appearance, creativity, and crowd appeal. Each category was rated by the LLM on a scale from 1 to 5, where 1 represents the lowest possible score and 5 the highest. The story component was assessed along three dimensions: how the recipe ties to the story, representation of family values or traditions, and expression of personal passion. Each of these was also rated on a scale from 1 to 5.
To establish a benchmark for model-based evaluation, we curated a dataset comprising the 2024 Pillsbury Bake-Off winning recipe and 29 additional entries selected at random from the same competition. This benchmark served to approximate human-level performance. All 30 recipes were evaluated by the models according to the official Pillsbury judging criteria, focusing exclusively on the recipe component, as the accompanying stories were not publicly available. The winning recipe from the Pillsbury Bake-Off is presented in Appendix 9.3.1.
We first evaluated whether the LLMs $meta-llama-Llama-4-Scout-17B-16E-Instruct$ $(Meta-17B)$ and $meta-llama-Meta-Llama-3.1-8B-Instruct$ $(Meta-8B)$ could serve as effective evaluators, and thus act as surrogate judges for the Pillsbury Bake-Off. To assess their ability to distinguish between high- and low-quality recipes, we applied the evaluation prompt described in Appendix 9.2.3 to the curated Pillsbury dataset. Following this, creativity was assessed using four LLMs: $gpt-4o-mini$ (GPT), $deepseek-ai-DeepSeek-R1-Distill-Llama-70B$ $(Llama)$, $microsoft-Phi-4-multimodal-instruct$ (Phi), and $Meta-17B$, with the prompt provided in Appendix 9.2.4. The use of multiple LLMs aimed to mitigate individual model bias and reduce the risk of hallucinated or inconsistent evaluations.

4.1 Measuring Creativity

Studies have investigated how to measure creativity by incorporating subjective dimensions such as self-expression, satisfaction, ease, enjoyment, uniqueness, ownership, and pride. In addition, concerns such as deceptive content, plagiarism, invasion of privacy, and discrimination are also considered [Li and Yin(2024)]. These aspects are typically

scored using Likert-scale ratings (e.g., 1 to 5), allowing for a multidimensional evaluation of perceived creativity.

A growing number of studies focus on LLM-generated narrative text, employing rubrics developed by literary experts. These often assess readability, plot structure, tone consistency, character development, and literary style [Gómez-Rodríguez and Williams(2023)]. For instance, one study scored $GPT-3.5$ generated short stories across narrative dimensions without fine-tuning or prompt engineering. However, qualitative feedback from professional writers highlights ongoing model limitations, such as overuse of clichés, predictable or moralistic endings, and lack of nuanced storytelling [Chakrabarty(2023)].

Recent work has begun to automate TTCT-style evaluation using LLMs, enabling scalable and efficient assessment of creativity [Zhao and Chen(2024)]. In this setup, one LLM, typically $GPT-4$, evaluates the outputs of other LLMs along the four TTCT dimensions. This method bypasses the need for manual raters and has proven effective for open-ended, language-based tasks such as story or recipe generation. Notably, research reports that $GPT-4$ not only serves as an evaluator, but also as a test subject [Guzik and Gilde(2023)]. In a comparative study involving 24 human participants, $GPT-4$ was assessed using six classic TTCT tasks: Asking Questions, Guessing Causes, Guessing Consequences, Product Improvement, Unusual Uses, and Just Suppose. $GPT-4$ consistently outperformed the human group across these tasks. This result underscores $GPT-4$'s capacity to generate responses that score highly across multiple dimensions of creativity and further supports the TTCT as a robust framework for evaluating both human and machine creativity.

Complementing TTCT, a newer framework, the Torrance Test of Creative Writing (TTCW), focuses on creativity as a product using 14 binary rubric, based evaluations mapped to TTCT's four core dimensions [Chakrabarty(2024)]. In this setup, professional evaluators scored human- and LLM-authored stories, revealing that LLMs passed 3–10 fewer TTCW criteria than human-written texts. Further, LLMs used as evaluators showed no correlation with expert judgment, raising concerns about using current models to assess creativity objectively.

Beyond TTCT, more experimental approaches include Generalized Additive Models (GAMs) to track creative search behavior across functional, visual, and data dimensions in a design context [Cheoh(2024)]. These models reveal how users shift their creative focus over time, providing a dynamic view of exploration during the creative process.

In Human-AI interaction studies, researchers have explored how AI impacts human-to-human creative performance and enhances individuals' creative capabilities through collaboration, particularly in a collaborative version of the Alternative Uses Task (AUT), which is a well-established and validated measure of creative potential [Bangerl(2025)]. The AUT is typically scored on four dimensions: fluency, flexibility, originality, and elaboration. Compared to the AUT, the TTCT assess a broader range of creative abilities. While the AUT focuses more narrowly on verbal and conceptual divergent thinking [Erwin(2022)]. In a separate study, the role of LLMs as co-creators was examined in the context of humorous content generation, specifically through the creation of internet memes [Wu(2025)]. The quality of the generated memes was evaluated through crowdsourced ratings based on creativity, humor, and shareability.

This approach contrasts with creativity evaluation in domains like programming, where outputs often have clear correctness metrics or "golden responses"
[DeLorenzo and Rajendran(2024)]. In such cases, creativity can be judged by how novel

or efficient a solution is relative to an ideal standard. However, in open-ended tasks without a single correct answer, like creative writing or recipe invention, such gold-standard evaluation is not applicable, making TTCT-based methods more suitable.
Alternative frameworks such as the Value-Novelty-Surprise (VNS) model have also been applied, using discriminators and distance-based methods to score creativity in structured domains like historical poetry [Franceschelli and Musolesi(2022)]. However, since such works rely on pre-established creativity benchmarks, which do not exist for domains like recipes, these methods are not directly transferable to our study.
Given the findings from previous studies, creativity in our research was assessed using four dimensions adapted from the TTCT. Each LLM evaluated the generated recipes on a five-point scale, ranging from 1 (low) to 5 (high), across the following categories: Fluency, Flexibility, Elaboration and Originality (see Subsection Appendix 9.2.4).
The four scores were averaged for each model, and the final creativity score was computed as the mean of these averages across all four LLMs. The experiments below were designed to compare FunSearch-generated recipes to this benchmark.

4.1.1 Prompt Design

In this research, we worked extensively on the design and refinement of prompts. To generate accurate and informative responses from the language model, we used Chain of Thought (CoT) prompting. This method encourages the model to produce intermediate reasoning steps that improve its understanding and logical coherence [Wei(2023)]. CoT prompting is especially effective for tasks involving complex reasoning, such as arithmetic, commonsense judgments, and symbolic logic.
In our context, we applied CoT prompting to guide qualitative assessments, such as evaluating taste quality or the appeal of a dish to a crowd. The language model was first asked to provide a written explanation for its evaluation and then to assign a quantitative score to the same attribute. For example, after reasoning about taste, the model would provide a numerical score for taste quality.
Each prompt also included a defined persona at the beginning. In our case, the model was instructed to adopt the role of a participant in the Pillsbury Bake-Off or a competition judge. This combination of persona-based role framing and step-by-step reasoning was inspired by previous work [Choudhury(2024)], which examined whether generative artificial intelligence could effectively replicate executive communication.
In addition, that study used a fixed JSON format as the expected output structure. Without specifying a structured output format, the model produced responses in a wide range of formats, making it difficult to extract the relevant values. The use of JSON ensured clarity and made the results easier to process and evaluate.

4.2 FunSearch Algorithm

This study employs the FunSearch (Searching in Function Space) algorithm, a method designed to efficiently explore complex solution spaces by iteratively generating and evaluating candidate programs [Romera-Paredes(2024)]. Previous research has shown that this approach can exceed the best-known results in significant problem domains, demonstrating the potential of large language models to contribute meaningfully to challenging computational tasks. For example, when applied to the cap set problem in extremal combinatorics, FunSearch discovered new constructions that improve upon existing results

in both finite-dimensional and asymptotic settings [Romera-Paredes(2024)]. This provides evidence that large language models can be used to make novel contributions to established open problems. In addition, FunSearch has been successfully applied to an algorithmic challenge in online bin packing, where it identified new heuristics that outperform commonly used baselines. A key distinction of FunSearch is that it searches for programs that describe how to solve a problem, rather than simply identifying solutions. This programmatic focus results in outputs that are often more interpretable and suitable for refinement in collaboration with domain experts [Romera-Paredes(2024)]. It also supports the practical deployment of these programs in real-world contexts. Given its effectiveness, general applicability, and support for interpretability and expert feedback, we consider FunSearch a promising technique for studying creativity in computational systems.

