Can Generalist Foundation Models Outcompete Special-Purpose Tuning? Case Study in Medicine

Harsha Nori[*‡], Yin Tat Lee[*], Sheng Zhang[*], Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, Renqian Luo, Scott Mayer McKinney[†], Robert Osazuwa Ness, Hoifung Poon, Tao Qin, Naoto Usuyama, Chris White, and Eric Horvitz[‡]

Microsoft

November 2023

Abstract

Generalist foundation models such as GPT-4 have displayed surprising capabilities in a wide variety of domains and tasks. Yet, there is a prevalent assumption that they cannot match specialist capabilities without intensive training of models with specialty knowledge. For example, most explorations to date on medical competency benchmarks have leveraged domain-specific training, as exemplified by efforts on BioGPT and Med-PaLM. We build on a prior study of the specialist capabilities of GPT-4 on medical challenge benchmarks in the absence of special training. In distinction to the intentional use of simple prompting to highlight the model's out-of-the-box capabilities, we perform a systematic exploration of prompt engineering to boost performance. We find that prompting innovation can unlock deeper specialist capabilities and show that GPT-4 easily tops prior leading results for medical question-answering datasets. The prompt engineering methods we explore are general purpose, and make no specific use of domain expertise, removing the need for expert-curated content. Our experimental design carefully controls for overfitting during the prompt engineering process. As a culmination of the study, we introduce Medprompt, based on a composition of several prompting strategies. Medprompt greatly enhances GPT-4's performance and achieves state of the art results on all nine of the benchmark datasets in the MultiMedQA suite. The method outperforms state-of-the-art specialist models such as Med-PaLM 2 by a large margin with an order of magnitude fewer calls to the model. Steering GPT-4 with Medprompt achieves a 27% reduction in error rate on the MedQA dataset (USMLE exam) over the best methods to date achieved with specialist models, and surpasses a score of 90% for the first time. Moving beyond medical challenge problems, we show the power of Medprompt to generalize to other domains and provide evidence for the broad applicability of the approach via studies of the strategy on competency exams in electrical engineering, machine learning, philosophy, accounting, law, nursing, and clinical psychology.

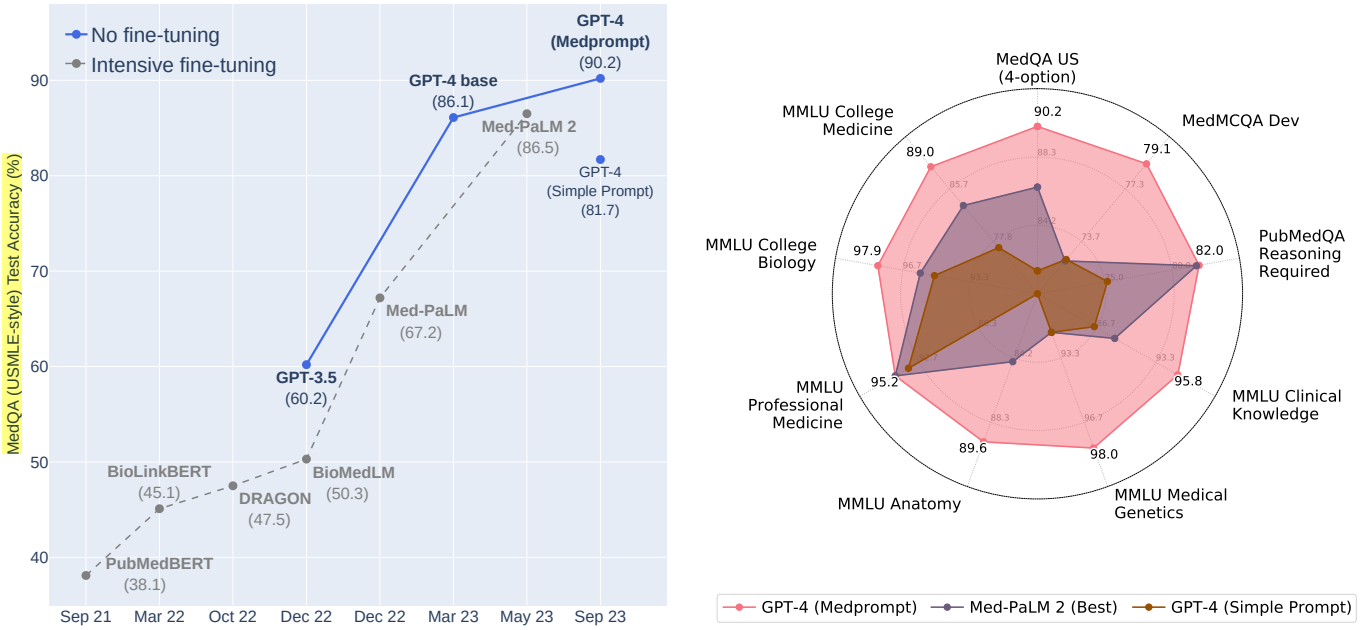
[*]These authors contributed equally.

[†]At OpenAI.

[‡]Correspondence: hanori@microsoft.com, horvitz@microsoft.com

1 Introduction

A long-term aspiration in AI research is to develop principles of computational intelligence and to harness these to build learning and reasoning systems that can perform general problem solving across a diversity of tasks [21, 22]. In line with this goal, large language models, also referred to as foundation models, such as GPT-3 [3] and GPT-4 [24], have demonstrated surprising competencies on a broad swath of tasks without requiring heavy specialized training [4]. These models build on the text-to-text paradigm [31] with investments in compute and data to learn at scale from indiscriminate consumption of large amounts of public web data. Some of these models are tuned via a learning objective to perform general instruction-following via prompts.



(a) (b)

Figure 1: (a) Comparison of performance on MedQA. (b) GPT-4 with Medprompt achieves SoTA on a wide range of medical challenge questions.

A core metric for characterizing the performance of foundation models is the accuracy of next word prediction. Accuracy with next word prediction is found to increase with scale in training data, model parameters, and compute, in accordance with empirically derived "neural model scaling laws" [3, 12]). However, beyond predictions of scaling laws on basic measures such as next word prediction, foundation models show the sudden emergence of numerous problem-solving capabilities at different thresholds of scale [33, 27, 24].

Despite the observed emergence of sets of general capabilities, questions remain about whether truly exceptional performance can be achieved on challenges within specialty areas like medicine in the absence of extensive specialized training or fine-tuning of the general models. Most explorations of foundation model capability on biomedical applications rely heavily on domain- and task-specific fine-tuning. With first-generation foundation models, the community found an unambiguous advantage with domain-specific pretraining, as exemplified by popular models in biomedicine such as

PubMedBERT [10] and BioGPT [19]. But it is unclear whether this is still the case with modern foundation models pretrained at much larger scale.

We focus in this paper on steering foundation models via prompt engineering to excel on a set of medical challenge benchmarks. Med-PaLM 2 attains competitive results on MedQA and other medical challenge problems, via expensive, task-specific fine-tuning of the general PaLM [6] foundation model [29, 30]. In addition to reliance on fine-tuning of the base PaLM model, results on the medical benchmarks for Med-PaLM 2 were generated via use of sophisticated, complex prompting strategies, leveraging exemplars crafted by experts. For example, many of the answers rely on an elaborate two-stage prompt scheme of 44 calls for answering each question.

