Aligning Agents with Experts in Material Selection: Prompting, Size, and Reasoning

Megan Y. Ying
Carnegie Mellon University,
Pittsburgh, PA USA
email: mying@alumni.cmu.edu

Daniele Grandi
Autodesk Research
San Francisco, CA USA
email: daniele.grandi@autodesk.com

Allin Groom
Autodesk Research
London, UK
email: allin.groom@autodesk.com

Christopher McComb[1]
Carnegie Mellon University,
Pittsburgh, PA 15213 USA
email: ccm@cmu.edu

Material selection is fundamental to the design process, as it significantly affects the cost, performance, appearance, manufacturability, and sustainability of a product. It is also a complex, open-ended challenge that forces designers to continuously adapt to new information, balance diverse stakeholder demands, weigh trade-offs, and navigate uncertainties to achieve the optimal outcome. Previous studies have explored the potential of large language models (LLMs) to assist in the material selection process, with findings suggesting that LLMs have the potential to provide valuable support. However, discrepancies between LLM outputs and expert recommendations indicate the need for further investigation. Recently agentic AI methods have been developed to address the limitations of standalone LLMs. AI agents integrate LLMs with external tools, allowing them to retrieve and analyze domain-specific information and iteratively refine responses. Other efforts to enhance LLM performance include the development of reasoning models that implicitly incorporate multi-step thinking processes for approaching complex tasks. This study compares standalone LLMs and agentic AI frameworks, both enhanced with a reasoning process, to evaluate how different approaches contribute to more effective emulation of expert decision-making in material selection. Our findings show that adding reasoning to LLMs increases token usage but may reduce performance differences across model sizes and prompting methods. We observed improved alignment to expert results with parallel prompting compared to previous non-thinking models, while the agentic framework showed reduced performance. These insights contribute to a broader understanding of AI integration in design workflows.

Keywords: Large Language Models (LLMs), Artificial Intelligence (AI), LLM-based Agents, Material Selection, Engineering Design

1 Introduction

Material selection is integral to the development cycle of engineering projects, influencing many aspects of the final product. The intrinsic properties of a material determine how it behaves in response to conditions such as stresses, temperatures, and environmental factors, impacting a design's performance and life cycle [1]. Mechanical and physical properties also affect compatibility with manufacturing processes, influencing the feasibility of achieving the desired geometries [2]. Incompatible material-process pairings can increase costs and reduce performance, while effectively optimized geometry can improve functionality, reduce material waste, and lower costs [3,4].

Beyond functionality, a material's aesthetic properties, such as color, texture, and finish, shape user perception [5]. Overall product cost is influenced not only by raw material expenses, but also by the complexity and number of manufacturing processes required [6]. Sustainability concerns are also shaping material choices as industries balance environmental impact, resource depletion, and consumer demand for eco-friendly solutions [4,7,8]. Ultimately, the interplay between material properties, performance, aesthetics, cost, sustainability, and manufacturing makes material selection a complex task.

The design process is often iterative, where initial design intent or product requirements guide material selection, which in turn refine those requirements [9]. With multiple trade-offs and stakeholder negotiations, this iterative process can become time and resource-intensive [10]. While databases and frameworks support material selection, critical insights often lie beyond datasheets, requiring expert knowledge and contextual information from suppliers, case studies, and in-house reports [11].

Advancements in artificial intelligence (AI) and natural language processing (NLP), particularly with the emergence of large language models (LLMs) trained on extensive textual datasets, have significantly enhanced the ability of computers to process and generate meaningful text [12]. Enhanced computational power and the growing availability of datasets in recent years have driven the widespread adoption of LLMs across diverse fields, with one potential application being their use as tools to recommend materials that align with the design intent [13].

While LLMs are capable of generating text based on input prompts, they struggle with complex reasoning and tasks in domains such as logic, spatial intelligence, and linguistic understanding [14,15]. Additionally, previous research has identified a specific challenge in material selection, where LLMs exhibit less variability in their responses for newer materials like composites, and offer less consistent scores for well-established materials like steel. This pattern suggests that older materials may be overrepresented in model outputs due to their historical prevalence in literature rather than their relevance to modern design [16].

AI agents are one approach to overcoming some LLM limitations by incorporating additional tools for perception and interaction with environments, enabling autonomy, planning, problem-solving, and decision-making [17,18]. In some agentic AI frameworks, the LLM acts as the core reasoning engine, while a percep-

[1]Corresponding Author.
Version 1.18, October 9, 2026

Copyright © 2026 by ASME

tion module interprets external information and an action module executes tasks [19]. Together, these modules enable agentic AI to go beyond passive text generation, allowing agents to perceive task instructions, create step-by-step plans, interact with external systems, and learn from feedback to improve performance [20].

Previous studies have collected material recommendations from human experts, LLMs, and LLM-based agents across various design cases and criteria to assess how closely AI-generated suggestions aligned with expert choices [16,21,22]. While AI was capable of generating material suggestions supported by relevant information, many did not closely align with expert recommendations. Researchers found that adjusting factors like model size, model type, prompting strategy, and parameter settings improved alignment with expert recommendations, highlighting the need for further research into replicating human decision-making [16,22].

Another recent approach to enhancing AI capabilities involves developing large reasoning models that build upon LLMs through test-time inference scaling techniques and post-training methods [23]. These methods encourage models to "think" more deeply by explicitly incorporating multi-step intermediate reasoning processes during problem solving [24].

This paper seeks to build on prior research [16,21,22] by exploring whether augmenting LLMs with search tools or reasoning capabilities can enhance their ability to emulate the decisions of human experts in material selection. Specifically, this study examines the impact of providing an LLM-driven agent with access to Wikipedia, alongside the effect of a thinking mode. We conduct a series of experiments to evaluate the role of model size and prompting techniques in aligning AI recommendations with expert decision-making.

In contrast to our prior study [22], which evaluated Qwen 2.5 models without explicit reasoning capabilities, the present work expands the experimental design to include reasoning-capable Qwen 3 models with thinking mode enabled. This allows us to systematically isolate the effects of reasoning, prompting strategy, and model size on expert alignment, and to directly assess how introducing multi-step reasoning alters both performance and efficiency relative to the earlier agentic and standalone LLM results. Moreover, to further measure the explanatory power of the models, we report R-squared values to describe model fit.

This paper seeks to address the following research questions:

(1) How does the use of an agentic AI framework and a reasoning process influence alignment with expert material selection behavior?
(2) How does the agent's behavior vary with respect to model size?
(3) To what extent does model size and prompting method impact the ability of LLMs to select appropriate materials?

2 Background

In this section, we review related works on the challenges of material selection, AI evaluation, and the use of AI in material selection.

2.1 Challenges in Material Selection. Material selection is crucial in product development, influencing the entire lifecycle beyond the early design stages [25,26]. With thousands of materials and processes available, filtering out unsuitable options early requires expertise. Just as material properties can be modified through various processes, impacting both performance and cost, design constraints can also be adjusted, changing the suitability thresholds of material candidates [13]. The tension between design requirements and materials necessitates a robust but flexible selection process. Researchers have developed various frameworks for material selection, all emphasizing screening and ranking as essential steps to refine material options [27].

Sustainability concerns are increasingly shaping material choices as industries respond to climate change, resource depletion, and pollution. Companies are financially incentivized to adopt sustainable materials and manufacturing methods to meet consumer demand, comply with environmental regulations, and reduce costs through more efficient processes [7]. Additionally, material shortages are making traditional materials less accessible, prompting engineers to explore alternative, less-developed options [8]. This shift affects engineering decisions by prioritizing durability, recyclability, and production efficiency [4].

The Ashby chart, which compares materials by two properties, is widely used but may be limited for designs with multiple constraints [9,27]. Other methods, such as the questionnaire method [28], case-based reasoning [29], multi-criteria decision analysis [30], and weighted properties [28], rely on measurable attributes, expert input, or prior applications. While quantitative analysis identifies material solutions quickly, it requires expertise in data analysis. Expert knowledge ensures accuracy but can limit innovation, while case-based selection depends on the quality of prior examples and may struggle with novel problems [11].

At the beginning of the design process, material selection focuses on defining core requirements and exploring materials that align with the design intent, considering factors like mechanical, physical, and manufacturing properties [31]. Trade-offs often arise, as materials may meet some but not all requirements [26]. AI is increasingly aiding material discovery by analyzing large datasets, predicting properties, and suggesting optimal combinations [9,14,32]. However, due to uncertain product requirements, designers often default to familiar or reverse-engineered materials, which may not always be optimal [32].

As the design progresses, product requirements become more refined, with broad limitations yielding too many material options and overly restrictive ones in too few. Finalizing constraints makes trade-offs more apparent, emphasizing the need for testing, simulation, and prototyping to validate material choices and address unexpected issues. Engineers must also consider manufacturing processes and system components as designs near fabrication [26]. Due to its multidisciplinary nature, material selection is often treated as an optimization problem, with designers iterating on constraints to identify the best material [10]. Advanced computational tools, such as neural networks, assist by analyzing material properties and design parameters to predict product performance under varying conditions [9,33].

2.2 Reasoning-Capable Models. LLMs have been trained on large datasets to generate text in response to a wide variety of prompts, excelling at System 1 tasks that can be answered quickly and automatically, such as natural language processing and question answering [24,34]. However, they often struggle with System 2 tasks, such as advanced mathematics, code comprehension, and logical reasoning, that require breaking problems down into smaller steps [34–36]. To address this, researchers have developed reasoning-capable models that generate sequences of intermediate steps, or chains of thought, which have been shown to substantially improve LLM performance across a wide range of complex tasks [35,37]. Approaches to developing this capability include incentivizing reasoning through reinforcement learning, incorporating step-by-step derivations into training data, and scaling test-time computation [23,24,38,39].

Reasoning models often produce excessively long and repetitive chains of thought, a phenomenon known as overthinking that arises when internal biases lead to premature guesses that conflict with subsequent reasoning steps [36,40]. This behavior significantly increases token usage, computational overhead, and processing time without consistently improving accuracy [36,39,41]. These models also tend to rely on larger architectures, further raising resource demands [39]. To improve efficiency, researchers are exploring strategies such as developing more concise models, shortening reasoning steps and outputs, using latent thinking tokens, and applying prompt engineering techniques [36,41–46]. Additionally, companies have developed models capable of switching between

fast, non-thinking and slow, thinking modes, allowing for adaptive resource allocation [47,48].

Although extended chains of thought can improve performance in some scenarios, other studies suggest that bypassing thinking mode may produce equally or more accurate results on certain tasks [47,49]. This may be because large reasoning models often overestimate the value of thinking tokens, and reliance on previous thinking tokens can trigger cascading generations that lead to unproductive reasoning loops [50]. Additional work indicates that although extended thinking initially enhances accuracy, performance can degrade beyond a certain point due to overthinking and error accumulation [36,51]. Ultimately, further research on reasoning-capable models is needed to identify optimal strategies that balance efficiency and accuracy.

2.3 Evaluating LLMs and Agentic AI. The emergent behaviors of LLMs, and their wide applicability to various domains, has led researchers to establish numerous benchmarks to evaluate LLM reasoning abilities in many different fields. Moreover, the rapid development of LLMs also demanded an equivalent development of new evaluation methods.

Starting with static test sets such as GLUE [52] and SuperGLUE [53], researchers further developed multi-domain and open-domain benchmarks for automatic evaluation of general language understanding [54–60]. Text-based question-answer benchmarks, or reference-dependent benchmarks, require the model to interpret additional information from a selected corpus, thus allowing for the evaluation of general knowledge, factoid-type questions, questions based on long-text documents, or from multiple sources in the model's pre-trained corpus [61–65]. Multimodal benchmarks and visual question-answer benchmarks typically evaluate a model's capability to answer questions about a specific image, focusing primarily on reasoning and perception abilities [66–70]. Despite much development around automatic evaluation, manual evaluation remains a common and trusted method, as it can provide more comprehensive and accurate feedback on model performance [56,71–74].

In AI research, the term *agent* originally referred to an artificial entity capable of perceiving its surroundings through sensors, making decisions, and subsequently taking actions in response [75,76]. Emergent LLM behaviors have further enhanced the capabilities of AI agents to plan and reason, and coupled with tool usage, modern AI agents can tackle very complex tasks [19,77–83]. Currently, the utility of LLM-powered agents is evaluated primarily by the success rate of task completion, as agents are typically implemented as human assistants [84,85]. This metric evaluates whether agents achieve predetermined objectives or expected scores [86–89]. Researchers have also conducted more detailed assessments of agents' foundational capabilities, such as environmental comprehension, reasoning, planning, tool usage, and decision-making [81,90–92]. Furthermore, the efficiency of the agent in completing the task has also been used as an evaluation criterion, as the agent should complete the task within an appropriate timeframe and with minimal resources [86]. In this work, we build on prior research that looked at material selection and evaluate the performance of an AI agent on the same task.

