Emergent Abilities of Large Language Models

Jason Wei[1] **Yi Tay**[1] **Rishi Bommasani**[2] **Colin Raffel**[3]
Barret Zoph[1] **Sebastian Borgeaud**[4] **Dani Yogatama**[4] **Maarten Bosma**[1]
Denny Zhou[1] **Donald Metzler**[1] **Ed H. Chi**[1] **Tatsunori Hashimoto**[2]
Oriol Vinyals[4] **Percy Liang**[2] **Jeff Dean**[1] **William Fedus**[1]

[1]Google Research [2]Stanford University [3]UNC Chapel Hill [4]DeepMind

Abstract

Scaling up language models has been shown to predictably improve performance and sample efficiency on a wide range of downstream tasks. This paper instead discusses an unpredictable phenomenon that we refer to as *emergent abilities* of large language models. We consider an ability to be emergent if it is not present in smaller models but is present in larger models. Thus, emergent abilities cannot be predicted simply by extrapolating the performance of smaller models. The existence of such emergence implies that additional scaling could further expand the range of capabilities of language models.

1 Introduction

Language models have revolutionized natural language processing (NLP) in recent years. It is now well-known that increasing the scale of language models (e.g., training compute, model parameters, etc.) can lead to better performance and sample efficiency on a range of downstream NLP tasks (Devlin et al., 2019; Brown et al., 2020, *inter alia*). In many cases, the effect of scale on performance can often be methodologically predicted via scaling laws—for example, scaling curves for cross-entropy loss have been shown to empirically span more than seven orders of magnitude (Kaplan et al., 2020; Hoffmann et al., 2022). On the other hand, performance for certain downstream tasks counterintuitively does not appear to continuously improve as a function of scale, and such tasks cannot be predicted ahead of time (Ganguli et al., 2022).

This paper is about the unpredictable phenomena of *emergent abilities* of large language models. Emergence as an idea has been long discussed in domains such as physics, biology, and computer science (Anderson, 1972; Hwang et al., 2012; Forrest, 1990; Corradini and O'Connor, 2010; Harper and Lewis, 2012, *inter alia*). We will consider the following general definition of emergence, adapted from Steinhardt (2022) and rooted in a 1972 essay called "More Is Different" by Nobel prize-winning physicist Philip Anderson (Anderson, 1972):

> *Emergence is when quantitative changes in a system result in qualitative changes in behavior.*

Here we will explore emergence with respect to model scale, as measured by training compute and number of model parameters. Specifically, we define *emergent abilities of large language models* as abilities that are not present in smaller-scale models but are present in large-scale models; thus they cannot be predicted by simply extrapolating the performance improvements on smaller-scale models (§2).[1] We survey emergent abilities as observed in a range of prior work, categorizing them in settings such as few-shot prompting (§3) and augmented prompting strategies (§4). Emergence motivates future research on why such abilities are acquired and whether more scaling will lead to further emergent abilities, which we highlight as important questions for the field (§5).

[1]This survey focuses on pre-trained Transformer language models. Emergent abilities in NLP more broadly, however, could go back to Miller et al. (2004), Liang (2005), or earlier.

2 Emergent Abilities Definition

As a broad concept, emergence is often used informally and can be reasonably interpreted in many different ways. In this paper, we will consider a focused definition of emergent abilities of large language models:

> *An ability is emergent if it is not present in smaller models but is present in larger models.*

Emergent abilities would not have been directly predicted by extrapolating a scaling law (i.e. consistent performance improvements) from small-scale models. When visualized via a scaling curve (x-axis: model scale, y-axis: performance), emergent abilities show a clear pattern—performance is near-random until a certain critical threshold of scale is reached, after which performance increases to substantially above random. This qualitative change is also known as a *phase transition*—a dramatic change in overall behavior that would not have been foreseen by examining smaller-scale systems (Huberman and Hogg, 1987).

Today's language models have been scaled primarily along three factors: amount of computation, number of model parameters, and training dataset size (Kaplan et al., 2020; Hoffmann et al., 2022). In this paper, we will analyze scaling curves by plotting the performance of different models where training compute for each model is measured in FLOPs on the x-axis (Hoffmann et al., 2022). Because language models trained with more compute tend to also have more parameters, we additionally show plots with number of model parameters as the x-axis in the Appendix C. Using training FLOPs or model parameters as the x-axis produces curves with similar shapes due to the fact that most dense Transformer language model families have scaled training compute roughly proportionally with model parameters (Kaplan et al., 2020).