Evaluating creative performance in subjective domains such as recipe generation presents significant methodological challenges. While human evaluation is often considered the golden standard, it is costly, time-consuming, and difficult to scale, particularly when iterative, fine-grained judgments are needed across a large number of generated outputs [Pandhare(2024)]. In this study, we employ LLMs not only for generation but also for initial evaluation of recipe ideas, following a structured prompting strategy. This approach aligns with recent advances in LLM-driven optimization frameworks, such as the FunSearch algorithm [Romera-Paredes(2024)], which relies on model-guided generation and evaluation cycles to explore high-dimensional solution spaces. Although concerns have been raised about potential circularity and bias—particularly the tendency of models to prefer their own outputs [Wataoka(2025), Panickssery(2024)], prior research provides a more nuanced picture. For instance, research found that self-evaluation by LLMs can improve the overall quality of selected outputs, suggesting that reflection-oriented evaluation prompts can support refinement rather than merely reinforce prior bias [Ren(2023)]. Similarly, research showed that high-performing models are capable of providing reliable self-evaluations, especially when guided by chain-of-thought prompting, though the risk of degraded judgment remains higher for smaller or underperforming models [Chen(2025)]. Moreover, several studies have demonstrated that LLM-based evaluations can closely align with human judgments in subjective domains, offering a practical and scalable alternative where human raters are infeasible [Ren(2023), Chen(2025)]. In our case, this initial evaluation served to identify high-potential recipes based on performance expectations within a subjective cooking competition. To mitigate any potential evaluation bias, we further validated the creativity of the winning recipes using four independently prompted LLMs, ensuring broader model diversity and reducing self-preference effects.

The core components of the FunSearch algorithm include a *skeleton*, *island method*, *best-shot prompting*, *evaluator*, *programs database* and *iterations*. In our implementation, the generator is based on the LLM model $meta-llama/Llama-4-Scout-17B-16E-Instruct$ ($Meta-17B$), while the evaluator is based on the models: $Meta-8B$ or $Meta-17B$. The algorithm further integrates the island method with best-shot prompting to promote diversity and optimization within the solution space. In this study, the *FunSearch* algorithm was employed within a structured framework to assess creativity, which can be seen in Figure 1. The algorithm was executed for n iterations to populate the programs database. After each set of n iterations, the best valid solution (i.e., recipe) was selected and stored in a separate list. This process was repeated 30 times, resulting in a final list of 30 top-performing recipes. These 30 recipes were subsequently evaluated by four LLMs

using criteria based on the TTCT, allowing for the determination of a creativity score. They were also compared to the 30 recipes from the Pillsbury Bake-Off, which serve as the benchmark.

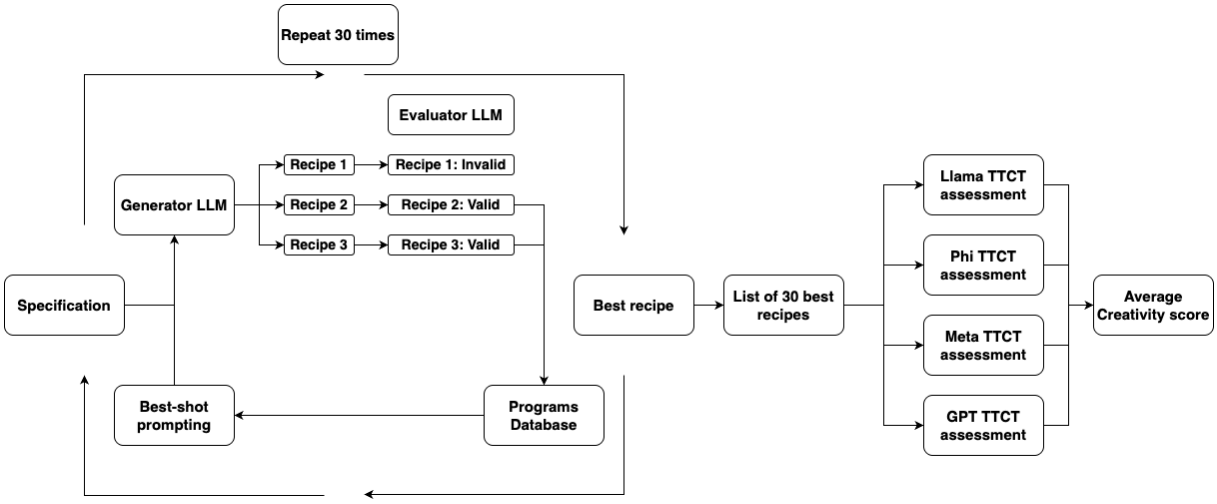

Figure 1: Conceptual framework illustrating how the FunSearch algorithm is integrated with the Torrance Tests of Creative Thinking (TTCT) to evaluate creative potential.

4.2.1 Skeleton

The skeleton defines the structural constraints that candidate solutions must satisfy. Since this study is centered around the Pillsbury Bake-Off challenge, the official competition rules were used to construct the skeleton. Each recipe submission must include the following: a title, a list of no more than ten ingredients (excluding common items such as salt and pepper), at least one and no more than one ingredient from the official Pillsbury ingredient list, and a preparation time of under 30 minutes. The submission must also include preparation instructions (up to 2000 characters) and a brief essay (maximum 500 characters) describing the story behind the dish.

4.2.2 Island Method

The island method enables parallel exploration of the solution space by evaluating distinct candidates in isolated environments. The evaluation is performed by an LLM, as described in 4.2.4. In this setting, each island corresponds to a unique initial recipe that adheres to the skeleton constraints. A population of n such islands is generated, and each candidate is evaluated independently. This structure supports exploration of diverse recipe variants without immediate cross-island interference. Each island is initialized with a valid recipe generated by the LLM, using the starter prompt provided in Appendix 9.2.1.

4.2.3 Best-shot prompting

To optimize individual islands, we apply a best-shot prompting strategy, using the prompt described in Appendix 9.2.2. A single island is randomly selected, from which a cluster of recipes is sampled using a softmax-based selection mechanism. This probabilistic approach

favors clusters with higher average ratings while preserving diversity by maintaining a non-zero selection probability for lower-rated clusters. Within the chosen cluster, the top two recipes (ranked by rating) are selected as exemplar prompts. This choice is motivated by the findings of [Romera-Paredes(2024)], where using the top two solutions proved effective for optimizing mathematical tasks. The language model is then prompted with these exemplars to generate an improved recipe variant.

4.2.4 Evaluator

The LLM evaluator assigns a score to each candidate solution using the prompt in Appendix 9.2.3, where higher scores indicate better performance. Scoring criteria depend on the specific task; in this study, evaluation is modeled after the official rules of the Pillsbury Bake-Off. Each recipe is scored based on two components: recipe quality (70%) and storytelling (30%), as defined by the competition's guidelines (see Section 4). The final score is given on a 1–5 scale.Recipe evaluation is conducted using large language models, specifically either $Meta - 8B$ or $Meta - 17B$.

4.2.5 Programs Database

The programs database stores only valid solutions, those that conform to the required structural format, or "skeleton" (see Subsection 4.2.1). Any candidate solution that does not follow this prescribed structure is considered invalid and is excluded from the database. Valid solutions are stored along with their evaluation scores in a sorted manner. In the context of this research, this means that a recipe must adhere to the official competition rules to be considered structurally valid.

4.2.6 Iteration

After each island is initialized with a valid solution generated from the starter prompt, the solution is evaluated and stored in the programs database. The algorithm then proceeds iteratively. In each iteration, every island uses its initial solution to generate a set of new candidate recipes using the best-shot prompting.
Each new candidate is evaluated by the LLM using the evaluation prompt (see Subsection Appendix 9.2.3). If a candidate is missing required structural elements, such as a name or score field as defined by the skeleton, it is considered invalid and excluded from the programs database. Valid candidates, on the other hand, are stored along with their scores for future selection.

4.2.7 Creativity score

After each FunSearch iteration, the best solution is selected and stored, resulting in a final list of 30 recipes after 30 iterations. These recipes are subsequently evaluated by four different LLMs using the prompt described in Appendix 9.2.4. The use of multiple LLMs helps mitigate variability in model outputs and reduces the risk of biased or inconsistent evaluations. A similar evaluation strategy is employed for the Pillsbury Bake-Off benchmark recipes to ensure a consistent basis for comparison.

4.3 Experiment 1: Effect of Iteration Count

In the first experiment, we investigated how the number of FunSearch iterations influences overall recipe quality. Each run involved generating and evaluating new candidate recipes over 5, 15, or 30 iterations. After each run, the highest scoring recipe, based on evaluations from the LLM evaluator, was retained and added to the final dataset. This process was repeated 30 times for each configuration. The resulting collection of top ranked recipes was then evaluated using the Pillsbury Bake-Off benchmark, with creativity assessed by four different LLMs.

4.4 Experiment 2: Effects of generation temperature and evaluator size on creativity

The second experiment examined the interaction between generator temperature and the evaluator model architecture. We tested temperatures of 0.5, 1.0, and 1.5, extending into higher ranges to investigate whether increased randomness contributes to greater creative potential. We compared two large language models used as evaluators in the FunSearch framework. The first was a high-capacity model, $Meta - 17B$, containing approximately 17 billion parameters. The second was a smaller model of similar architecture, $Meta - 8B$, which consists of 8 billion parameters. This comparison allowed us to examine how model size influences the evaluation of creative outputs. These evaluators were responsible for assessing recipe quality using the Pillsbury Bake-Off rubric, which includes criteria such as taste, creativity, appearance, and storytelling.