2.4 Automating Material Selection with AI. Over the years, the material science field has kept evolving to meet the demand to characterize and develop novel materials, such as composites, and engineer entirely new ones with unique property combinations, such as metamaterials [93,94]. Recent advances in NLP have led the field towards the use of LLMs for material discovery [95], experimental solid mechanics [96], predictive material science [97], biologically inspired materials design [98], and various other applications in the material science field [99].

Understanding a material's mechanics plays an important role in determining the feasibility of the material for a design problem, however, as discussed in the previous section, many other factors play a role in material selection for product design. To this end, a few review papers have evaluated LLM capabilities against mechanical engineering design tasks, including material selection and other manufacturing-related tasks [100,101]. LLMs have been applied to forward and inverse material design problems [102] and knowledge retrieval from a material failure mechanics corpus [103]. Buehler and colleagues expanded the approach by leveraging multi-agent systems to solve mechanics problems and automate alloy design [104,105].

Despite recent progress, automating material selection with AI remains difficult because the task is inherently contextual, multi-objective and influenced by experience/domain knowledge [9–11,21,27]. Candidate suitability depends not only on nominal properties, but also on processing route, joining methods, geometry, service environment (e.g., corrosion, fatigue, creep), and organizational constraints such as cost targets, supplier availability, and standards compliance [2,6,11,21,27] Much of this information is distributed across heterogeneous sources (handbooks, datasheets, test reports, and internal lessons learned) and may be incomplete or contradictory [11]. LLM-based approaches also introduce risks that are especially salient for engineering decisions, including hallucinated properties/mechanisms and overconfident recommendations without traceable evidence [14,15,106]. Consequently, AI systems are best positioned as decision-support tools for early-stage screening and ranking. Thus, helping designers explore trade-offs and narrow the option space. Rather than fully autonomous selectors without human verification.

In this paper, we bridge recent work on agentic AI frameworks with the material selection task in product design, improving on prior work and reducing the bias that LLMs exhibit on this task [16].

3 Methods

In this section, we describe the methodology used for our study. Our approach involves using a previously collected dataset of expert material selection decisions, developing a tool-augmented LLM-based agent, testing prompting strategies across model sizes, and conducting analyses to assess performance and behavior.

3.1 Expert Data Collection. To address our research questions, we used the MSEval[2] dataset collected by Jain *et al.* [21]. MSEval was obtained through a survey conducted among material selection professionals, including materials scientists, materials engineers, design engineers, and others in related fields, via the Autodesk Research Community.

The experts were asked to assess a set of nine materials (*steel, aluminum, titanium, glass, wood, thermoplastic, elastomer, thermoset, and composite*) based on their suitability for sixteen different scenarios, assigning a score between 0 and 10, with 0 representing an unsatisfactory choice and 10 representing an excellent choice. These scenarios included all possible combinations of four design applications (*kitchen utensil grip, safety helmet, spacecraft component, and underwater component*) and four design criteria (*lightweight, heat resistant, corrosion resistant, and high strength*).

The selection of these applications, criteria, and materials aimed to cover a wide range of use cases and maximize the likelihood of identifying both suitable and unsuitable design options. Only responses from participants who indicated experience in material selection were retained. Although not all participants answered every question, all responses were included, resulting in a data set collected from 138 material selection experts.

3.2 Design of LLM-Based Agent. To build our LLM-based agent, we used the Hugging Face Transformers library. To align with expert decision-making, we designed the agent to evaluate multiple factors critically and solve problems effectively. We chose

[2] https://github.com/grndnl/llm_material_selection_jcise

the ReAct framework for its step-by-step approach, where reasoning and actions dynamically refine plans, enhancing accuracy and reducing hallucinations [107]. Furthermore, research suggests that LLM-based agents are better equipped to manage complex tasks and interact with external systems when using executable code [17]. Therefore, we chose an agent that generates Python code instead of JSON for tool calls.

To initialize a Hugging Face agent, key parameters must be defined, including selecting an LLM [108]. We chose Qwen 3 for its strong performance and range of model sizes, measured by the number of parameters or adjustable values that determine how the model understands and generates language. This range allows us to examine the impact of model size on replicating expert decision-making. The Qwen 3 models were trained on a dataset of 36 trillion tokens, the smallest units of text that an LLM can process, doubling the 18 trillion used in the Qwen 2.5 series. Additionally, unlike earlier model families, the Qwen 3 series are able to switch between a multi-step thinking mode for complex problems and a fast non-thinking mode for simpler question, enabling us to examine how activating thinking mode affects the results. [48]. As a result of these changes, Qwen 3 models, across both thinking and non-thinking modes, match the performance of larger Qwen 2.5 models and even surpass them in STEM, coding, and reasoning benchmarks [48].

Furthermore, the agent requires a toolbox from which it can select tools to use. While the Hugging Face Transformers library includes default tools, we determined that our agent needed internet access to search for material characteristics, identify comparable designs, and retrieve real-time data. We developed a custom tool class that utilizes the Wikipedia API to perform searches and retrieve summaries of the top results [108].

Finally, the agent must be initialized with a system prompt that defines its input, expected behavior, available tools, and output format [108]. Clear instructions and examples within the prompt are crucial for achieving accurate results [109]. We customized the default system prompt to include examples of using the Wikipedia search tool to evaluate material effectiveness in various design cases, avoiding survey data to prevent bias. An example of these different components interacting with each other in the agentic framework is shown in Figure 1.

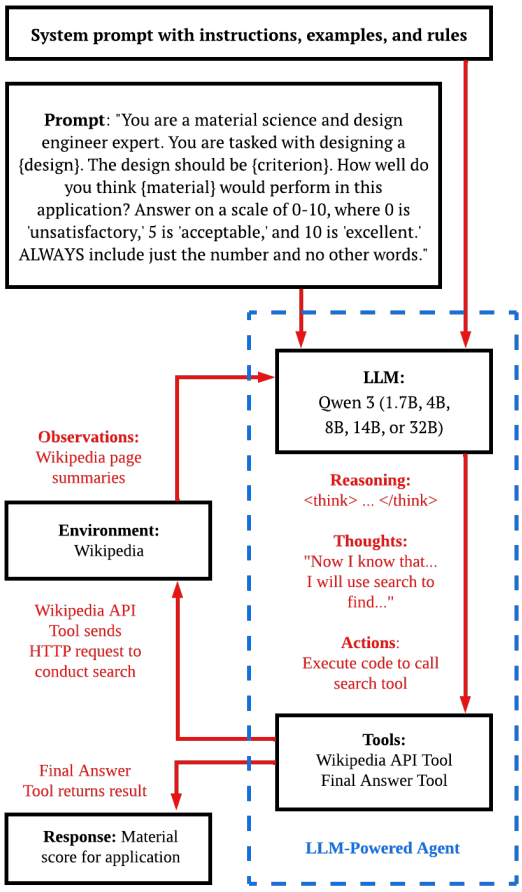


Fig. 1 A visualization depicting the dynamic workflow of the agentic AI used in this work, illustrating how the prompt, LLM, tools, and environment interact through an iterative process of reasoning, thoughts, actions, and observations to produce a final response.

3.3 LLM Experiments. In order to address our research questions, we perform an experiment in which we compare our agentic framework using zero-shot prompting to standalone LLMs using zero-shot, few-shot, parallel, and chain-of-thought prompting methods. Zero-shot prompting provides only task instructions, offering high interpretability, the absence of examples, and a more flexible prompt [110]. In contrast, few-shot prompting involves one or more examples for better context, which helps optimize model performance [111]. For the few-shot evaluation, we used two examples not included in our survey.

A parallel approach directs the model to generate all nine scores in a single query, providing faster inference, reduced computational costs, and better memory efficiency. However, it requires additional post-processing and may increase the risk of model hallucinations [106,112]. We also evaluate a two-step chain-of-thought approach, first prompting the model to explain how a material would perform in a specific design scenario, then integrating that explanation into the final query where the model provides a score. The exact prompts for each approach (*zero-shot, few-shot, parallel, and chain-of-thought*) can be found in the paper by Grandi *et al.* [16].

For each prompting method, we observed instances where the LLM failed to follow instructions correctly. As a result, we excluded responses that failed to return an answer within the specified 0-10 range. Furthermore, in the agentic experiments, we retained only runs where the final answer tool was successfully called and the last search query did not produce an error, ensuring the results reflected the agent's reasoning process rather than a fallback mechanism. If a failure occurred during a run, we reran the prompt up to five times with the same design, material, and criterion combination before excluding it from analysis. For the agentic prompting method, the number of reruns, completion tokens, and types of search queries made by the agent were logged for further analysis of the agent's internal behavior.

We tested these five methods (*agentic, zero-shot, few-shot, parallel, and chain-of-thought*) across six different Qwen 3 model sizes (*1.7B, 4B, 8B, 14B, and 32B*) with thinking mode enabled. The results were then saved for further analysis to evaluate the effects of both model size and prompting strategy.

We adapted the code developed in the paper by Ying *et al.* in their study of how model size and prompting strategy affected the ability of Qwen 2.5 models to emulate professional decision-making [22]. To further support development of LLM methods, the data and code are available on Github [3].

3.4 Evaluation. In order to measure alignment with material selection professionals, we compute the average z-score, mean absolute error (MAE), and R-squared value of the generated scores relative to the distribution of survey data, using a sample size of 144. We assess how these metrics vary across model sizes and prompting strategies.

We compute the z-score for each LLM-generated score to measure its deviation from survey responses for the same material. The

[3] https://github.com/cmudrc/Agents-for-Material-Selection

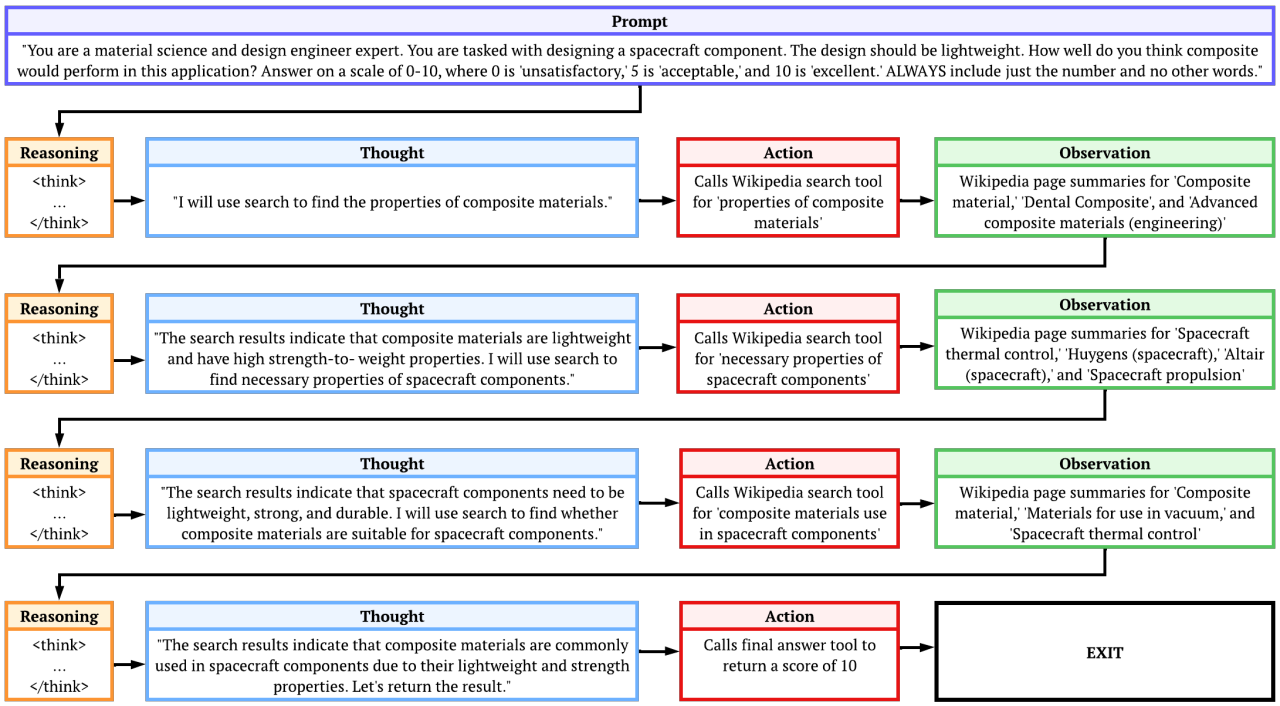


Fig. 2 A flowchart of the LLM-based agent's iterative process when asked to assess the suitability of composite materials for a lightweight spacecraft component. The agent can interact with Wikipedia, determine when and what to search, and use findings to draw conclusions.

