Few-shot Inference Analysis of Automatic Prompt Generation Capabilities of GPT-3 via the Lens of GPT-Challenging Arithmetics

Brian Lui
Department of Computer Science
Stanford University
Stanford, CA 94305
brianlui@stanford.edu

Abstract

Automatic prompt generation of large language models like GPT-3 is crucial to the scalability and user-friendliness of conversational artificial intelligence (AI). This paper examines the capabilities of prompt generation of instruction-finetuned 175-B GPT-3 using Stanford's CRFM HELM platform [4] on 10 sets of arithmetic tasks easy to humans solvers but challenging to GPT-3 via a series of evaluation experiments, including multi-step few-shot prompting. We find that under two-step zero-shot prompting, GPT-3 is able to provide a helpful prompt for itself of comparable value to an additional example from the target task.

1 Introduction

Prompt generation empowered the recent resurgence in popularity of large language models (LLMs) like ChatGPT from OpenAI but still entails significant human design efforts and manual trial-and-error experimentation. For user-friendliness and long-term scalability of successful conversational artificial intelligence (AI), there is a need for large language models to automate generating the best prompt to correctly answer any user-specified question. Achieving automatic prompt generation would empower both users and researchers to reallocate time away from iterative prompt engineering efforts and raise the bar on the fluidity and user-friendliness of natural conversational AI.

This paper analyzes the capabilities of GPT-3 (with 175 billion parameters, instruction-finetuned) to automate prompt generation for itself to best answer user-specified arithmetic questions. We leverage a token-level computational budget on Stanford's CRFM HELM platform [4] to run inference on this model in a multi-step process and evaluates the accuracy of GPT-3's responses to various prompt structures.

After initial experiments, we target the instruction-finetuned 175-B (OpenAI's Davinci) version of GPT-3 for its aptitude in digesting more complex prompts and richer model complexity compared to smaller models (e.g. OpenAI's Ada) to return better-phrased responses to arithmetic questions.

This work on arithmetic datasets helps researchers tackle more complex question-answer datasets, e.g. GradeSchoolMath (GSM8K) [2], that require much longer prompts and solutions and hence a much more extension computational budget for GPT-3 to perform lengthy, step-by-step deduction.

Preprint. Under review.

2 Related work

Recent work in automatic prompt generation reflects an ongoing trend to pre-design templates, impose constraints on the format of prompts, and train new architectures to fill out masked values in such otherwise incomplete prompts:

1. Instruction induction [3] presents a new challenge to learn the best instruction that precedes a list of example input-output pairs;
2. Automatic Prompt Engineer [7] uses large language models for inference, scoring, and resampling;
3. AutoPrompt [5] appends trigger tokens and templates to prompt masked language models; and
4. PromptGen [6] trains a new encoder-decoder model for prompt generation.

In contrast to the aforementioned related work, this research project emphasizes the following differentiating aspects:

1. **Frugality of architectures:** The focus of this research is not to train and explore new architectures for prompt generation, but to analyze how well existing language models, e.g. GPT-3, can self-learn prompt generation. We believe that reliable language models should not rely on separate architectures to output helpful prompts for themselves and should be self-sufficient. This spirit of frugality is crucial to learning an end-to-end language model that takes in any user-given question, generates effective prompts itself, and produces accurate answers.
2. **Robustness against adversarial prompts:** This paper also aims to assess how resilient and robust large language models are against adversarial example output-input pairs from other tasks in prompts. This study would help inform researchers of how well existing large language models defend against injection attacks and uphold robustness.

For datasets introduced in the next section, we draw from two separate pieces of prior work by OpenAI on:

1. Training verifiers to solve mathematical questions in the newly introduced GradeSchoolMath (GSM8K) dataset [2] and
2. Demonstrating that language models are few-shot learners [1].