The goal of this experiment was to assess whether the size of the evaluator model and the sampling temperature of the generator influence the creative quality of the generated recipes. To assess the impact of model configuration on creative performance, we conducted a controlled experiment using the FunSearch algorithm. We fixed the number of islands to seven and the batch size to five. For each configuration, the algorithm was executed thirty times, retaining the top-performing recipe from each run. This procedure yielded thirty recipes per condition, with six experimental conditions in total. These conditions varied along three binary factors: the temperature setting of the generator during the low-level search (LowTemp), the temperature setting during the high-level search (HighTemp), and the size of the evaluator model (LargeEvaluator), which was either $Meta - 17B$ or $Meta - 8B$.

To quantify the influence of these variables on creative output, we estimated an ordinary least squares (OLS) regression of the following form:

$Creativity = \alpha + \beta \cdot HighTemp + \gamma \cdot LowTemp + \delta \cdot LargeEvaluator + \varepsilon$

Here, HighTemp is equal to 1 if the temperature during high-level search was set to 1.0, and 0 otherwise. LowTemp equals 1 if the temperature during low-level search was set to 0.5, and 0 otherwise. LargeEvaluator is a binary variable equal to 1 if the evaluator was the larger 17-billion-parameter model, and 0 if it was the 8-billion-parameter model. The total sample consisted of 180 recipes, reflecting thirty observations for each of the six experimental conditions. This regression framework was applied not only to the overall creativity score but also independently to each of the four dimensions of the TTCT framework: fluency, flexibility, originality, and elaboration. As a result, we estimated a total of five separate linear regression models.

5 Results

To evaluate whether the model-based evaluator could distinguish between high- and low-quality recipes, we applied the prompt described in Appendix 9.2.3 to the Pillsbury dataset. In this assessment, the actual winning recipe was ranked third by $Meta-17B$ and fourth by $Meta-8B$. These results indicate that the models were capable of identifying higher-quality recipes, demonstrating partial alignment with human judgment based on the recipe content alone. Creativity was subsequently assessed using four LLMs: GPT, $Llama$, Phi, and $Meta-17B$. The winning entry's creativity score exceeded the mean of the other 29 entries by more than one standard deviation, confirming its status as an outlier.

5.1 Experiment 1: Effect of Iteration Count

In the first experiment, we evaluated the impact of the number of iterations on the performance of the FunSearch algorithm. For each configuration, the algorithm was run for a predetermined number of iterations, during which valid recipes were continuously added to the database. After each complete run, the best-performing recipe from the database was selected. This process was repeated 30 times to produce a dataset of comparable size to that of the Pillsbury competition. Appendix 9.3.2 presents one of the 30 generated recipes, notable for its exceptionally high creativity score.

The experiment was conducted with iteration counts of 5, 15, and 30. The corresponding results are reported in Figure 2 and Table 1. The mean creativity scores produced by the FunSearch algorithm were 3.921, 3.835, and 3.927 for 5, 15, and 30 iterations, respectively. All three scores exceed the average creativity score of the Pillsbury benchmark dataset, which was 3.638. In addition, the standard deviations of the FunSearch outputs were 0.445, 0.456, and 0.411 for 5, 15, and 30 iterations, respectively. These values are higher than the standard deviation of the Pillsbury dataset, which was 0.328, indicating greater variability in the generated content.

Given the marginal differences in performance across iteration counts, the second experiment proceeded with 5 iterations. This choice was motivated by the desire to reduce computational time and resource costs without compromising effectiveness.

Iterations	**Mean**	**SD**
Pillsbury	3.638	0.328
5	3.921	0.445
15	3.835	0.456
30	3.927	0.411

Table 1: Mean and standard deviation (SD) of creativity scores for the best recipe selected by the FunSearch algorithm across different iteration counts.

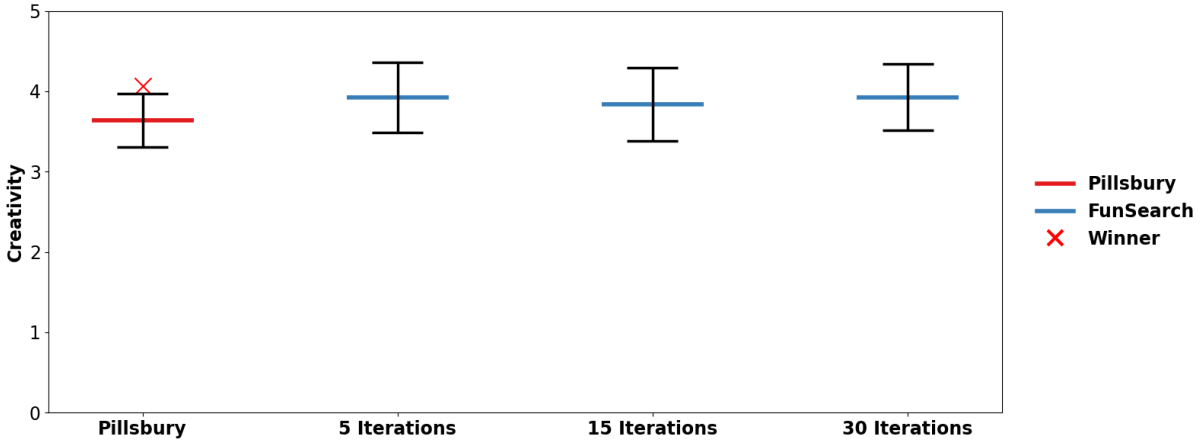

Figure 2: Comparison of FunSearch algorithm performance across different iteration counts against the Pillsbury Bake-Off dataset, measured by average Creativity (y-axis). The red bar represents the average score of the 30 Pillsbury recipes, with a red cross indicating the score of the winning Pillsbury recipe. Blue bars correspond to the 30 FunSearch-generated recipes, where each bar reflects a different number of algorithm iterations.

5.2 Experiment 2: Effects of generation temperature and evaluator size on creativity

The second experiment examined the interaction between the evaluator model and the generator temperature within the FunSearch algorithm. Specifically, we assessed the impact of using a less capable language model as the evaluator, characterized by a smaller parameter count, under the assumption that it would yield lower evaluation quality. This model was compared against the higher-capacity LLM used in the previous experiments. Simultaneously, we varied the temperature of the generator LLM across three settings: 0.5, 1.0, and 1.5.

In this experiment, we analyzed not only the overall creativity scores but also the individual dimensions of the TTCT: Fluency, Flexibility, Originality, and Elaboration. The results are shown in the figures below. All configurations in this experiment used a batch size of 5, island size of 7, and 5 iterations, parameters selected for their balance between performance and efficiency.

Table 2 presents coefficient estimates from five separate linear regressions, each predicting one of the divergent thinking scores: Creativity, Fluency, Flexibility, Originality, or Elaboration. The models follow the specification:

$$Creativity = \alpha + \beta \cdot HighTemp + \gamma \cdot LowTemp + \delta \cdot LargeEvaluator + \varepsilon.$$

with binary indicators for generator temperature and evaluator model size as independent variables.

The variable *LargeEvaluator*, which equals one when the $Meta - 17B$ model is used for evaluation, shows a statistically significant negative association with four of the five outcome measures. Specifically, the coefficient is -0.161 for Creativity, -0.128 for Fluency, -0.307 for Flexibility, and -0.231 for Elaboration. Each of these effects is significant at the five percent level.