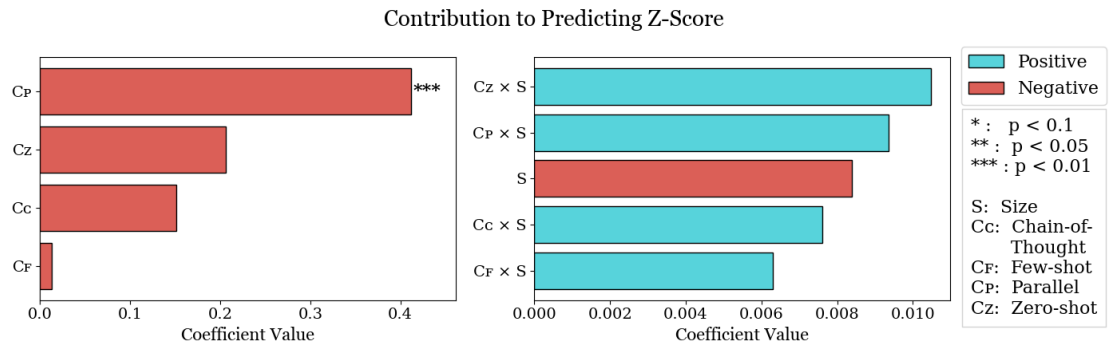


Fig. 3 The contributions of model size and prompt type to predicting z-score are represented as coefficients in the linear regression model. The p-values and coefficients show that parallel prompting with standalone LLMs significantly decreases z-score compared to the agentic framework, while other strategies have negligible effects.

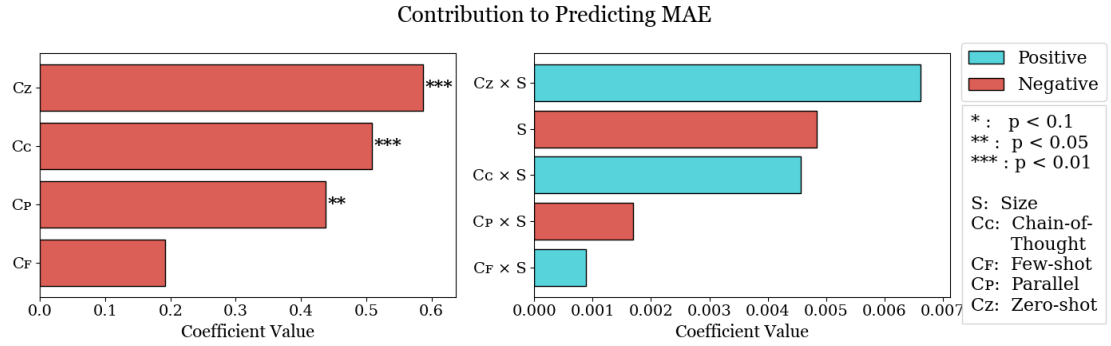


Fig. 4 The contributions of model size and prompt type to predicting MAE relative to survey data are represented as coefficients in the linear regression model. The p-values and coefficients indicate that all prompting strategies used with standalone LLMs, except for few-shot prompting, result in a significant decrease in MAE compared to the agentic framework.

z-score is calculated as

$$z = \frac{x - \mu}{\sigma} \tag{1}$$

where x is the score, μ is the mean of survey responses evaluating that material, and σ is the standard deviation of survey responses for that material.

The MAE is calculated as

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{2}$$

where n is the number of responses, y_i is the ith expert score, and $\hat{y}_i$ is the ith predicted score.

To explore the agent's internal behavior for our second research question, we analyzed its performance across different model sizes using several metrics. For each model size, we measured the number of reruns per prompt, both overall and for successfully retained responses. We also examined search behavior by calculating the total number of search queries per design, criterion, and material combination, along with the proportion of unique queries to assess search diversity. Lastly, we quantified token usage by computing the number of completion tokens generated by the agent. We averaged these metrics across all test cases.

To determine the similarity between search query terms, we used sentence embeddings, which represent words as continuous-valued vectors capturing their syntactic and semantic properties and thereby enabling effective measurement of textual similarity [113,114]. We used Hugging Face's Sentence Transformers, a class of transformer models designed for generating sentence embeddings and selected MiniLM as the most effective model for clustering our data points. To prevent clustering based solely on specific design, criterion, or material, we filtered queries by replacing design applications with '{design},' criteria with '{criterion},' and materials with '{material}' before representing the queries as word embeddings.

To mitigate computational costs associated with the high dimensionality of word embeddings, we apply Principal Component Analysis (PCA), a linear method that reduces dimensionality by projecting the data onto principal components while preserving overall variance [115]. After performing dimensionality reduction, we visualize the word embeddings of agent queries alongside system prompt queries to evaluate whether the agent queries cluster around the prompt queries.

To answer our third research question, we used multiple linear regression to model the relationship between the predicting factors and the results. We ran several regression models, with MAE or z-score as the dependent variable, and prompting strategy and model size as independent variables. Since our independent variables include both categorical and numerical data, we applied one-hot encoding to convert categorical data into a binary format suitable for linear regression. Each prompting method (*agentic, zero-shot, few-shot, parallel, and chain-of-thought*) is represented by a separate column, with a value of 1 indicating the presence of that method and 0 indicating its absence.

We then assessed the statistical relationship between the predictor and response variables using the regression coefficients and associated p-values. For each predictor, our null hypothesis is that it has no effect on the dependent variable after controlling for the other predictors (i.e., the corresponding regression coefficient equals zero). We treat the agentic condition as the reference level; thus, coefficients for other prompting strategies are interpreted as changes in MAE or z-score relative to the agentic baseline. A low p-value indicates evidence against the null hypothesis and suggests that the predictor is associated with a statistically significant shift in the outcome.

A coefficient in our regression model represents the change in the dependent variable for a unit increase in an independent variable, with all other predictors held constant. By setting the agentic approach as the baseline with a coefficient of 0, we interpret other coefficients as the change in the dependent variable relative to this baseline. The p-value indicates the likelihood of obtaining a result as extreme as the one observed, assuming the independent variable has no effect on the dependent variable. A low p-value indicates statistical significance, allowing us to reject the null hypothesis and conclude that the predictor impacts the response, while a high p-value suggests no significant relationship between the variables.

4 Results and Discussion

In this section, we present results from our experiments, focusing on the inference process of the LLM-based agent, the impact of model size and prompting strategy on alignment with expert decisions, and behavioral trends observed across agentic AI.

4.1 Example of LLM-Based Agent Inference Process. The agent's inference process begins by determining what information to search for and executing the necessary search queries. This may involve searching for key material properties and essential characteristics required for the target application. After gathering this information from Wikipedia, the agent evaluates the material's compatibility with the design requirements. Finally, based on its findings, it assigns a numerical rating.

During each iteration of its inference process, the agent generated descriptive chains of thought. It summarized its current findings, identified additional necessary searches, explained their relevance to the prompt, and noted items requiring further confirmation. After gathering results, it explained how the information supported its decision before producing the final answer.

Figure 2 illustrates an example of the agent's inference process in evaluating composites for lightweight spacecraft components. It performs several unique, targeted Wikipedia searches based on the prompt and newly acquired information, refining its assessment through multiple iterations. By synthesizing material properties and engineering constraints, the agent ensures that its final conclusion aligns with key requirements specified in the prompt, such as strength-to-weight ratio. This approach brings the agent closer to replicating the reasoning process of a material selection expert, who may draw on domain knowledge and prior research to iteratively refine conclusions.

However, some of the Wikipedia results retrieved by the agent are repeated or irrelevant, suggesting that Wikipedia may not always provide the necessary information. Additionally, in this example, the agent primarily focuses on weight and strength, while other critical factors, such as thermal stability and environmental resistance, should also be considered when evaluating the suitability of a material for a given application and criterion.

4.2 Effects of Model Size and Prompting Strategy on LLM Performance. Regression analysis is used to develop models that relate experimental parameters to key LLM performance metrics. Specifically, we aim to predict proximity to survey results, measured by z-score and MAE, based on model size and prompting strategy. This relationship is mathematically expressed as $\hat{z} = f(C, S)$ and $\hat{MAE} = f(C, S)$, where $\hat{z}$ is the predicted z-score, $\hat{MAE}$ is the predicted MAE, S is the model size, and C denotes prompt type. S is treated as a continuous variable. We represent C using several binary categorical variable; C_A represents the use of an agentic framework, C_Z represents zero-shot prompting, C_F represents few-shot prompting, C_P represents parallel prompting, and C_C represents chain-of-thought prompting. We consider both main effects and secondary effects in the regression model.

The linear regression model for predicting z-scores is shown in Figure 3 ($p = 0.014$, $R^2 = 0.091$), while those for predicting MAE metrics are shown in Figure 4 ($p = 0.002$, $R^2 = 0.115$). The explanatory power of these regression models is low due to the high degree of spread across problems, but several primary trends still emerge.

Coefficient values for C are relative to the effect of C_A. We observe that parallel prompting leads to a statistically significant



Fig. 5 The mean z-scores from experiments across all prompting strategies and model sizes. On average, parallel prompting produces the lowest z-scores, while the other prompt types yield similar values. Average z-scores remain relatively consistent across model sizes.

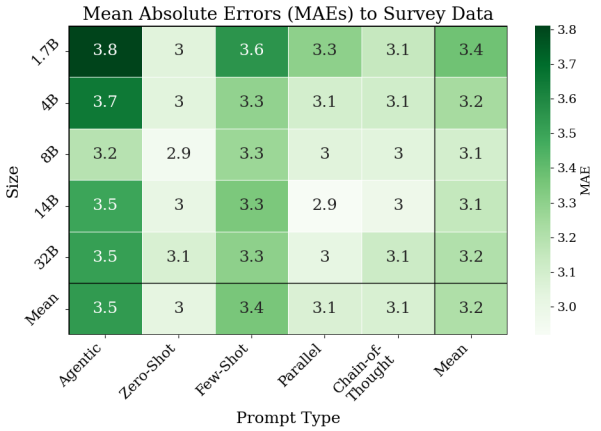


Fig. 6 The MAE relative to survey data across all prompting strategies and model sizes. The greatest error is observed using the agentic and few-shot methods.

decrease in z-score compared to the agentic framework, suggesting improved model performance, while zero-shot, few-shot, and chain-of-thought prompting show no statistically significant differences. In contrast, all prompting methods using standalone LLMs, except for few-shot prompting, result in a significant reduction in MAE compared to the agentic framework, with zero-shot prompting yielding the largest decrease. These findings contrast with those reported for Qwen 2.5 in prior work (Appendix B) , where all prompting types except parallel produced a significant increase in z-score, and all prompting types resulted in a significant decrease in MAE [22]. One possible explanation for the reduced performance differences across prompting strategies is the addition of a thinking mode, although differences in model architecture may also contribute.

We do not observe a statistically significant decrease in either z-score or MAE with increasing model size, as the corresponding coefficients have p-values greater than 0.1, suggesting that larger models do not necessarily produce more accurate predictions. Additionally, the minimal impact of the interaction between model size and prompt type further supports that prompting strategy plays a larger role in aligning with expert responses than model size. Similar trends were observed in Qwen 2.5 studies [22].

The average z-score and MAE across prompting strategies and model sizes are presented in Figures 5 and 6. We assume that a closer proximity to the survey data indicates greater accuracy, with z-scores and MAEs closer to 0 being ideal.

Our z-score analysis reveals no clear trend between model size and alignment with expert decision-making, consistent with our linear regression results. This contrasts with previous Qwen 2.5 experiments, which observed a non-linear trend where the z-score initially decreased with model size before increasing beyond a certain point [22]. We also find that, compared to Qwen 2.5 results, Qwen 3 exhibits a substantial increase in z-score for agentic AI [22]. This may be due to the combined effects of thinking mode and the agentic framework, both of which encourage extended reasoning that can lead to overthinking, a factor shown to negatively affect model performance [50].