3 Datasets

OpenAI introduced a new dataset called GradeSchoolMath (GSM8K) [2] that contains 8,000 question-answer pairs where each human-langauge mathematical question asks for explicit step-by-step deduction culminating in a numeric final answer. While this dataset is an initial target of this project, the lengthy prompts and step-by-step solutions in this dataset have proven to consume many more tokens than the given computational budget. Besides, we observe from initial experiments of GSM8K that the requirement of step-by-step solutions before a final answer seems to confound evaluation of GPT-3's responses against the ground-truth solutions. These practical concerns regarding the extensive demand on computational resources and confounding factors in evaluation of GPT-3's responses to prompts motivate us to pivot away from GSM8K to a smaller, more customizable dataset from another piece of work also by OpenAI: "Language Models are Few-shot Learners" [1] in 2020.

We refer to Section 3.9.1 of OpenAI's paper "Language Models are Few-shot Learners" [1] and generate 10 sets of 2,000 question-answer pairs in natural language for 10 arithmetic tasks according to the exact sampling procedure given in that section, namely:

1. **2D+, 3D+, 4D+, and 5D+** for addition of two non-negative integers each of at most 2-5 digits, sampled uniformly from the corresponding range, e.g. "Q: What is 79 plus 4? A: 83";
2. **2D-, 3D-, 4D-, and 5D-** for subtraction between two non-negative integers each of at most 2-5 digits, sampled uniformly from the corresponding range, e.g. "Q: What is 274 minus 831? A: -557";

3. **2Dx**, multiplication of two 2-digit non-negative integers, each sampled uniformly from [0, 100), e.g. "Q: What is 14 times 29? A: 406"; and
4. **1DC**, a composite operation between three 1-digit integers, with parantheses around the last two, where each operation is sampled uniformly from $\{+, -, \times\}$ and each 1-digit integer is sampled uniformly from [0, 10), e.g. "Q: What is 5 × (7-9)? A: -10".

For each of the 10 arithematic tasks above, among the 2,000 question-answer pairs we generate, we treat the first 1,900 pairs as our training set from which we sample example output-input pairs to form few-shot prompts, and evaluate GPT-3's responses to such prompts on the test set of the last 100 pairs in terms of exact-match accuracies of the final numeric answers.

While the above arithmetic tasks look intuitively simple to human solvers, they pose a significant challenge to the 175-B version of GPT-3 before any instruction fine-tuning according to OpenAI's prior work [1], giving:

1. Sub-10% accuracies for addition and subtraction between long integers of at most 4-5 digits ({4,5}D{+,-}) and composite operations among three 1-digit integers (1DC); and
2. A sub-20% accuracy for 2-digit integer mulitplication (2Dx) (Table 3.9 of [1]).

Readers should note that 175-B GPT-3 achieves such performances without any aid from an outer wrapper around GPT-3 that handles preprocessing and postprocessing for simple arithmetic tasks like OpenAI's online ChatGPT product.

These arithmetic tasks that are easy to human solvers but challenging to 175-B GPT-3 make it meaningful for this project to find the best prompt for GPT-3 to correctly answer these arithmetic questions. This smaller dataset compared to GSM8K offer a distinct advantage for us to examine the effect of using example question-answer pairs from other tasks to form few-shot prompts for questions in a certain task.

Instead of querying 175-B GPT-3 before any instruction finetuning as in OpenAI's paper [1], we target the instruction-finetuned version of 175-B GPT-3 to evaluate GPT-3's full potential to digest prompt-generating instructions and output potentially helpful prompts it can understand.

4 Methods

For each test question in each of the 10 arithmetic tasks, we prompt instruction-finetuned 175-B GPT-3 for a response and compute its exact-match accuracy in each arithmetic task in a series of experiments:

1. **Zero-shot baseline experiments**: Query GPT-3 with a 0-shot prompt, e.g. "Q: What is 19 plus 34? A:" for the question "What is 19 plus 34?" in the 2D+ task.
2. **Few-shot cross-task experiments**: Query GPT-3 with a {1,2,3}-shot prompt, in which the target question is preceded by {1,2,3} example question-answer pair(s) sampled uniformly from the training set of the same task or a different task. For example, a 1-shot prompt for the 2D- question "What is 72 minus 35?" using 1 example question-answer pair from the 3D+ task could be "Q: What is 190 plus 423? A: 613. Q: What is 72 minus 35? A:".
3. **Two-step zero-shot experiments**: Query GPT-3 twice with the following prompting procedure:
   (a) First, query GPT-3 with a 0-shot prompt, in which the target question is preceded by a fixed instruction "Write a helpful prompt for the question", for a prompt-like response.
   (b) Then, append this prompt-like response with a 0-shot prompt for the target question, and use this augmented prompt to query GPT-3 again for a final numeric answer to the target question.

