



Evaluation of the Effectiveness of Prompts and Generative AI Responses

Ajay Bandi¹(✉) and Ruida Zeng²

¹ School of Computer Science and Information Systems, Northwest Missouri State University, 800 University Dr, Maryville, MO 64468, USA

AJAY@nwmissouri.edu

² Department of Computer Science, Brown University, Providence, RI 02912, USA

Abstract. Our paper proposes a comprehensive framework to evaluate the effectiveness of prompts and the corresponding responses generated by Generative Artificial Intelligence (GenAI) systems. To do so, our evaluation framework incorporates both objective metrics (accuracy, speed, relevancy, and format) and subjective metrics (coherence, tone, clarity, verbosity, and user satisfaction). A sample evaluation is performed on prompts sent to Gemini and ChatGPT GenAI models. Additionally, our evaluation framework employs various feedback mechanisms, such as surveys, expert interviews, and automated reinforcement learning from human feedback (RLHF), to iteratively enhance the performance and reliability of GenAI models. By providing a holistic approach to evaluating and improving prompt-response effectiveness, our evaluation framework contributes to the development of more credible and user-friendly AI systems.

Keywords: Generative AI (GenAI) · Prompt Engineering · Evaluation Framework · Reinforcement Learning from Human Feedback (RLHF)

1 Introduction

Generative Artificial Intelligence (GenAI) represents a significant advancement in the field of AI in recent years, and is looking to further revolutionize various aspects of internet, technology, and even humanity [4].

Nevertheless, despite the consensus that Generative AI (GenAI) will play a pivotal role in shaping the future with incredible transformative potentials, there are many limitations in the current GenAI models [4]. A notable concern here is “hallucinations”, where AI confidently gives out a factually incorrect response [29]. This is very concerning since the GenAI will also word the response to appear truthful and credible, and it rarely gives out a “Sorry I don’t know...” response since it heavily prefers answers (regardless of correctness) to non-answers [29]. This is because at this time, no GenAI models actually understand the responses that it provide to the users, they just use probabilistic statistical model to select the correct words from the massive natural language database, and regurgitate those words back to the users [6, 22]. In fact, researchers have coined the term “stochastic parrot” to criticize the phenomenon

of GenAI not genuinely understanding the underlying meaning of the words they read and write [6]. Sometimes, users are not satisfied with GenAI’s responses due to a variety of reason, ranging from inaccurate information in the response, to the answer simply not being the one that user is looking for and want to see [16].

There are many ways to address this issue, such as continual improvement of the GenAI models by adding more training data, incorporating more advanced machine learning methods and techniques, and using better foundational large language models [5]. In addition, researchers have discovered that writing better prompts can often times result in better responses from the GenAI [9, 32].

In order to improve the GenAI responses, our paper propose a holistic and comprehensive evaluation framework to evaluate the prompt-response effectiveness by measuring the performance of the GenAI responses to align with objective expectations and user experiences. While most of the framework focuses on the GenAI responses, it also considers metrics that affect both prompts and responses, such as semantic correctness, clarity, and verbosity. With proper and uniformed evaluations, methods such as reinforcement learning can then be used to enhance the overall effectiveness and reliability of the GenAI systems [19].

The remainder of the paper is organized as follows: Section 2 will contain works related to prompt engineering and motivation for the holistic evaluation framework; Sect. 3 will contain our proposed framework in detail, including objective metrics, subjective metrics, and feedback mechanisms; Sect. 4 will contain a sample evaluation of our proposed framework using Gemini and ChatGPT; Sect. 5 will contain our conclusion and future works.

2 Related Work

In recent years, prompt engineering has been an important topic for many academic researchers, and many work has been done on how writing better prompts can result in better responses from the GenAI, with most of the work focusing on the ChatGPT model specifically [9, 11, 30, 32]. Other GenAI models such as Gemini were also explored for prompt engineering [26]. Overall, recent research has pointed out that prompt engineering has many applications such as in academic writing, medical education, and legal research where factually accurate responses are crucial [11, 13, 18, 20, 21]. However, most of the past work uses various different, sometimes arbitrary, evaluation to determine whether or not the GenAI responses are considered to be good or not, which is why our paper presents a holistic evaluation framework to assist with the evaluation of prompt-response effectiveness.