Shortly after GPT-4 was made public in March 2023, several co-authors of this study showed that the model had impressive biomedical competencies "out-of-the-box" on medical challenge benchmarks. To demonstrate the latent power of GPT-4 on specialty medical expertise, the co-authors purposefully employed a rudimentary prompting strategy [23]. Despite the strong results demonstrated in that study, questions remain about the depth of GPT-4's domain-specific capabilities in the absence of additional special training or tuning.

We present results and methods of a case study on steering GPT-4 to answer medical challenge questions with innovative prompting strategies. We include a consideration of best practices for studying prompting in an evaluative setting, including the holding out of a true eyes-off evaluation set. We discover that GPT-4 indeed possesses deep specialist capabilities that can be evoked via prompt innovation. The performance was achieved via a systematic exploration of prompting strategies. As a design principle, we chose to explore prompting strategies that were inexpensive to execute and not customized for our benchmarking workload. We converged on a top prompting strategy for GPT-4 for medical challenge problems, which we refer to as Medprompt. Medprompt unleashes medical specialist skills in GPT-4 in the absence of expert crafting, easily topping existing benchmarks for all standard medical question-answering datasets. The approach outperforms GPT-4 with the simple prompting strategy and state-of-the-art specialist models such as Med-PaLM 2 by large margins. On the MedQA dataset (USMLE exam), Medprompt produces a 9 absolute point gain in accuracy, surpassing 90% for the first time on this benchmark.

As part of our investigation, we undertake a comprehensive ablation study that reveals the relative significance for the contributing components of Medprompt. We discover that a combination of methods, including in-context learning and chain-of-thought, can yield synergistic effects. Perhaps most interestingly, we find that the best strategy in steering a generalist model like GPT-4 to excel on the medical specialist workload that we study is to use a generalist prompt. We find that GPT-4 benefits significantly from being allowed to design its prompt, specifically with coming up with its own chain-of-thought to be used for in-context learning. This observation echoes other reports that GPT-4 has an emergent self-improving capability via introspection, such as self-verification [9].

We note that the automated chain-of-thought reasoning removes dependency on special human expertise and medical datasets. Thus, despite the name Medprompt, extending from the framing context and research trajectory of our investigation of the capabilities of GPT-4 on medical challenge problems, the methodology doesn't include any components specifically oriented towards medicine. As we explore in Section 5.3, the approach can be applied readily to other domains. We present details on Medprompt to facilitate future studies on steering generalist foundation models to provide specialist advice.

2 Background

2.1 Foundation Models on Medical Challenge Problems

In the era of first-generation foundation models, limited model size and computational resources made domain-specific pretraining advantageous. Models such as PubMedBERT [10], BioLinkBERT [37], DRAGON [36], BioGPT [19], and BioMedLM [2] were pretrained with self-supervised objectives using domain-specific data sources, such as the PubMed corpus and UMLS knowledge graph. Despite their small size and limited computational power, these models demonstrate strong performance in biomedical NLP tasks. More powerful, general-domain foundation models have demonstrated significantly elevated performance in medical challenges without requiring domain-specific pretraining.

Several studies have explored the performance of generalist foundation models on medical challenge problems. In [17], ChatGPT-3.5 was evaluated on questions drawn from United States Medical Licensing Exam (USMLE), and performed at or near the passing threshold without any specialized training. In [23], GPT-4 was shown to exceed the USMLE passing score by over 20 points using simple 5-shot prompting. Other studies have explored the use of foundation models that are specially fine-tuned with medical knowledge.

Other studies have explored the power of relying on explicit tuning with medical knowledge. Med-PaLM [29] and Med-PaLM 2 [30] leverage fine-tuning of the 540B-parameter Flan-PaLM, using instruction prompt tuning. With Med-PaLM, authors asked a panel of five clinicians to prepare their instruction prompt tuning dataset. Med-PaLM 2, built similarly on PaLM 2, relied on instruction-following full fine-tuning and achieved the state-of-the-art performance on medical QA datasets.

We re-examine the capabilities of generalist foundation models without resorting to extensive fine-tuning. We explore diverse prompting strategies to best steer powerful generalist foundation models toward delivering strong performance in specialized domains.

2.2 Prompting Strategies

Prompting in the context of language models refers to the input given to a model to guide the output that it generates. Empirical studies have shown that the performance of foundation models on a specific task can be heavily influenced by the prompt, often in surprising ways. For example, recent work shows that model performance on the GSM8K benchmark dataset can vary by over 10% without any changes to the model's learned parameters [35]. *Prompt engineering* refers to the process of developing effective prompting techniques that enable foundation models to better solve specific tasks. Here, we briefly introduce a few key concepts that serve as building blocks for our Medprompt approach.

In-Context Learning (ICL) is a key capability of foundation models, allowing the models to solve new tasks from just a few task demonstrations [3]. For example, an ICL prompt can be created by preceding a test question with several different examples of questions and desired results. ICL does not require updating model parameters but can offer effects similar to fine-tuning. The choice of examples used in few-shot prompting can substantially influence model performance. In our prior investigation of the performance of GPT-4 on medical challenge problems [23], we expressly limited prompting to basic in-context learning methods such as fixed one-shot and five-shot prompting to demonstrate the ease with which GPT-4 could be steered to perform with excellence.

Chain of Thought (CoT) is a prompting methodology that employs intermediate reasoning steps prior to introducing the sample answer [34]. By breaking down complex problems into a series

of smaller steps, CoT is thought to help a foundation model to generate a more accurate answer. CoT ICL prompting integrates the intermediate reasoning steps of CoT directly into the few-shot demonstrations. As an example, in the Med-PaLM work, a panel of clinicians was asked to craft CoT prompts tailored for complex medical challenge problems [29]. Building on this work, we explore in this paper the possibility of moving beyond reliance on human specialist expertise to mechanisms for generating CoT demonstrations automatically using GPT-4 itself. As we shall describe in more detail, we can do this successfully by providing [question, correct answer] pairs from a training dataset. We find that GPT-4 is capable of autonomously generating high-quality, detailed CoT prompts, even for the most complex medical challenges.

Ensembling is a technique for combining the outputs of multiple model runs to arrive at a more robust or accurate result via combining the separate outputs with functions like averaging, consensus, or majority vote. Ensembling methods employing a technique referred to as *self-consistency* [32] use a sampling method to produce multiple outputs that are then consolidated to identify a consensus output. The diversity of the outputs can be controlled by shifting the "temperature" parameter in a model's generation, where higher temperatures can be viewed as injecting greater amounts of randomness into the generation process. By reordering or *shuffling* components of a few-shot prompt, ensembling techniques can also address the order sensitivity commonly found with foundation models [26, 39], thus improving robustness.

