Project Proposal: The Cost of Prompt Engineering

Anonymous ACL submission

Abstract

What are the attributes of text generated by large language models, and which ones make them useful?

1 Introduction

Prompt engineering, the practice of developing specialized prompts and queries to improve the accuracy of Large Language Models after training, is a prominent topic of interest in the NLP community, and among the general public. The practice is believed to allow for improvements in LLM performance a variety of domains without investment in underlying training (Martineau, 2021). It is not, however, without its critics. Some commentators believe that the practice will become irrelevant as models grow larger and more powerful, becoming more capable of directly interpreting a user's intent. (Ethan Mollick [@emollick], 2023). Others question the need for specialized professionals or traiing to attain minimal improvements which are often not repeatable across domain types and contexts (Shackell, 2023; Acar, 2023).

Despite such controversy, it is difficult to find empirical and quantitative analyses of the tradeoff between costs and accuracy benefits associated with advanced prompting. Papers introducing new prompting techniques often only include performance benchmarks concerning the techniques' efficacy, typically within a limited question domain. Some authors briefly mention problems associated with human-tailored problems, such as the increased complexity induced by prompt-chaining and limitations on creativity and randomness (Wu et al., 2022), and others suggest the automation of prompting (Diao et al., 2023).

This paper uses several metrics to evaluate the costs of prompt engineering methods systematically, and analyzes the tradeoffs inherent in their application to standardized data. Such an assessment is valuable on several dimensions. Beyond testing the practicability of prompt engineering as a whole, it can be used to compare the performance of different approaches, in a world where so many competing techniques are available. I also offer a new assessment in a period long after ideas were introduced and in an enviroment with greater capabilities in underlying models. Finally, I survey and introduce some useful measures of costs and complexity such as the ratio of interaction length with prompting to the length of an accepted human-generated answer.

Costs are generally in terms of length and complexity. Length imposes several costs. Potential for quality to decline, see Roose. Extra time to review output and check each step in real world scenarios, when some steps may be obvious to human reviewers. Token costs.

(noa) Promptbase offers prompts for sale, and some token/cost estimates

2 Metrics

2.1 Accuracy

Accuracy at the point a human is confident in the answer (modelling the real world)

Accuracy (according to the papers initially presenting the technique, in whatever domain it was tested)

2.2 Costs (Automated Metrics)

Motivation Roose conversation - length can lead to significant deterioration in quality. Microsoft limited converation/response length for Bing Chat afterwards. (Roose, 2023)

Length of the entire interaction in tokens

Length of the entire interaction in tokens relative to the length of a human/solved out/generally accepted as correct answer. How much is the engineering stretching the interaction out? Is this stretching adding value/improving accuracy?

Length of the entire interaction in time (seconds)

Response complexity using automated metrics: vocabulary, sentence length, readability scores.

Cost of this length

Run all automated metrics on the prompt, dialogue and responses, and the accepted/outside correct answer.

2.3 Costs and Complexity (Human Metrics)

Human assessment of need for specialized knowledge/difficulty of implementation of the technique. This could be task specific (done for some real world example questions/prompting scenarios), or it could be done overall based on a simple description of the technique. Perhaps a balance of both is best.

Human assessment of output complexity (ease of evaluating results). This could be task specific (done for some real world example questions/prompting scenarios), or it could be done overall based on a simple description of the technique. Perhaps a balance of both is best.