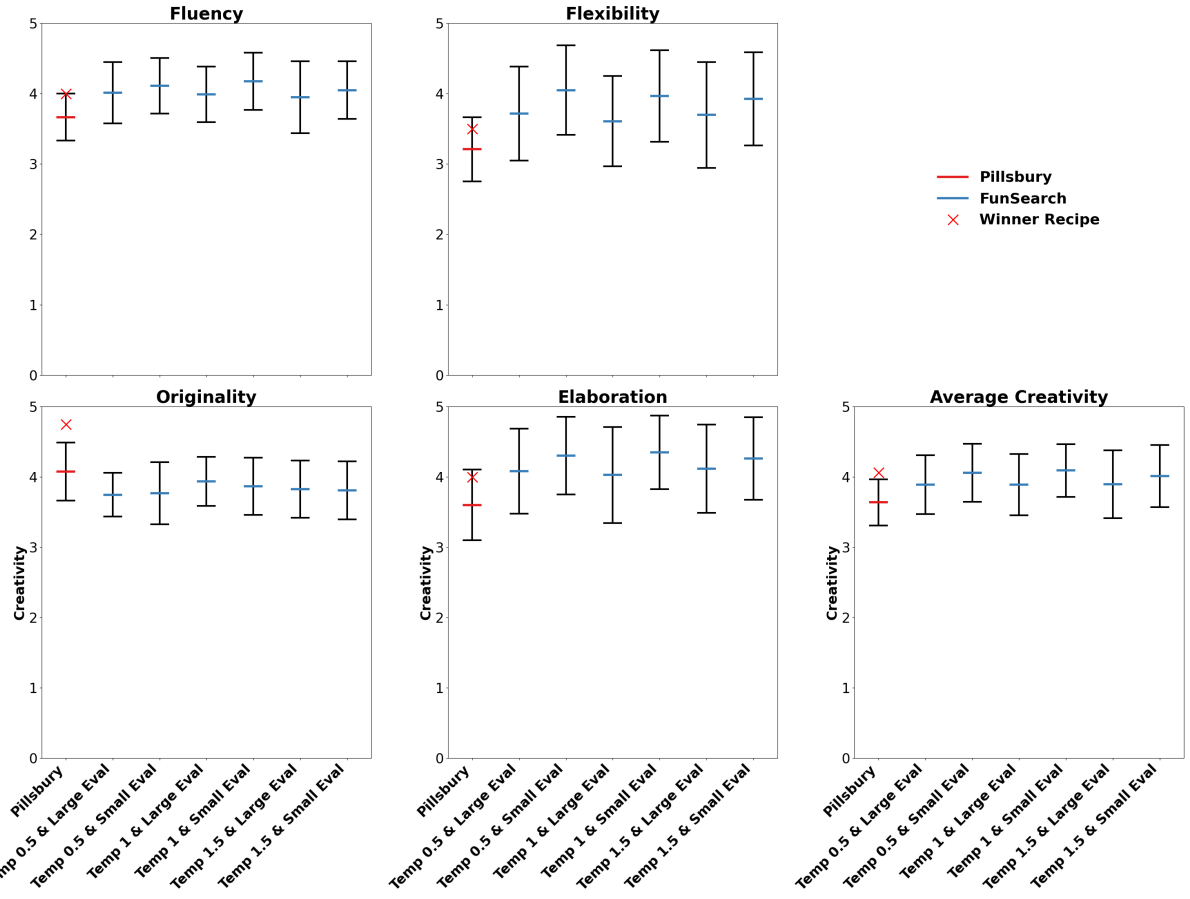

Figure 3: Average creativity scores (on a scale from 1 to 5) across five creativity dimensions: Average Creativity, Fluency, Flexibility, Originality, and Elaboration, for recipes generated by the FunSearch algorithm, compared to the baseline from the Pillsbury dataset. 'Temp' refers to the generator temperature, and 'Eval' specifies the evaluation model. 'Large' uses the 17B parameter LLM, while 'Small' uses the 8B parameter LLM.

Variable	(1) Creativity	(2) Fluency	(3) Flexibility	(4) Originality	(5) Elaboration
LowTemp	-0.017	-0.021	0.096	-0.144*	0.002
	(0.079)	(0.079)	(0.124)	(0.072)	(0.111)
HighTemp	-0.037	-0.083	0.023	-0.083	-0.002
	(0.079)	(0.079)	(0.124)	(0.072)	(0.111)
LargeEvaluator	-0.161*	-0.128*	-0.307*	0.021	-0.231*
	(0.065)	(0.065)	(0.101)	(0.059)	(0.090)
Constant	4.070	4.147	3.941	3.890	4.303
	(0.065)	(0.065)	(0.101)	(0.059)	(0.090)
N	180	180	180	180	180
Adj R^2	0.019	0.012	0.037	0.006	0.019

Table 2: *Notes.* This table reports coefficient estimates from the following regression model applied to the full sample of evaluated recipes:
$Creativity = \alpha + \beta \cdot HighTemp + \gamma \cdot LowTemp + \delta \cdot LargeEvaluator + \varepsilon$.
HighTemp is a binary indicator equal to 1 when the generator temperature is 1, and 0 otherwise. *LowTemp* equals 1 when the temperature is 0.5, and 0 otherwise. *LargeEvaluator* equals 1 if the evaluator is $Meta - 17B$, and 0 if it is $Meta - 8B$. The outcome variable is one of the four divergent thinking scores (Fluency, Flexibility, Originality, and Elaboration) or their average (Creativity). The table reports estimated coefficients, model constants, standard deviations, sample size (N), and adjusted R^2 values. Asterisks (*) denote statistical significance at the 5% level.

In the Originality model, the coefficient for *LargeEvaluator* is 0.021 and is not statistically significant. The standard errors for these coefficients are 0.065, 0.065, 0.101, and 0.090, respectively.

The temperature conditions are represented by the *LowTemp* and *HighTemp* indicators. These variables do not exhibit consistent or statistically significant effects across the models. The only exception is a negative coefficient of -0.144 for *LowTemp* in the Originality model, which is statistically significant at the five percent level. The standard error for this coefficient is 0.072. All other coefficients for temperature variables range from -0.083 to 0.096, and their corresponding standard errors range from 0.079 to 0.124. These coefficients are not statistically significant.

The constant terms reflect the average outcome when all binary predictors are equal to zero. These values range from 3.890 for Originality to 4.303 for Elaboration, with standard errors of 0.059 and 0.090, respectively. These relatively high baseline scores suggest a consistent average across all dimensions.

Adjusted R^2 values range from 0.006 to 0.037, indicating that the explanatory variables account for only a small proportion of the variance in the outcome variables.

Figure 3 visually presents the average scores across all five dimensions for each experimental condition, compared to the Pillsbury dataset baseline.

6 Discussion

Previous research has largely focused on the performance of Large Language Models in objective tasks, where improvements are strongly associated with increases in model size [Taylor(2024)]. This has led to the assumption that larger models perform better across all domains, including creative ones. However, creativity is commonly defined as the ability to produce novel and valuable ideas or artifacts [Doshi and Hauser(2024)], which makes it fundamentally different from tasks with clear or factual outcomes.

A dimension we explored was iteration, which plays a central role in many human creative processes. The FunSearch algorithm, which uses multiple cycles of generation and evaluation, was used to test whether more iterations would improve the creative quality of outputs. Despite its effectiveness in objective problem-solving, FunSearch did not yield a noticeable improvement in creativity as the number of cycles increased. This suggests that iterative generation alone is not sufficient.

Our findings also challenge the idea that scaling model size results in better creative output. Contrary to expectations based on prior work that links larger models to better performance [Broadhead(2023)], the smaller model with 8 billion parameters significantly outperformed the larger 16 billion parameter model creative tasks. This was especially evident in scores for average creativity, fluency, flexibility and elaboration based on the Torrance Tests of Creative Thinking. This suggests that performance advantages found in larger models may not extend to domains that require subjective or imaginative capabilities.

We also explored the effect of temperature, a parameter often linked to creativity in LLMs. While earlier studies have suggested that increasing temperature can boost novelty by introducing more randomness into the model's output [Peeperkorn(2024)], our results present a more limited effect. Our analysis revealed that lower temperature settings were associated with a decline in originality, which aligns with existing research suggesting that reduced randomness in model outputs (i.e., lower temperature) often leads to more predictable and conventional results. In contrast, higher temperature settings, which introduce greater randomness and diversity in the generation process, did not consistently lead to improvements across other creativity dimensions measured by the TTCT. This may indicate that while diversity in model outputs (higher temperature) can promote novelty, it does not necessarily enhance other dimensions of creativity, such as fluency, flexibility, or elaboration, without careful calibration of the temperature parameter.

This research contributes to a relatively underexplored area in LLM studies by examining the relationship between model features and creative performance. While earlier studies have emphasized gains in performance through increased scale [Taylor(2024)], few have investigated how those strategies translate to creativity. Our findings indicate that creative output does not follow the same patterns observed in standard LLM benchmarks. Our results show that, among the variables tested, only model size had a significant effect, with the smaller language model producing more creative output. Additionally, lower temperature settings had a significant negative coefficient on originality, suggesting that reduced randomness may limit creative output in terms of originality.

7 Limitations and Future Work

One limitation of this study is that both the generator and evaluator components of the FunSearch algorithm were based on the same language model. While this approach ensures consistency, it may also introduce bias or limit the diversity of evaluation. Future research could investigate the effects of using a separate model as the evaluator to see whether a different perspective results in higher-quality or more varied creative outputs.
Another limitation is that although the iterative structure of FunSearch mimics important aspects of human creativity, it may not be sufficient for improving subjective outcomes. Iteration without more sophisticated evaluation criteria or feedback does not necessarily lead to better results. Future research could explore generation strategies that incorporate adaptive evaluation, learning from human feedback, or interactive human input throughout the process.
These limitations highlight the need for continued research into how we improve creativity in artificial systems. A deeper understanding of these factors will be essential for developing models that can support and collaborate in genuinely creative work.

8 Conclusion

This study introduces a quantitative framework for evaluating creativity and applies it to the domain of recipe generation. By benchmarking outputs from a large language model against entries from the 2024 Pillsbury Bake-Off, we examine the role of model size and temperature settings in shaping creative output. In particular, we observe that a smaller evaluation model was associated with higher creativity ratings, raising questions about how model capacity interacts with perceptions of creativity. Additionally, temperature settings played a role in influencing creativity, with lower temperatures being associated with a decline in originality. While the scope of this study was limited, it offers a foundation for further investigation into how creativity emerges in artificial systems and how it can be meaningfully assessed. Understanding when and why smaller models outperform larger ones in creative domains, and exploring generation strategies that go beyond simple iteration, will be key to advancing our understanding of creativity in artificial systems.

References

[Adler(1996)] P. Adler. 1996. The Dynamic Relationship Between Tacit and Codified Knowledge: Comments on Ikujiro Nonaka's, "Managing Innovation as an Organizational Knowledge Creation Process". *Pogorel, G. and Allouche, J. (eds.) International Handbook of Technology Managemen* (1996).