Training dataset size is also an important factor, but we do not plot capabilities against it because many language model families use a fixed number of training examples for all model sizes (Brown et al., 2020; Rae et al., 2021; Chowdhery et al., 2022). Although we focus on training computation and model size here, there is not a single proxy that adequately captures all aspects of scale. For example, Chinchilla (Hoffmann et al., 2022) has one-fourth as many parameters as Gopher (Rae et al., 2021) but uses similar training compute; and sparse mixture-of-expert models have more parameters per training/inference compute than dense models (Fedus et al., 2021; Du et al., 2021). Overall, it may be wise to view emergence as a function of many correlated variables. For example, later in Figure 4 we will also plot emergence as a function of WikiText103 perplexity (Merity et al., 2016), which happens to closely correlate with training computation for Gopher/ Chinchilla (though this correlation may not hold in the long-run).

Note that the scale at which an ability is first observed to emerge depends on a number of factors and is not an immutable property of the ability. For instance, emergence may occur with less training compute or fewer model parameters for models trained on higher-quality data. Conversely, emergent abilities also crucially depend on other factors such as not being limited by the amount of data, its quality, or the number of parameters in the model. Today's language models are likely not trained optimally (Hoffmann et al., 2022), and our understanding of how to best train models will evolve over time. Our goal in this paper is not to characterize or claim that a specific scale is required to observe emergent abilities, but rather, we aim to discuss examples of emergent behavior in prior work.

3 Few-Shot Prompted Tasks

We first discuss emergent abilities in the *prompting* paradigm, as popularized by GPT-3 (Brown et al., 2020).[2] In prompting, a pre-trained language model is given a prompt (e.g. a natural language instruction) of a task and completes the response without any further training or gradient updates to its parameters. Brown et al. (2020) proposed *few-shot prompting*, which includes a few input-output examples in the model's context (input) as a preamble before asking the model to perform the task for an unseen inference-time example. An example prompt is shown in Figure 1 below.

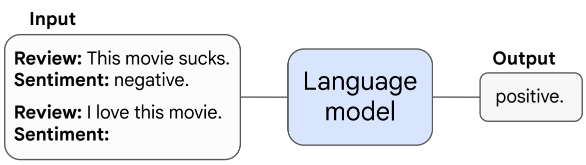

Figure 1: Example of an input and output for few-shot prompting.

[2]Though GPT-3 popularized prompting, the task setup has existed since before GPT-3 (Trinh and Le, 2018; McCann et al., 2018; Radford et al., 2019; Raffel et al., 2020).