While ensembling can enhance performance, it comes at the cost of increased computational demands. For example, Med-PaLM 2's Ensemble Refinement method used as many as 44 separate inferences for a single question. Due to this computational overhead, we have pursued as a design principle using simpler techniques to avoid excessive inference costs. We report an ablation study in Section 5.2 which explores the potential of further increased performance under increased computational load.

3 Experimental Design

We start with an overview of the medical challenge problem datasets and then outline our testing methodology, designed to avoid overfitting that can occur with intensive iteration on a fixed evaluation dataset.

3.1 Datasets

Our benchmarks, as reported in Section 5 are primarily based on performance of GPT-4 on 9 multiple-choice, biomedical datasets from the MultiMedQA benchmark suite [29]. Specifically, the benchmarks include the following:

- **MedQA** [14] contains multiple choice questions in the style of the Medical Licensing Examination questions used to test medical specialist competency in the United States, Mainland China, and Taiwan. For fair comparison with prior work [29, 30, 23], we focus on the United States subset of the dataset, which has questions in English in the style of the United States Medical Licensing Exam (USMLE). This dataset contains 1273 questions with four multiple choice answers each.

- **MedMCQA** [25] presents mock and historic exam questions in the style of two Indian medical school entrance exams—the AIIMS and NEET-PG. The "dev" subset of the dataset, upon

which we report benchmark results (consistent with prior studies), contains 4183 questions, each with four multiple choice answers.

- **PubMedQA** [15] contains tests requiring a yes, no, or maybe answer to biomedical research questions when given context provided from PubMed abstracts. There are two settings for PubMedQA tests called *reasoning-required* and *reasoning-free*. In the reasoning-free setting, a long-form answer that contains explanations of the abstracts is provided. We report results for the reasoning-required setting, in which the model is only given context from abstracts to use when answering the question. This dataset contains a total of 500 questions.

- **MMLU** [11] is a multitask benchmark suite of 57 different datasets spanning domains across STEM, humanities, and social sciences. We follow prior work [29] and benchmark against a medically relevant subset of MMLU tasks: clinical knowledge, medical genetics, anatomy, professional medicine, college biology, and college medicine.

As we shall see in Section 5.3, we can test the generality of the Medprompt approach by studying its efficacy for competency exams outside the primary focus on medical challenge problems. We test our methodology on two nursing datasets focused on answering NCLEX (National Council Licensure Examinaton) questions and six additional datasets from MMLU covering topics like accounting and law. Details of these datasets are presented in Section 5.3.

3.2 Sound Testing Methodology

While prompting and in-context learning does not change model parameters, a specific choice of prompting strategy can be viewed as a high-level setting or *hyperparameter* of the end-to-end testing process. As a result, we must be cautious about overfitting as part of training and testing, thus providing results that would not generalize out of the training and test sets under consideration. Concerns about overfitting with studies of foundation model performance are similar to the valid concerns in traditional machine learning with overfitting during the hyperparameter optimization process [8]. We wish to avoid analogous overfitting in the prompt engineering process.

Intuitively, a prompt harnessing for examples a lookup table of specific benchmark questions will naturally perform much better on those questions than on unseen problems. A common technique to address this problem in traditional machine learning is to create "test" sets, *which are only evaluated against at the end of the model selection process*. We adopt this important aspect of sound testing methodology for machine learning studies and randomly carved out 20% of each benchmark dataset as an "eyes-off" split that is completely held out from consideration until the final testing phase. That is, the eyes-off data is kept hidden until the end-stage. The data is not examined or optimized against during the prompt engineering process. For simplicity, we apply the same methodology to every dataset in MultiMedQA, as many of the datasets were not published with dedicated train/test splits by the authors. In Section 5.1, we show the stratified performance of Medprompt on "eyes-on" vs. "eyes-off" splits of the MultiMedQA datasets. We find that our performance is quite similar between the two, and that GPT-4 with Medprompt actually performs marginally better on the eyes-off, held out data suggesting that the methods will generalize well to similar questions in the "open world." We have not seen evidence of the use of a similar eyes-off approach in prior studies.

4 Power of Prompting: Exploration and Results

In this section, we detail the three major techniques employed in Medprompt: Dynamic few-shot selection, self-generated chain of thought, and choice shuffle ensembling. After discussing each technique, we review our approach to composing the three methods into the integrated Medprompt.

4.1 Dynamic Few-shot

Few-shot learning [3] is arguably the most effective in-context learning method. With the prompting approach, through a few demonstrations, foundation models quickly adapt to a specific domain and learn to follow the task format. For simplicity and efficiency, the few-shot examples applied in prompting for a particular task are typically fixed; they are unchanged across test examples. This necessitates that the few-shot examples selected are broadly representative and relevant to a wide distribution of text examples. One approach to meeting these requirements is to have domain experts carefully hand-craft *exemplars* [29]. Even so, this approach cannot guarantee that the curated, fixed few-shot examples will be appropriately representative of every test example. In comparison, when available, the task training set can serve as an inexpensive, high-quality source for few-shot examples. If the training set is sufficiently large, we can select different few-shot examples for different task inputs. We refer to this approach as employing dynamic few-shot examples. The method makes use of a mechanism to identify examples based on their similarity to the case at hand [18]. For Medprompt, we did the following to identify representative few shot examples: Given a test example, we choose k training examples that are semantically similar using a k-NN clustering in the embedding space. Specifically, we first use `text-embedding-ada-002`* to embed training questions and test questions as vector representations. Then, for each test question x, we retrieve its nearest k neighbors $x_1, x_2, ..., x_k$ from the training set (according to distance in the embedding space of `text-embedding-ada-002`). Given a pre-defined similarity measure d such as cosine similarity, the neighbors are ordered in such a way that $d(x_i, x) \leq d(x_j, x)$ when $i < j$. Compared with fine-tuning, dynamic few-shot leverages the training data, but does not require billions of updates to model parameters.

*https://openai.com/blog/new-and-improved-embedding-model

4.2 Self-Generated Chain of Thought

Question: A 22-year-old male marathon runner presents to the office with the complaint of right-sided rib pain when he runs long distances. Physical examination reveals normal heart and lung findings and an exhalation dysfunction at ribs 4-5 on the right. Which of the following muscles or muscle groups will be most useful in correcting this dysfunction utilizing a direct method?

(A) anterior scalene (B) latissimus dorsi (C) pectoralis minor (D) quadratus lumborum

Hand-crafted CoT from Med PaLM 2

Expert

Let's solve this step-bystep, referring to authoritative sources as needed. Among the options, only pectoralis minor muscle origins from the outer surfaces of the 3rd to 5th ribs.

GPT-4 generated CoT

GPT-4

The primary muscle involved in rib exhalation is the internal intercostal muscle. However, this option is not listed among the answer choices. Among the provided options, the pectoralis minor can contribute to rib movement, as it originates from the 3rd-5th ribs and can act to depress the ribs during exhalation when the scapula is fixed. Therefore, it could potentially be useful in correcting an exhalation dysfunction at ribs 4-5.

Figure 2: Comparison of expert-crafted and GPT-4-generated chain-of-thought (CoT) prompts. Using a [question, correct answer] pair from a training set, GPT-4 is capable of generating a detailed explanation suitable for use in few-shot CoT demonstrations.

