EvalLM: Interactive Evaluation of Large Language Model Prompts on User-Defined Criteria

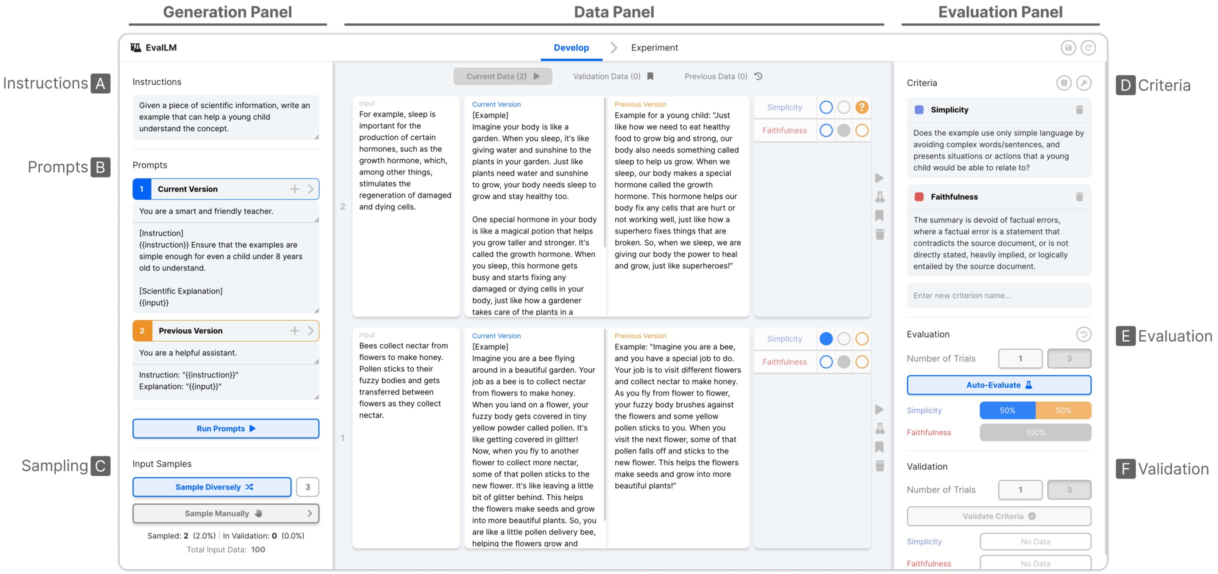
Tae Soo Kim
taesoo.kim@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Yoonjoo Lee
yoonjoo.lee@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Jamin Shin
jayshin.nlp@gmail.com
NAVER AI Lab
Seongnam, Republic of Korea

Young-Ho Kim
yghokim@younghokim.net
NAVER AI Lab
Seongnam, Republic of Korea

Juho Kim
juhokim@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea



Instructions

Given a piece of scientific information, write an example that can help a young child understand the concept.

Prompts

1 Current Version

You are a smart and friendly teacher.

[Instruction]
{{instruction}} Ensure that the examples are simple enough for even a child under 8 years old to understand.

[Scientific Explanation]
{{input}}

2 Previous Version

You are a helpful assistant.

Instruction: "{{instruction}}"
Explanation: "{{input}}"

Input Samples

Sample Diversely: 3

Sampled: 2 (2.0%) | In Validation: 0 (0.0%)
Total Input Data: 100

2

Input
For example, sleep is important for the production of certain hormones, such as the growth hormone, which, among other things, stimulates the regeneration of damaged and dying cells.

Current Version
[Example]
Imagine your body is like a garden. When you sleep, it's like giving water and sunshine to the plants in your garden. Just like plants need water and sunshine to grow, your body needs sleep to grow and stay healthy too.

One special hormone in your body is like a magical potion that helps you grow taller and stronger. It's called the growth hormone. When you sleep, this hormone gets busy and starts fixing any damaged or dying cells in your body, just like how a gardener takes care of the plants in a

Previous Version
Example for a young child: "Just like how we need to eat healthy food to grow big and strong, our body also needs something called sleep to help us grow. When we sleep, our body makes a special hormone called the growth hormone. This hormone helps our body fix any cells that are hurt or not working well, just like how a superhero fixes things that are broken. So, when we sleep, we are giving our body the power to heal and grow, just like superheroes!"

Simplicity ○ ○ ?
Faithfulness ○ ● ○

1

Input
Bees collect nectar from flowers to make honey. Pollen sticks to their fuzzy bodies and gets transferred between flowers as they collect nectar.