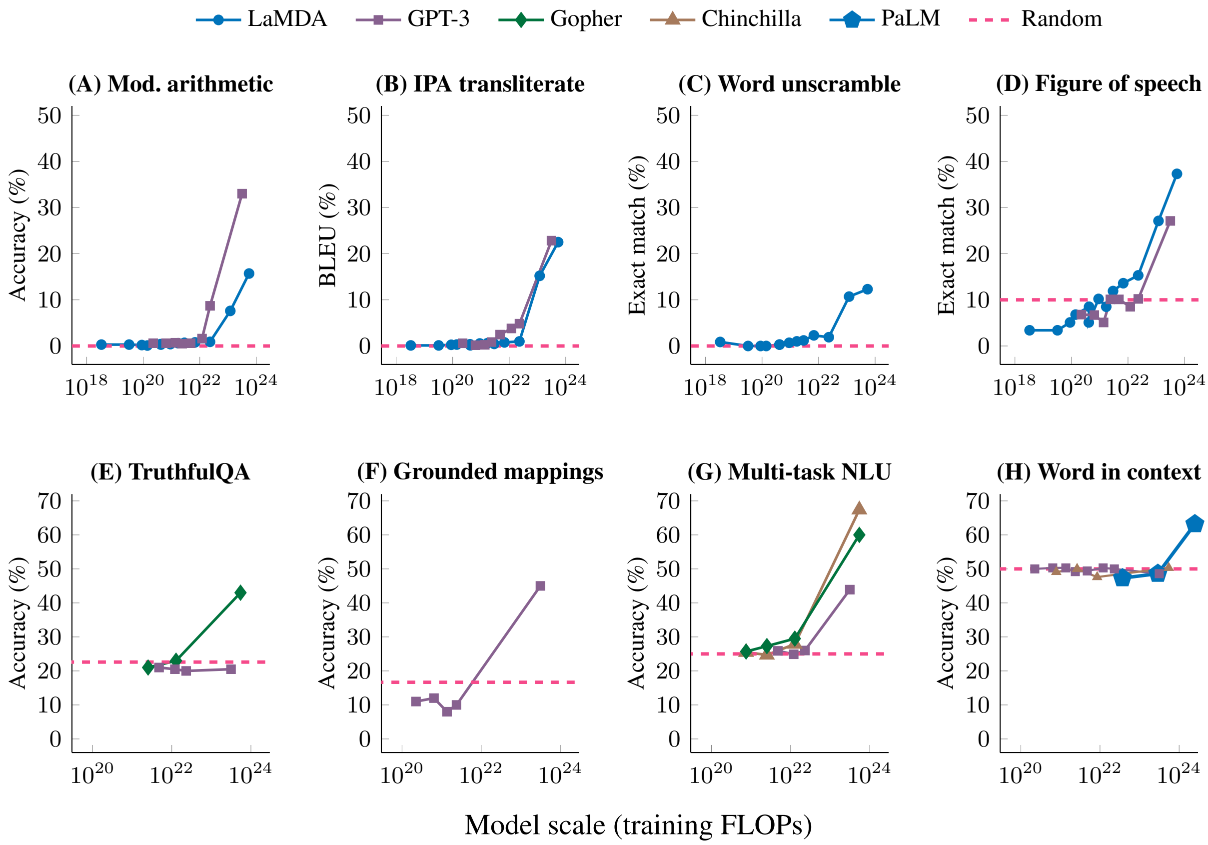

Figure 2: Eight examples of emergence in the few-shot prompting setting. Each point is a separate model. The ability to perform a task via few-shot prompting is emergent when a language model achieves random performance until a certain scale, after which performance significantly increases to well-above random. Note that models that used more training compute also typically have more parameters—hence, we show an analogous figure with number of model parameters instead of training FLOPs as the x-axis in Figure 7. A–D: BIG-Bench (2022), 2-shot. E: Lin et al. (2021) and Rae et al. (2021). F: Patel and Pavlick (2022). G: Hendrycks et al. (2021), Rae et al. (2021), and Hoffmann et al. (2022). H: Brown et al. (2020), Hoffmann et al. (2022), and Chowdhery et al. (2022) on the WiC benchmark (Pilehvar and Camacho-Collados, 2019).

The ability to perform a task via few-shot prompting is emergent when a model has random performance until a certain scale, after which performance increases to well-above random. Figure 2 shows eight such emergent abilities spanning five language model families from various work.

BIG-Bench. Figure 2A–D depicts four emergent few-shot prompted tasks from BIG-Bench, a crowd-sourced suite of over 200 benchmarks for language model evaluation (BIG-Bench, 2022). Figure 2A shows an arithmetic benchmark that tests 3-digit addition and subtraction, as well as 2-digit multiplication. GPT-3 and LaMDA (Thoppilan et al., 2022) have close-to-zero performance for several orders of magnitude of training compute, before performance jumps to sharply above random at $2 \cdot 10^{22}$ training FLOPs (13B parameters) for GPT-3, and 10^{23} training FLOPs (68B parameters) for LaMDA. Similar emergent behavior also occurs at around the same model scale for other tasks, such as transliterating from the International Phonetic Alphabet (Figure 2B), recovering a word from its scrambled letters (Figure 2C), and detecting figures of speech (Figure 2D). Even more emergent abilities from BIG-Bench are given in Table 1.

TruthfulQA. Figure 2E shows few-shot prompted performance on the TruthfulQA benchmark, which measures the ability to answer questions truthfully (Lin et al., 2021). This benchmark is adversarially curated against GPT-3 models, which do not perform above random, even when scaled to the largest model size. Small Gopher models also do not perform above random until scaled up to the largest model of $5 \cdot 10^{23}$ training FLOPs (280B parameters), for which performance jumps to more than 20% above random (Rae et al., 2021).