Chain-of-thought (CoT) [34] uses natural language statements, such as "*Let's think step by step*," to explicitly encourage the model to generate a series of intermediate reasoning steps. The approach has been found to significantly improve the ability of foundation models to perform complex reasoning. Most approaches to chain-of-thought center on the use of experts to manually compose few-shot examples with chains of thought for prompting [30]. Rather than rely on human experts, we pursued a mechanism to automate the creation of chain-of-thought examples. We found that we could simply ask GPT-4 to generate chain-of-thought for the training examples using the following prompt:

Self-generated Chain-of-thought Template

```
## Question: {{question}}
{{answer_choices}}
## Answer
```
model generated chain of thought explanation
Therefore, the answer is [*final model answer (e.g. A,B,C,D)*]

Figure 3: Template used to prompt foundation model to generate chain-of-thought explanations automatically (detailed in Section 4.2).

A key challenge with this approach is that self-generated CoT rationales have an implicit risk of including hallucinated or incorrect reasoning chains. We mitigate this concern by having GPT-4 generate both a rationale and an estimation of the most likely answer to follow from that reasoning chain. If this answer does not match the ground truth label, we discard the sample entirely, under the assumption that we cannot trust the reasoning. While hallucinated or incorrect reasoning can still yield the correct final answer (i.e. false positives), we found that this simple label-verification step acts as an effective filter for false negatives.

We observe that, compared with the CoT examples used in Med-PaLM 2 [30], which are hand-crafted by clinical experts, CoT rationales generated by GPT-4 are longer and provide finer-grained step-by-step reasoning logic. Concurrent with our study, recent works [35, 7] also find that foundation models write better prompts than experts do.

4.3 Choice Shuffling Ensemble

While less severe than other foundation models, GPT-4 can exhibit a propensity to favor certain options in multiple choice answers over others (regardless of the option content), i.e., the model can show position bias [1, 16, 40]. To reduce this bias, we propose shuffling the choices and then checking consistency of the answers for the different sort orders of the multiple choice. As a result, we perform choice shuffle and self-consistency prompting. Self-consistency [32] replaces the naive single-path or *greedy* decoding with a diverse set of reasoning paths when prompted multiple times at some temperature> 0, a setting that introduces a degree of randomness in generations. With choice shuffling, we shuffle the relative order of the answer choices before generating each reasoning path. We then select the most consistent answer, i.e., the one that is least sensitive to choice shuffling. Choice shuffling has an additional benefit of increasing the diversity of each reasoning path beyond temperature sampling, thereby also improving the quality of the final ensemble [5]. We also apply this technique in generating intermediate CoT steps for training examples. For each example, we shuffle the choices some number of times and generate a CoT for each variant. We only keep the examples with the correct answer.

4.4 Putting it all together: Medprompt

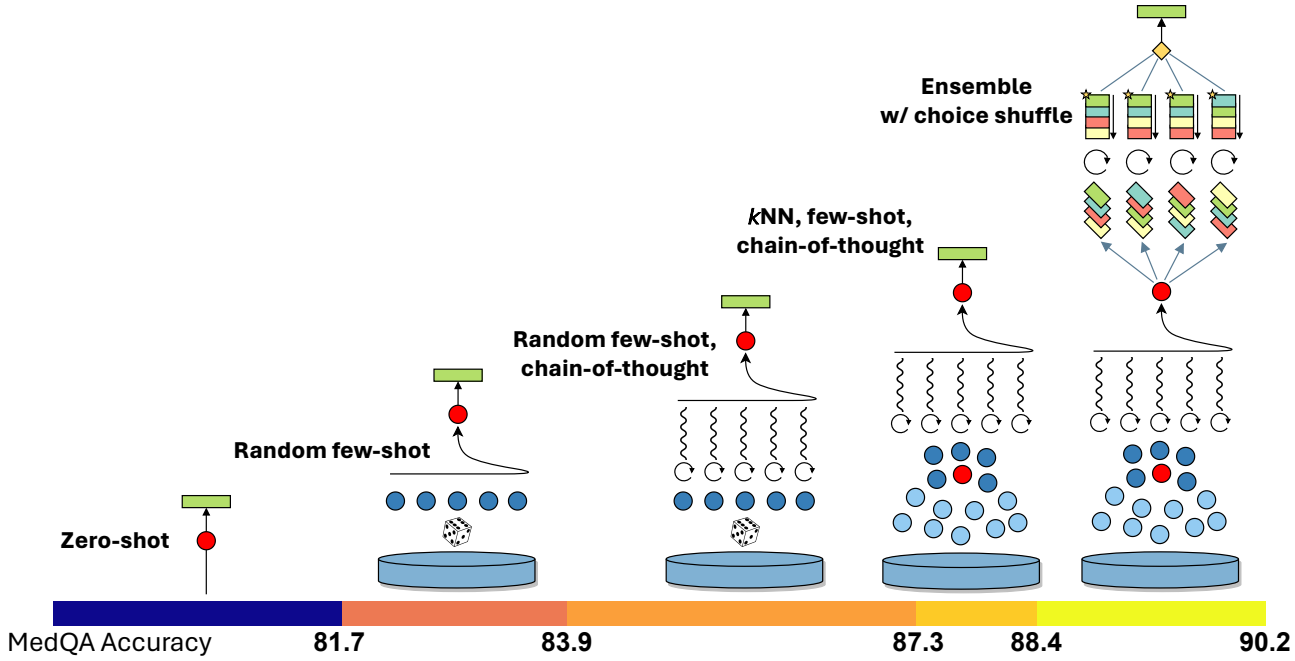


Figure 4: Visual illustration of Medprompt components and additive contributions to performance on the MedQA benchmark. The prompting strategy combines kNN-based few-shot example selection, GPT-4–generated chain-of-thought prompting, and answer-choice shuffled ensembling (see details in Section 4). Relative contributions of each component are shown at the bottom (details in Section 5.2).

Medprompt combines intelligent few-shot exemplar selection, self-generated chain of thought steps, and a majority vote ensemble, as detailed above in Sections 4.1, 4.2, and 4.3, respectively. The composition of these methods yields a general purpose prompt-engineering strategy. A visual depiction of the performance of the Medprompt strategy on the MedQA benchmark, with the additive contributions of each component, is displayed in Figure 4. We provide an a corresponding algorithmic description in Algorithm 1.

Medprompt consists of two stages: a *preprocessing* phase and an *inference* step, where a final prediction is produced on a test case. During preprocessing, each question in the training dataset is passed through a lightweight embedding model to generate an embedding vector (Line 4 in Algorithm 1). We employed OpenAI's text-embedding-ada-002 to create an embedding. For each question, GPT-4 is harnessed to create a chain of thought and a prediction of the final answer (Line 5). If the generated answer is correct and matches the ground truth label, we store the associated question, its embedding vector, the chain of thought, and the answer. Otherwise, we discard the question entirely from our retrieval pool, with the assumption that we cannot trust the reasoning if the model ultimately arrives at the wrong final answer (Lines 6-7).