3 Proposed Framework

The proposed framework for the evaluation of effectiveness for prompts and their responses shown in Fig. 1 aims to systematically assess and enhance the quality of interactions between users and GenAI. It incorporates both objective and subjective metrics to provide a comprehensive evaluation. Objective metrics

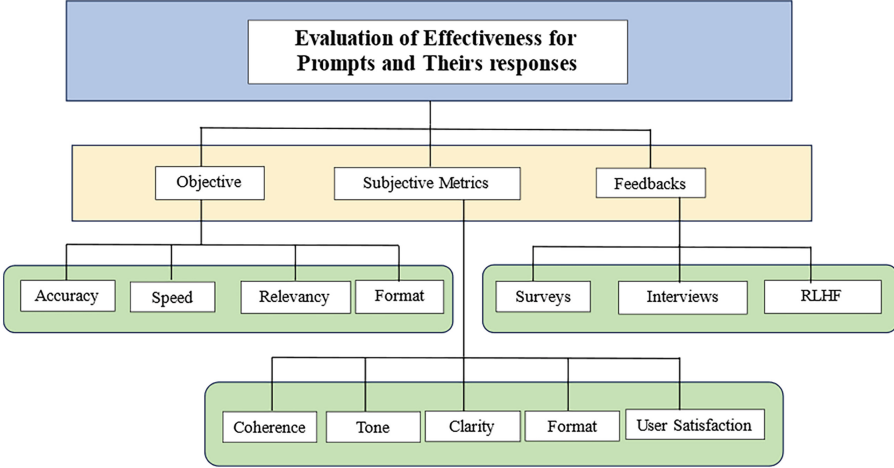


Fig. 1. Framework for Evaluating Prompt-Response Effectiveness

include accuracy, speed, relevancy, and format, ensuring that responses meet the standards. Subjective metrics focus on coherence, tone, clarify, verbosity, and user satisfaction, focusing more on the nuanced human experience of the interaction.

Additionally, we also introduce several feedback mechanisms, allowing for iterative improvements based both the objective and subjective metrics mentioned. This holistic approach ensures that the evaluation framework not only measures the responses both quantitatively and qualitatively with feedbacks that are designed for continuous improvement and enhancement of the GenAI.

3.1 Objective Metrics

Objective metrics provide quantifiable measures of performance, allowing for a clear and unbiased assessment of the GenAI system’s capabilities. These metrics are essential for evaluating the technical aspects of the prompts and responses, ensuring that the system meets specific standards of correctness, quality, and efficiency. The key objective metrics here include accuracy, speed, relevancy, and format, as shown in Fig. 2.

Accuracy. Accuracy refers to both the factual correctness and the semantic correctness of the information provided.

It is crucial that the GenAI system delivers factually correct and precise answers that are not results of AI hallucinations to user queries [29]. Incorrect answers, especially incorrect answers delivered by the GenAI using confident languages, can be detrimental depending on the context and may cause legal and ethical concerns [21, 29]. Semantic correctness is also important and ensures that the response is in the intended languages and not gibberish (unless specifically requested), and conveys the intended meaning accurately and logically.