When assessed by MAE, it appears that proximity to expert results initially improves with model size before remaining steady, contrasting with the z-score behavior. These differences stem from their calculation methods. The z-score measures direction and magnitude, normalizing for scale, while the MAE focuses solely on the magnitude of the distance between points. Although a similar trend is observed with Qwen 2.5 models, the Qwen 3 series exhibit a less steep initial change in average MAE with model size [22]. The non-linear relationship between our model size and alignment with expert decision-making may be due to the uneven scaling of architectural factors, such as layers, attention heads, and context length, with model size [48].

We observe that MAE values appear nearly identical across all model sizes and prompt types, with all values between 2.9 and 3.8. This can be attributed to several factors. MAE is computed by averaging absolute error across all design, criteria, and material combinations, which can mask differences when models produce scores with similar central tendencies. Additionally, the scoring scale is bounded between 1 and 10, limiting the range over which MAE can vary. Consequently, MAE alone may be insufficient to evaluate performance differences, motivating the use of complementary analyses.

Furthermore, across both metrics, we observe smaller differences in average values between prompting types compared to previous Qwen 2.5 results, with the exception of parallel prompting for average z-score. This is consistent with our linear regression findings, which indicate that only parallel prompting yields a statistically significant change in z-score. Compared to prior work, these trends suggest that Qwen 3 models may produce more standardized results than Qwen 2.5, potentially due to the incorporation of a reasoning process [22]. They also align with prior research showing that parallel thinking, distributing the inference budget across multiple reasoning paths, can maintain or even improve model performance relative to extended reasoning [51]. One potential reason is that, because overthinking has been shown to degrade model performance, parallel prompting may outperform other methods in reasoning-capable models by mitigating this effect.

4.3 LLM-Based Agent Behavior. Figure 7 illustrates the distribution of scores by model size for our agentic experiments, while Figure 8 presents the distribution for standalone LLM experiments. The distribution of scores across all model sizes, categorized by prompt type, is shown in Figure 9.

Across all model sizes, AI generally rates materials as more suitable than human professionals. Standalone LLMs generate scores with lower variability than material selection experts, as indicated by the smaller interquartile range. In contrast, the agentic framework produces the highest variance, surpassing that of the survey data and suggesting that LLM-based agents better capture expert decision-making variability. However, compared to Qwen 2.5 experiments (Appendix B), the response spread from the agentic framework is smaller [22]. This may be because the reasoning process reduces variability, making agent behavior more similar to that of standalone LLMs, although differences in model architecture may also have influenced the results.

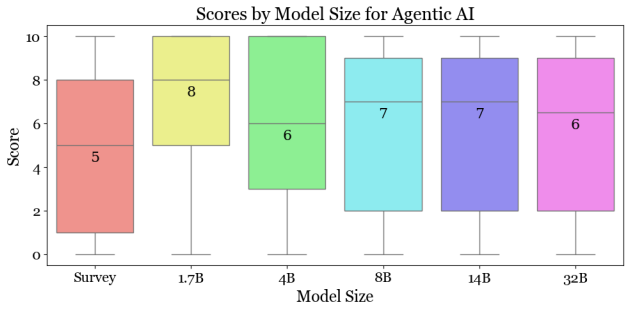


Fig. 7 The scores generated by the LLM-based agent across all design, criteria, and material combinations from our experiments, grouped by model size, with median values labeled.

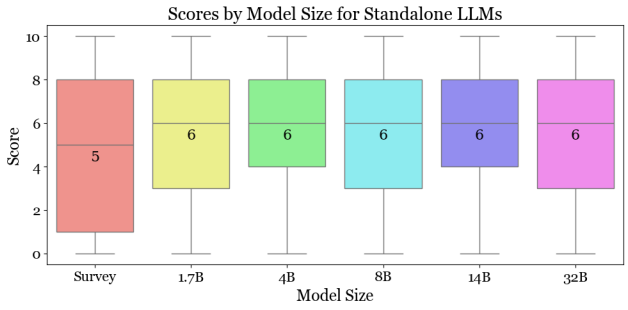


Fig. 8 The scores generated by standalone LLMs across all design, criteria, and material combinations from our experiments, grouped by model size, with median values labeled.

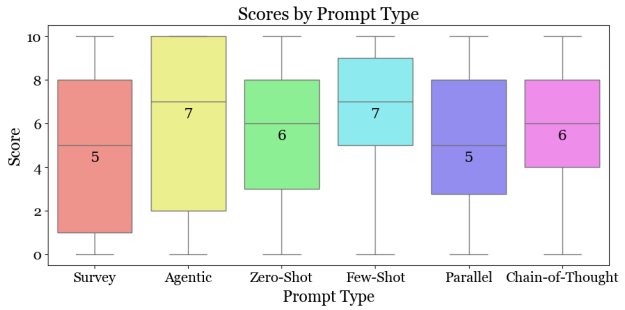


Fig. 9 The scores across all design, criteria, and material combinations from our experiments, grouped by prompt type, with median values labeled.

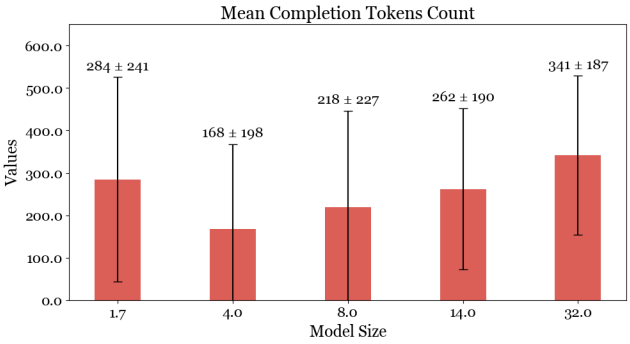


Fig. 10 The mean number of completion tokens used for each iteration of the agent workflow, categorized by model size, with error bars representing standard deviation. The 32B model required the most tokens, followed by the 1.7B model, with the 4B model using the fewest tokens.

We also observe that the median and spread of scores remain relatively consistent across model sizes for standalone LLMs, while agentic AI exhibits greater variance. Earlier experiments with Qwen 2.5 models, which lacked a thinking mode, showed more variability in score spread across model sizes for standalone LLMs. This suggests that differences between the model series, which may include the use of thinking mode, have contributed to a reduction in performance variability by model size [22].

Upon reviewing the agent logs, we found that the LLM-powered agent occasionally encountered repeated code parsing errors. After reaching the maximum number of query iterations, it defaulted to returning a response based on the error-handling mechanism rather than the intended reasoning process. Further analysis revealed that the final answer tool was not called, and an error was reported in the last iteration before a response. During development, we modified our code to rerun a prompt up to five times in an attempt to achieve successful execution before omitting it.

To assess agent behavior, we measured prompt omissions, total and successful reruns, total and unique queries, and completion tokens across model sizes. Figure 10 presents the average completion token count. Discussion of the remaining metrics can be found in Table 1 and Figures 12 and 13 in Appendix A.

The average number of completion tokens generally increased with model size, with the 1.7B model as an exception, which unexpectedly produced the second-highest token count. A review of agent logs revealed that as model size increased, the reasoning generated by the models became more detailed, resulting in longer outputs and higher token usage for the larger models. Interestingly, the 1.7B model exhibited the greatest amount of backtracking and self-reflection during its reasoning process, more frequently revisiting earlier decisions and generating thinking phrases, which contributed to its increased token count. This suggests that smaller reasoning-capable models may be less efficient and more prone to reasoning traps. Prior research supports this, indicating that smaller models (under 3B parameters) tend to perform better when fine-tuned on shorter reasoning chains that align with their intrinsic learning capacity, rather than on lengthy chains of thought [116].

Our study also found that average token usage in Qwen 3 models increased significantly compared to Qwen 2.5 results (Appendix B), due to the generation of more thinking tokens such as "wait," "but," "okay," "however," "hmm," "alternatively," and "maybe" [22]. This aligns with prior research showing that the use of reasoning models leads to higher token consumption [49,50]. However, these reasons for increased token usage differ from previous Qwen 2.5 experiments, where increased token counts were attributed to returning justifications within final answers or generating code with more extensive comments [22]. These results suggest an improvement in instruction-following, which may stem from the reasoning process itself, enhancements between the Qwen 2.5 and Qwen 3 series, or a combination of both.

In the agent logs, we observed that many of the agent's queries closely followed the predefined query formats in the system prompt, such as *properties of {material}, necessary properties of {design},* and *{material} use in {design}.*This finding aligns with our embedding analysis, shown in Figure 11 after applying PCA to reduce the dimensionality.

The data points primarily group into three main clusters near the embeddings associated with the example searches provided in the system prompt, which are represented by black dots. This indicates that the models often generate similar terms or structures. However, there is still significant variability around these points, suggesting variability in query construction. Upon review-

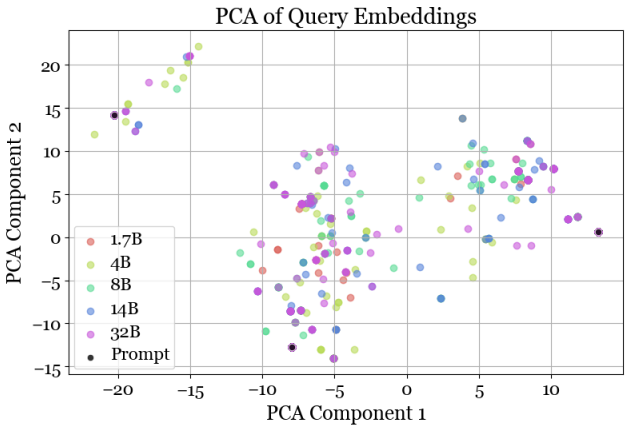


Fig. 11 The results of the word embeddings analysis after PCA, color-coded based on model size, reveal three main clusters around the system prompt embeddings. This suggests that the models generally generate query responses that align with the system prompt.

ing the queries in each cluster, we find that the clusters correspond to distinct themes: one focuses on material properties related to a criterion, another centers on design-related properties and requirements, and the third examines how materials perform within specific applications and criteria. These findings are consistent with results observed in Qwen 2.5 models [22].

Interpreted in a material selection workflow, lower MAE and z-score indicate closer agreement with expert scoring patterns and therefore greater potential utility for early-stage screening and shortlist generation [21,27]. In this context, the primary value of improved alignment is not "winning" a benchmark. But, rather, reducing the risk that an assistant systematically over- or underprioritizes particular material classes under common conceptual design criteria [16,21]. Our results therefore inform how LLM-based tools might be configured for practical material screening.

5 Limitations and Future Work

While our analysis provides insights into prompting strategies and model size, these results should be interpreted primarily as evidence about AI-assisted material screening and ranking on MSEval [21], rather than as end-to-end automation of real-world material selection decisions [26,27]. Our paper focused exclusively on the Qwen 3 and Qwen 2.5 models, which already demonstrated key differences in results. However, given that different architectures may exhibit varying performance patterns based on size and prompt type, the generalizability of our results to other model series is unknown. Future research should investigate a broader range of models to better understand how these factors interact across diverse LLMs. Additionally, a more quantitative comparison between Qwen 2.5 and Qwen 3 results could provide clearer insights into how changes in model architecture impact performance metrics.

In our experiment, we collected results from Qwen 3 models using *thinking mode* and compared them to Qwen 2.5 models, which did not use a reasoning process. Since other differences exist between these model families, future research should conduct controlled comparisons within a single model series that include both the inclusion and exclusion of thinking mode to better isolate the impact of a reasoning process on model performance.

We observed slight differences in the behavior of the LLM-based agent, leading to variations in the final scores generated across different runs. Similar fluctuations were noted in standalone LLMs, reflecting the inherent stochasticity of LLM inference. In this study, we focus on aggregate alignment trends across the full MSEval benchmark, which comprises 144 distinct design scenarios, rather than on individual predictions, allowing run-to-run variability to be partially mitigated through aggregation. Since our analysis was based on a single run per configuration, future work should conduct systematic sensitivity analyses across repeated trials to more explicitly quantify output variance and assess the reproducibility of alignment trends.

We equipped our LLM-based agent with a Wikipedia search tool. While Wikipedia provides general information, it is not considered a scholarly resource and has a limited knowledge base, as evidenced by its often uninformative page summaries. In practice, material selection professionals often rely on peer-reviewed journals, industry reports, and past project data. Future research could explore retrieval-augmented generation (RAG) tools for accessing specialized databases or alternative search APIs to enhance result quality.

6 Conclusion

The agentic framework effectively enables LLMs to execute actions and interact with external environments using tools. Our research demonstrates that this structured approach allows LLMs to break down complex tasks and generate informed responses, facilitating a reasoning process that more closely resembles how a material selection expert might approach decision-making. Our LLM-based agent tends to replicate search queries similar to those provided in the system prompt, emphasizing the importance of strong examples in guiding AI behavior for complex design tasks.