At inference time, given a test question, we re-embed the test sample with the same embedding model used during pre-processing, and utilize kNN to retrieve similar examples from the preprocessed pool (Lines 12-13). These examples, and their corresponding GPT-4 generated reasoning chains, are structured as context for GPT-4 (Line 14). The test question and corresponding answer choices are

then appended at the end, which serves as the final prompt (Line 17). The model, following the few shot exemplars, then outputs a chain of thought and a candidate answer. Finally, we perform an ensembling process consisting of repeating the steps described above multiple times. We increase diversity by shuffling the answer choices of the test question (Lines 15-16), as detailed in Section 4.3 and Figure 4. To determine the final predicted answer, we select the most frequent answer (Line 20).

Algorithm 1 Algorithmic specification of Medprompt, corresponding to the visual representation of the strategy in Figure 4.

1: **Input**: Development data $\mathcal{D}$, Test question Q
2: **Preprocessing**:
3: **for** `each question` q `in` $\mathcal{D}$ **do**
4: Get an embedding vector v_q for q.
5: Generate a chain-of-thought C_q and an answer A_q with the LLM.
6: **if** Answer A_q is correct **then**
7: Store the embedding vector v_q, chain-of-thought C_q, and answer A_q.
8: **end if**
9: **end for**
10:
11: **Inference Time**:
12: Compute the embedding v_Q for the test question Q.
13: Select the 5 most similar examples $\{(v_{Q_i}, C_{Q_i}, A_{Q_i})\}_{i=1}^{5}$ from the preprocessed training data using KNN, with the distance function as the cosine similarity: $\text{dist}(v_q, v_Q) = 1 - \frac{\langle v_q, v_Q \rangle}{\|v_q\| \|v_Q\|}$.
14: Format the 5 examples as context $\mathcal{C}$ for the LLM.
15: **for** `5 times` **do**
16: Shuffle the answer choices of the test question.
17: Generate a chain-of-thought C_q^k and an answer A_q^k with the LLM and context $\mathcal{C}$.
18: **end for**
19: Compute the majority vote of the generated answers $\{A_q^k\}_{k=1}^{K}$:

$$A^{\text{Final}} = \text{mode}(\{A_q^k\}_{k=1}^{K}),$$

where $\text{mode}(X)$ denotes the most common element in the set X.
20: **Output**: Final answer A^{Final}.

The Medprompt results we report here are configured to use **5** kNN selected few shot exemplars and **5** parallel API calls as part of the choice-shuffle ensemble procedure, which we find strikes a reasonable balance between minimizing inference cost and maximizing accuracy.

Our ablation studies, detailed in Section 5.2, suggest that further improvements may be achieved by increasing these hyperparameter values. For example, by increasing to 20 few-shot exemplars and 11 ensemble items, we achieve a further +0.4% performance on MedQA, setting a new state-of-the-art performance threshold of **90.6**%.

We note that, while Medprompt achieves record performance on medical benchmark datasets, the algorithm is general purpose and is not restricted to the medical domain or to multiple choice question answering. We believe the general paradigm of combining intelligent few-shot exemplar selection, self-generated chain of thought reasoning steps, and majority vote ensembling can be broadly applied

to other problem domains, including less constrained problem solving tasks (see Section 5.3 for details on how this framework can be extended beyond multiple choice questions).

5 Results

Table 1: Performance of different foundation models on multiple choice components of MultiMedQA [29]. GPT-4 with Medprompt outperforms all other models on every benchmark.

Dataset	Flan-PaLM 540B* (choose best)	Med-PaLM 2* (choose best)	GPT-4 (5 shot)	GPT-4 (Medprompt)
MedQA				
US (4-option)	67.6	86.5	81.4	**90.2****
PubMedQA				
Reasoning Required	79.0	81.8	75.2	**82.0**
MedMCQA				
Dev	57.6	72.3	72.4	**79.1**
MMLU				
Clinical Knowledge	80.4	88.7	86.4	**95.8**
Medical Genetics	75.0	92.0	92.0	**98.0**
Anatomy	63.7	84.4	80.0	**89.6**
Professional Medicine	83.8	**95.2**	93.8	**95.2**
College Biology	88.9	95.8	95.1	**97.9**
College Medicine	76.3	83.2	76.9	**89.0**

* Sourced directly from [29] and [30]. "Choose best" refers to a process used in the Med-Palm studies of executing several distinct approaches and selecting the best performing strategy for each dataset among the variety of experimental methods tried. Flan-PaLM 540B and Med-PaLM 2 are also both fine-tuned on subsets of these benchmark datasets. By contrast, every GPT-4 reported number uses a single, consistent strategy across all datasets.

** We achieve 90.6%, as discussed in Section 5.2, with $k = 20$ and 11x ensemble steps. The 90.2% represents "standard" Medprompt performance with $k = 5$ few shot examples and a 5x ensemble.

With harnessing the prompt engineering methods described in Section 4 and their effective combination as Medprompt, GPT-4 achieves state-of-the-art performance on every one of the nine benchmark datasets in MultiMedQA.

5.1 Performance on Eyes-Off Data

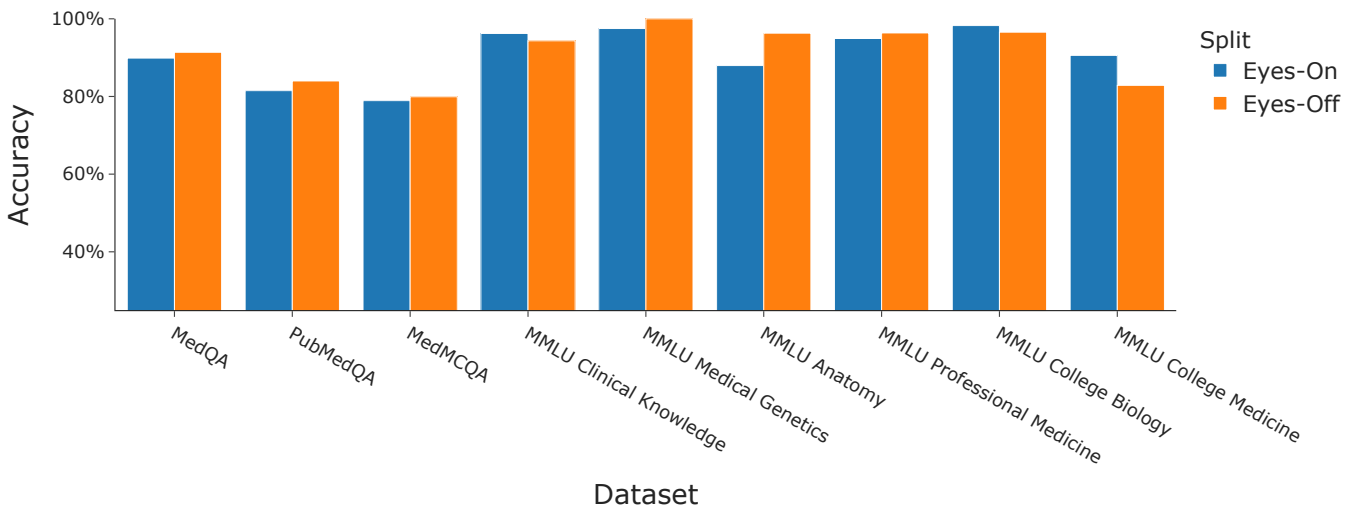


Figure 5: Medprompt evaluation against 20% eyes-off holdout. Medprompt performs better on the eyes-off dataset in the majority of cases.