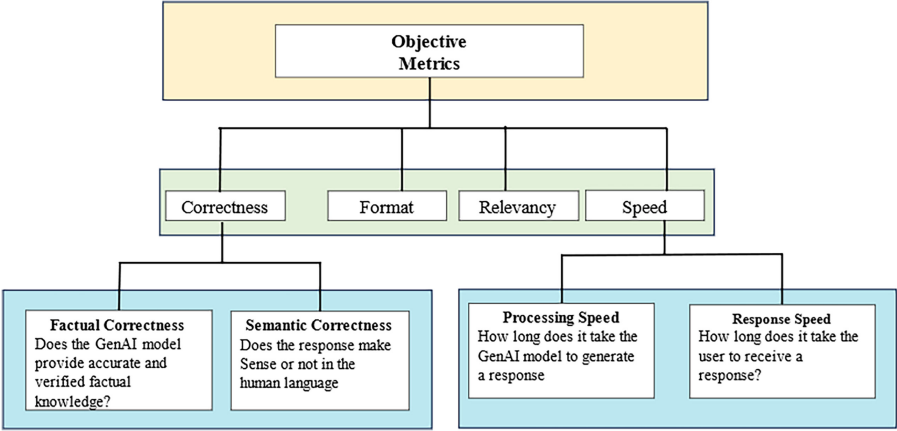


Fig. 2. Objective Metrics: Accuracy, Speed, Relevance, Format Assessment

It is worth nothing that factual incorrectness in the prompt itself might lead to hallucinations, and semantic incorrectness, such as stating the prompt in a language not supported by GenAI, would lead to GenAI not understanding the prompt at all [29].

Generally speaking, factual accuracy can be measured based on how closely the responses align with the established knowledge, data, or observable reality. This type of correctness is often verifiable through evidence, research, or documentation. For example, bogus law cases submitted to court by lawyers as part of their GenAI assisted legal research are clearly factually incorrect and inaccurate [21]. For AI generated source codes, this can be measured by whether or not the source codes correctly achieve the desired functionalities or produce the desired outputs when executed. For outputs such as images or voices, assessing the accuracy are a bit trickier. Semantic accuracy, on the other hands, can be measured based on whether or the generated languages use the correct grammar.

For AI generated source codes, semantic correctness can be measured by whether the source codes can compile successful in the given programming language. Unlike textual responses, visual and auditory outputs are inherently more subjective and difficult to measure their accuracy using objective metrics. For example, an image generated to match a prompt might be accurate in terms of content but the style and production may not meet user expectations.

Overall, accuracy is an extremely important metric and high accuracy ensures that users can trust the information provided by the system, which is essential for maintaining GenAI’s credibility and user confidence in the responses [2].

Speed. The speed metric includes two major aspects: processing speed and response speed. Processing speed measures how long it takes the GenAI model to generate a response, and response speed measures how long it takes for the user to receive the response [1].

As a standalone metric, the higher the speed (both processing and response) the better. High speed GenAI can help speed up critical decision making and information retrieval in time sensitive situations for the users.

Relevancy. Relevancy can be measured by whether the responses answer the correct question, and not a completely unrelated question, or even a related question that does not address the main point of the prompt. For example, a question about `std::array` in C++ should not focus on C++ `std::vector`, even though they are similar constructs [25]. Another way relevancy can be defined would be how “up to date” the responses are, particularly in fields where knowledge is rapidly advancing, such as in science, technology, and medicine. An example of this would be the COVID-19 virus that caused a global pandemic a few years ago. Our knowledge and understanding about the virology of COVID-19 is now built upon a few years of solid worldwide medical research, and GenAI responses that align with information from contemporary sources is definitively more relevant than information from sources from 2021, even if both sources come from reliable medical professionals [8]. Needless to say, the more relevant the responses are with respect to the prompts, the more effective the responses are.

Format. Last but not least, we can measure whether or not the responses are delivered in the correct format. Some of the formats available include short sentences, paragraphs, bullet points, tables, equations, source code blocks, images, and even voices [1]. If the prompt asks the GenAI to generate an image, the GenAI’s response should be in the image format, and not in the paragraph format describing the image textually. And if the prompt asks the GenAI to describe an algorithm that sorts a list, the GenAI’s response should be in paragraph or bullet points format, rather than a code block that sorts a list using the built-in “`sort()`” method in Python [27].

Note that in the case where multiple response formats are acceptable, we can score the responses based on whether or not it is delivered in the optimal format. However, there could be cases where the “optimal format” is decided subjectively as opposed to objectively.

3.2 Subjective Metrics

Subjective metrics provide a qualitative assessment of the GenAI system’s performance, focusing on the end user’s experience aspects of human-computer interactions. These metrics capture the human elements of communication, such as how the responses are perceived and understood by users. It also evaluates whether the humans are satisfied with the interactions with the AI regardless of the accuracy or efficiency of the responses - although those factors can play a substantive part in the user satisfaction metric, different users might perceive and value them differently. By evaluating coherence, tone, clarity, verbosity, and user satisfaction, subjective metrics help ensure that the responses overall create a good experience for the users.