Grounded conceptual mappings. Figure 2F shows the task of grounded conceptual mappings, where language models must learn to map a conceptual domain, such as a cardinal direction, represented in a textual grid world (Patel and Pavlick, 2022). Again, performance only jumps to above random using the largest GPT-3 model.

Multi-task language understanding. Figure 2G shows the Massive Multi-task Language Understanding (MMLU) benchmark, which aggregates 57 tests covering a range of topics including math, history, law, and more (Hendrycks et al., 2021). For GPT-3, Gopher, and Chinchilla, models of $\sim 10^{22}$ training FLOPs ($\sim$10B parameters) or smaller do not perform better than guessing on average over all the topics, scaling up to 3–5 $\cdot 10^{23}$ training FLOPs (70B–280B parameters) enables performance to substantially surpass random. This result is striking because it could imply that the ability to solve knowledge-based questions spanning a large collection of topics might require scaling up past this threshold (for dense language models without retrieval or access to external memory).

Word in Context. Finally, Figure 2H shows the Word in Context (WiC) benchmark (Pilehvar and Camacho-Collados, 2019), which is a semantic understanding benchmark. Notably, GPT-3 and Chinchilla fail to achieve one-shot performance of better than random, even when scaled to their largest model size of $\sim 5 \cdot 10^{23}$ FLOPs. Although these results so far may suggest that scaling alone may not enable models to solve WiC, above-random performance eventually emerged when PaLM was scaled to $2.5 \cdot 10^{24}$ FLOPs (540B parameters), which was much larger than GPT-3 and Chinchilla.

4 Augmented Prompting Strategies

Although few-shot prompting is perhaps currently the most common way of interacting with large language models, recent work has proposed several other prompting and finetuning strategies to further augment the abilities of language models. If a technique shows no improvement or is harmful when compared to the baseline of not using the technique until applied to a model of a large-enough scale, we also consider the technique an emergent ability.

Multi-step reasoning. Reasoning tasks, especially those involving multiple steps, have been challenging for language models and NLP models more broadly (Rae et al., 2021; Bommasani et al.,

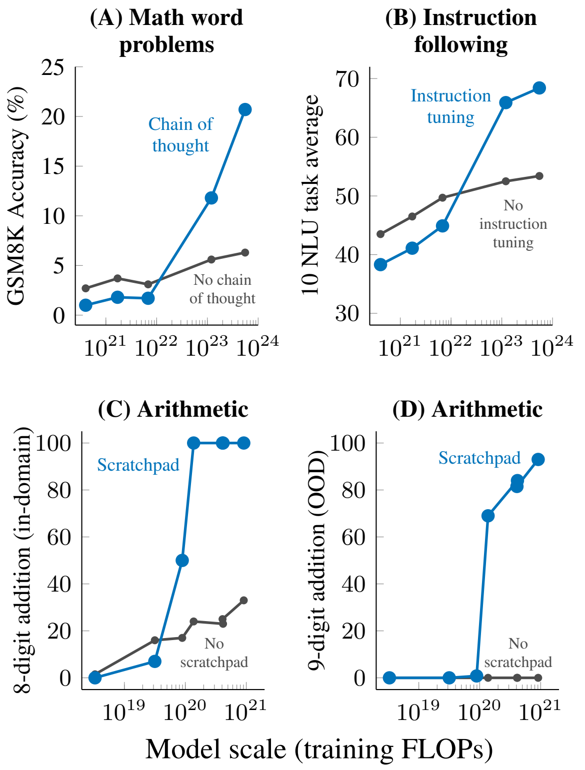

Figure 3: Specialized prompting or finetuning methods can be emergent in that they do not have a positive effect until a certain model scale. A: Wei et al. (2022b). B: Wei et al. (2022a). C & D: Nye et al. (2021). An analogous figure with number of parameters on the x-axis instead of training FLOPs is given in Figure 8. The model shown here is LaMDA (Thoppilan et al., 2022).

2021; Nye et al., 2021). A recent prompting strategy called chain-of-thought prompting enables language models to solve such problems by guiding them to produce a sequence of intermediate steps before giving the final answer (Cobbe et al., 2021; Wei et al., 2022b; Zhou et al., 2022). As shown in Figure 3A, chain of thought prompting only surpasses standard prompting without intermediate steps when scaled to 10^{23} training FLOPs ($\sim$100B parameters). A similar emergence in performance gain was also observed when augmenting few-shot prompting with explanations that came after the final answer (Lampinen et al., 2022).