As introduced in Section 5.1, we evaluated the Medprompt prompting design on a held-out "eyes-off" subset of each benchmark dataset to check for overfitting risk. GPT-4 with Medprompt achieved an average performance of 90.6% on the eyes-on data, and an average performance of 91.3% on the eyes-off data, suggesting that the prompt engineering process likely did not lead to overfitting on MultiMedQA datasets. As additional evidence, the performance on eyes-off data was better in 6/9 of the benchmark datasets (Figure 5).

5.2 Insights about Medprompt Components via Ablation Studies

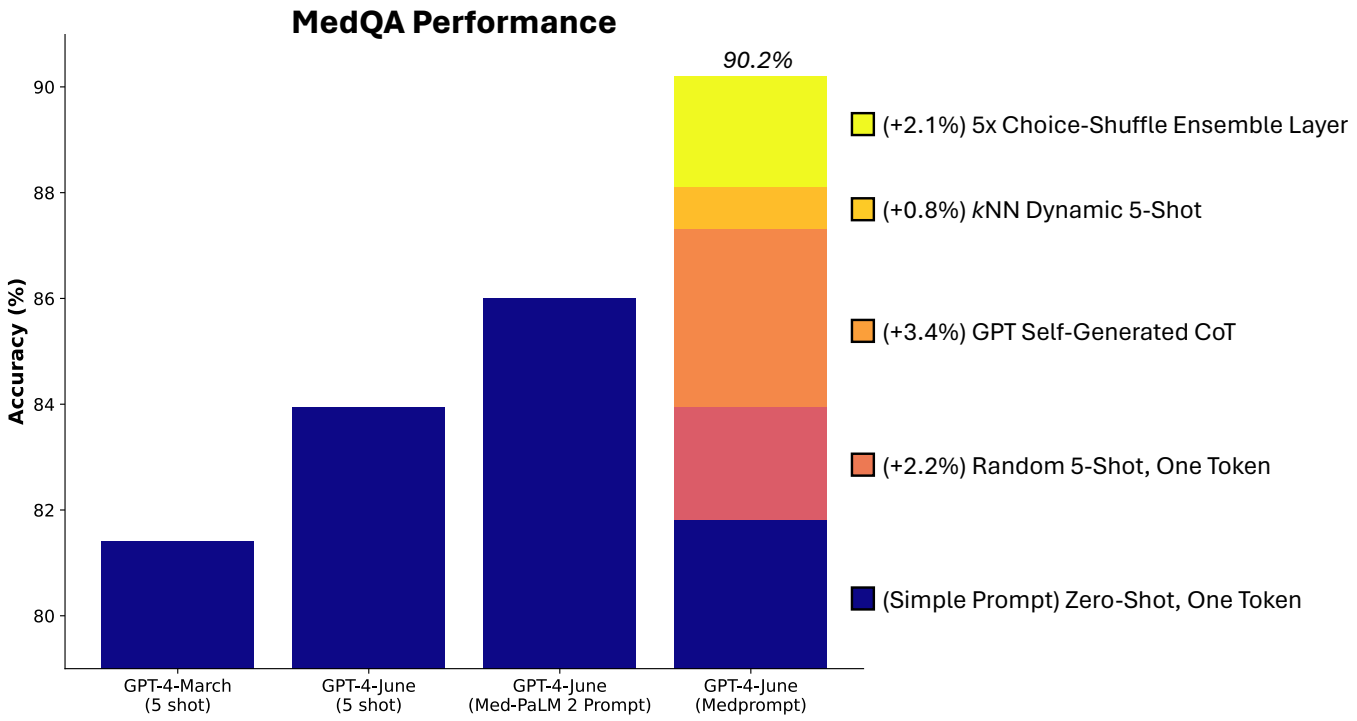


Figure 6: Identification of the relative contributions of different components of Medprompt via an ablation study.

Figure 6 shows the results of an ablation study conducted on the MedQA dataset, in an attempt to understand the relative contributions of each technique in Medprompt. The blue bars represent prior work from [23], and establish baselines for the Medprompt methodology. We then iteratively layered in each technique, and measured the relative difference in performance from each incremental change. As outlined in Section 4.4, our base Medprompt strategy uses 5 kNN-curated few-shot exemplars and ensembles 5 API-calls together. We also experimented with setting up to 20 few-shot exemplars and up to 11 steps in the ensemble. We found that performance does increase marginally to 90.6%, with additional few-shot exemplars and more ensemble steps. This suggests that further improvements on benchmarks may yet be possible, with a corresponding increase in inference time cost and complexity. The introduction of chain-of-thought steps, as described in Section 4, contributed the most to performance (+3.4%), followed by few-shot prompting and choice shuffle ensembling (+2.2% each).

The techniques we use are not statistically independent – therefore, the order in which we test the contribution of each method matters. Our choice of ordering for this ablation study is subjective and based on the relative complexity of the technique introduced. A more theoretically sound method for credit allocation in the ablation study would involve the calculation of game-theoretic Shapley values [28], which takes exponentially more model evaluations to test every potential permutation of orderings. We leave this to future work and encourage readers to think of the specific numbers in the ablation studies as reasonable approximations of relative contributions.

Table 2: Ablation study on expert-crafted chain-of-thought (CoT) vs. GPT-4 self-generated CoT. Both use fixed 5-shot examples, with no ensemble.

	MedQA US (4-option)
Expert-crafted CoT prompt from [30]	83.8
GPT-4's self-generated CoT prompt	86.9 (+3.1)

Apart from the stack of incremental changes, we compare the expert-crafted chain-of-thought (CoT) prompt used in Med-PaLM 2 [30] with the CoT prompt automatically generated by GPT-4 (Section 4.2). We evaluate GPT-4 using both prompts, with fixed 5-shot examples, no ensemble. Table 2 reports their accuracy on the MedQA dataset. GPT-4's self-generated CoT outperforms the expert-crafted one by 3.1 absolute points. We notice that compared with the expert-crafted CoT used in Med-PaLM 2, CoT rationales generated by GPT-4 are longer and provide finer-grained step-by-step reasoning logic. One potential explanation is that GPT-4 generated CoT may be better suited to the model's own strengths and limitations, which could lead to improved performance when compared to the expert-crafted one. Another potential explanation is that expert-crafted CoT may contain implicit biases or assumptions that may not hold for all questions in the MedQA dataset, whereas GPT-4 generated CoT may be more neutral and generalizable across different questions.