   For example, for the target 2D- question "What is 72 minus 35?", the first 0-shot prompt is "Write a helpful prompt for the question. Q: What is 72 minus 35? Prompt:", while the second prompt to GPT-3 is GPT-3's response to the first prompt appended with the 0-shot prompt "Q: What is 72 minus 35? A:" simply to ensure that our second query to GPT-3 always asks for an explicit answer to the target question no matter what kind of prompt GPT-3 outputs in response to our first query.

4. **Two-step one-shot cross-task experiments**: Query GPT-3 twice with the following prompting procedure:
   (a) First, query GPT-3 with a 1-shot prompt, in which the target question is preceded by a fixed instruction "Write a helpful prompt for the question" (same as the above) and 1 example question-answer pair sampled uniformly from the training set of the same task or a different task, for a prompt-like response.
   (b) Then, append this prompt-like response with a 0-shot prompt for the target question, and use this augmented prompt to query GPT-3 again for a final numeric answer to the target question.

   For example, for the target 2D- question "What is 72 minus 35?", the first 1-shot prompt aided by 1 example question-answer pair from the training set of the 1DC task could be "Write a helpful prompt for the question. Q: What is 7 × (2+1)? A: 21. Q: What is 72 minus 35? Prompt:", while the second prompt to GPT-3 is GPT-3's response to the first prompt appended with the 0-shot prompt "Q: What is 72 minus 35? A:" for the same reasons given above.

For each of the 10 arithmetic tasks, we compare GPT-3's final numeric answers to the 100 test questions of the task against the ground-truth answers to compute the exact-match accuracy in various prompting scenarios parameterized by the number of shots (K=0,1,2,3) and the helper task the example question-answer pairs in the prompts are sampled from.

5 Results

We analyze the test accuracies of instruction-finetuned 175-B GPT-3 for each of the 10 arithemtic tasks in various prompting scenarios.

5.1 Zero-shot baseline experiments

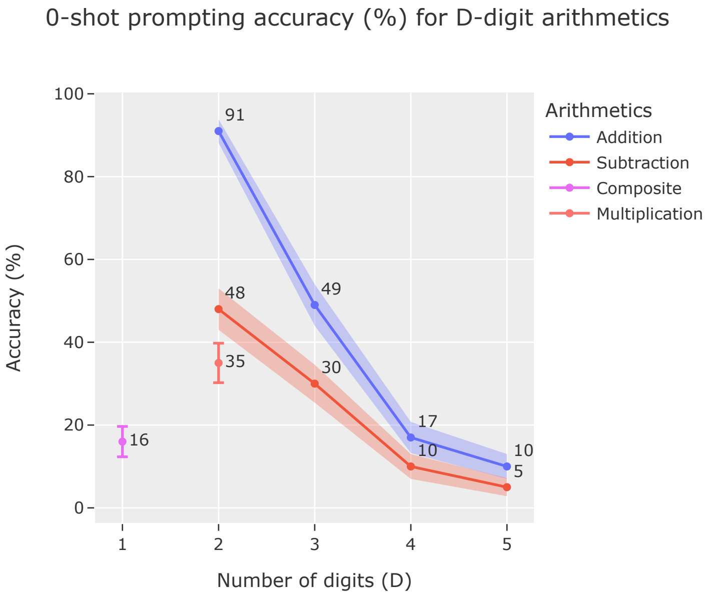


Figure 1: Accuracy of instruction-finetuned 175-B GPT-3 in each arithmetic task in zero-shot baseline experiments. Error bands correspond to accuracies one standard deviation above and below the reported mean accuracy.

In Figure 1, we see the impressive performance of GPT-3 in 2-digit integer addition (91%) quickly deteriorates by almost a half to 49% for 3-digit integer addition and even lower figures (17% and

10%) for {4,5}-digit integer addition. This negative relationship of the accuracy with the number of digits per integer involved in integer addition (blue curve) is echoed by integer subtraction (red curve) as well.