Instruction following. Another growing line of work aims to better enable language models to perform new tasks simply by reading instructions describing the task (without few-shot exemplars). By finetuning on a mixture of tasks phrased as instructions, language models have been shown to respond appropriately to instructions describing an unseen task (Ouyang et al., 2022; Wei et al., 2022a;

	Emergent scale			
	Train. FLOPs	Params.	Model	Reference
Few-shot prompting abilities				
• Addition/subtraction (3 digit)	2.3E+22	13B	GPT-3	Brown et al. (2020)
• Addition/subtraction (4-5 digit)	3.1E+23	175B		
• MMLU Benchmark (57 topic avg.)	3.1E+23	175B	GPT-3	Hendrycks et al. (2021)
• Toxicity classification (CivilComments)	1.3E+22	7.1B	Gopher	Rae et al. (2021)
• Truthfulness (Truthful QA)	5.0E+23	280B		
• MMLU Benchmark (26 topics)	5.0E+23	280B		
• Grounded conceptual mappings	3.1E+23	175B	GPT-3	Patel and Pavlick (2022)
• Modified arithmetic	1.3E+23	68B	LaMDA	BIG-Bench (2022)
• IPA transliterate	1.3E+23	68B		
• Word unscramble	1.3E+23	68B		
• Figure of speech detection	1.3E+23	68B		
• Logical arguments	1.3E+23	68B		
• Sports understanding	5.5E+23	137B		
• MMLU Benchmark (30 topics)	5.0E+23	70B	Chinchilla	Hoffmann et al. (2022)
• Word in Context (WiC) benchmark	2.5E+24	540B	PaLM	Chowdhery et al. (2022)
Augmented prompting abilities				
• Instruction following (finetuning)	1.3E+23	68B	FLAN	Wei et al. (2022a)
• Scratchpad: 8-digit addition (finetuning)	8.9E+19	40M	LaMDA	Nye et al. (2021)
• Scratchpad: 9-digit addition (finetuning; OOD eval)	1.4E+20	130M		
• Using open-book knowledge for fact checking	1.3E+22	7.1B	Gopher	Rae et al. (2021)
• Chain of thought: Math word problems	1.3E+23	68B	LaMDA	Wei et al. (2022b)
• Chain of thought: StrategyQA	2.9E+23	62B	PaLM	Chowdhery et al. (2022)
• Differentiable search index	3.3E+22	11B	T5	Tay et al. (2022b)
• Self-consistency decoding	1.3E+23	68B	LaMDA	Wang et al. (2022b)
• Leveraging explanations in prompting	5.0E+23	280B	Gopher	Lampinen et al. (2022)
• Least-to-most prompting	3.1E+23	175B	GPT-3	Zhou et al. (2022)
• Zero-shot chain of thought reasoning	3.1E+23	175B	GPT-3	Kojima et al. (2022)

Table 1: List of emergent abilities of large language models and the scale (both training FLOPs and number of model parameters) at which the abilities emerge. The BIG-Bench tasks also include GPT-3 and internal Google sparse models, though we did not list them here for space reasons.

Sanh et al., 2022). As shown in Figure 3B, Wei et al. (2022a) found that this instruction-finetuning technique hurts performance for models of $7 \cdot 10^{21}$ training FLOPs (8B parameters) or smaller, and only improves performance when scaled to 10^{23} training FLOPs ($\sim$100B parameters) (though Sanh et al. (2022) found shortly after that this instruction-following behavior could be also induced by finetuning smaller encoder-decoder T5 models).

Program execution. Finally, consider tasks involving multi-step computation, such as adding large numbers or executing computer programs. Nye et al. (2021) show that finetuning language models to predict intermediate outputs ("scratchpad") enables them to successfully execute such multi-step computations. As shown in Figure 3C, on an in-domain evaluation of 8-digit addition, using a scratchpad only helps for models of $\sim 9 \cdot 10^{19}$ training FLOPs (40M parameters) or larger. Figure 3D shows that such models also generalized to out-of-domain 9-digit addition, which emerged at $\sim 1.3 \cdot 10^{20}$ training FLOPs (100M parameters).