Our linear regression results demonstrated that choice of prompting style played a larger role in predicting both z-score and MAE in comparison to model size. Although larger models tend to produce more detailed reasoning and consume more tokens, these factors do not translate into significantly improved accuracy. This suggests that prompt engineering has a greater influence than model size in driving accurate AI behavior across various design applications.

Furthermore, our results demonstrate greater standardization of performance across model sizes and prompting types, with reduced variability compared to earlier non-thinking models. This suggests that designers may have increased flexibility in choosing models and prompting strategies to achieve superior performance. While these differences may be partly attributed to changes in the model series, the implementation of a thinking process may be a contributing factor.

We observe that agents based on smaller models exhibit less efficient reasoning, characterized by increased backtracking and higher token consumption. Parallel prompting with reasoning-capable models improves alignment with expert recommendations, likely by mitigating overthinking. In contrast, integrating reasoning within the agentic framework tends to reduce accuracy, potentially due to increased overthinking and error accumulation. However, these differences may not be solely attributable to reasoning integration, as factors such as model architecture and training protocols may also influence outcomes.

Practically, these findings suggest that for bounded material screening tasks (fixed candidate sets and criteria), standalone prompting, especially parallel prompting, may offer the most efficient route to expert-like shortlists. In contrast, tool-augmented agents should not be assumed to improve material recommendations unless retrieval is constrained to criterion-relevant, engineering-grade sources (e.g., datasheets, standards, handbooks) and the agent is guided to justify scores with traceable evidence. Finally, since thinking mode increased token usage without clear accuracy gains across sizes, selective use of reasoning (e.g., only for ambiguous cases or conflicting constraints) may be preferable in interactive design workflows.

Acknowledgment

We would like to thank the anonymous survey participants in the Autodesk Research Community for their contribution.

Appendix A: Additional Analysis

Table 1 shows the percentage of LLM-based agent responses that were omitted in our analysis. Figure 12 displays the average number of reruns, categorized by successful prompts and all prompts, while Figure 13 illustrates the average number of searches conducted by the agent, both total and unique.

Table 1 The percentage of responses omitted by our program across all model sizes, categorized by whether the final answer tool was called and whether the last agent iteration resulted in an error. Omissions occurred only for the 8B model, with 22 omissions total, and no omissions for any other model size or prompt type.

ine **1.7B** ine 0.0% ine	**4B** 0.0%	**8B** 15.3%	**14B** 0.0%	**32B** 0.0%

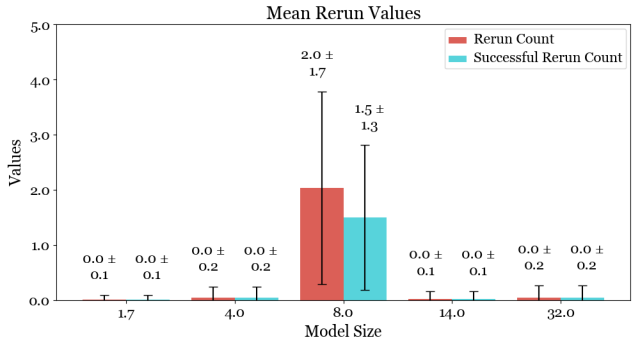


Fig. 12 The average number of reruns per prompt, both overall and for successful prompts, grouped by model size. The 8B model shows the highest mean reruns in both categories, while other model sizes required much fewer reruns.

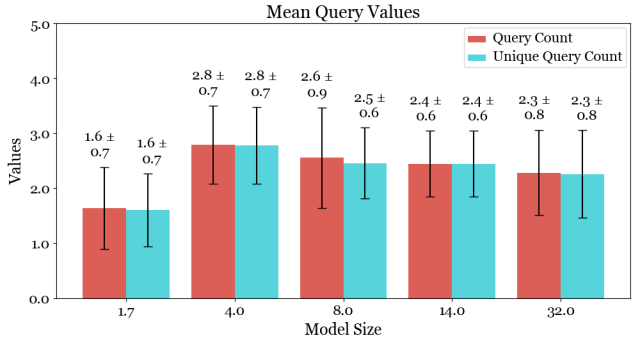


Fig. 13 The average number of total and unique queries for successful prompts, categorized by model size. The 4B model generated the highest number of queries, while the 1.7B model produced the fewest. Most queries were unique across all model sizes, with the 8B model showing some repeated searches.

Our results reveal nonlinear effects of model size on agent behavior. Both smaller models (1.7B and 4B) and larger models (14B and 32B) exhibited stable performance, with no omissions, minimal reruns, and high proportions of unique queries. In contrast, the 8B model showed a decline in performance, characterized by a higher rate of omissions, increased reruns, and a slightly lower proportion of unique queries.

These results align with previous Qwen 2.5 experiments, where performance measured by omissions, reruns, and unique queries initially improved with increasing model size but dropped sharply for mid-sized models [22]. Despite these similarities, the Qwen 3 series demonstrates clear improvements. Notably, the 8B model exhibited a much lower proportion of omissions and fewer reruns compared to the Qwen 2.5 14B model [22]. Moreover, no other Qwen 3 model sizes showed high rates of omissions or reruns, and duplicate query searches were reduced relative to Qwen 2.5 [22].

Appendix B: Qwen 2.5 Results

To facilitate comparisons with the Qwen 2.5 family of models referenced in Section 4, we reproduce in this section the key findings from our prior work [22]. We include the z-scores (Figure 14), and MAE scores (Figure 15); the distributions of scores grouped by model size for agentic AI (Figure 16) and standalone LLMs (Figure 17), as well as prompt type (Figure 18) [22]; and token consumption (Figure 19).

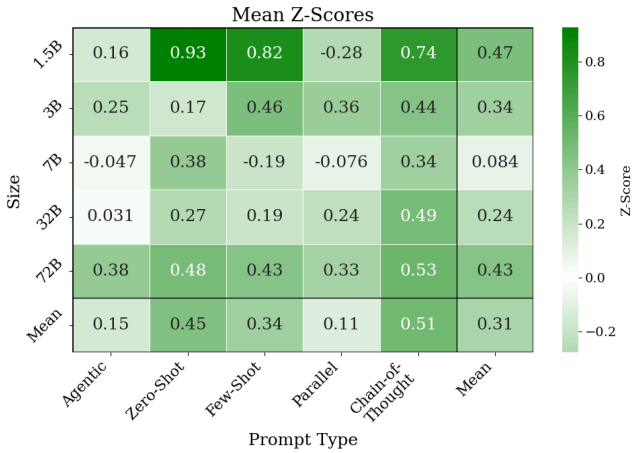


Fig. 14 Qwen 2.5 mean z-scores.

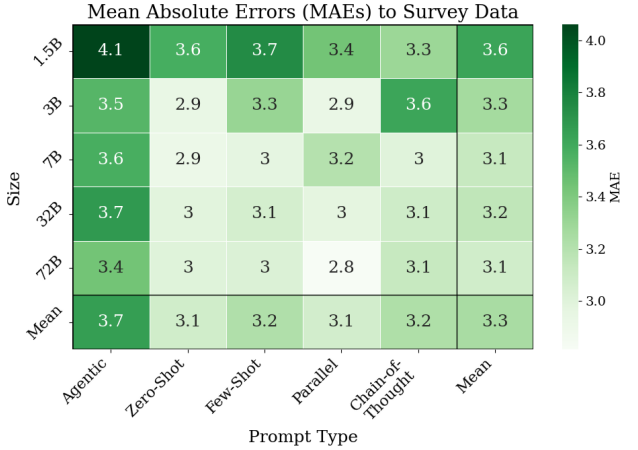


Fig. 15 Qwen 2.5 MAE.

References

[1] Wiederhorn, S., Fields, R., Low, S., Bahng, G.-W., Wehrstedt, A., Hahn, J., Tomota, Y., Miyata, T., Lin, H., Freeman, B., Aihara, S., Hagihara, Y., and Tagawa, T., 2006, "Mechanical Properties," *Springer Handbook of Materials Measurement Methods*, H. Czichos, T. Saito, and L. Smith, eds., Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 283–397.

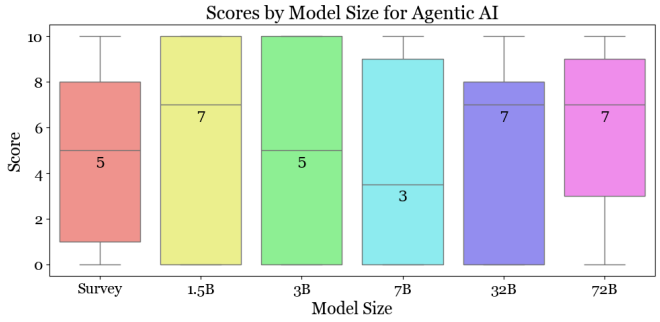


Fig. 16 Qwen 2.5 agentic AI score distributions grouped by model size.

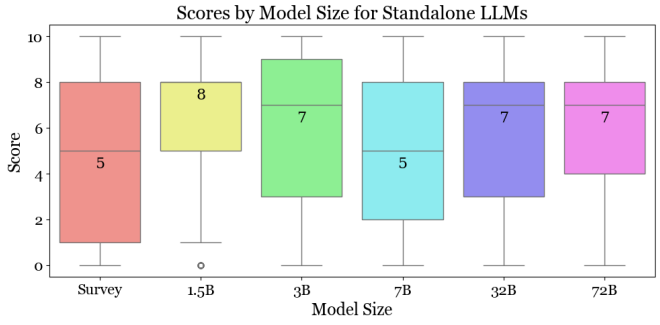


Fig. 17 Qwen 2.5 standalone LLM score distributions grouped by model size.

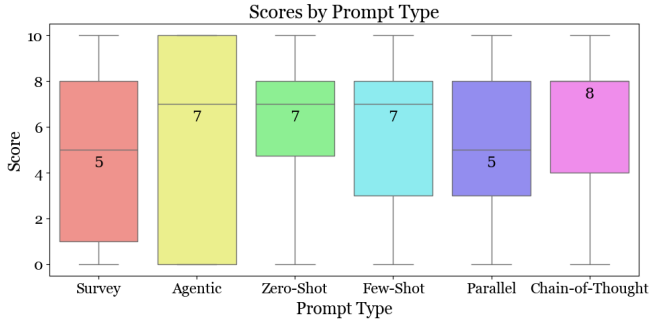


Fig. 18 Qwen 2.5 score distributions grouped by prompt type.

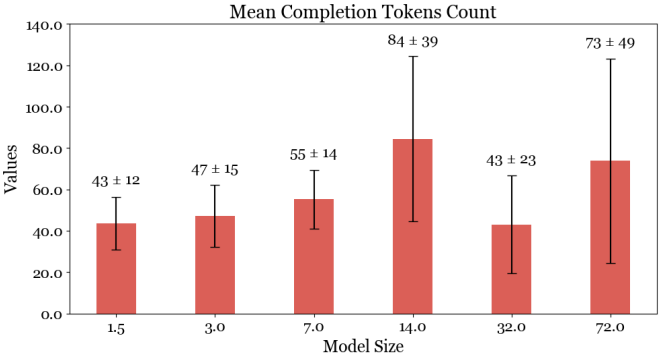


Fig. 19 Qwen 2.5 mean number of completion tokens.

[2] William D. Callister, J. and Rethwisch, D. G., 2007, *Materials Science and Engineering: An Introduction*, Vol. 7, John Wiley Sons, Inc., New York.

[3] Giachetti, R., 1997, "Manufacturing Process and Material Selection During Conceptual Design," *Proceedings of the 6th Industrial Engineering Research Conference*, Miami, FL, https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=821161

[4] Clark, G., Kosoris, J., Hong, L. N., and Crul, M., 2009, "Design for Sustainability: Current Trends in Sustainable Product Design and Development," Sustainability, **1**(3), pp. 409–424.

[5] Karana, E. and Kesteren, I. V., 2008, "Materials Affect: The Role of Materials in Product Experience," *Design and Emotion Moves*, P. Desmet, J. van Erp, and M. Karlsson, eds., Cambridge Scholars Publishing, UK, Chap. 11, pp. 221–245.

[6] Farag, M. M., 2020, *Materials and Process Selection for Engineering Design*, CRC Press, Boca Raton.

[7] Bissler, A. and Ghahestani, S., 2018, "Key Factors in Driving Sustainability Initiatives in the Supply Chain: A Multiple Case Study of Manufacturing Companies," Journal of Supply Chain Management, **45**(2), pp. 78–92.