We also observe from Figure 1 that GPT-3 performs addition (blue curve) better than subtraction (red curve) when the number of digits per integer involved is fixed.

GPT-3 gives relatively underwhelming accuracies (16% for 1DC and 35% for 2Dx) for complex multiplicative and composite arithmetic tasks.

5.2 Few-shot cross-task experiments

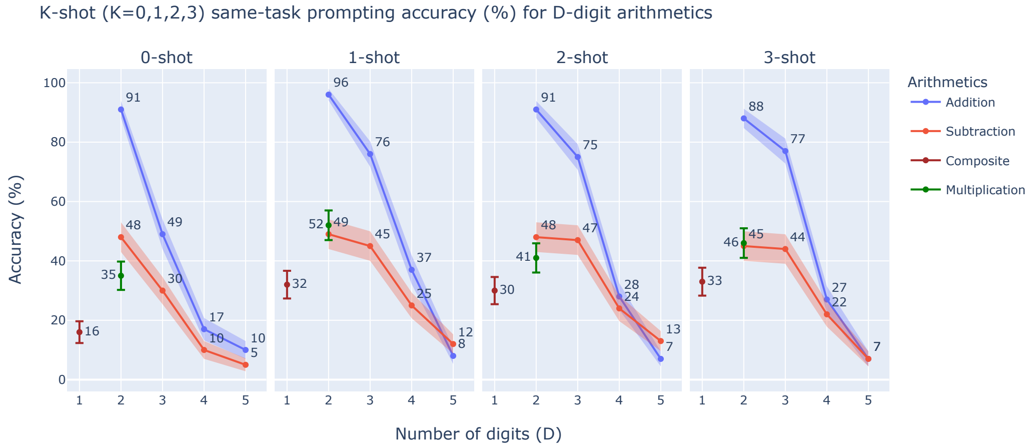


Figure 2: Accuracy of instruction-finetuned 175-B GPT-3 in each arithmetic task in {0,1,2,3}-shot same-task experiments, where the helper task is the same as the target task of the target question. Error bands correspond to accuracies one standard deviation above and below the reported mean accuracy.

In Figure 2, we visualize the test accuracy in each arithmetic task when 1-shot prompts are used with helper question-answer pairs sampled from the same task as that of the target question.

Complex tasks involving multiplication (2Dx) and composite operations (1DC) seem to benefit substantially from the 1-shot same-task prompting (up from 16% to 32% for 1DC and up from 35% to 52% for 2Dx) despite lackluster performance from 0-shot prompting in baseline experiments.

The most prominent take-away is that this few-shot prompting paradigm benefits the most when we use 1-shot prompts but suffers from diminishing marginal returns as we add more example question-answer pairs from the same task. This statement is visualized more precisely in Figure 3.

In Figure 3, we notice that in most arithmetic tasks, the marginal return from adding an additional shot to the few-shot same-task prompts is maximized when we add the first shot but quickly dwindles as we add beyond the first shot. 3-digit and 4-digit integer addition (3D+ and 4D+) seem to benefit from adding the first same-task example question-answer pair in the few-shot prompts the most (up by 27% for 3D+ and up by 20% for 4D+ in absolute percentage units).

In Figure 4, we see the test accuracy of each arithmetic task (row) under 1-shot cross-task prompting aided by each helper task.

The key take-away is that the best helper task for each target task is not necessarily the same as the target task; otherwise, we would see a clear red diagonal in the 1-shot cross-task heatmap in Figure 4. Therefore, there might be value to researchers to explore using examples from other tasks in forming few-shot prompts.

For example, 2-digit integer multiplication (2Dx) is aided best by itself (52%), but 3-digit integer subtraction is aided best by composite operations (1DC at 48%) and {2,3}-digit integer addition (2D+ and 3D+ at 48%).