Coherence. Coherence refers to the logical sense and linguistic consistency of the response on its own without taking into other objective metrics. We can measure coherence based on the response’s logical structure and whether or not it is easy to follow for the particular user. A coherent response would be a response

that has clear connection between different parts and the overall message is cohesive. A response that would score poorly in the coherence category would be a response that abruptly jumps from one point to another with no transition, or presents random information throughout the response without mentioning how the information relate to the overall message that the response is trying to convey.

Tone. Tone is an important metric since an appropriate tone can make the responses seem more human-like, and GenAI should be able to adapt to different tones based on the context and the audience. Using deep learning techniques, GenAI models are able to detect user's emotions from the prompts and take that into consideration when generating responses [7]. If the prompt is just a casual question, then GenAI should use a more relaxed and friendly tone, and make the conversation more engaging. If the prompt is a professional inquiry, then GenAI should respond formally without using any slang. If the prompt addresses a sensitive topic, then GenAI should deliver the response with an empathetic and supportive tone, with lots of respect and consideration to the user's emotional feelings [31].

Clarity. Clarify refers to how understandable and clear the prompts and responses are. Clarify in prompts mean that the users should provide clear and specific instructions so that the GenAI understand exact what the users want [33]. With better prompts, the generated responses more accurate and relevant [33]. Clarity in responses means that GenAI should avoid using professional jargons or overly complicated technical terms unless they are essential to the responses. In cases where they are necessary, appropriate explanations and definitions should be provided to avoid confusions.

Verbosity. Verbosity refers to the conciseness of the response. Users generally have the options to specify the verbosity by explicitly requesting responses of a specific length or by asking the model to provide more or less information [1,9]. However, when unspecified, we want GenAI to provide useful information with sufficient details that answers the prompts fully with enough supporting evidence and reasoning, without omitting any important details, and without being overly verbose. Lengthy descriptions with unnecessary elaboration, excessive explanation for the obvious, or repetitions can cause frustration for the users, especially if they are purposefully looking for a concise and straightforward answer.

User Satisfaction. For our framework, "user satisfaction" serves as an umbrella term that encompasses all the other subtle subjective metrics and factors not already mentioned that contribute to the overall user experience and how pleased the users are with the prompt-response interaction. This can vary between different users and include things like preferences, where one user simply like the response more than another user, even when it is the exact same

response to the exact same prompt. As mentioned before, sometimes even factors that are usually considered to be objective metrics can play a substantive part in the user satisfaction metric, since different users might perceive and value each objective metric differently.

3.3 Feedback

Feedback mechanisms are crucial for the continuous improvement and enhancement of the GenAI system [24, 28]. In our framework, multiple feedback channels are employed to measure different aspects of user interaction and system performance to achieve comprehensive insights. This section outlines the key feedback mechanisms: surveys, interviews, and reinforcement learning from human feedback (RLHF).

Surveys. Surveys provided to the users after each response are a great way to collect instant feedback from the users about the interactions and measure the subjective metrics of the GenAI system. For example, the survey if the user like the responses or not (for user satisfaction metric), and also the user the rate the coherence, tone, clarity, verbosity of the responses from a scale of 1 to 10.

Interviews. Interviews with experts with knowledge in the fields can help measure the objective metrics of the GenAI responses. Some of the objective metric such as accuracy and relevancy may require academic or professional experts to verify how closely the responses align with the established knowledge, data, and observed reality, especially if the ground truth can not be easily determined by an average person. These kinds of interviews require that the experts involved are indeed experts and can be trusted to make a judgmental decision on the responses.

We can also ask the regular users about the subjective metrics in the interviews as well or even inquire users about potential areas for improvements in general.

RLHF. The results from both the surveys and the interviews can be converted into a “score” and fed into a model that uses reinforcement learning from human feedback (RLHF) techniques that use reward models to iteratively improve the prompt-response effectiveness overtime [14, 17, 24, 28].