Current Version
[Example]
Imagine you are a bee flying around in a beautiful garden. Your job as a bee is to collect nectar from flowers to make honey. When you land on a flower, your fuzzy body gets covered in tiny yellow powder called pollen. It's like getting covered in glitter! Now, when you fly to another flower to collect more nectar, some of that pollen sticks to the new flower. It's like leaving a little bit of glitter behind. This helps the flowers make seeds and grow into more beautiful plants. So, you are like a little pollen delivery bee, helping the flowers grow and

Previous Version
Example: "Imagine you are a bee, and you have a special job to do. Your job is to visit different flowers and collect nectar to make honey. As you fly from flower to flower, your fuzzy body brushes against the flowers and some yellow pollen sticks to you. When you visit the next flower, some of that pollen falls off and sticks to the new flower. This helps the flowers make seeds and grow into more beautiful plants!"

Simplicity ● ○ ○
Faithfulness ○ ● ○

Criteria

Simplicity

Does the example use only simple language by avoiding complex words/sentences, and presents situations or actions that a young child would be able to relate to?

Faithfulness

The summary is devoid of factual errors, where a factual error is a statement that contradicts the source document, or is not directly stated, heavily implied, or logically entailed by the source document.

Evaluation

Number of Trials: 1 | 3

Simplicity: 50% | 50%
Faithfulness: 100%

Validation

Number of Trials: 1 | 3

Simplicity: No Data
Faithfulness: No Data

Figure 1: EvalLM is composed of three main panels: generation, data, and evaluation. In the generation panel, the user can compose the overall instructions for their task (A), two prompt templates they want to compare (B), and sample inputs from their dataset (C). To evaluate outputs, the user first defines their criteria set (D) and can see an overview of evaluation results (E). If the user has added samples to their validation set, they can also check the accuracy of the evaluations in this panel (F). The data panel shows a series of rows, where each row presents an input sample, the outputs generated on this input, and the evaluation results for these outputs.

ABSTRACT

By simply composing prompts, developers can prototype novel generative applications with Large Language Models (LLMs). To refine prototypes into products, however, developers must iteratively revise prompts by evaluating outputs to diagnose weaknesses. In this work, we present `EvalLM`, an interactive system for iteratively refining prompts by evaluating multiple outputs on user-defined criteria. By describing criteria in natural language, users can employ the system's LLM-based evaluator to get an overview of where prompts excel or fail, and improve these based on the evaluator's feedback. We discuss directions for future work: (1) investigating how to audit LLM-based evaluators, (2) supporting criteria design, and (3) constructing criteria hierarchies to compile evaluation results.

KEYWORDS

Large Language Models, Natural Language Generation, Evaluation, Human-AI Interaction

1 INTRODUCTION

Large Language Models (LLMs) have catalyzed the creation of a wide array of novel applications. Composed of billions of parameters and trained on billions of tokens, LLMs can interpret a natural language description of a task, a **prompt**, and generate coherent human-like outputs for diverse purposes [2, 13, 16] (e.g., summarization [22], dialogue [21], story writing [3]). By composing a prompt,

1st HEAL Workshop at CHI Conference on Human Factors in Computing Systems, May 12, Honolulu, HI, USA
2024.

developers and researchers (i.e., prompt designers) can guide LLMs to perform novel tasks that satisfy desired requirements and support specific application settings. For example, HCI researchers have leveraged LLMs to ideate possible journalistic angles for a given event [17], generate questions to quiz children about information they learned [11], or simplify research papers into plain language [1].

Although prompt designers can easily bootstrap AI-based applications by simply composing a prompt, developing a prototype into a polished application that consistently produces high-quality outputs requires more dedicated effort. As LLMs are non-deterministic and even partial changes in a prompt can significantly influence generated outputs [12, 15], designers need to iterate on their prompts multiple times to achieve satisfactory results [7, 13, 20, 22, 24, 25]. In this iterative process, designers test their prompt with sample inputs (e.g., paragraphs to summarize), inspect the generated outputs to identify areas for improvement, revise their prompts (e.g., change structure, wording, content), and repeat.

However, as designers are increasingly adopting LLMs for novel and more open-ended tasks, the evaluation of outputs and, consequently, prompts becomes significantly more challenging. Specifically, open-ended tasks require outputs that satisfy certain subjective qualities, but designing automatic metrics that can adequately encode and measure these subjective qualities is challenging [4]. While NLP researchers have designed or trained automatic metrics that can approximately measure subjective qualities of text [10, 18, 19, 26], as the tasks of prompt designers can be novel, designing new metrics would require excessive effort. While prompt designers can alternatively conduct human evaluations where human annotators or experts assess the quality of outputs [6], the significant cost involved would be impractical during early development stages when prompts must be iterated on quickly and frequently. As a result, prompt designers may resort to manually evaluating outputs themselves—a time-consuming and effortful task.