[Bangerl(2025)] David T. Disch L. Pammer-Schindler V. Bangerl, M. 2025. CreAItive Collaboration? Users' Misjudgment of AI-Creativity Afects Their Collaborative Performance. *CHI* (2025).

[Bellemare-Pepin(2024)] Lespinasse F. Thölke P. Harel-Y. Mathewson K. Olson J.-Bengio Y Jerbi K. Bellemare-Pepin, A. 2024. Divergent Creativity in Humans and Large Language Models. *arXiv* (2024).

[Boden(2004)] M. A. Boden. 2004. The Creative Mind: Myths and Mechanisms. *Psychology Press* (2004).

[Broadhead(2023)] G. Broadhead. 2023. A Brief Guide To LLM Numbers: Parameter Count vs. Training Size. *Medium* (2023).

[Burtsev(2023)] Reeves M. Job A. Burtsev, M. 2023. The Working Limitations of Large Language Models. *MIT Sloan* (2023).

[Chakrabarty(2024)] Laban P. Agarwal D. Muresan S. Wu S. Chakrabarty, T. 2024. Art or Artifice? Large Language Models and the False Promise of Creativity. *CHI* (2024).

[Chakrabarty(2023)] Padmakumar V. Brahman F. Muresan S. Chakrabarty, T. 2023. Creativity Support in the Age of Large Language Models: An Empirical Study Involving Emerging Writers. *ACM* (2023).

[Chen(2025)] Wei Z. Zhu X. Feng S. Meng Y. Chen, W. 2025. Do LLM Evaluators Prefer Themselves for a Reason? *arXiv* (2025).

[Cheoh(2024)] J. Cheoh. 2024. Modeling Multidimensional Cognitive Search in Creativity with Generalized Additive Modely. *ACM* (2024).

[Choudhury(2024)] Vanneste B. Zohrehvand A. Choudhury, P. 2024. The Wade Test: Generative AI and CEO Communication. *Harvard Business School* (2024).

[DeLorenzo and Rajendran(2024)] Gohil V. DeLorenzo, M. and J. Rajendran. 2024. CreativEval: Evaluating Creativity of LLM-Based Hardware Code Generation. *Texas AM University, USA* (2024).

[Doshi and Hauser(2024)] A. Doshi and O. Hauser. 2024. Generative AI enhances individual creativity butreduces the collective diversity of novel content. *Science Advances* (2024).

[Durmus and Isaac(2024)] Giretti A. Ashkenazi O. Carbonari A. Durmus, D. and S. Isaac. 2024. The Role of Large Language Models for Decision Support in Fire Safety Planning. *ISARC* (2024).

[Erwin(2022)] Tran Kl. Koutstaal W. Erwin, E. 2022. Evaluating the predictive validity of four divergent thinking tasks for the originality of design product ideation. *PLOS ONE* (2022).

[Franceschelli and Musolesi(2022)] G. Franceschelli and M. Musolesi. 2022. DeepCreativity: measuring creativity with deep learning techniques. *Intelligenza Artificiale* (2022).

[Guzik and Gilde(2023)] Byrge C. Guzik, E. and C. Gilde. 2023. The originality of machines: AI takes the Torrance Test. *Elsevier* (2023).

[Gómez-Rodríguez and Williams(2023)] C. Gómez-Rodríguez and P. Williams. 2023. A Confederacy of Models: a Comprehensive Evaluation of LLMs on Creative Writing. *arXiv* (2023).

[Hsueh(2024)] Felice M. Alaoui S. Mackay W. Hsueh, S. 2024. What Counts as 'Creative' Work? Articulating Four Epistemic Positions in Creativity-Oriented HCI Research. *CHI* (2024).

[Li and Yin(2024)] Liang C. Peng J. Li, Z. and M. Yin. 2024. The Value, Benefits, and Concerns of Generative AI-Powered Assistance in Writing. *CHI Conference on Human Factors in Computing Systems* (2024).

[Li(2019)] S. Li. 2019. Food.com Recipes and Interactions. (2019). https://doi.org/10.34740/KAGGLE/DSV/783630doi:10.34740/KAGGLE/DSV/783630

[Makin(2024)] A. Makin. 2024. Ontology-Driven Knowledge Management Systems Enhanced by Large Language Models. *Northeastern University* (2024).

[Marco(2025)] Rello L. Gonzalo J. Marco, G. 2025. Small Language Models can Outperform Humans in Short Creative Writing: A Study Comparing SLMs with Humans and LLMs. *arXiv* (2025).

[Nonaka(2007)] I. Nonaka. 2007. The Knowledge-Creating Company. *Harvard Business Review* (2007).

[Pandhare(2024)] H. Pandhare. 2024. Evaluating Large Language Models: Frameworks and Methodologies for AI/ML System Testing. *International Journal of Scientific Research and Management* (2024).

[Panickssery(2024)] Bowman S. Feng S. Panickssery, A. 2024. LLM Evaluators Recognize and Favor Their Own Generations. *Conference on Neural Information Processing Systems* (2024).

[Peeperkorn(2024)] Kouwenhoven T. Brown D. Jordanous A. Peeperkorn, M. 2024. Is Temperature the Creativity Parameter of Large Language Models? *ICCC* (2024).

[Ren(2023)] Zhao Y. Liu P. Lakshminarayanan B. Ren, J. 2023. Self-Evaluation Improves Selective Generation in Large Language Models. *Google Research* (2023).

[Renze(2024)] Guven E. Renze, M. 2024. The Effect of Sampling Temperature on Problem Solving in Large Language Models. *arXiv* (2024).

[Rhodes(1961)] M. Rhodes. 1961. An analysis of creativity. *Phi Beta Kappen* (1961).

[Romanovska and Balina(2024)] Birzniece I. Romanovska, M. and S. Balina. 2024. Contextualization of Information Objects Towards Supporting Knowledge Management in Digital Workspaces. *RTU* (2024).

[Romera-Paredes(2024)] Barekatain M. Novikov A. Balog M. Kumar M. Dupont E.-Ruiz F. Ellenberg J. Wang P. Fawzi O. Kohli P. Fawzi A. Romera-Paredes, B. 2024. Mathematical discoveries from program search with large language models. *Nature* (2024).

[Runco and Jaeger(2012)] M. A. Runco and G. J. Jaeger. 2012. The standard definition of creativity. Creativity Research Journal. *Elsevier* (2012).

[Sawyer(2021)] R. Sawyer. 2021. The iterative and improvisational nature of the creative process. *Elsevier* (2021).

[Sumbal and Tsui(2024)] Amber Q. Tariq A. Raziq M.M. Sumbal, M.S. and E. Tsui. 2024. Wind of change: how ChatGPT and big data can reshape the knowledge management paradigm? *Industrial Management Data Systems* (2024).

[Taylor(2024)] Ghose U. Rohanian O. Nouriborji M. Kormilitzin A. Clifton U. Nevado-Holgado A. Taylor, N. 2024. Efficiency at scale: Investigating the performance of diminutive language models in clinical tasks. *Elsevier* (2024).

[Wataoka(2025)] Takahashi T. Ri R. Wataoka, K. 2025. SELF-PREFERENCE BIAS IN LLM-AS-A-JUDGE. *arXiv* (2025).

[Wei(2023)] Wang X. Schuurmans D. Bosma M. Ichter B. Xia F. Chi-E. Le Q. Zhou D. Wei, J. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Google Research* (2023).

[Wu(2025)] Weber T. Müller F. Wu, Z. 2025. One Does Not Simply Meme Alone: Evaluating Co-Creativity Between LLMs and Humans in the Generation of Humor. *IUI* (2025).

[Zhang(2024)] Liu Z. Cherry C. Firat O. Zhang, B. 2024. When scaling meets LLM finetuning: The effect of data, model and finetuning method. *ICLR* (2024).

[Zhao and Chen(2024)] Zhang R. Li W. Huang D. Guo J. Peng S. Hao-Y. Wen Y Hu X Du Z. Guo Q. Li L. Zhao, Y. and Y. Chen. 2024. Assessing and Understanding Creativity in Large Language Models. *University of Chinese Academy of Sciences, Beijing, China.* (2024).

9 Appendix

9.1 Neural networks as evaluator

As part of our exploration into improving the FunSearch algorithm, we attempted to implement a neural network-based evaluator. The goal was to leverage the neural network's ability to learn complex patterns in recipe creativity and provide more nuanced evaluations. However, despite our efforts, this approach did not yield the expected results. The neural network failed to provide reliable or consistent assessments, leading us to reconsider its use within the FunSearch framework. As a result, we reverted to using pre-existing language models for evaluation.

9.1.1 Data

This study used the *Food.com* dataset [Li(2019)], which is publicly available on Kaggle. The dataset consists of two parts: a recipe dataset and an interactions dataset. These were merged using the common `Recipe ID`. After merging, the data was reduced to three key columns: `ingredients`, `steps`, and `ratings`, as these features were essential for both recipe generation and quality prediction.

The combined dataset contains 230186 distinct recipes, each associated with a user rating. However, the rating distribution is highly skewed. As shown in Table 3, ratings between 0 and 3 constitute only 10.17% of the data, while ratings between 4 and 5 account for 89.93%.