5.3 Generalization: Cross-Domain Exploration of Medprompt

We argue that the composition of prompt engineering techniques employed in Medprompt, based on a combination of dynamic few shot selection, self-generated chain of thought, and choice shuffle ensembling, have general purpose application. They are not custom-tailored to the MultiMedQA benchmark datasets. To validate this, we further tested the final Medprompt methodology on six additional, diverse datasets from the MMLU benchmark suite covering challenge problems in the following subjects: electrical engineering, machine learning, philosophy, professional accounting, professional law, and professional psychology. We further sourced two additional datasets answering NCLEX (National Council Licensure Examination) style questions, the exam required to practice as a registered nurse in the United States.

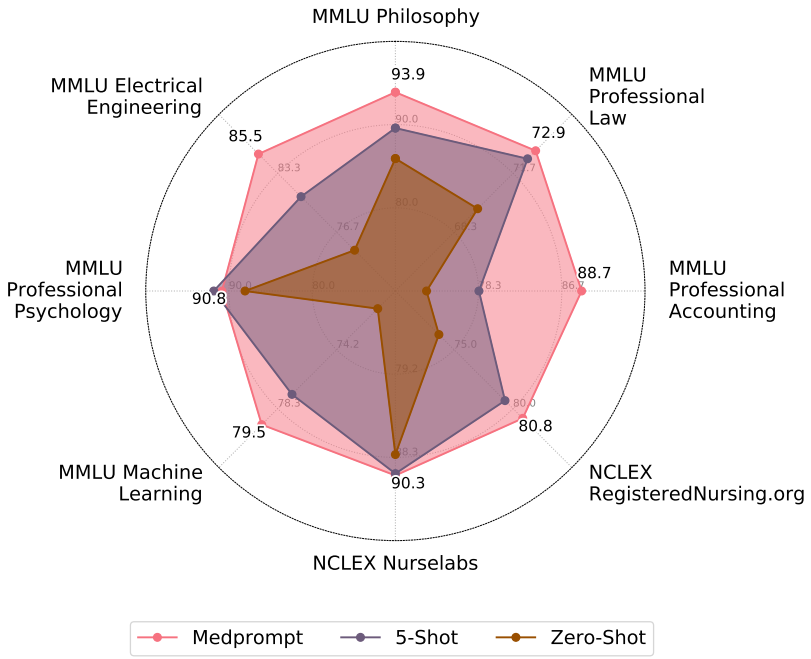


Figure 7: GPT-4 performance with three different prompting strategies on out of domain datasets. Zero-shot and five-shot approaches represent baselines and mirror the methodology followed in [23].

Figure 7 shows GPT-4's performance on these diverse, out of domain dataset with Medprompt alongside zero-shot and five-shot prompts (with random exemplar selection). Across these datasets, Medprompt provides an average improvement of +7.3% over baseline zero-shot prompting. By comparison, Medprompt provided a +7.1% improvement over the same zero-shot baseline on the MultiMedQA datasets studied in this paper. We emphasize that the similarity of improvement across datasets from different distributions demonstrates the generality of the Medprompt approach. While beyond the scope of this paper, we believe the general framework underlying MedPrompt—a combination of few shot learning and chain-of-thought reasoning wrapped in an ensemble layer—can further generalize in applicability beyond the multiple choice question/answer setting with minor algorithmic modifications. For example, in an open-text generation setting, the ensemble layer may not be able to rely on a direct majority vote, but instead may aggregate by selecting the answer closest to all other answers in an embedding space. Another option would be to concatenate each of the K generated pieces of text in a structured format and ask the model to select the most likely option, in the style of Ensemble Refinement [30]. We leave as future work exploration of the space of algorithmic modifications to other settings.

6 Limitations and Risks

Our paper highlights the power of systematic prompt engineering for steering generalist foundation models to amplify the specialist abilities of GPT-4 on medical challenge problems. We now share reflections on limitations and future directions from our assessment.

As foundation models are trained on massive, internet-scale datasets, strong performance on benchmark problems may be due to memorization or leakage effects, where direct test samples have previously been observed by the model during training. In our previous study, which assessed the

performance of GPT-4 on the datasets studied in this work with basic prompting [23], we introduced and ran a blackbox testing algorithm (MELD) which was unable to discover evidence of memorization. However, blackbox testing approaches like MELD are unable to guarantee that data has not been seen before. We also separately assessed GPT-4's performance on USMLE questions that were behind a paywall and, thus, not available on the public internet, and saw similarly strong performance [23]. In this study, we adopted standard machine learning best practices to control for overfitting and leakage during the prompt engineering process (Section 5.1). However, concerns of benchmark contamination during training remain.

Further, we note that the strong performance of GPT-4 with Medprompt cannot be taken to demonstrate real-world efficacy of the model and methods on open-world healthcare tasks [23]. While we are excited about the ability to steer foundations models to become top specialists on the benchmarks, we are cautious about taking the performance of the prompting strategies and model output to mean that the methods will be valuable in the practice of medicine in the open world, whether for automated or assisting healthcare professionals with administrative tasks, clinical decision support, or patient engagement in the open world. To be clear, the medical challenge problems that we and others have studied are designed for testing human competencies in selected domains. Such competency tests are typically framed as sets of multiple choice questions. Although such challenge problems are a common evaluation method and cover diverse topics, they do not capture the range and complexity of medical tasks that healthcare professionals face in actual practice. Thus, the pursuit of tests as proxies for real-world competency and the focus on multiple-choice style answers are limitations when it comes to transferring strong performance on speciality benchmarks to real-world performance. Futhermore, while we believe that the MedPrompt strategy can be adapted to non-multiple choice settings, we did not explicitly test these proposed adaptations on benchmarks in this work.

We note that foundation models can generate erroneous information (sometimes referred to as *hallucinations*) which may compromise generations and advice. While improvements in prompting strategies may lead to reductions in hallucinations and better overall accuracy, they may also make any remaining hallucinations even harder to detect. Promising directions include efforts on probabilistic calibration of generations, providing end-users with trustworthy measures of confidence in output. In our prior study, we found that GPT-4 was well-calibrated and could provide trustable measures of its confidence on multiple choice test questions [23].

We must also remain aware of biases in the output of foundation models. We do not yet understand how optimization in pursuit of top-level performance could influence other goals, such as equitable performance. It is vital to balance the pursuit of overall accuracy with equitable performance across different subpopulations to avoid exacerbating existing disparities in healthcare. Prior work has highlighted the need to understand and address biases in AI systems. The challenge of bias and fairness remains relevant and pressing in the context of model optimization, fine-tuning, and prompt engineering [13, 20, 38].

7 Summary and Conclusions

We presented background, methods, and results of a study of the power of prompting to unleash top-performing specialist capabilities of GPT-4 on medical challenge problems, without resorting to special fine-tuning nor reliance on human specialist expertise for prompt construction. We shared best practices for evaluating performance, including the importance of evaluating model capabilities

on an eyes-off dataset. We reviewed a constellation of prompting strategies and showed how they could be studied and combined via a systematic exploration. We found a significant amount of headroom in boosting specialist performance via steering GPT-4 with a highly capable and efficient prompting strategy.