However, sometimes GenAI’s surveys with long and excessive multiple choice questions can lead to survey fatigue and frustrate the user, causing the users to quickly select an inaccurate response and the users might spam click a random response that is the most convenient for them ergonomically, similar to how people might select January 1 as their birthday when registering an account online [15]. This will lead to the survey results being inaccurate and counterproductive for improving the prompt-response effectiveness.

Instead, we propose an automated feedback process with a chain of large language models (LLMs) and Non-LLMs that are RLHF for the user’s usage

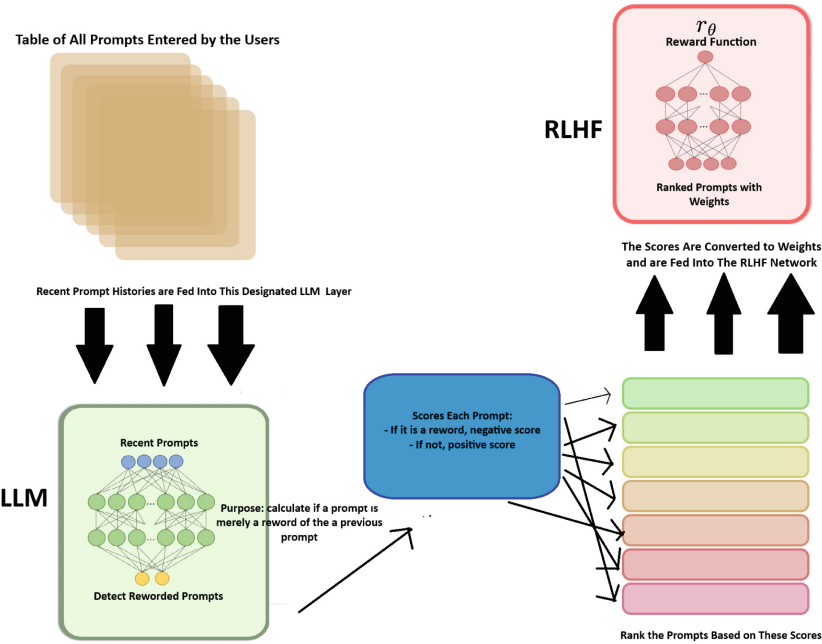


Fig. 3. Integrated Process of LLM with RLHF Utilizing User Feedback for Continuous Performance Enhancement

over time. One simple approach of doing this practically would be keeping a table of all the prompts entered by the users. Then, a layer of LLMs is used to calculate if a prompt is merely a reword of the a previous prompt by analyzing recent prompt history in the table. If so, this indicates the previous response corresponding to the previous prompt is a failed and unproductive response, and receives a negative score in the RLHF. Otherwise, it receives a positive score in the RLHF. Although this is done under the naive assumption that not rewording a prompt means that the response are “okay” and avoids the frustration of choice prompts. This method is shown in Fig. 3 [10].

Additionally, we can also use the same tools alongside a human sampling model to automate the scoring of expert-level prompts and responses by assessing how knowledgeable the users are with respect to the topic and complexity [3]. A secondary LLM can be used to categorize the questions by topic and complexity, then the human sampling framework is used to assess the user’s expertise and historical performances. The GenAI can also measure the accuracy of user’s judgment on information with established ground truth. If the user’s decision on the accuracy of a response differs from the established ground truth, then the user gets a negative score for the topic and complexity in the human sampling RLHF, and vice versa. This way, we can determine which user’s feedbacks are more valuable and should be associated with a higher confidence level score.

Further research could explore the use of more advanced RLHF to automate feedback mechanisms, eliminating the need for surveys and interviews.

4 Evaluation of Framework

In order to demonstrate an evaluation of the framework, we will be evaluating the GenAI responses to carefully designed prompts related to computer science for simplicity purpose, using our proposed framework. For our purpose, Gemini 1.0 and ChatGPT 3.5 will be used on the same sets of standardized prompts to ensure that the GenAI responses are comparable [12, 23]. Relevant objective and subjective metrics will be collected and evaluated.