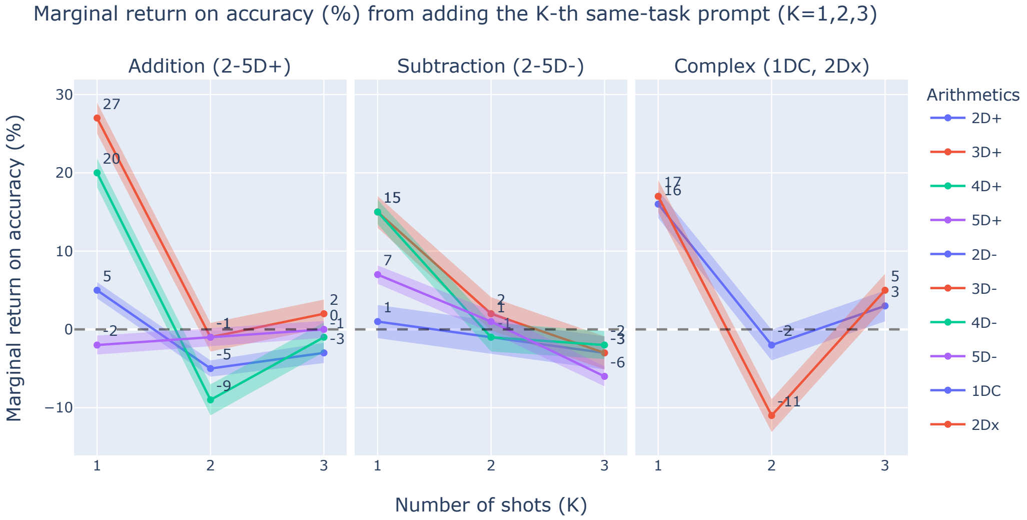


Figure 3: Marginal return from adding the K-th (K=1,2,3) shot (example question-answer pair) to the few-shot same-task prompts on the accuracy of instruction-finetuned 175-B GPT-3 in each arithmetic task in {0,1,2,3}-shot same-task experiments, where the helper task is the same as the target task of the target question. Error bands correspond to accuracies 0.3 standard deviation above and below the reported mean accuracy. The black dotted horizontal line at 0 indicates no marginal benefit to adding an extra shot to the few-shot prompts.

Some helper tasks are clearly adversarial to certain target tasks. For example, using 1-digit composite operations (1DC) as a helper tasks hurt the accuracy for 2-digit integer multiplication (2Dx) by 15% (down from 52% to 37%) and 4-digit integer addition (4D+) by 19% (down from 37% to 18% by more than a half) in absolute percentage units.

However, we also note that the overall variation of the target-task performance over various helper tasks is not too drastic for high-performing target tasks, e.g. 2-digit addition (2D+).

5.3 Two-step {0,1}-shot same-task experiments

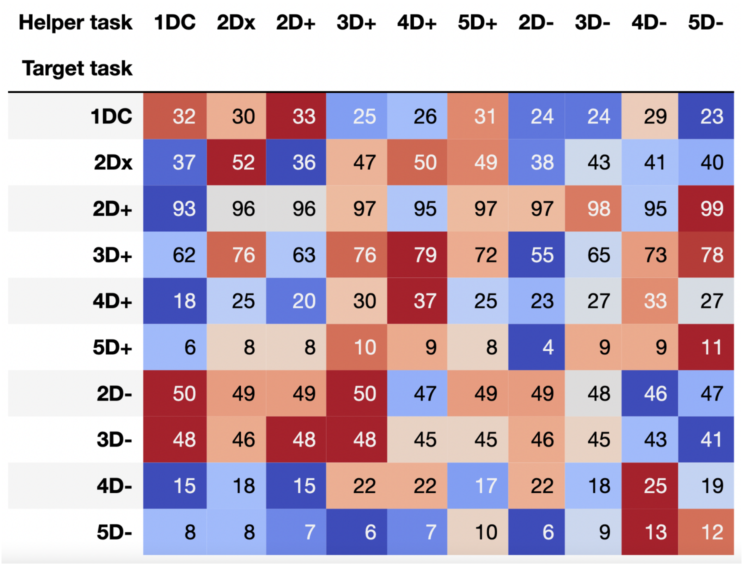
In Figure 5, we observe that in the two-step prompting procedure where we first ask GPT-3 to provide a helpful prompt for itself, GPT-3 actually performs comparably well in both 0-shot and 1-shot same-task prompting scenarios. This is in stark contrast to the one-step few-shot same-task experiments, where we observed that adding the first shot in few-shot prompts brings significant improvement in the test accuracies. This contrast implies that GPT-3 is able to offer a helpful prompt with no help from other examples that provides comparable deductive value to an example from the same task in the 1-shot same-task prompting scenario.