To address these challenges, we propose EvalLM [9] (Fig. 1), an interactive system that supports prompt iterations by facilitating the evaluation of outputs on user-defined and application-specific criteria. In the interface, a designer composes two prompts that they want to evaluate and compare, and then generates outputs with each of these prompts for the same set of sampled inputs. Instead of relying on incompatible metrics or manually assessing outputs, EvalLM enables designers to design their own "metrics" by simply defining criteria through natural language—e.g., a designer defines the criteria "Familiar Language" for an application that explains scientific concepts to children. Inspired by recent techniques on LLM-based evaluations [14, 23, 27], EvalLM employs an LLM as an *evaluation assistant*, which evaluates generated outputs on each of the criteria defined by the user. To aid users in revising prompts and criteria, the evaluation assistant explains its assessments to allow users to identify where outputs fell short or where the assistant's interpretation of criteria misaligned with the user's intent. Furthermore, the system aids prompt designers in defining effective criteria through an LLM-based *criteria reviewer*, which analyses the user's criteria to identify revisions that can lead to evaluations that assess more specific and fine-grained dimensions. By iterating with EvalLM, designers can co-evolve their prompts and criteria by improving their prompts to satisfy criteria and improving their criteria to discern prompt quality—ultimately leading to more polished applications.

2 FUTURE DIRECTIONS

In our work, we propose how a collaborative human-LLM workflow for more robustly evaluating long-tail LLM applications: a user defines criteria through natural language, an LLM automatically evaluates large samples of outputs on these criteria, and then the user audits the evaluator by assessing a smaller sample of evaluations. In this section, we discuss potential opportunities for future work on leveraging LLMs to interactively evaluate LLMs.

2.1 Auditing the Evaluator

The main advantage of LLMs as evaluators is that they allow for scaling up evaluation (i.e., assessing more outputs on more criteria) with less human effort and cost. During early development stages, this enables developers to more robustly assess the effect of prompt changes and make more informed decisions—one of the main challenges in developing with non-deterministic and black-box LLMs [24]. However, LLMs are not perfect evaluators. Our technical evaluation found that around 10% of LLM evaluations were illogical and around 15% were not self-contained. As developers and researchers use LLMs to assess larger samples of outputs, it can also be challenging to adequately audit the LLM evaluations. In EvalLM, we aimed to facilitate this by surfacing samples where evaluations from the same or different models differed to point out potentially faulty evaluations, and by highlighting what fragments from an output were evaluated by an LLM to help the user verify the evaluation's faithfulness. However, future work could investigate more sophisticated methods for auditing LLM evaluators: visualization techniques, automatic text analysis, and human-LLM-crowd workflows where a small set of crowdworkers verify samples of the LLM evaluations.

2.2 Designing Effective Criteria

Through our work and user studies, we identified that the LLM evaluations are only as effective as the quality of the criteria designed by the user. Similar to difficulties with prompt engineering, however, this can be challenging as evaluation criteria may have multiple interpretations and the user should be able to clearly express what aspects or features they intend to measure. In our work, we aimed to address this challenge by providing an LLM-based criteria reviewer that could help users identify and revise criteria that were unclear, vague, or overloaded. Future work can investigate approaches for designing more reliable and effective evaluation criteria. For example, as recent NLP work found that providing scoring rubrics alongside criteria can increase the reliability of LLM evaluations [8, 23], future work could investigate how to help users in designing rubrics for their criteria.

2.3 Criteria Overload to Criteria Hierarchy

While NLP researchers commonly employed general metrics or criteria to assess performance (e.g., "coherency", "relevance" [5, 28]) as they focused on general or broad tasks, LLMs are increasingly used for more specific and *long-tail* tasks where performance needs to be measured on bespoke criteria. While evaluating LLMs on more diverse criteria and metrics can provide a more comprehensive and in-depth understanding of performance [6], this diversification can also introduce challenges in how to compile evaluation results and overall model performance. However, as shown by our formative interviews and user study, most of these task-specific criteria are frequently subordinate to more general criteria—meaning that results on specific criteria can present insights about performance on general criteria. Future work could investigate how to collect and aggregate criteria from diverse evaluations into a criteria hierarchy that can represent model performance at a macroscopic (i.e., general criteria) or microscopic level (i.e., specific criteria)—enabling practitioners to more adequately compare models and make more informed model choices.

ACKNOWLEDGMENTS

This work was supported by NAVER-KAIST Hypercreative AI Center. This work was also supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-01347,Video Interaction Technologies Using Object-Oriented Video Modeling).