[8] Alonso, E., Gregory, J., Field, F., and Kirchain, R., 2007, "Material Availability and the Supply Chain: Risks, Effects, and Responses," Environmental Science Technology, **41**(19), pp. 6649–6656.

[9] Brechet, Y., Bassetti, D., Landru, D., and Salvo, L., 2001, "Challenges in materials and process selection," Progress in Materials Science, **46**(3–4), pp. 407–428.

[10] Jahan, A., Ismail, M., Sapuan, S., and Mustapha, F., 2010, "Material screening and choosing methods – A review," Materials Design, **31**(2), pp. 696–705.

[11] Ashby, M., Bréchet, Y., Cebon, D., and c, L. S., 2004, "Selection strategies for materials and processes," Materials Design, **25**(1), pp. 51–67.

[12] Chen, K., Fei, C., Bi, Z., Liu, J., Peng, B., Zhang, S., Pan, X., Xu, J., Wang, J., Yin, C. H., Zhang, Y., Feng, P., Wen, Y., Wang, T., Li, M., Ren, J., Niu, Q., Chen, S., Hsieh, W., Yan, L. K. Q., Liang, C. X., Xu, H., Tseng, H.-M., Song, X., and Liu, M., 2024, "Deep Learning and Machine Learning – Natural Language Processing: From Theory to Application," arXiv, 2411.05026, https://arxiv.org/abs/2411.05026

[13] Chiner, M., 1988, "Planning of expert systems for materials selection," Materials Design, **9**(4), pp. 195–203.

[14] Huang, J. and Chang, K. C.-C., 2023, "Towards Reasoning in Large Language Models: A Survey," *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1049–1065, doi: 10.48550/arXiv.2212.10403.

[15] Williams, S. and Huckle, J., 2024, "Easy Problems That LLMs Get Wrong," arXiv, 2212.10403, https://arxiv.org/abs/2405.19616

[16] Grandi, D., Jain, Y. P., Groom, A., Cramer, B., and McComb, C., 2024, "Evaluating Large Language Models for Material Selection," Journal of Computing and Information Science in Engineering, **25**(2), pp. 1–33.

[17] Wang, Q., Wang, Z., Su, Y., Tong, H., and Song, Y., 2024, "Rethinking the Bounds of LLM Reasoning: Are Multi-Agent Discussions the Key?" *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*, Association for Computational Linguistics, Bangkok, Thailand, pp. 6106–6131, doi: 10.18653/v1/2024.acl-long.331.

[18] Mukherjee, A. and Chang, H. H., 2025, "Agentic AI: Autonomy, Accountability, and the Algorithmic Society," arXiv, doi: 10.2139/ssrn.5123621, https://arxiv.org/abs/2309.07870

[19] Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Wang, J., Jin, S., Zhou, E., Zheng, R., Fan, X., Wang, X., Xiong, L., Zhou, Y., Wang, W., Jiang, C., Zou, Y., Liu, X., Yin, Z., Dou, S., Weng, R., Cheng, W., Zhang, Q., Qin, W., Zheng, Y., Qiu, X., Huang, X., and Gui, T., 2023, "The Rise and Potential of Large Language Model Based Agents: A Survey," arXiv, 2309.07864, https://arxiv.org/abs/2309.07864

[20] Ruan, J., Chen, Y., Zhang, B., Xu, Z., Bao, T., Du, G., Shi, S., Mao, H., Li, Z., Zeng, X., and Zhao, R., 2023, "TPTU: Large Language Model-based AI Agents for Task Planning and Tool Usage," arXiv, 2308.03427, https://arxiv.org/abs/2308.03427

[21] Jain, Y. P., Grandi, D., Groom, A., Cramer, B., and McComb, C., 2025, "MSEval: A Dataset for Material Selection in Conceptual Design to Evaluate Algorithmic Models," Journal of Mechanical Design, **147**(4).

[22] Ying, M. Y., Grandi, D., Groom, A., and McComb, C., 2025, "Evaluating the Role of Model Size in Agentic AI for Expert-Like Material Selection," p. V03AT03A017, doi: 10.1115/DETC2025-168873.

[23] Xu, F., Hao, Q., Zong, Z., Wang, J., Zhang, Y., Wang, J., Lan, X., Gong, J., Ouyang, T., Meng, F., Shao, C., Yan, Y., Yang, Q., Song, Y., Ren, S., Hu, X., Li, Y., Feng, J., Gao, C., and Li, Y., 2025, "Towards Large Reasoning Models: A Survey of Reinforced Reasoning with Large Language Models," 2501.09686, https://arxiv.org/abs/2501.09686

[24] Kambhampati, S., Stechly, K., and Valmeekam, K., 2025, "(How) Do reasoning models reason?" Annals of the New York Academy of Sciences, **1547**(1), p. 33–40.

[25] Zarandi, M. H. F., Mansour, S., Hosseinijou, S. A., and Avazbeigi, M., 2011, "A material selection methodology and expert system for sustainable product design," The International Journal of Advanced Manufacturing Technology, **57**(9–12), pp. 885–903.

[26] Deng, Y.-M. and Edwards, K., 2007, "The role of materials identification and selection in engineering design," Materials Design, **28**(1), pp. 131–139.

[27] Ashby, M. F., 2010, *Materials Selection in Mechanical Design (Fourth Edition)*, Butterworth-Heinemann.

[28] Rahim, A. A., Musa, S. N., Ramesh, S., and Lim, M. K., 2020, "A systematic review on material selection methods," Proceedings of the Institution of Mechanical Engineers, Part L: Journal of Materials: Design and Applications, **234**(7), pp. 1032–1059.

[29] Amen, R. and Vomacka, P., 2001, "Case-based reasoning as a tool for materials selection," Materials Design, **22**(5), pp. 353–358.

[30] Emovon, I. and Oghenenyerovwho, O. S., 2020, "Application of MCDM method in material selection for optimal design: A review," Results in Materials, **7**.

[31] Karandikar, H. M. and Mistree, F., 1992, "An Approach for Concurrent and Integrated Material Selection and Dimensional Synthesis," Journal of Mechanical Design, **114**(4), pp. 633–641.

[32] Rao, R. V., 2008, "A decision making methodology for material selection using an improved compromise ranking method," Materials Design, **29**(10), pp. 1949–1954.

[33] Chandrasekhar, A., Sridhara, S., and Suresh, K., 2022, "Integrating material selection with design optimization via neural networks," Engineering with Computers, **38**(5), pp. 4715–4730.

[34] Plaat, A., Wong, A., Verberne, S., Broekens, J., van Stein, N., and Back, T., 2024, "Reasoning with Large Language Models, a Survey," 2407.11511, https://arxiv.org/abs/2407.11511

[35] Li, Z.-Z., Zhang, D., Zhang, M.-L., Zhang, J., Liu, Z., Yao, Y., Xu, H., Zheng, J., Wang, P.-J., Chen, X., Zhang, Y., Yin, F., Dong, J., Li, Z., Bi, B.-L., Mei, L.-R., Fang, J., Liang, X., Guo, Z., Song, L., and Liu, C.-L., 2025, "From System 1 to System 2: A Survey of Reasoning Large Language Models," 2502.17419, https://arxiv.org/abs/2502.17419

[36] Sui, Y., Chuang, Y.-N., Wang, G., Zhang, J., Zhang, T., Yuan, J., Liu, H., Wen, A., Zhong, S., Chen, H., and Hu, X., 2025, "Stop Overthinking: A Survey on Efficient Reasoning for Large Language Models," 2503.16419, https://arxiv.org/abs/2503.16419

[37] 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," *Proceedings of the 36th International Conference on Neural Information Processing Systems*, Curran Associates Inc.

[38] Wang, Y., Yang, Q., Zeng, Z., Ren, L., Liu, L., Peng, B., Cheng, H., He, X., Wang, K., Gao, J., Chen, W., Wang, S., Du, S. S., and Shen, Y., 2025, "Reinforcement Learning for Reasoning in Large Language Models with One Training Example," 2504.20571, https://arxiv.org/abs/2504.20571

[39] Feng, S., Fang, G., Ma, X., and Wang, X., 2025, "Efficient Reasoning Models: A Survey," 2504.10903, https://arxiv.org/abs/2504.10903

[40] Dang, R., Huang, S., and Chen, J., 2025, "Internal Bias in Reasoning Models leads to Overthinking," 2505.16448, https://arxiv.org/abs/2505.16448

[41] Liu, Y., Wu, J., He, Y., Gao, H., Chen, H., Bi, B., Gong, R., Zhang, J., Huang, Z., and Hooi, B., 2025, "Efficient Inference for Large Reasoning Models: A Survey," 2503.23077, https://arxiv.org/abs/2503.23077

[42] Han, T., Wang, Z., Fang, C., Zhao, S., Ma, S., and Chen, Z., 2025, "Token-Budget-Aware LLM Reasoning," 2412.18547, https://arxiv.org/abs/2412.18547

[43] Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J., and Tian, Y., 2024, "Training Large Language Models to Reason in a Continuous Latent Space," 2412.06769, https://arxiv.org/abs/2412.06769

[44] Luo, H., Shen, L., He, H., Wang, Y., Liu, S., Li, W., Tan, N., Cao, X., and Tao, D., 2025, "O1-Pruner: Length-Harmonizing Fine-Tuning for O1-Like Reasoning Pruning," 2501.12570, https://arxiv.org/abs/2501.12570

[45] Ma, X., Wan, G., Yu, R., Fang, G., and Wang, X., 2025, "CoT-Valve: Length-Compressible Chain-of-Thought Tuning," 2502.09601, https://arxiv.org/abs/2502.09601

[46] Arora, D. and Zanette, A., 2025, "Training Language Models to Reason Efficiently," 2502.04463, https://arxiv.org/abs/2502.04463

[47] Zhang, W., Qiao, S., Luo, L., Li, Y., Zheng, C., Xu, Q., Li, M., Gui, Y., He, Y., Qiu, J., Hong, J., and Sun, J., 2025, "SynapseRoute: An Auto-Route Switching Framework on Dual-State Large Language Model," 2507.02822, https://arxiv.org/abs/2507.02822

[48] Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., Zheng, C., Liu, D., Zhou, F., Huang, F., Hu, F., Ge, H., Wei, H., Lin, H., Tang, J., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Zhou, J., Lin, J., Dang, K., Bao, K., Yang, K., Yu, L., Deng, L., Li, M., Xue, M., Li, M., Zhang, P., Wang, P., Zhu, Q., Men, R., Gao, R., Liu, S., Luo, S., Li, T., Tang, T., Yin, W., Ren, X., Wang, X., Zhang, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Zhang, Y., Wan, Y., Liu, Y., Wang, Z., Cui, Z., Zhang, Z., Zhou, Z., and Qiu, Z., 2025, "Qwen3 Technical Report," 2505.09388, https://arxiv.org/abs/2505.09388

[49] Ma, W., He, J., Snell, C., Griggs, T., Min, S., and Zaharia, M., 2025, "Reasoning Models Can Be Effective Without Thinking," 2504.09858, https://arxiv.org/abs/2504.09858

[50] Ding, B., Chen, Y., Wang, F., Ming, L., and Lin, T., 2025, "Do Thinking Tokens Help or Trap? Towards More Efficient Large Reasoning Model," 2506.23840, https://arxiv.org/abs/2506.23840

[51] Ghosal, S. S., Chakraborty, S., Reddy, A., Lu, Y., Wang, M., Manocha, D., Huang, F., Ghavamzadeh, M., and Bedi, A. S., 2025, "Does Thinking More always Help? Understanding Test-Time Scaling in Reasoning Models," 2506.04210, https://arxiv.org/abs/2506.04210

[52] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R., 2018, "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding," arXiv, 1804.07461, https://arxiv.org/abs/1804.07461

[53] Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R., 2019, "SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems," arXiv, 1905.00537, https://arxiv.org/abs/1905.00537

[54] Lin, Y.-T. and Chen, Y.-N., 2023, "LLM-Eval: Unified Multi-Dimensional Automatic Evaluation for Open-Domain Conversations with Large Language Models," *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, Association for Computational Linguistics, Toronto, Canada, pp. 47–58, doi: 10.18653/v1/2023.nlp4convai-1.5.

