CS-552 Final project - Milestone 1 - Paper Summary

Mathieu Desponds - 283229

May 2023

1 Introduction

The paper, published in February 2021 by Laria Reynolds and Kyle McDonell on arXiv, delves into the realm of prompting and presents novel insights, particularly in the form of meta prompts to improve the actual prompting strategies. The central premise of the paper is that 0-shot strategies can outperform few-shot strategies, with the authors asserting that few-shot strategies primarily serve to locate the task rather than facilitate learning from the provided examples. They underscore the significance of prompts and their potential to enhance model performance.

The paper's motivation to enhance prompting strategies is natable. Although language models (LLMs) are highly capable, effectively using their full potential remains a challenge. The exploration of novel techniques like metaprompting, which provide general approaches applicable to multiple specific tasks, can greatly benefit LLM users and facilitate improved utilization of these powerful models.

2 Summary

The authors devote considerable attention to the exploration of prompts, highlighting techniques to incorporate cultural anchors and encourage problem deconstruction. They argue that prompts can be more effective than fine-tuning or few-shot strategies. Additionally, they emphasize the importance of understanding how models are trained and leveraging this knowledge to design effective prompts. By drawing parallels to human cognition, they propose prompt programming as a form of natural language programming, urging the adaptation of existing language comprehension knowledge to interact with autoregressive language models. To improve prompting, the authors propose several techniques, all rooted in the way humans think and understand language. These techniques include:

1. Direct task specification: Instructing the model explicitly, such as using phrases like "translate..." or "give me three dead French authors."

2. Specification by demonstration: Providing examples of tasks in context, distinct from traditional few-shot approaches. This approach ensures that the examples are treated as independent entities.

3. Task specification by memetic proxy: Using analogies or well-known figures to convey cultural information that may be more effective than explicit task descriptions. This technique is particularly useful for complex tasks. For instance, instead of instructing the model to answer a question with care and attentiveness, a prompt that sets up a dialogue between a student and a teacher can provide richer contextual information. By leveraging the dynamics of a student-teacher interaction, the model gains a deeper understanding of the desired task. This example illustrates how employing memetic proxies can capture nuanced aspects of a task more comprehensively than explicit instructions.

4. Prompt programming as constraining behavior: crafting prompts that maintain consistency with the desired outcome and deviate from undesired continuations. Few-shot examples can help establish patterns that the model should adhere to. For example, instead of using a prompt like "French: [this sentence]. English:", a more effective approach would be to use a prompt like "Translate [this sentence] into English." This formulation provides clearer instructions and encourages the model to generate the desired translation.

5. Serializing reasoning for closed-ended questions: This approach aims to replicate the way humans break down complex questions into smaller subtasks before providing an answer. By discouraging premature verdicts, the prompt programming process mimics the act of using scratch paper for reasoning. To achieve this, a prompt fragment like "therefore, the final answer is" is injected to signal the end of generation and enforce a more deliberate approach. The challenge lies in determining the optimal timing for injecting this fragment. The authors suggest utilizing a likelihood signal that reaches its maximum before injecting the prompt fragment. However, the specific method for computing this likelihood is not detailed in the paper. Further research and experimentation are needed to refine and validate this technique's effectiveness.

6. Meta prompting programming: The authors acknowledge that using a generic prompt for every task yields suboptimal results. To address this, they propose the concept of metaprompts, which aim to automate the generation of task-specific prompts. Metaprompts serve as seed prompts that encapsulate a broader intention, allowing for customization based on the specific task requirements. The paper presents several metaprompts such as "this problem asks us to" and "Let's solve this problem by splitting it into steps." Additionally, fill-in-the-blank sentences are employed to provide more flexibility. However, it is worth noting that no quantitative analysis is conducted to evaluate the performance or effectiveness of these

different metaprompts. Further research and experimentation are needed to assess their impact and potential benefits in prompt programming.

In conclusion, the paper highlights the importance of future directions in prompt engineering. Specifically, it emphasizes the need for new benchmarks to assess the effectiveness of meta prompts. The authors acknowledge the limitations of current evaluation metrics, such as BLUE, and suggest using language models to evaluate the quality of generated answers.

3 Strength and Critics

While the paper's strengths lie in providing a comprehensive overview of current prompting methods and introducing the concept of meta prompts for task-specific prompts, some criticisms need to be raised. The main one is the absence of comprehensive results (qualitative and quantitative) and experiments that makes it difficult to fully evaluate and validate the effectiveness of the proposed prompt strategies. This limitation, particularly the lack of comparative analysis and concrete evidence, weakens the paper's ability to robustly support its claims and showcase the advantages and disadvantages of different prompt techniques. Additionally, the paper lacks specific details on the process of finding meta prompts that would be useful to develop new ones and does not offer a thorough comparison of different prompt strategies or their respective advantages and disadvantages. Furthermore, the authors primarily demonstrate the superiority of 0-shot strategies over few-shot strategies using an example of translation. We might argue that translation is not the most representative case, as learning to translate requires more than a few examples. The impact of few shots in translation might be lower than in a sentiment analysis task. Moreover, we can question the assertion that anyone fluent in natural language can become a prompt programmer, contending that a deeper understanding of model architectures and transformers is necessary for effective prompt design.

4 Conclusion

In summary, while the paper offers valuable insights into prompting methods, placing a particular emphasis on meta prompts as a generalized approach and by relating prompt design to human language comprehension, the lack of detailed results and experimentation poses a challenge in fully assessing the efficacy and practical implications of the proposed techniques. Further research and concrete evidence are necessary to strengthen the paper's arguments and demonstrate the advantages and limitations of the various prompt strategies.