We described the composition of a set of prompting methods into Medprompt, the best performing prompting strategy we found for steering GPT-4 on medical challenge problems. We showed how Medprompt can steer GPT-4 to handily top existing charts for all standard medical question-answering datasets, including the performance by Med-PaLM 2, a specialist model built via fine-tuning with specialist medical data and guided with handcrafted prompts authored by expert clinicians. Medprompt unlocks specialty skills on MedQA delivering significant gains in accuracy over the best performing model to date, surpassing 90% for the first time on the benchmark.

During our exploration, we found that GPT-4 can be tasked with authoring sets of custom-tailored chain-of-thought prompts that outperform hand-crafted expert prompts. We pursued insights about the individual contributions of the distinct components of the Medprompt strategy via ablation studies that demonstrate the relative importance of each component. We set aside eyes-off evaluation case libraries to avoid overfitting and found that the strong results by Medprompt are not due to overfitting. We explored the generality of Medprompt via performing studies of its performance on a set of competency evaluations in six fields outside of medicine, including electrical engineering, machine learning, philosophy, accounting, law, nursing, and clinical psychology. The findings in disparate fields suggests that Medprompt and its derivatives will be valuable in unleashing specialist capabilities of foundation models for numerous disciplines. We see further possibilities for refining prompts to unleash speciality capabilities from generalist foundation models, particularly in the space of adapting the general MedPrompt strategy to non multiple choice questions. For example, we see an opportunity to build on the Medprompt strategy of using GPT-4 to compose its own powerful chain of thought examples and then employ them in prompting. Research directions moving forward include further investigation of the abilities of foundation models to reflect about and compose few-shot examples and to weave these into prompts.

While our investigation focuses on exploring the power of prompting generalist models, we believe that fine-tuning, and other methods of making parametric updates to foundation models are important research avenues to explore, and may offer synergistic benefits to prompt engineering. We maintain that both approaches should be judiciously explored for unleashing the potential of foundation models in high-stakes domains like healthcare.

Acknowledgments

We thank Sébastien Bubeck, Peter Durlach, Peter Lee, Matthew Lungren, Satya Nadella, Joe Petro, Kevin Scott, Desney Tan, and Paul Vozila for discussion and feedback.

References

[1] Niels J. Blunch. Position bias in multiple-choice questions. *Journal of Marketing Research*, 21(2):216–220, 1984.

[2] Elliot Bolton, David Hall, Michihiro Yasunaga, Tony Lee, Chris Manning, and Percy Liang. Biomedlm, 2022. Stanford Center for Research on Foundation Models.

[3] 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. Language models are few-shot learners, 2020.

[4] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

[5] Rich Caruana, Alexandru Niculescu-Mizil, Geoff Crew, and Alex Ksikes. Ensemble selection from libraries of models. In *Proceedings of the twenty-first international conference on Machine learning*, page 18, 2004.

[6] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

[7] Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution. *arXiv preprint arXiv:2309.16797*, 2023.

[8] Matthias Feurer and Frank Hutter. Hyperparameter optimization. *Automated machine learning: Methods, systems, challenges*, pages 3–33, 2019.

[9] Zelalem Gero, Chandan Singh, Hao Cheng, Tristan Naumann, Michel Galley, Jianfeng Gao, and Hoifung Poon. Self-verification improves few-shot clinical information extraction. In *ICML 3rd Workshop on Interpretable Machine Learning in Healthcare (IMLH)*, 2023.

[10] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23, 2021.

[11] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

[12] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[13] Ayanna M. Howard, Cha Zhang, and Eric Horvitz. Addressing bias in machine learning algorithms: A pilot study on emotion recognition for intelligent systems. *2017 IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, pages 1–7, 2017.

[14] Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421, 2021.

[15] Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William W Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering. *arXiv preprint arXiv:1909.06146*, 2019.

[16] Miyoung Ko, Jinhyuk Lee, Hyunjae Kim, Gangwoo Kim, and Jaewoo Kang. Look at the first sentence: Position bias in question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1109–1121, Online, November 2020. Association for Computational Linguistics.

[17] Tiffany H Kung, Morgan Cheatham, Arielle Medenilla, Czarina Sillos, Lorie De Leon, Camille Elepaño, Maria Madriaga, Rimel Aggabao, Giezel Diaz-Candido, James Maningo, et al. Performance of chatgpt on usmle: Potential for ai-assisted medical education using large language models. *PLoS digital health*, 2(2):e0000198, 2023.

[18] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3?, 2021.

[19] Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in Bioinformatics*, 23(6):bbac409, 2022.

[20] Michael Madaio, Lisa Egede, Hariharan Subramonyam, Jennifer Wortman Vaughan, and Hanna Wallach. Assessing the fairness of ai systems: AI practitioners' processes, challenges, and needs for support. In *25th ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW 2022)*, February 2022.

[21] John McCarthy, Marvin L Minsky, Nathaniel Rochester, and Claude E Shannon. A proposal for the Dartmouth summer research project on artificial intelligence, August 31, 1955. *AI magazine*, 27(4):12–12, 2006.

[22] Allen Newell, John C Shaw, and Herbert A Simon. Report on a general problem solving program. In *IFIP congress*, volume 256, page 64. Pittsburgh, PA, 1959.

[23] Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. Capabilities of GPT-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*, 2023.

[24] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023.

[25] Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on Health, Inference, and Learning*, pages 248–260. PMLR, 2022.

[26] Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. *arXiv preprint arXiv:2308.11483*, 2023.

[27] Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage?, 2023.

[28] Lloyd S Shapley et al. A value for n-person games. 1953.

[29] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *arXiv preprint arXiv:2212.13138*, 2022.

[30] Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Le Hou, Kevin Clark, Stephen Pfohl, Heather Cole-Lewis, Darlene Neal, Mike Schaekermann, Amy Wang, Mohamed Amin, Sami Lachgar, Philip Mansfield, Sushant Prakash, Bradley Green, Ewa Dominowska, Blaise Aguera y Arcas, Nenad Tomasev, Yun Liu, Renee Wong, Christopher Semturs, S. Sara Mahdavi, Joelle Barral, Dale Webster, Greg S. Corrado, Yossi Matias, Shekoofeh Azizi, Alan Karthikesalingam, and Vivek Natarajan. Towards expert-level medical question answering with large language models, 2023.

[31] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.

[32] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023.

[33] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. Survey Certification.

[34] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

[35] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*, 2023.

[36] Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D. Manning, Percy Liang, and Jure Leskovec. Deep bidirectional language-knowledge graph pretraining. In *Neural Information Processing Systems (NeurIPS)*, 2022.

[37] Michihiro Yasunaga, Jure Leskovec, and Percy Liang. Linkbert: Pretraining language models with document links. In *Association for Computational Linguistics (ACL)*, 2022.

[38] Travis Zack, Eric Lehman, Mirac Suzgun, Jorge A. Rodriguez, Leo Anthony Celi, Judy Gichoya, Dan Jurafsky, Peter Szolovits, David W. Bates, Raja-Elie E. Abdulnour, Atul J. Butte, and Emily Alsentzer. Coding inequity: Assessing gpt-4's potential for perpetuating racial and gender biases in healthcare. *medRxiv*, 2023.

[39] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors, 2023.

[40] 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. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.