Rating	Number of Recipes
0	9,240
1	2,334
2	2,572
3	9,497
4	48,026
5	159,931

Table 3: Distribution of recipe ratings in the Food.com dataset.

To address this imbalance, several preprocessing techniques were tested: `MinMaxScaler`, `RobustScaler`, `Box-Cox Transformation`, `SMOTE`, `Logarithmic Transformation`, and `Quantile Transformation`. Among these, the `MinMaxScaler` yielded the best results by normalizing all ratings between 0 and 1 (via division by 5).
Additionally, the `ingredients` and `steps` columns were transformed into embeddings to be used as model features. Three types of embeddings were tested: `Vectorizer`, `BERT`, and `STELLA`. Their key properties are summarized in Table 4.

Embedding	Parameters	Max Input Length
Vectorizer	10,000	100
BERT	110,000,000	512
STELLA	435,000,000	512

Table 4: Comparison of embedding models used for feature extraction.

9.1.2 Experiments

Initial experiments focused on integrating a prediction model within the FunSearch algorithm to assess its effect on recipe generation. The prediction model performance was initially evaluated using a shallow model with BERT embeddings. However, this model predicted a constant rating of 4, indicating its inability to capture variations in recipe quality effectively. For the retention strategy, after each iteration of FunSearch, a subset of top-rated recipes, ranging from 8 to 100, was retained to guide the next round of optimization. In terms of prompt engineering, two different prompt styles were tested. The first, a simple prompt, instructed: "Keep the ingredients list under 10 items. Use essential, clear, and concise steps." The second, a more creative prompt, asked: "Design a recipe that must be served upside down and tastes better when eaten backwards." In both scenarios, the LLMs were able to generate coherent and occasionally imaginative recipes, demonstrating their adaptability to varied prompt constraints.

9.1.3 Model Creation

This section outlines the process used to build the best-performing prediction model for the Food.com dataset. Several configurations were systematically evaluated to optimize performance. In terms of embeddings, different methods were explored, including Vectorizer, BERT, and STELLA. For preprocessing, various normalization techniques were applied, as discussed earlier in the methodology. The model architecture was refined by adjusting the number of layers, ranging from 3 to 15, and the number of units per layer, which varied between 64 and 2048. The layer types considered included Dense, BatchNormalization, Dropout, ReLU, and LeakyReLU. Additionally, key learning parameters such as the learning rate, loss functions, and the choice between regression and classification were tested to find the most effective combination.
The final model selected was a deep regression neural network featuring residual connections, LeakyReLU activations, and dropout for regularization. This model was implemented using TensorFlow/Keras, as detailed in the following section.

```
def residual_block(x, units, l2_reg=1e-4):
    shortcut = x
    x = Dense(units, activation=None,
    kernel_regularizer=regularizers.l2(l2_reg))(x)
    x = LeakyReLU(alpha=0.3)(x)
    x = BatchNormalization()(x)
    x = Dense(units, kernel_regularizer=
    regularizers.l2(l2_reg))(x)
    x = Add()([x, shortcut])
    x = LeakyReLU(alpha=0.3)(x)
    return x

def build_model(input_shape):
    inputs = Input(shape=(input_shape,))
    x = Dense(512, activation=None,
    kernel_regularizer=regularizers.l2(1e-4))(inputs)
    x = LeakyReLU(alpha=0.3)(x)
    x = residual_block(x, 512)
    x = Dropout(0.5)(x)
    x = Dense(256, activation=None,
    kernel_regularizer=regularizers.l2(1e-4))(x)
    x = LeakyReLU(alpha=0.3)(x)
    x = BatchNormalization()(x)
    x = Dropout(0.4)(x)
    outputs = Dense(1, activation='linear')(x)
    model = Model(inputs, outputs)
    return model
```

Listing 1: Optimal neural network architecture for predicting recipe ratings using BERT and STELLA embeddings on the Food.com dataset

The model design incorporates several key features to enhance its performance. Residual connections are used to improve gradient flow and ensure training stability, particularly in deep networks. LeakyReLU activations are employed to prevent dead neurons, allowing for smoother training and improved learning dynamics. To mitigate overfitting, dropout layers with rates of 0.5 and 0.4 are included, helping to regularize the model. L2 regularization is applied to control the magnitudes of the weights, contributing to better generalization. Finally, the linear layer at the output enables regression on the normalized rating scale,

providing the model with the ability to predict continuous values.
Regression was selected over classification due to better alignment with the continuous rating scale. Model selection was based on three performance metrics:Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Validation Loss, summarized in Table 5.

Model	MAE	RMSE	Loss
BERT	0.3173	0.3703	0.3351
STELLA	0.3161	0.3509	0.3358

Table 5: Performance of top regression models on the creativity prediction task.

While lower error values (e.g., MAE = 0.15) were technically achievable, these models exhibited poor predictive spread, consistently outputting ratings between 4 and 5. For example, one such model had RMSE = 0.356 and Loss = 0.05 but lacked discriminatory power to identify lower-quality recipes.
While the final selected model struck a reasonable balance between predictive accuracy and generalizability, achieving relatively low error and modest rating differentiation, it ultimately proved inadequate for reliably evaluating recipe quality. Despite extensive tuning, the neural network could not capture the full nuance of recipe assessment, especially in edge cases or more creative outputs. As a result, we chose to shift away from relying on predictive models for evaluation and instead employed LLMs as evaluators, given their ability to reason over ingredient combinations, step clarity, and culinary plausibility.

9.2 Prompts Used in the Experiments

This appendix presents the prompts used in the different stages of the FunSearch algorithm pipeline. Each prompt is labeled according to its role in the process and formatted as it was presented to the language model, including placeholders and structural instructions.

9.2.1 Initial Recipe Generation Prompt

> You are a world-renowned, highly innovative chef competing in The Pillsbury Bake-Off.
> Your challenge is to **create a completely new and unique recipe**—something that has never been seen before!
> Your response must strictly follow the `JSON format` as outlined in the `<output>` section.
>
> `<instructions>`
> 1. **Invent a completely original, competition-worthy recipe** that follows the official contest rules.
> 2. **Tell the story behind the dish** (max 500 characters). Make it emotional, engaging, and authentic.
> 3. **List the ingredients** (max 10, excluding pantry staples). Include 1 Pillsbury™ products from: {`Pillsbury_ingredient_list`}.
> 4. **Provide step-by-step instructions** (max 2,000 characters). Must be prepped in 30 minutes or less (excluding cooking/baking time).
> 5. Recipes will be judged on:

(70%) Recipe Score — Taste, Appearance, Creativity, Crowd Appeal.
(30%) Story Score — Connection, Emotion, Passion.
6. **Required fields**: `Recipe Idea`, `Essay`, `Recipe Name`, `Ingredients`, `Instructions`.

Please provide your response strictly in the following JSON format, without any extra commentary.
IMPORTANT: Ensure your response **fully completes** the recipe and ends with }}.

`</instructions>`

`<output>`
{
"recipe_idea": `<your_recipe_idea>`,
"essay": `<your_essay>`,
"recipe_name": `<your_recipe_name>`,
"ingredients": `<your_recipe_ingredients>`,
"instructions": `<your_recipe_instructions>`
}
`</output>`

9.2.2 Evolve Recipe Prompt

You are a renowned chef participating in a high-profile international contest: The Pillsbury Bake-Off.
Please create a new recipe according to the instructions in the `<instructions>` tag and provide your response in the JSON format as specified in the `<output_format>` tag.

`<instructions>`
You should create and return a **better, more creative, and different version** of the following recipe and essay: {`previous_versions`}. The new recipe must surpass the previous one in creativity, originality, and presentation, while still strictly adhering to all contest rules outlined in the {`template`}.
Your answer must include, without exception, the following components:

- Recipe Idea
- Essay
- Recipe Name
- Ingredients (max 10, excluding pantry staples)
- Instructions (clear, concise, and within 2,000 characters)

For each component (Recipe Idea, Essay, Recipe Name, Ingredients, Instructions), you must provide your response immediately following the qualitative assessment.

Please provide your response strictly in the following JSON format, without any extra commentary.
IMPORTANT: Ensure your response **fully completes** the recipe and ends with }}.

</output_format>
</instructions>

<output>
{
"recipe_idea": <your_recipe_idea>,
"essay": <your_essay>,
"recipe_name": <your_recipe_name>,
"ingredients": <your_recipe_ingredients>,
"instructions": <your_recipe_instructions>
}
</output>

9.2.3 B.3 Evaluator Pillsbury Bake-Off Prompt

You are a highly critical judge in the Pillsbury Bake-Off, and your standards are exceptionally high. Your role is to rate recipes with great scrutiny, focusing on both the technical and emotional aspects. Please analyze the following recipe in the <recipe> tag according to the instructions in the <instructions> tag and provide your response in the JSON format specified in the <output_format> tag.

<recipe>
{recipe}
</recipe>

<instructions>
For each of the following dimensions, **first provide a detailed qualitative assessment** followed by a score (1 to 5). **It is mandatory that you provide a score immediately after each qualitative assessment for all dimensions.** You **must** provide a score for each dimension, even if the qualitative assessment is brief.