Remarkably, for both 2-digit integer addition (2D+) and 1-digit composite operation (1DC), GPT-3 actually performs better (99% $>$ 95% for 2D+ and 52% $>$ 46% for 1DC) by generating a helpful prompt for itself without any example (0-shot) than with an example (1-shot).

In Figure 6, we notice that in the 0-shot prompting scenario, asking GPT-3 first to generate a helpful prompt for itself instead of using the original target question only actually brings a boost to the test accuracies in most arithmetic tasks. However, in the 1-shot setting, doing so does not generally bring much improvement to the test accuracies except in complex tasks like 1-digit composite operations (1DC).

5.4 Two-step one-shot cross-task experiments

In Figure 7, we display the test accuracy of each arithmetic task in two-step 1-shot prompting scenarios aided by each helper task. The main take-away is that the best helper task for each target task is



Helper task	1DC	2Dx	2D+	3D+	4D+	5D+	2D-	3D-	4D-	5D-
Target task										
1DC	32	30	33	25	26	31	24	24	29	23
2Dx	37	52	36	47	50	49	38	43	41	40
2D+	93	96	96	97	95	97	97	98	95	99
3D+	62	76	63	76	79	72	55	65	73	78
4D+	18	25	20	30	37	25	23	27	33	27
5D+	6	8	8	10	9	8	4	9	9	11
2D-	50	49	49	50	47	49	49	48	46	47
3D-	48	46	48	48	45	45	46	45	43	41
4D-	15	18	15	22	22	17	22	18	25	19
5D-	8	8	7	6	7	10	6	9	13	12

Figure 4: Accuracy (%) of instruction-finetuned 175-B GPT-3 in each arithmetic task in 1-shot cross-task prompting experiments where each prompt is formed by including 1 example question-answer pair from a helper task. Each row corresponds to a target task, while each column corresponds to a helper task. Red colors and blue colors refer to high and low values respectively for each target task in each row.

again not necessarily the same as the target task itself; otherwise, we would observe a prominent red diagonal on the heatmap. However, we note that integer addition and integer subtraction seem to aid each other, since we notice that two 4-by-4 off-diagonal blocks corresponding to the interaction between these two types of tasks seem to be dominated by red colors (high test accuracies).

6 Conclusion

Automatic prompt generation of large language models like GPT-3 is crucial to the scalability and user-friendliness of conversational artificial intelligence (AI). We assess the capabilities of prompt generation of instruction-finetuned 175-B GPT-3 using Stanford's CRFM HELM platform [4] on 10 sets of arithmetic tasks easy to humans solvers but deceptively challenging to GPT-3 via a series of evaluation experiments, including multi-step few-shot prompting.

We find that there is value to researchers to explore helper tasks different from the task of the target question they seek to answer to provide more informative example question-answer pairs to form few-shot prompts for GPT-3. We also observe that without any example from any helper task, GPT-3, in two-step zero-shot prompting scenarios, is capable of providing a helpful prompt of comparable value to an additional example from the training set of the target task.

7 Contributions

This is the work of a 1-person team. 100% of this project is work by Brian Lui. The code for this project can be found at the GitHub repository at