3 Prompting Methods Assessed

Zero-Shot Control Baseline Zero-Shot Chain of Thought Prompting https://arxiv.org/pdf/2305.02897.pdf provides several prompt examples. Original paper https://papers.nips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf Few-Shot Prompting. On page 4, we see a comparison of the returns to in-context learning by model size https://arxiv.org/pdf/2005.14165.pdf. But this paper is from a time of smaller models (and indeed I don't think the newest GPT-4 tech report even considers few-shot/chain-of-thought)! Also, here we see that the correctness of examples is not important - instead, info about the distribution of the label space, input text dist, output format is key https://arxiv.org/pdf/2202.12837.pdf. Manually provided chain-of-thought prompting. It was at one time claimed that this method could produce superior performance https://arxiv.org/pdf/2210.03493.pdf. Note some of these advanced LMs automatically do CoT Chain of verification prompting: https://arxiv.org/pdf/2309.11495.pdf Finish going through repo and also look up most popular techniques online... Generated Knowledge Prompting: https://arxiv.org/pdf/2110.08387.pdf (possibly with knowledge coming from the GPT model itself) Tree of thought prompting: https://github.com/dave1010/tree-of-thought-prompting

4 Data

Probably real-world/professional, rather than academic tasks. Coding/Leetcode, then perhaps some very generalized standard tests (All GRE Sections)

For some of the simpler metrics, in limited domains, it may be possible to use reported accuracy scores on the domain dataset coming from the original paper. It may also be possible to use prompts, LLM responses, and correct responses provided along with original papers.

However, the more complex metrics will require human evaluation.

I perform the analysis on one cutting-edge model, to get the current state of things, and one older model, closer to the time that these techniques were introduced. This will allow one picture of the changing costs and benefits of advanced prompting, a trend that may even be extrapolated into the future if current LLM scaling laws continue to hold. As the most widely used models and the ones behind much original work in the field, I select two models from the OpenAI series: GPT-4 and text-davinci-003 on OpenAI playground. Should text-davinci-003 (a legacy model) become unavailable during the course of the project, or should I encounter other difficulties, I will use the simplest/smallest model available, likely GPT-3.5. All of these models are available both via the OpenAI API and in web interfaces - where possible I will use web interfaces to limit resources required.

5 Responsiblities

I am the only author of this proposal, but I am open to working with others with similar interests. Expanding the group would be helpful in order to improve the project.

Limitations

It was difficult to select prompt engineering methods to try for this paper, and there is potential for my choice of methods to be somewhat biased. I mostly picked methods based my perception of their popularity and on their ease of implementation. If anything, this may lead to an underestimation of costs.

Just as my evaluation comes at a time with significantly more capable LLMs relative to that where much work began on prompting, I expect the underlying calculus concerning prompting to continue to change in the future. However, I again only expect relative costs of complex engineering to increase as models get better.

Another potential problem is the extent that prompting techniques have been absorbed into default LLM behavior, likely through reinforcement learning. GPT-4 in particular does seem to implement chain-of-thought methods when prompted with a sufficiently complex problem. In this environment, this paper become less of an evaluation of prompting techniques themselves, but more of an evaluation of their manual implementation.

Acknowledgements

The template for this document was adapted by Jordan Boyd-Graber, Naoaki Okazaki, and Anna Rogers.

References

PromptBase.

Oguz A. Acar. 2023. AI Prompt Engineering Isn't the Future. *Harvard Business Review*. Section: Technology and analytics.

Shizhe Diao, Pengcheng Wang, Yong Lin, and Tong Zhang. 2023. Active Prompting with Chain-of-Thought for Large Language Models. ArXiv:2302.12246 [cs].

Ethan Mollick [@emollick]. 2023. I have a strong suspicion that "prompt engineering" is not going to be a big deal in the long-term & prompt engineer is not the job of the future AI gets easier. You can already see in Midjourney how basic prompts went from complex in v3 to easy in v4. Same with ChatGPT to Bing. https://t.co/BTtSN4oVF4.

Kim Martineau. 2021. What is prompt tuning?

Kevin Roose. 2023. A Conversation With Bing's Chatbot Left Me Deeply Unsettled. *The New York Times*.

Cameron Shackell. 2023. Prompt engineering: is being an AI 'whisperer' the job of the future or a short-lived fad?

Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022. AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts. In *CHI Conference on Human Factors in Computing Systems*, pages 1–22, New Orleans LA USA. ACM.