Recipe Judging:

1. **Taste**: Evaluate how well-balanced and pleasing the flavors are in the dish, considering aspects like seasoning, texture, and overall flavor profile. (1 low - 5 high)
2. **Appearance**: Rate how visually appealing the dish is, considering factors like color contrast, plating, and presentation. (1 low - 5 high)
3. **Creativity**: Assess the innovation and originality of the recipe, considering ingredient combinations, cooking techniques, and presentation. (1 low - 5 high)
4. **Crowd Appeal**: Determine how likely the dish is to be enjoyed by a wide range of people, considering its familiarity, comfort, and versatility. (1 low - 5 high)

Story Judging:

1. **How the recipe ties to the story**: Does the recipe reflect the story behind it, making the dish feel authentic to the narrative? (1 low - 5 high)

2. **How the story brings to life a family value, tradition, or memory**: Does the story evoke emotions tied to family or tradition, adding depth to the recipe? (1 low - 5 high)
3. **Demonstration of Passion**: Does the story showcase a deep, genuine emotional connection to the recipe? Does it convey the chef's personal love for the dish, the culinary tradition, or cooking in general? **This score is absolutely crucial** and must be clearly articulated in your assessment. (1 low - 5 high)

Overall score:

1. **How would you rate the overall recipe with respect to all these dimensions?** (1 low - 5 high)

Important Notes:

- **You MUST provide scores immediately after each quantitative assessment: taste, appearance, creativity, crowd_appeal, recipe_ties_story, story_brings_to_life, passion and overall. You must rate all these assessments.**
- Provide your response **only** in this strict JSON format, without any extra commentary.

</instructions>

<output_format>
{
"taste_quality_assess": <your_assessment>,
"taste": <score between 1-5>,
"appearance_quality_assess": <your_assessment>,
"appearance": <score between 1-5>,
"creativity_quality_assess": <your_assessment>,
"creativity": <score between 1-5>,
"crowd_appeal_quality_assess": <your_assessment>,
"crowd_appeal": <score between 1-5>,
"recipe_ties_story_quality_assess": <your_assessment>,
"recipe_ties_story": <score between 1-5>,
"story_brings_to_life_quality_assess": <your_assessment>,
"story_brings_to_life": <score between 1-5>,
"passion_quality_assess": <your_assessment>,
"passion": <score between 1-5>,
"overall_quality_assess": <your_assessment>,
"overall": <score between 1-5> }
</output_format>

9.2.4 Evaluator TTCT Prompt

You are an expert in evaluating recipe creativity based on the Torrance Tests of Creative Thinking (TTCT). Your task is to assess the following recipe according to the four dimensions of creativity: Fluency, Flexibility, Elaboration, and Originality.

Recipe: 'recipe'

Please evaluate this recipe based on the following dimensions:

1. **Fluency** – Does the recipe contain multiple distinct creative elements, such as innovative ingredient combinations or unique preparation techniques? (Score: 1 = low, 5 = high)
2. **Flexibility** – Does the recipe showcase versatility in ingredient use, cooking methods, or cultural fusion? (Score: 1 = low, 5 = high)
3. **Elaboration** – How well does the recipe provide depth, explanation, and clarity in its preparation steps and ingredient choices? (Score: 1 = low, 5 = high)
4. **Originality** – How unique is this recipe compared to traditional versions? Does it introduce new concepts, techniques, or ingredient uses? (Score: 1 = low, 5 = high)

For each dimension, please:

- First, provide a **detailed qualitative assessment** of the recipe's creativity.
- Immediately **assign a score** from 1 to 5 (1 being low creativity, 5 being high creativity) for each dimension.
- The **score must be included immediately after the qualitative assessment**.

Your response **must follow this exact JSON format below** with no additional explanations, comments, or text. Do not include any other tags like `</instructions>` or `</output_format>`. The response should be **only** the JSON object.

```
<output_format>
{
"fluency_quality_assess": "Your qualitative assessment for Fluency",
"fluency": "Score between 1-5",
"flexibility_quality_assess": "Your qualitative assessment for Flexibility",
"flexibility": "Score between 1-5",
"elaboration_quality_assess": "Your qualitative assessment for Elaboration",
"elaboration": "Score between 1-5",
"originality_quality_assess": "Your qualitative assessment for Originality",
"originality": "Score between 1-5"
}
</output_format>
```

9.3 Example Recipe Generated by an LLM and winner Pillsbury

9.3.1 Pillsbury Bake-off winning Recipe 2024

The winning recipe from the 2024 Pillsbury Bake-Off achieved a creativity score of 4.063, as evaluated by the four LLMs. This score reflects a high level of originality.
Ingredients:

- 3/4 lb beef tenderloin, cut into 24 (1-inch) pieces

- 1/2 teaspoon salt
- 1/4 teaspoon pepper
- 1 tablespoon olive oil
- 5 oz (1 1/4 cups) frozen spinach, thawed, squeezed to drain
- 2 tablespoons sour cream
- 1 tablespoon dry onion soup mix (from 1-oz package)
- 2 tablespoons saltine cracker crumbs (about three 2x2-inch squares)
- 1 can (8 oz) refrigerated Pillsbury™ Original Crescent Dough Sheet
- 1 egg, slightly beaten
- 4 oz smoked gouda cheese, shredded (1 cup)
- 1 teaspoon dry onion soup mix (from 1-oz package)
- 3/4 cup sour cream
- 2 1/2 teaspoons olive oil
- Dash pepper
- 6 (4-inch) thyme sprigs

Instructions:

1. Heat oven to 375°F. Line 18x13-inch sheet pan with cooking parchment paper.
2. Season beef with salt and pepper. In 10-inch nonstick skillet, heat 1 tablespoon oil over high heat. Sear beef in oil 1 minute, turning to brown sides. Remove beef from skillet; set aside. Save skillet and drippings for dipping sauce.
3. In small bowl, mix spinach, 2 tablespoons sour cream, 1 tablespoon onion soup mix, and the cracker crumbs; set aside.
4. Unroll dough sheet on work surface. Press or roll into 13 1/2x9-inch rectangle. Using pizza cutter or sharp knife, cut dough lengthwise into 4 rows, then cut crosswise into 6 rows to make 24 pieces.
5. To form each wellington, place 1 teaspoon of the spinach mixture onto center of each dough piece. Top with 1 piece of beef; press in lightly. Pull up 2 opposite corners of dough stretching slightly; pinch together. Repeat with 2 remaining corners, pressing all seams to seal. Place on sheet pan. Repeat with remaining ingredients. Brush wellingtons lightly with egg.
6. Bake 17 to 19 minutes or until golden brown.

7. Meanwhile, heat skillet with drippings over medium-low heat. Cook cheese and 1 teaspoon onion soup mix in drippings 2 to 3 minutes, stirring occasionally, until cheese is melted. Stir in 3/4 cup sour cream and 2 1/2 teaspoons oil. Continue cooking 2 minutes, stirring frequently, until mixture is well blended. Remove from heat; place in small serving bowl. Sprinkle with dash of pepper. Serve appetizers and bowl of dipping sauce on large platter. Garnish with thyme.

9.3.2 Example Recipe Generated by an LLM with Story

The following recipe was generated using the $Meta{-}17$ model, which served both as the generator and evaluator. It was produced through the FunSearch algorithm over five iterations. In each iteration, FunSearch generated seven recipes, and the best one was selected based on evaluation scores. This process was repeated 30 times in total.
The recipe shown below is entry 22 out of the 30 generated recipes. It received an overall score of 4.65 out of 5, which is considered high within the context of the Pillsbury Bake-Off. In terms of creativity, it achieved a score of 4.438, indicating an exceptionally high level of originality.
Ingredients:

- 1 package Pillsbury Original Crescent Rolls
- 1 cup cherry tomatoes
- 2 tablespoons Yuzu juice
- 1/4 cup crispy prosciutto
- 1/2 cup heavy cream
- 1/2 teaspoon matcha powder
- 1/4 cup grated parmesan cheese
- 1/4 teaspoon salt
- 1 tablespoon olive oil
- 1 clove garlic

Instructions:

1. Preheat oven to 375°F. Bake Pillsbury Crescent Rolls according to package instructions (8-10 minutes).
2. While rolls are baking, mix Yuzu juice, cherry tomatoes, garlic, and olive oil in a small saucepan. Cook over medium heat, stirring constantly, until tomatoes are caramelized and mixture is syrupy.
3. In a separate bowl, whip heavy cream with matcha powder until stiff peaks form. Fold in parmesan cheese.
4. Once rolls are done, let them cool for 2 minutes. Then, slice each roll in half horizontally.

5. Top with Yuzu-infused tomato sauce, crispy prosciutto, and a dollop of matcha-parmesan cream.

6. Serve immediately and enjoy!

Story:
Growing up in a multicultural city, I was inspired by the vibrant flavors and traditions of Japanese and Italian cuisine. This recipe combines the flaky, buttery texture of Pillsbury Crescent Rolls with the spicy, citrusy essence of Yuzu and the bright, grassy flavor of matcha, creating a unique and captivating treat that celebrates my love for international cuisine.