`https://github.com/briancylui/prompt.`

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

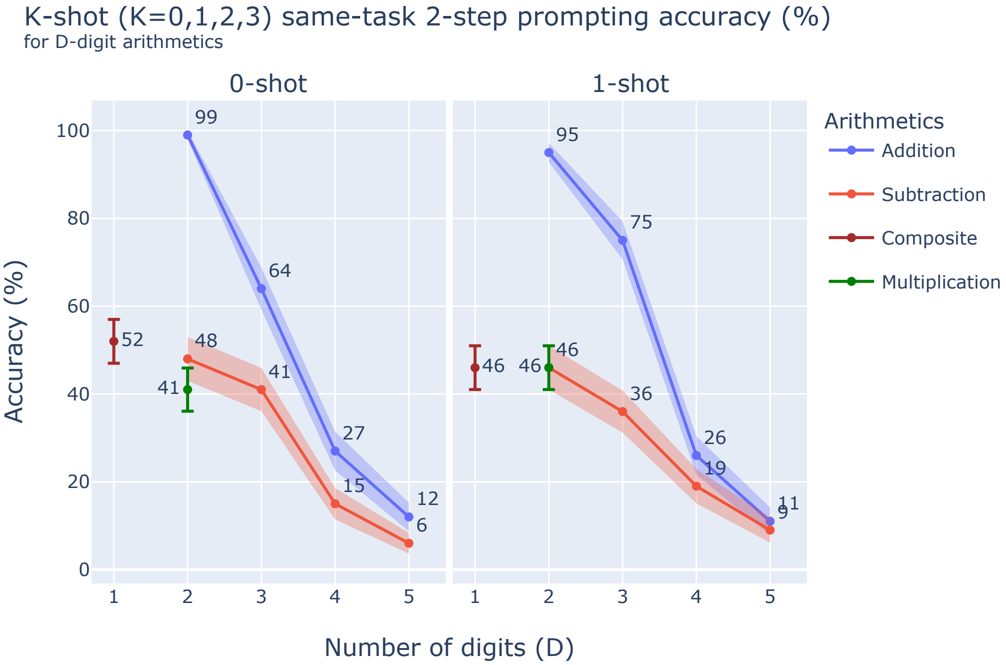


Figure 5: Accuracy of instruction-finetuned 175-B GPT-3 in each arithmetic task in two-step {0,1}-shot same-task experiments, where the helper task is the same as the target task of the target question. In this two-step procedure, GPT-3 is first asked to provide a helpful prompt for itself to answer the target question in the second query. Error bands correspond to accuracies one standard deviation above and below the reported mean accuracy.

[2] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

[3] Or Honovich, Uri Shaham, Samuel R Bowman, and Omer Levy. Instruction induction: From few examples to natural language task descriptions. *arXiv preprint arXiv:2205.10782*, 2022.

[4] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

[5] Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*, 2020.

[6] Yue Zhang, Hongliang Fei, Dingcheng Li, and Ping Li. Promptgen: Automatically generate prompts using generative models. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 30–37, 2022.

[7] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*, 2022.