5 Discussion

We have seen that a range of abilities—in the few-shot prompting setup or otherwise—have thus far only been observed when evaluated on a sufficiently large language model. Hence, their emergence cannot be predicted by simply extrapolating performance on smaller-scale models. Emergent few-shot prompted tasks are also unpredictable in the sense that these tasks are not explicitly included in pre-training, and we likely do not know the full scope of few-shot prompted tasks that language models can perform. The overall implication is that further scaling will likely endow even-larger language models with new emergent abilities. Tasks that language models cannot currently do are prime candidates for future emergence; for instance, there are dozens of tasks in BIG-Bench for which even the largest LaMDA and GPT-3 models do not achieve above-random performance.

The ability for scale to unpredictably enable new techniques is not just theoretical. Consider the Word in Context (WiC) benchmark (Pilehvar and

Camacho-Collados, 2019) shown in Figure 2H, as a historical example. Here, scaling GPT-3 to around $3 \cdot 10^{23}$ training FLOPs (175B parameters) failed to unlock above-random one-shot prompting performance.[3] Regarding this negative result, Brown et al. (2020) cited the model architecture of GPT-3 or the use of an autoregressive language modeling objective (rather than using a denoising training objective) as potential reasons, and suggested training a model of comparable size with bidirectional architecture as a remedy. However, later work found that further scaling a decoder-only language model was actually enough to enable above-random performance on this task. As is shown in Figure 2H, scaling PaLM (Chowdhery et al., 2022) from $3 \cdot 10^{23}$ training FLOPs (62B parameters) to $3 \cdot 10^{24}$ training FLOPs (540B parameters) led to a significant jump in performance, without the significant architectural changes suggested by Brown et al. (2020).

5.1 Potential explanations of emergence

Although there are dozens of examples of emergent abilities, there are currently few compelling explanations for why such abilities emerge in the way they do. For certain tasks, there may be natural intuitions for why emergence requires a model larger than a particular threshold scale. For instance, if a multi-step reasoning task requires l steps of sequential computation, this might require a model with a depth of at least $O(l)$ layers. It is also reasonable to assume that more parameters and more training enable better memorization that could be helpful for tasks requiring world knowledge.[4] As an example, good performance on closed-book question-answering may require a model with enough parameters to capture the compressed knowledge base itself (though language model-based compressors can have higher compression ratios than conventional compressors (Bellard, 2021)).

It is also important to consider the evaluation metrics used to measure emergent abilities (BIG-Bench, 2022). For instance, using exact string match as the evaluation metric for long-sequence targets may disguise compounding incremental improvements as emergence. Similar logic may apply for multi-step or arithmetic reasoning problems, where models are only scored on whether they get the final answer to a multi-step problem correct, without any credit given to partially correct solutions. However, the jump in final answer accuracy does not explain why the quality of intermediate steps suddenly emerges to above random, and using evaluation metrics that do not give partial credit are at best an incomplete explanation, because emergent abilities are still observed on many classification tasks (e.g., the tasks in Figure 2D–H).

As an alternative evaluation, we measure cross-entropy loss, which is used in scaling laws for pre-training, for the six emergent BIG-Bench tasks, as detailed in Appendix A. This analysis follows the same experimental setup from BIG-Bench (2022) and affirms their conclusions for the six emergent tasks we consider. Namely, cross-entropy loss improves even for small model scales where the downstream metrics (exact match, BLEU, and accuracy) are close to random and do not improve, which shows that improvements in the log-likelihood of the target sequence can be masked by such downstream metrics. However, this analysis does not explain why downstream metrics are emergent or enable us to predict the scale at which emergence occurs. Overall, more work is needed to tease apart what enables scale to unlock emergent abilities.

5.2 Beyond scaling

Although we may observe an emergent ability to occur at a certain scale, it is possible that the ability could be later achieved at a smaller scale—in other words, model scale is not the singular factor for unlocking an emergent ability. As the science of training large language models progresses, certain abilities may be unlocked for smaller models with new architectures, higher-quality data, or improved training procedures. For example, the zero-shot SuperGLUE performance of UL2 20B (Tay et al., 2022a) is comparable to GPT-3 175B despite UL2 being significantly smaller ($6 \cdot 10^{22}$ versus $3.1 \cdot 10^{23}$ training FLOPs), potentially because it was trained using multiple denoising objectives instead of only a single objective such as next word prediction.