As a demonstration, we input the same prompt to both Gemini 1.0 and ChatGPT 3.5 for evaluation: **Please generate a short, concise, functioning Python code that demonstrates how recursion works.**

Below are the verbatim generated responses from the two GenAI models:

4.1 Gemini

Absolutely, here's a concise Python code that demonstrates recursion by calculating the factorial of a number:

```
def factorial(n):
    """
    This function calculates the factorial of a number using recursion.
    """
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)

# Print the factorial of 5
print(factorial(5))
```

In this code, the *factorial* function takes a number n as input. It has two parts:

- Base Case: If n is equal to 0, the function returns 1. This is the stopping condition for the recursion. Factorial of 0 is defined as 1.
- Recursive Case: If n is not 0, the function returns n multiplied by the factorial of $n - 1$. This is the recursive call where the function calls itself with a smaller value of ' n '.

When you call *factorial*(5), the function breaks down the problem into smaller subproblems:

- $factorial(5) = 5 * factorial(4)$
- $factorial(4) = 4 * factorial(3)$
- ...
- $factorial(1) = 1 * factorial(0)$ (base case reached)

The recursion continues until it reaches the base case (factorial of 0), then the results are returned back up the chain, calculating the factorial of 5.

This is a simple example of recursion, but it demonstrates the core concept of a function calling itself to solve a problem by breaking it down into smaller, similar subproblems.

4.2 ChatGPT

Sure, here's a simple example of a recursive function in Python that calculates the factorial of a number:

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

# Example usage:
number = 5
print(f"The factorial of {number} is {factorial(number)}")
```

In this code:

- The factorial function is defined to calculate the factorial of a number n .
- The base case (if $n == 0$;) returns 1, which is necessary to stop the recursion.
- The recursive case (else:) computes $n * factorial(n - 1)$ which calls the *factorial* function again with a smaller argument until it reaches the base case.

When you run this code with $number = 5$, it will output:

The factorial of 5 is 120

This demonstrates how recursion works by breaking down a problem into smaller, simpler instances of the same problem until a base case is reached.

4.3 Evaluation of Gemini and ChatGPT

Objective Metrics. Both Gemini and ChatGPT seems to be accurate in terms of factual correctness and semantic correctness. Both models generate relevant Python codes as instructed that demonstrate recursion, and both generated codes compiled and ran with the expected outputs and no errors. In terms of speed, from sending the prompt to the response fully finishing, ChatGPT was a lot faster. Gemini took 8.38 s, and ChatGPT only needed 3.69 s. For the format, both models generated both source codes and textual paragraphs explaining the source codes in detail, which do not fully align with what the prompt asked for, since the prompt only asked for “Python code” and not explanation in text format.

Subjective Metrics. Both Gemini and ChatGPT delivered a cohesive and logical response with professional tone that one would normally see in academia. Both responses are clear with no ambiguity. However, both responses do not satisfy the “verbosity” criteria fully. The prompt asked for “short, concise, functioning Python code”. Although the code portions are short, both GenAI models generated very detailed and verbose explanation about how the recursive function work, including the base case and the recursive case, and also examples of what the output would look like once executed.

4.4 Survey Questionnaire

To better demonstrate our framework’s performance on GenAI prompt-response effectiveness on prompts related to computer science (again for simplicity purpose), we have come up with more potential prompts that can be used for evaluation via our framework.

- Please generate a Python code snippet that demonstrates how a binary search algorithm works on a sorted list.
- Write a concise C++ program to implement the quicksort algorithm, ensuring it is clear and efficient.
- Generate a Java function that utilizes a stack data structure to evaluate a postfix expression (Reverse Polish Notation).
- Create a JavaScript code example that illustrates the use of a breadth-first search (BFS) algorithm in a graph.
- Provide a short C# script that demonstrates how to implement a linked list, including insertion and deletion of nodes.
- Write a Java function to find the shortest path in a weighted graph using Dijkstra’s algorithm.
- Generate a C program that implements the merge sort algorithm and includes a brief explanation of its time complexity.
- Create a Ruby code snippet that demonstrates how to use a hash table to store and retrieve data efficiently.
- Write a MATLAB function that performs matrix multiplication and explain its computational complexity.
- Provide a Swift example of how to use dynamic programming to solve the knapsack problem.
- Generate a concise Kotlin program that demonstrates how to implement a trie (prefix tree) for storing strings.
- Write a Rust function that uses recursion to solve the Tower of Hanoi problem and include an explanation of the solution.
- Create a JavaScript script that demonstrates the use of depth-first search (DFS) in finding connected components in an undirected graph.
- Provide a short Go code example that shows how to implement a binary search tree (BST) with insert and search operations.
- Write a Haskell function that implements the Floyd-Warshall algorithm for finding shortest paths in a weighted graph.

Note that conducting these surveys and interviews require human subject participation, and hence approval from Institutional Review Boards (IRBs) at universities and institutions.

5 Conclusion and Future Works

In this paper, we presented holistic and comprehensive framework for evaluating the effectiveness of prompts and responses. It integrates both objective metrics and subjective metrics to ensure a balanced assessment for both the technical performance of the AI and the nuanced human interaction experience with the users. Through the application of our evaluation framework (as demonstrated in our sample evaluation using Gemini and ChatGPT), GenAI researchers and developers can address the current limitations in GenAI and refine the AI generated content by better understanding the strengths and weaknesses of their respective GenAI model according to our evaluation framework. We can incorporate multiple feedback mechanisms that enable GenAI researchers and developers to use the evaluated metrics directly to improve GenAI through RLHF.

In order to fully evaluate the framework, future research work can be done to evaluate the GenAI responses to carefully designed prompts in all aspects and not just computer science, using our proposed framework. For our purpose, we suggest using two different GenAI models on the same sets of standardized prompts to ensure that the GenAI responses are comparable. Relevant objective and subjective metrics collected and measured using both interviews and surveys on different users as described in Sect. 3 to fully evaluate and improve the prompt-response effectiveness. Ultimately, our goal is to contribute to the development of GenAI systems that are not only technologically advanced but also trusted and valued by their users.