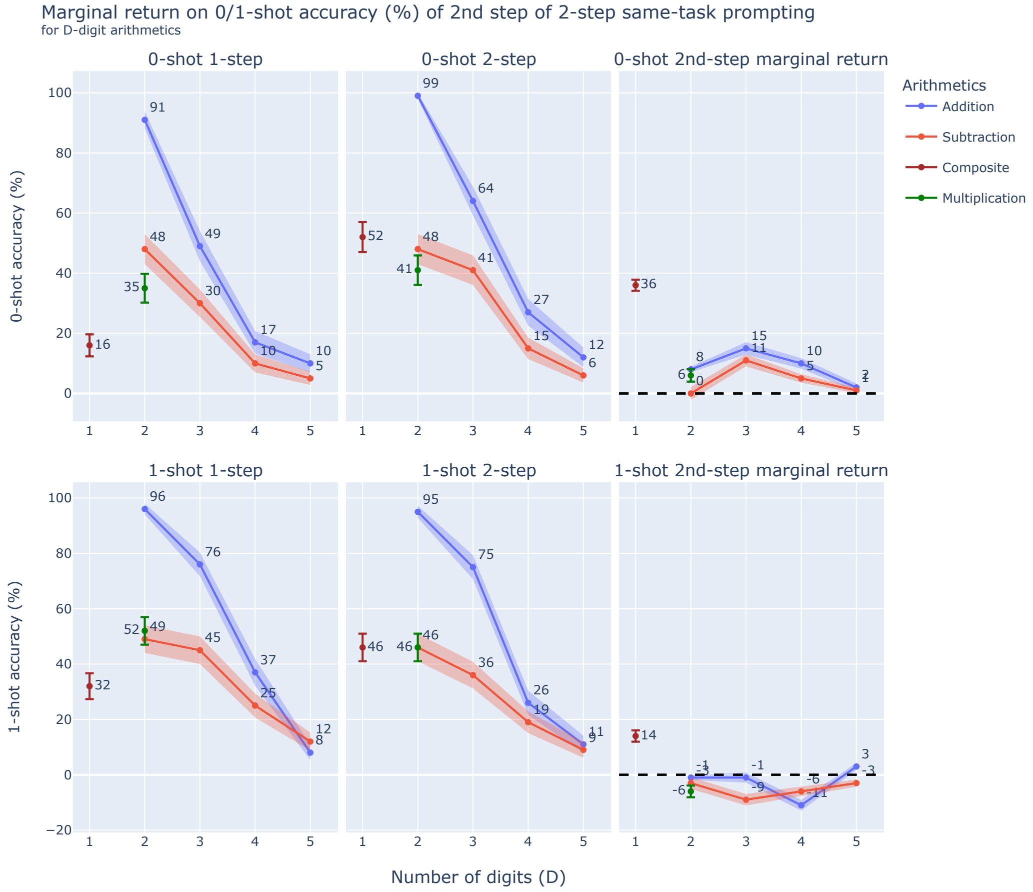
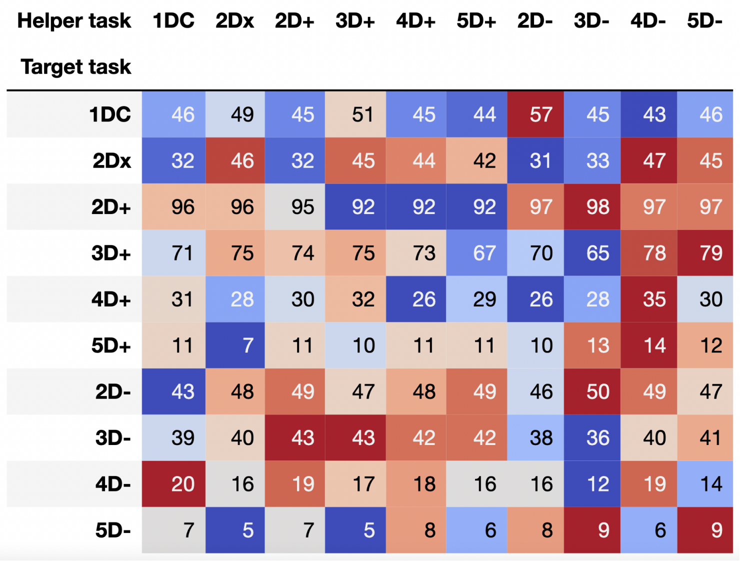


Figure 6: Marginal return from adding the first shot (example question-answer pair) to the two-step {0,1}-shot same-task prompts on the accuracy of instruction-finetuned 175-B GPT-3 in each arithmetic task in {0,1}-shot same-task experiments, where the helper task is the same as the target task of the target question. In this two-step procedure, GPT-3 is first asked to provide a helpful prompt for itself to answer the target question in the second query. Error bands correspond to accuracies 0.3 standard deviation above and below the reported mean accuracy. The black dotted horizontal line at 0 indicates no marginal benefit to adding an extra shot to the few-shot prompts.



Helper task	1DC	2Dx	2D+	3D+	4D+	5D+	2D-	3D-	4D-	5D-
Target task										
1DC	46	49	45	51	45	44	57	45	43	46
2Dx	32	46	32	45	44	42	31	33	47	45
2D+	96	96	95	92	92	92	97	98	97	97
3D+	71	75	74	75	73	67	70	65	78	79
4D+	31	28	30	32	26	29	26	28	35	30
5D+	11	7	11	10	11	11	10	13	14	12
2D-	43	48	49	47	48	49	46	50	49	47
3D-	39	40	43	43	42	42	38	36	40	41
4D-	20	16	19	17	18	16	16	12	19	14
5D-	7	5	7	5	8	6	8	9	6	9

Figure 7: Accuracy (%) of instruction-finetuned 175-B GPT-3 in each arithmetic task in two-step 1-shot cross-task prompting experiments where each prompt is formed by including 1 example question-answer pair from a helper task. In this two-step procedure, GPT-3 is first asked to provide a helpful prompt for itself to answer the target question in the second query. Each row corresponds to a target task, while each column corresponds to a helper task. Red colors and blue colors refer to high and low values respectively for each target task in each row.