REFERENCES

[1] Tal August, Lucy Lu Wang, Jonathan Bragg, Marti A. Hearst, Andrew Head, and Kyle Lo. 2023. Paper Plain: Making Medical Research Papers Approachable to Healthcare Consumers with Natural Language Processing. *ACM Trans. Comput.-Hum. Interact.* (apr 2023). https://doi.org/10.1145/3589955 Just Accepted.

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL]

[3] John Joon Young Chung, Wooseok Kim, Kang Min Yoo, Hwaran Lee, Eytan Adar, and Minsuk Chang. 2022. TaleBrush: Sketching Stories with Generative Pretrained Language Models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 209, 19 pages. https://doi.org/10.1145/3491102.3501819

[4] Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. All That's 'Human' Is Not Gold: Evaluating Human Evaluation of Generated Text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Online, 7282–7296. https://doi.org/10.18653/v1/2021.acl-long.565

[5] Alexander R Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics* 9 (2021), 391–409.

[6] Sebastian Gehrmann, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research* 77 (2023), 103–166.

[7] Ellen Jiang, Kristen Olson, Edwin Toh, Alejandra Molina, Aaron Donsbach, Michael Terry, and Carrie J Cai. 2022. PromptMaker: Prompt-Based Prototyping with Large Language Models. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, Article 35, 8 pages. https://doi.org/10.1145/3491101.3503564

[8] Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, et al. 2023. Prometheus: Inducing fine-grained evaluation capability in language models. *arXiv preprint arXiv:2310.08491* (2023).

[9] Tae Soo Kim, Yoonjoo Lee, Jamin Shin, Young-Ho Kim, and Juho Kim. 2023. EvalLM: Interactive Evaluation of Large Language Model Prompts on User-Defined Criteria. *arXiv preprint arXiv:2309.13633* (2023).

[10] Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840* (2019).

[11] Yoonjoo Lee, Tae Soo Kim, Sungdong Kim, Yohan Yun, and Juho Kim. 2023. DAPIE: Interactive Step-by-Step Explanatory Dialogues to Answer Children's Why and How Questions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 450, 22 pages. https://doi.org/10.1145/3544548.3581369

[12] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What Makes Good In-Context Examples for GPT-3? *arXiv preprint arXiv:2101.06804* (2021).

[13] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-Train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Comput. Surv.* 55, 9, Article 195 (jan 2023), 35 pages. https://doi.org/10.1145/3560815

[14] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. arXiv:2303.16634 [cs.CL]

[15] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2021. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786* (2021).

[16] OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL]

[17] Savvas Petridis, Nicholas Diakopoulos, Kevin Crowston, Mark Hansen, Keren Henderson, Stan Jastrzebski, Jeffrey V Nickerson, and Lydia B Chilton. 2023. AngleKindling: Supporting Journalistic Angle Ideation with Large Language Models. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 225, 16 pages. https://doi.org/10.1145/3544548.3580907

[18] Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems* 34 (2021), 4816–4828.

[19] Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. BLEURT: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696* (2020).

[20] Hendrik Strobelt, Albert Webson, Victor Sanh, Benjamin Hoover, Johanna Beyer, Hanspeter Pfister, and Alexander M. Rush. 2023. Interactive and Visual Prompt Engineering for Ad-hoc Task Adaptation with Large Language Models. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2023), 1146–1156. https://doi.org/10.1109/TVCG.2022.3209479

[21] Jing Wei, Sungdong Kim, Hyunhoon Jung, and Young-Ho Kim. 2023. Leveraging Large Language Models to Power Chatbots for Collecting User Self-Reported Data. arXiv:2301.05843 [cs.HC]

[22] Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022. AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 385, 22 pages. https://doi.org/10.1145/3491102.3517582

[23] Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2023. FLASK: Fine-grained Language Model Evaluation based on Alignment Skill Sets. *arXiv preprint arXiv:2307.10928* (2023).

[24] J.D. Zamfirescu-Pereira, Heather Wei, Amy Xiao, Kitty Gu, Grace Jung, Matthew G Lee, Bjoern Hartmann, and Qian Yang. 2023. Herding AI Cats: Lessons from Designing a Chatbot by Prompting GPT-3. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference* (Pittsburgh, PA, USA) *(DIS '23)*. Association for Computing Machinery, New York, NY, USA, 2206–2220. https://doi.org/10.1145/3563657.3596138

[25] J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 437, 21 pages. https://doi.org/10.1145/3544548.3581388

[26] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* (2019).

[27] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. arXiv:2306.05685 [cs.CL]

[28] Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. *arXiv preprint arXiv:2210.07197* (2022).