References

1. Ali, I., Ahmad, S., Usama, M., Muhammad, M.: ChatGPT: a comprehensive review on background, applications, key challenges, bias, ethical implications, limitations, and future directions. *J. Innov. Digit. Ecosyst.* **3**, 121–154 (2023)
2. Amoozadeh, M., et al.: Trust in generative AI among students: an exploratory study. arXiv preprint [arXiv:2310.04631v2](https://arxiv.org/abs/2310.04631v2) (2023)
3. Argyle, L.P., Busby, E.C., Fulda, N., Gubler, J.R., Rytting, C., Wingate, D.: Out of one, many: using language models to simulate human samples. arXiv preprint [arXiv:2209.06899](https://arxiv.org/abs/2209.06899) (2022)
4. Bandi, A., Adapa, P.V.S.R., Kuchi, Y.E.V.P.K.: The power of generative AI: a review of requirements, models, input–output formats, evaluation metrics, and challenges. *Future Internet* **15**(8), 260 (2023)
5. Bandi, A., Kagitha, H.: A case study on the generative AI project life cycle using large language models. In: Bandi, A., Hossain, M., Jin, Y. (eds.) *Proceedings of 39th International Conference on Computers and Their Applications, EPiC Series in Computing*, vol. 98, pp. 189–199. EasyChair (2024)
6. Bender, E.M., Gebru, T., McMillan-Major, A., Shmitchell, S.: On the dangers of stochastic parrots: Can language models be too big? In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (FAccT 2021)*, pp. 610–623, New York, NY, USA (2021). Association for Computing Machinery
7. Bharti, S.K., et al.: Text-based emotion recognition using deep learning approach. *Comput. Intell. Neurosci.* **2022**, 2645381 (2022)

8. Centers for disease control and prevention. CDC museum Covid-19 timeline 2023. Accessed 11 June 2024
9. Ekin, S.: Prompt engineering for ChatGPT: a quick guide to techniques, tips, and best practices. TechRxiv, Version 2 (2023)
10. Hugging Face: Illustrating reinforcement learning from human feedback (RLHF). Hugging Face Blog (2023)
11. Giray, L.: Prompt engineering with ChatGPT: a guide for academic writers. *Ann. Biomed. Eng.* **51**(12), 2629–2633 (2023)
12. Google. Google Gemini API documentation (2023). <https://ai.google.dev/gemini-api/docs>. Accessed 12 June 2024
13. Heston, T.F., Khun, C.: Prompt engineering in medical education. *Med. Educ. J.* **2**(3), 19–25 (2023)
14. Huyen, C.: RLHF: reinforcement learning from human feedback (2023). Accessed 11 June 2024
15. Jeong, D., Aggarwal, S., Robinson, J., Kumar, N., Spearot, A., Park, D.S.: Exhaustive or exhausting? Evidence on respondent fatigue in long surveys. *J. Dev. Econ.* **161**, 102992 (2022)
16. Kim, Y., Lee, J., Kim, S., Park, J., Kim, J.: Understanding users' dissatisfaction with ChatGPT responses: types, resolving tactics, and the effect of knowledge level. In: *Proceedings of the ACM on Human-Computer Interaction*, vol. 7, no. CSCW2 (2023)
17. Kirk, R., et al.: Understanding the effects of RLHF on LLM generalisation and diversity. arXiv preprint [arXiv:2310.06452](https://arxiv.org/abs/2310.06452) (2023)
18. Meskó, B.: Prompt engineering as an important emerging skill for medical professionals: tutorial. *J. Med. Internet Res.* **25**(1), e50638 (2023)
19. Mungoli, N.: Exploring the synergy of prompt engineering and reinforcement learning for enhanced control and responsiveness in chat gpt. *J. Electr. Electron. Eng.* **2**(3), 201–205 (2023)
20. Naseem, U., Bandi, A., Raza, S., Rashid, J., Chakravarthi, B.R.: Incorporating medical knowledge to transformer-based language models for medical dialogue generation. In: *Proceedings of the 21st Workshop on Biomedical Language Processing*, pp. 110–115 (2022)
21. Neumeister, L.: Lawyers submitted bogus case law created by ChatGPT. a judge fined them \$5,000. *AP News* (2023)
22. University of Michigan: Prompt literacy (2024). <https://genai.umich.edu/resources/prompt-literacy>. Accessed 12 June 2024
23. OpenAI. Openai platform: Text generation guide (2023). <https://platform.openai.com/docs/guides/text-generation>. Accessed 12 June 2024
24. Ouyang, L., et al.: Training language models to follow instructions with human feedback. arXiv preprint [arXiv:2203.02155](https://arxiv.org/abs/2203.02155) (2022)
25. Stack Overflow. std::vector versus std::array in c++ (2023). Accessed 11 June 2024
26. Patel, H., Shah, K.A., Mondal, S.: Do large language models generate similar codes from mutated prompts? A case study of Gemini Pro. In: *Proceedings of the ACM International Conference on the Foundations of Software Engineering (FSE)* (2024)
27. Python software foundation. Sorting HOWTO (2023). Accessed 11 June 2024
28. Jack, W.R., et al.: Scaling language models: methods, analysis & insights from training gopher. arXiv preprint [arXiv:2112.11446](https://arxiv.org/abs/2112.11446) (2021)
29. Shan, M., Zhang, W., Li, T., Wang, H., Sun, L.: Artificial hallucinations in ChatGPT: implications in scientific writing. *Cureus* **15**(2), e35021 (2023)

30. Short, C.E., Short, J.C.: The artificially intelligent entrepreneur: Chatgpt, prompt engineering, and entrepreneurial rhetoric creation. *J. Bus. Ventur. Insights* **19**, e00388 (2023)
31. Wang, J., et al.: The good, the bad, and why: unveiling emotions in generative AI. arXiv preprint [arXiv:2312.11111](https://arxiv.org/abs/2312.11111) (2023)
32. White, J., et al.: A prompt pattern catalog to enhance prompt engineering with ChatGPT. *Artif. Intell. Rev.* **5**(2), 101–110 (2023)
33. Fangjun, Yu., Quartey, L., Schilder, F.: Exploring the effectiveness of prompt engineering for legal reasoning tasks. *Find. Assoc. Comput. Linguist. ACL* **2023**, 13582–13596 (2023)