[55] Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J., 2021, "Measuring Massive Multitask Language Understanding," 2009.03300, https://arxiv.org/abs/2009.03300

[56] Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C. D., Ré, C., Acosta-Navas, D., Hudson, D. A., Zelikman, E., Durmus, E., Ladhak, F., Rong, F., Ren, H., Yao, H., Wang, J., Santhanam, K., Orr, L., Zheng, L., Yuksekgonul, M., Suzgun, M., Kim, N., Guha, N., Chatterji, N., Khattab, O., Henderson, P., Huang, Q., Chi, R., Xie, S. M., Santurkar, S., Ganguli, S., Hashimoto, T., Icard, T., Zhang, T., Chaudhary, V., Wang, W., Li, X., Mai, Y., Zhang, Y., and Koreeda, Y., 2023, "Holistic Evaluation of Language Models," 2211.09110, https://arxiv.org/abs/2211.09110

[57] Huang, Y., Bai, Y., Zhu, Z., Zhang, J., Zhang, J., Su, T., Liu, J., Lv, C., Zhang, Y., Lei, J., Fu, Y., Sun, M., and He, J., 2023, "C-Eval: A Multi-Level Multi-Discipline Chinese Evaluation Suite for Foundation Models," 2305.08322, https://arxiv.org/abs/2305.08322

[58] Zhong, W., Cui, R., Guo, Y., Liang, Y., Lu, S., Wang, Y., Saied, A., Chen, W., and Duan, N., 2023, "AGIEval: A Human-Centric Benchmark for Evaluating Foundation Models," 2304.06364, https://arxiv.org/abs/2304.06364

[59] Dubois, Y., Li, X., Taori, R., Zhang, T., Gulrajani, I., Ba, J., Guestrin, C., Liang, P., and Hashimoto, T. B., 2024, "AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback," 2305.14387, https://arxiv.org/abs/2305.14387

[60] Chiang, W.-L., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D., Zhang, H., Zhu, B., Jordan, M., Gonzalez, J. E., and Stoica, I., 2024, "Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference," 2403.04132, https://arxiv.org/abs/2403.04132

[61] Richardson, M., Burges, C. J. C., and Renshaw, E., 2013, "MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text," *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, Association for Computational Linguistics, Seattle, Washington, USA, pp. 193–203.

[62] Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P., 2016, "SQuAD: 100,000+ Questions for Machine Comprehension of Text," 1606.05250, https://arxiv.org/abs/1606.05250

[63] Yang, Y., tau Yih, W., and Meek, C., 2015, "WikiQA: A Challenge Dataset for Open-Domain Question Answering," *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, Association for Computational Linguistics, Lisbon, Portugal, pp. 2013–2018, doi: 10.18653/v1/D15-1237.

[64] Dasigi, P., Lo, K., Beltagy, I., Cohan, A., Smith, N. A., and Gardner, M., 2021, "A Dataset of Information-Seeking Questions and Answers Anchored in Research Papers," 2105.03011, https://arxiv.org/abs/2105.03011

[65] Shaham, U., Ivgi, M., Efrat, A., Berant, J., and Levy, O., 2023, "ZeroSCROLLS: A Zero-Shot Benchmark for Long Text Understanding," 2305.14196, https://arxiv.org/abs/2305.14196

[66] Doris, A. C., Grandi, D., Tomich, R., Alam, M. F., Ataei, M., Cheong, H., and Ahmed, F., 2024, "DesignQA: A Multimodal Benchmark for Evaluating Large Language Models' Understanding of Engineering Documentation," Journal of Computing and Information Science in Engineering, **25**(2), p. 021009.

[67] Liu, Y., Duan, H., Zhang, Y., Li, B., Zhang, S., Zhao, W., Yuan, Y., Wang, J., He, C., Liu, Z., Chen, K., and Lin, D., 2024, "MMBench: Is Your Multi-modal Model an All-around Player?" 2307.06281, https://arxiv.org/abs/2307.06281

[68] Fu, C., Chen, P., Shen, Y., Qin, Y., Zhang, M., Lin, X., Yang, J., Zheng, X., Li, K., Sun, X., Wu, Y., and Ji, R., 2024, "MME: A Comprehensive Evaluation Benchmark for Multimodal Large Language Models," 2306.13394, https://arxiv.org/abs/2306.13394

[69] Chen, Y., Hu, H., Luan, Y., Sun, H., Changpinyo, S., Ritter, A., and Chang, M.-W., 2023, "Can Pre-trained Vision and Language Models Answer Visual Information-Seeking Questions?" 2302.11713, https://arxiv.org/abs/2302.11713

[70] Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.-W., Zhu, S.-C., Tafjord, O., Clark, P., and Kalyan, A., 2022, "Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering," 2209.09513, https://arxiv.org/abs/2209.09513

[71] Ma, K., Grandi, D., McComb, C., and Goucher-Lambert, K., 2023, "Conceptual Design Generation Using Large Language Models," *Proceedings of the ASME 2023 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE)*, Boston, Massachusetts, USA, p. V006T06A021, doi: 10.1115/DETC2023-116838.

[72] Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., Nori, H., Palangi, H., Ribeiro, M. T., and Zhang, Y., 2023, "Sparks of Artificial General Intelligence: Early experiments with GPT-4," 2303.12712, https://arxiv.org/abs/2303.12712

[73] Bang, Y., Cahyawijaya, S., Lee, N., Dai, W., Su, D., Wilie, B., Lovenia, H., Ji, Z., Yu, T., Chung, W., Do, Q. V., Xu, Y., and Fung, P., 2023, "A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity," 2302.04023, https://arxiv.org/abs/2302.04023

[74] Ziems, C., Held, W., Shaikh, O., Chen, J., Zhang, Z., and Yang, D., 2024, "Can Large Language Models Transform Computational Social Science?" Computational Linguistics, **50**(1), pp. 237–291.

[75] Russell, S., Norvig, P., and Davis, E., 2010, *Artificial Intelligence: A Modern Approach*, Prentice Hall series in artificial intelligence, Prentice Hall.

[76] Wooldridge, M. and Jennings, N. R., 1995, "Intelligent agents: theory and practice," The Knowledge Engineering Review, **10**(2), p. 115–152.

[77] Park, J. S., O'Brien, J., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S., 2023, "Generative Agents: Interactive Simulacra of Human Behavior," Association for Computing Machinery, doi: 10.1145/3586183.3606763.

[78] Liu, R., Yang, R., Jia, C., Zhang, G., Zhou, D., Dai, A. M., Yang, D., and Vosoughi, S., 2023, "Training Socially Aligned Language Models on Simulated Social Interactions," 2305.16960, https://arxiv.org/abs/2305.16960

[79] Sumers, T. R., Yao, S., Narasimhan, K., and Griffiths, T. L., 2024, "Cognitive Architectures for Language Agents," 2309.02427, https://arxiv.org/abs/2309.02427

[80] Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W. X., Wei, Z., and Wen, J., 2024, "A survey on large language model based autonomous agents," Frontiers of Computer Science, **18**(6).

[81] Qin, Y., Hu, S., Lin, Y., Chen, W., Ding, N., Cui, G., Zeng, Z., Zhou, X., Huang, Y., Xiao, C., Han, C., Fung, Y. R., Su, Y., Wang, H., Qian, C., Tian, R., Zhu, K., Liang, S., Shen, X., Xu, B., Zhang, Z., Ye, Y., Li, B., Tang, Z., Yi, J., Zhu, Y., Dai, Z., Yan, L., Cong, X., Lu, Y., Zhao, W., Huang, Y., Yan, J., Han, X., Sun, X., Li, D., Phang, J., Yang, C., Wu, T., Ji, H., Li, G., Liu, Z., and Sun, M., 2024, "Tool Learning with Foundation Models," ACM Comput. Surv., **57**(4).

[82] Mialon, G., Dessì, R., Lomeli, M., Nalmpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., Grave, E., LeCun, Y., and Scialom, T., 2023, "Augmented Language Models: a Survey," 2302.07842, https://arxiv.org/abs/2302.07842

[83] Ren, R., Wang, Y., Qu, Y., Zhao, W. X., Liu, J., Tian, H., Wu, H., Wen, J.-R., and Wang, H., 2024, "Investigating the Factual Knowledge Boundary of Large Language Models with Retrieval Augmentation," 2307.11019, https://arxiv.org/abs/2307.11019

[84] Liu, B., Jiang, Y., Zhang, X., Liu, Q., Zhang, S., Biswas, J., and Stone, P., 2023, "LLM+P: Empowering Large Language Models with Optimal Planning Proficiency," 2304.11477, https://arxiv.org/abs/2304.11477

[85] Zhang, H., Du, W., Shan, J., Zhou, Q., Du, Y., Tenenbaum, J. B., Shu, T., and Gan, C., 2024, "Building Cooperative Embodied Agents Modularly with Large Language Models," 2307.02485, https://arxiv.org/abs/2307.02485

[86] Qian, C., Liu, W., Liu, H., Chen, N., Dang, Y., Li, J., Yang, C., Chen, W., Su, Y., Cong, X., Xu, J., Li, D., Liu, Z., and Sun, M., 2024, "ChatDev: Communicative Agents for Software Development," 2307.07924, https://arxiv.org/abs/2307.07924

[87] Mehta, N., Teruel, M., Sanz, P. F., Deng, X., Awadallah, A. H., and Kiseleva, J., 2024, "Improving Grounded Language Understanding in a Collaborative Environment by Interacting with Agents Through Help Feedback," 2304.10750, https://arxiv.org/abs/2304.10750

[88] Aher, G., Arriaga, R. I., and Kalai, A. T., 2023, "Using large language models to simulate multiple humans and replicate human subject studies," *Proceedings of the 40th International Conference on Machine Learning*, JMLR.org, doi: 10.48550/arXiv.2208.10264.

[89] Liu, X., Yu, H., Zhang, H., Xu, Y., Lei, X., Lai, H., Gu, Y., Ding, H., Men, K., Yang, K., Zhang, S., Deng, X., Zeng, A., Du, Z., Zhang, C., Shen, S., Zhang, T., Su, Y., Sun, H., Huang, M., Dong, Y., and Tang, J., 2023, "AgentBench: Evaluating LLMs as Agents," 2308.03688, https://arxiv.org/abs/2308.03688

[90] Lin, J., Tomlin, N., Andreas, J., and Eisner, J., 2024, "Decision-Oriented Dialogue for Human-AI Collaboration," Transactions of the Association for Computational Linguistics, **12**, pp. 892–911.

[91] Liang, Y., Zhu, L., and Yang, Y., 2023, "Tachikuma: Understading Complex Interactions with Multi-Character and Novel Objects by Large Language Models," 2307.12573, https://arxiv.org/abs/2307.12573

[92] Xu, B., Liu, X., Shen, H., Han, Z., Li, Y., Yue, M., Peng, Z., Liu, Y., Yao, Z., and Xu, D., 2023, "Gentopia: A Collaborative Platform for Tool-Augmented LLMs," 2308.04030, https://arxiv.org/abs/2308.04030

[93] Sciammarella, C. and Sciammarella, F., 2012, *Experimental Mechanics of Solids*, Wiley.

[94] Jenkins, C. and Khanna, S., 2005, *Mechanics of Materials: A Modern Integration of Mechanics and Materials in Structural Design*, Elsevier Science.

[95] Zaki, M., Jayadeva, Mausam, and Krishnan, N. M. A., 2024, "MaScQA: investigating materials science knowledge of large language models," Digital Discovery, **3**(2), pp. 313–327.

[96] Jin, H., Zhang, E., and Espinosa, H. D., 2023, "Recent Advances and Applications of Machine Learning in Experimental Solid Mechanics: A Review," Applied Mechanics Reviews, **75**(6), p. 061001.

[97] Hu, Y. and Buehler, M. J., 2023, "Deep language models for interpretative and predictive materials science," APL Machine Learning, **1**(1), p. 010901.

[98] Luu, R. and Buehler, M., 2024, "BioinspiredLLM: Conversational Large Language Model for the Mechanics of Biological and Bio-Inspired Materials," Advanced Science, **11**(2), p. 2306724.

[99] Lei, G., Docherty, R., and Cooper, S. J., 2024, "Materials science in the era of large language models: a perspective," Digital Discovery, **3**(7), pp. 1257–1272.

[100] Picard, C., Edwards, K. M., Doris, A. C., Man, B., Giannone, G., Alam, M. F., and Ahmed, F., 2024, "From Concept to Manufacturing: Evaluating Vision-Language Models for Engineering Design," 2311.12668, https://arxiv.org/abs/2311.12668

[101] Mustapha, K., 2025, "A survey of emerging applications of large language models for problems in mechanics, product design, and manufacturing," Advanced Engineering Informatics, **64**, p. 103066.