9.4 Experiment: Effect of Batch Size, Generator Temperature, and Island Count

9.4.1 Research Design & Methodology

One of the experiments investigated how three key parameters, batch size, generator temperature, and island count, influence the generator's performance in producing creative recipes. It is important to note that this experiment was conducted when the FunSearch implementation was not fully optimized. As a result, rather than selecting the best recipe after a set number of iterations, the output consisted of the top 30 recipes from the program's database.
Batch size refers to the number of recipes generated by each island per iteration. We tested batch sizes of 5, 15, and 25 to observe whether a larger solution pool leads to more diverse and higher-quality outputs.
Generator temperature controls the randomness of the LLM's output. Lower temperatures (e.g., 0.2) tend to produce more deterministic outputs, while higher temperatures (e.g., 0.8) promote exploration and creativity. We tested temperatures of 0.2, 0.5, and 0.8.
Island count determines how many independent solution populations are explored in parallel. This allows for variation between isolated solution paths. We evaluated 5, 7, and 10 island configurations.
For each parameter combination, outputs were evaluated using the creativity scoring framework described above (i.e., four external LLMs rating recipes based on TTCT-style metrics). Performance was measured by comparing creativity scores to the benchmark dataset.

9.4.2 Findings & Recommendations

This experiment investigated the effectiveness of the FunSearch algorithm in generating original recipes, using creativity as the primary evaluation criterion. The resulting recipes were compared against entries from the Pillsbury Bake-Off competition. The Pillsbury dataset is represented by the red bar in Figure 4, with the winner of the competition indicated by a red cross. This winner is one of 30 participants in the sample, each of whom submitted a recipe evaluated by both human judges and four LLMs based on creativity metrics.
In the figure, the bar height represents the mean creativity score of the 30 Pillsbury entries, while the black cross indicates one standard deviation from the mean. The winning recipe

clearly surpasses this threshold, suggesting that it is not only favored by human judges but also stands out in terms of creativity as measured by the TTCT framework.
To assess FunSearch, we generated recipes under different parameter settings by varying generator temperature, batch size, and island count. These generated datasets were then compared to the Pillsbury benchmark. Interestingly, the average creativity scores of most FunSearch configurations exceed that of the Pillsbury winner, suggesting that the algorithm consistently outperforms human participants in terms of creative output.
However, no consistent or linear relationship was observed between creativity and any of the three parameters (temperature, batch size, island size). This indicates that increasing these values does not directly translate to more creative outputs. The top three performing configurations are reported in Table 6. Given the minimal differences between their average scores, we selected a setting of generator temperature = 0.8, batch size = 5, and island size = 7 for further experiments. This choice was made for computational efficiency, as smaller batch sizes significantly reduce the required computation without compromising performance.

Parameter Setting	**Mean**	**St. Dev.**
Gen 0.2, Batch 25, Island 5	4.50	0.20
Gen 0.2, Batch 15, Island 10	4.47	0.18
Gen 0.8, Batch 5, Island 7	4.47	0.27

Table 6: Mean and standard deviation of creativity scores for the top three performing FunSearch parameter configurations. *Gen* refers to generator temperature, *Batch* to batch size, and *Island* to island count.

9.5 Codified and Tacit Knowledge in LLMs

The performance of LLMs in creative or domain-specific tasks depends heavily on their ability to handle both codified and tacit knowledge. Codified knowledge refers to formalized, explicit information, such as scientific facts or structured manuals. Tacit knowledge, in contrast, involves context-specific, intuitive, and experiential understanding, often difficult to articulate [Adler(1996)].
Modern AI systems are predominantly trained on codified, general-purpose knowledge, with limited exposure to industrial or organizational knowledge due to privacy and access limitations [Makin(2024)]. This poses challenges in domains like cooking, where tacit knowledge is crucial, such as understanding how ingredients interact or how cultural preferences influence flavor profiles.
The SECI model, developed by Nonaka and Takeuchi [Nonaka(2007)], offers a useful lens for understanding how tacit and explicit knowledge transform within organizations. It includes four stages: *socialization* (tacit to tacit), *externalization* (tacit to explicit), *combination* (explicit to explicit), and *internalization* (explicit to tacit). LLMs, through exposure and generation, may mimic parts of this knowledge transformation cycle but often struggle with genuine tacit insight.
Recent research explores how LLMs can be leveraged to enhance organizational knowledge processes. For example, contextualizing information objects with meta-models and LLMs has been shown to improve digital workplace efficiency [Romanovska and Balina(2024)].

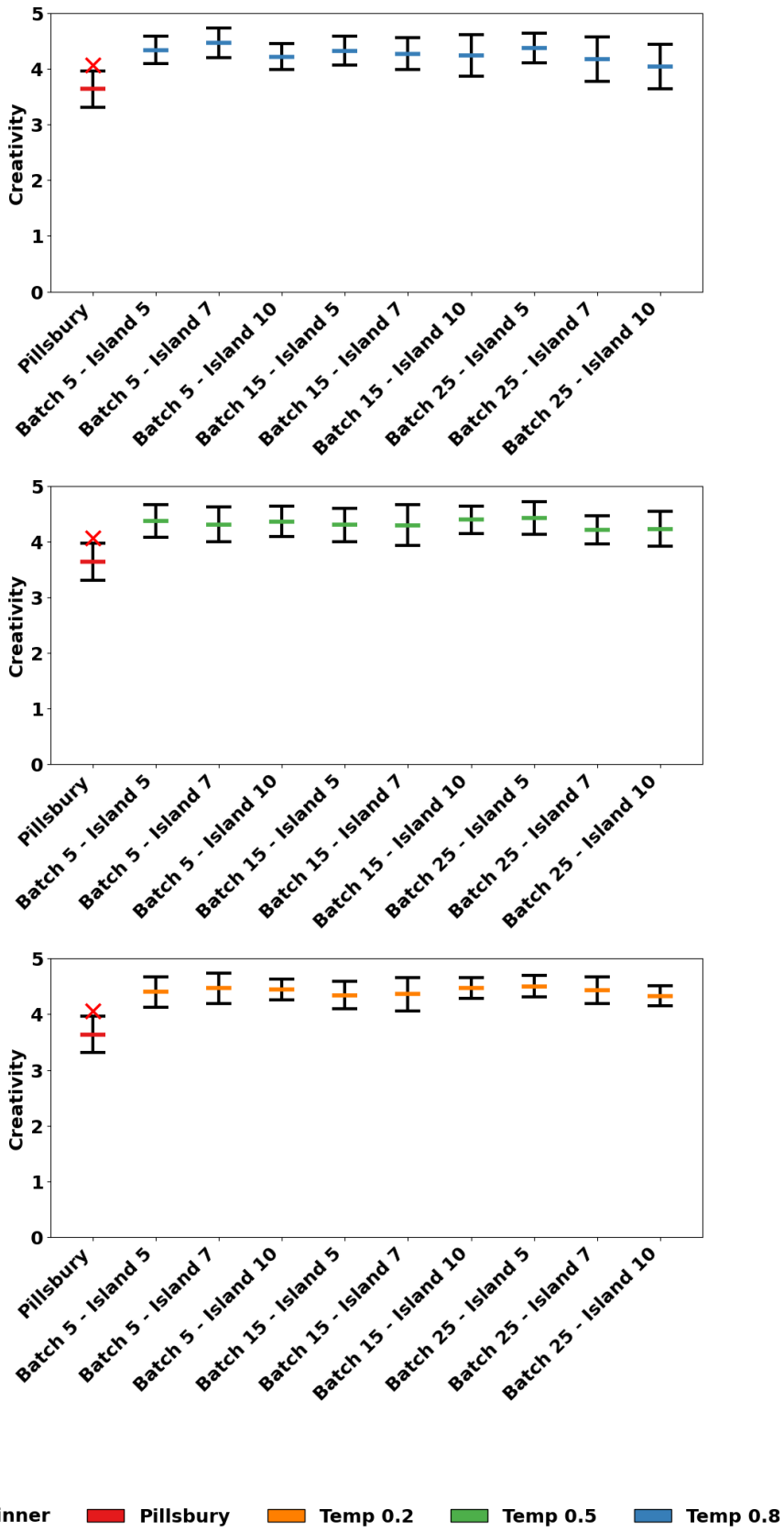

Figure 4: Performance of the FunSearch algorithm under varying generator temperature (Temp), batch size (Batch), and island size (Island), measured by comparing creativity scores assigned by four LLM judges to a benchmark derived from the Pillsbury dataset. The red bar indicates the Pillsbury dataset mean; the red cross shows the winning entry's score.

Similarly, LLMs have been applied in planning tasks, such as emergency response, showcasing their abilities in reasoning, adaptation, and knowledge retrieval, although they are known to have limitations, such as hallucinations [Durmus and Isaac(2024)].
Tools like ChatGPT are now being explored for their potential to support real-time decision-making and collaborative knowledge sharing in organizations
[Sumbal and Tsui(2024)]. While promising, these tools require careful calibration to account for limitations in domain-specific understanding, especially where tacit knowledge is essential.

9.6 Code Repository

The code used to generate and evaluate recipes for this research is available in the following public repository:
https://github.com/RensAnderson/llm-creativity-thesis
This repository includes all scripts for recipe generation, creativity evaluation, and model configurations. The code is designed to support reproducibility of the experiments and can be used to replicate the study or explore different model settings.