[102] Buehler, M. J., 2023, "MeLM, a generative pretrained language modeling framework that solves forward and inverse mechanics problems," Journal of the Mechanics and Physics of Solids, **181**, p. 105454.

[103] Buehler, M. J., 2024, "MechGPT, a Language-Based Strategy for Mechanics and Materials Modeling That Connects Knowledge Across Scales, Disciplines, and Modalities," Applied Mechanics Reviews, **76**(2), p. 021001.

[104] Ni, B. and Buehler, M. J., 2024, "MechAgents: Large language model multi-agent collaborations can solve mechanics problems, generate new data, and integrate knowledge," Extreme Mechanics Letters, **67**, p. 102131.

[105] Ghafarollahi, A. and Buehler, M., 2024, "Rapid and Automated Alloy Design with Graph Neural Network-Powered LLM-Driven Multi-Agent Systems," doi: 10.48550/arXiv.2410.13768.

[106] Rawte, V., Sheth, A., and Das, A., 2023, "A Survey of Hallucination in Large Foundation Models," 2309.05922, https://arxiv.org/abs/2309.05922

[107] Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y., 2023, "ReAct: Synergizing Reasoning and Acting in Language Models," 2210.03629, https://arxiv.org/abs/2210.03629

[108] Zhou, W., Jiang, Y. E., Li, L., Wu, J., Wang, T., Qiu, S., Zhang, J., Chen, J., Wu, R., Wang, S., Zhu, S., Chen, J., Zhang, W., Tang, X., Zhang, N., Chen, H., Cui, P., and Sachan, M., 2023, "Agents: An Open-source Framework for Autonomous Language Agents," 2309.07870, https://arxiv.org/abs/2309.07870

[109] Marvin, G., Hellen, N., Jjingo, D., and Nakatumba-Nabende, J., 2024, "Prompt Engineering in Large Language Models," *Data Intelligence and Cognitive Informatics*, I. J. Jacob, S. Piramuthu, and P. Falkowski-Gilski, eds., Springer Nature Singapore, Singapore, pp. 387–402, https://arxiv.org/abs/2402.07927

[110] Jalham, I. S., 2006, "Decision-making integrated information technology (IIT) approach for material selection," Int. J. Comput. Appl. Technol., **25**(1), p. 65–71.

[111] Sinhababu, N., Parry, A., Ganguly, D., Samanta, D., and Mitra, P., 2024, "Few-shot Prompting for Pairwise Ranking: An Effective Non-Parametric Retrieval Model," *Findings of the Association for Computational Linguistics: EMNLP 2024*, Association for Computational Linguistics, Miami, Florida, USA, pp. 12363–12377, doi: 10.18653/v1/2024.findings-emnlp.720.

[112] Chen, H. M., Luk, W., Yiu, K. F. C., Li, R., Mishchenko, K., Venieris, S. I., and Fan, H., 2024, "Hardware-Aware Parallel Prompt Decoding for Memory-Efficient Acceleration of LLM Inference," 2405.18628, https://arxiv.org/abs/2405.18628

[113] Mikolov, T., Chen, K., Corrado, G. S., and Dean, J., 2013, "Efficient Estimation of Word Representations in Vector Space," *International Conference on Learning Representations*, https://api.semanticscholar.org/CorpusID:5959482

[114] Şenel, L. K., Utlu, , Yücesoy, V., Koç, A., and Çukur, T., 2018, "Semantic Structure and Interpretability of Word Embeddings," IEEE/ACM Transactions on Audio, Speech, and Language Processing, **26**(10), pp. 1769–1779.

[115] Raunak, V., Gupta, V., and Metze, F., 2019, "Effective Dimensionality Reduction for Word Embeddings," *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, I. Augenstein, S. Gella, S. Ruder, K. Kann, B. Can, J. Welbl, A. Conneau, X. Ren, and M. Rei, eds., Association for Computational Linguistics, pp. 235–243, doi: 10.18653/v1/W19-4328.

[116] Li, Y., Yue, X., Xu, Z., Jiang, F., Niu, L., Lin, B. Y., Ramasubramanian, B., and Poovendran, R., 2025, "Small Models Struggle to Learn from Strong Reasoners," 2502.12143, https://arxiv.org/abs/2502.12143

[117] Jain, N., Saifullah, K., Wen, Y., Kirchenbauer, J., Shu, M., Saha, A., Goldblum, M., Geiping, J., and Goldstein, T., 2023, "Bring Your Own Data! Self-Supervised Evaluation for Large Language Models," 2306.13651, https://arxiv.org/abs/2306.13651

[118] Ferrari, A., Spagnolo, G. O., and Gnesi, S., 2017, "Pure: A Dataset of Public Requirements Documents," *Proceedings of the 2017 IEEE 25th International Requirements Engineering Conference (RE 2017)*, IEEE, Lisbon, Portugal, pp. 502–505, doi: 10.1109/RE.2017.29.

[119] Ulrich, K. and Eppinger, S., 2012, *Product Design and Development*, McGraw-Hill/Irwin.

[120] Xiong, W., Liu, J., Molybog, I., Zhang, H., Bhargava, P., Hou, R., Martin, L., Rungta, R., Sankararaman, K. A., Oguz, B., Khabsa, M., Fang, H., Mehdad, Y., Narang, S., Malik, K., Fan, A., Bhosale, S., Edunov, S., Lewis, M., Wang, S., and Ma, H., 2023, "Effective Long-Context Scaling of Foundation Models," 2309.16039, https://arxiv.org/abs/2309.16039

[121] Wang, L., Hu, Y., He, J., Xu, X., Liu, N., Liu, H., and Shen, H. T., 2023, "T-SciQ: Teaching Multimodal Chain-of-Thought Reasoning via Mixed Large Language Model Signals for Science Question Answering," 2305.03453, https://arxiv.org/abs/2305.03453

[122] Zeng, Y., Zhang, H., Zheng, J., Xia, J., Wei, G., Wei, Y., Zhang, Y., and Kong, T., 2023, "What Matters in Training a GPT4-Style Language Model with Multimodal Inputs?" 2307.02469, https://arxiv.org/abs/2307.02469

[123] Thirunavukarasu, A. J., Ting, D. S. J., Elangovan, K., Gutierrez, L., Tan, T. F., and Ting, D. S. W., 2024, "Large Language Models in Medicine: The Potentials and Pitfalls," Annals of Internal Medicine, **177**(2), pp. 210–220, PMID: 38285984.

[124] Clusmann, J., Kolbinger, F. R., Muti, H. S., Carrero, Z. I., Eckardt, J.-N., Laleh, N. G., Löffler, C. M. L., Schwarzkopf, S.-C., Unger, M., Veldhuizen, G. P., et al., 2023, "The future landscape of large language models in medicine," Communications Medicine, **3**(1), p. 141.

[125] Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., and Kasneci, G., 2023, "ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education," Learning and Individual Differences, **103**, p. 102274.

[126] Novikova, J., Dušek, O., Curry, A. C., and Rieser, V., 2017, "Why We Need New Evaluation Metrics for NLG," *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, Association for Computational Linguistics, Copenhagen, Denmark, pp. 2241–2252, doi: 10.18653/v1/D17-1238.

[127] Peng, K., Nisbett, R., and Wong, N., 1997, "Validity Problems Comparing Values Across Cultures and Possible Solutions," Psychological Methods, **2**, pp. 329–344.

[128] Berladir, K., Trojanowska, J., Ivanov, V., and Pavlenko, I., 2022, *Materials Selection in Product Development: Challenges and Quality Management Tools*, pp. 72–86.

[129] Rajak, D. K. and Gupta, M., 2020, "Materials Selection and Design Considerations," *An Insight Into Metal Based Foams: Processing, Properties and Applications*, Springer, Singapore, pp. 53–80.

[130] Aires, R. F. d. F. and Ferreira, L., 2022, "A New Multi-Criteria Approach for Sustainable Material Selection Problem," Sustainability, **14**(18).

[131] Hopkins, A. K., Renda, A., and Carbin, M., 2023, "Can LLMs Generate Random Numbers? Evaluating LLM Sampling in Controlled Domains," *Sampling and Optimization in Discrete Space Workshop, ICML 2023*, Honolulu, Hawaii, USA, https://openreview.net/forum?id=Vhh1K9LjVI

[132] Tjuatja, L., Chen, V., Wu, S. T., Talwalkar, A., and Neubig, G., 2024, "Do LLMs exhibit human-like response biases? A case study in survey design," 2311.04076, https://arxiv.org/abs/2311.04076

[133] Gomes, C. P., Selman, B., and Gregoire, J. M., 2019, "Artificial intelligence for materials discovery," MRS Bulletin, **44**(7), pp. 538–544.

[134] Merchant, A., Batzner, S., Schoenholz, S., Aykol, M., Cheon, G., and Cubuk, E., 2023, "Scaling deep learning for materials discovery," Nature, **624**(7990), pp. 1–6.

[135] Chen, C., Nguyen, D. T., Lee, S. J., Baker, N. A., Karakoti, A. S., Lauw, L., Owen, C., Mueller, K. T., Bilodeau, B. A., Murugesan, V., and Troyer, M., 2024, "Accelerating Computational Materials Discovery with Machine Learning and Cloud High-Performance Computing: from Large-Scale Screening to Experimental Validation," Journal of the American Chemical Society, **146**(29), p. 20009–20018.

[136] Miret, S. and Krishnan, N. M. A., 2024, "Are LLMs Ready for Real-World Materials Discovery?" 2402.05200, https://arxiv.org/abs/2402.05200

List of Figures

1 A visualization depicting the dynamic workflow of the agentic AI used in this work, illustrating how the prompt, LLM, tools, and environment interact through an iterative process of reasoning, thoughts, actions, and observations to produce a final response. . . . 4
2 A flowchart of the LLM-based agent's iterative process when asked to assess the suitability of composite materials for a lightweight spacecraft component. The agent can interact with Wikipedia, determine when and what to search, and use findings to draw conclusions. . . . 5
3 The contributions of model size and prompt type to predicting z-score are represented as coefficients in the linear regression model. The p-values and coefficients show that parallel prompting with standalone LLMs significantly decreases z-score compared to the agentic framework, while other strategies have negligible effects. . . . 5
4 The contributions of model size and prompt type to predicting MAE relative to survey data are represented as coefficients in the linear regression model. The p-values and coefficients indicate that all prompting strategies used with standalone LLMs, except for few-shot prompting, result in a significant decrease in MAE compared to the agentic framework. . . . 5
5 The mean z-scores from experiments across all prompting strategies and model sizes. On average, parallel prompting produces the lowest z-scores, while the other prompt types yield similar values. Average z-scores remain relatively consistent across model sizes. . . . 7
6 The MAE relative to survey data across all prompting strategies and model sizes. The greatest error is observed using the agentic and few-shot methods. . . . 7
7 The scores generated by the LLM-based agent across all design, criteria, and material combinations from our experiments, grouped by model size, with median values labeled. . . . 8
8 The scores generated by standalone LLMs across all design, criteria, and material combinations from our experiments, grouped by model size, with median values labeled. . . . 8
9 The scores across all design, criteria, and material combinations from our experiments, grouped by prompt type, with median values labeled. . . . 8
10 The mean number of completion tokens used for each iteration of the agent workflow, categorized by model size, with error bars representing standard deviation. The 32B model required the most tokens, followed by the 1.7B model, with the 4B model using the fewest tokens. . . . 8
11 The results of the word embeddings analysis after PCA, color-coded based on model size, reveal three main clusters around the system prompt embeddings. This suggests that the models generally generate query responses that align with the system prompt. . . . 9
12 The average number of reruns per prompt, both overall and for successful prompts, grouped by model size. The 8B model shows the highest mean reruns in both categories, while other model sizes required much fewer reruns. . . . 10
13 The average number of total and unique queries for successful prompts, categorized by model size. The 4B model generated the highest number of queries, while the 1.7B model produced the fewest. Most queries were unique across all model sizes, with the 8B model showing some repeated searches. . . . 10
14 Qwen 2.5 mean z-scores. . . . 10
15 Qwen 2.5 MAE. . . . 10
16 Qwen 2.5 agentic AI score distributions grouped by model size. . . . 11
17 Qwen 2.5 standalone LLM score distributions grouped by model size. . . . 11
18 Qwen 2.5 score distributions grouped by prompt type. . . . 11
19 Qwen 2.5 mean number of completion tokens. . . . 11

List of Tables

1 The percentage of responses omitted by our program across all model sizes, categorized by whether the final answer tool was called and whether the last agent iteration resulted in an error. Omissions occurred only for the 8B model, with 22 omissions total, and no omissions for any other model size or prompt type. . . . 10