Moreover, once an ability is discovered, further research may make the ability available for smaller scale models. Consider the nascent direction of enabling language models to follow natural language instructions describing a task (Wei et al., 2022a; Sanh et al., 2022; Ouyang et al., 2022, *inter alia*). Although Wei et al. (2022a) ini-

[3] GPT-3 does achieve slightly above-random performance on the dev set with few-shot instead of one-shot prompting ($\sim$55%), but this above-random performance did not appear to be a result of scale and did not hold on the test set server.

[4] Though note that encoding world knowledge in parameters is just one approach; there are others (e.g., Guu et al., 2020; Borgeaud et al., 2021).

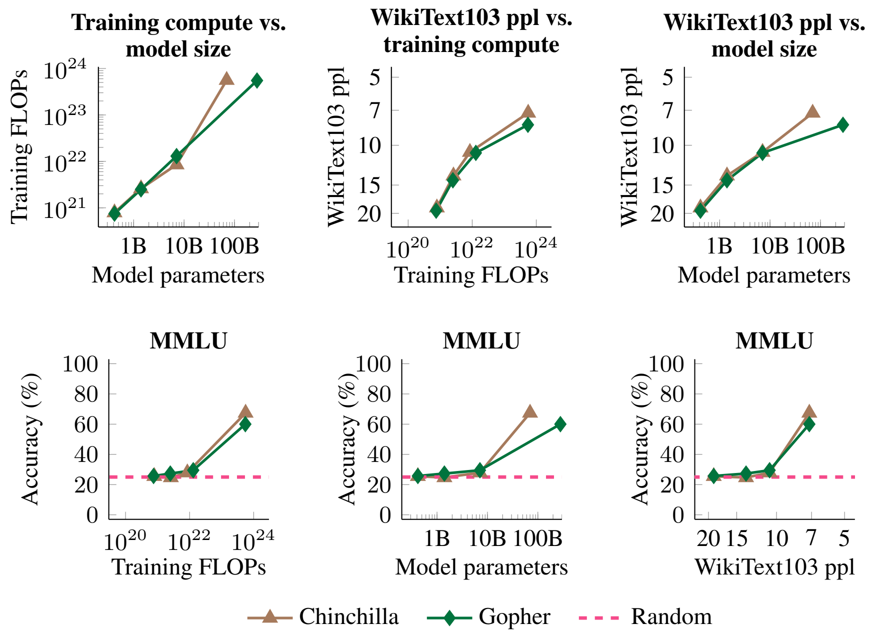

Figure 4: Top row: the relationships between training FLOPs, model parameters, and perplexity (ppl) on WikiText103 (Merity et al., 2016) for Chinchilla and Gopher. Bottom row: Overall performance on the massively multi-task language understanding benchmark (MMLU; Hendrycks et al., 2021) as a function of training FLOPs, model parameters, and WikiText103 perplexity.

tially found that instruction-based finetuning only worked for 68B parameter or larger decoder-only models, Sanh et al. (2022) induced similar behavior in a 11B model with an encoder-decoder architecture, which typically has higher performance after finetuning than decoder-only architectures (Wang et al., 2022a). As another example, Ouyang et al. (2022) proposed a finetuning and reinforcement learning from human feedback approach for the InstructGPT models, which enabled a 1.3B model to outperform much larger models in human-rater evaluations on a broad set of use cases.

There has also been work on improving the general few-shot prompting abilities of language models, (Gao et al., 2021; Schick and Schütze, 2021, *inter alia*). Certain distributional features of training data have also been observed to explain emergent few-shot prompting and could potentially enable it in smaller models (Xie et al., 2022; Chan et al., 2022). As we continue to train ever-larger language models, lowering the scale threshold for emergent abilities will become more important for allowing research on such abilities to available to the community broadly (Bommasani et al., 2021; Ganguli et al., 2022; Liang et al., 2022).

Finally, there are limitations to a program consisting only of increasing scale (training compute, model parameters, and dataset size). For instance, scaling may eventually be bottle-necked by hardware constraints, and some abilities may not have emerged at this point. Other abilities may never emerge—for instance, tasks that are far out of the distribution of even a very large training dataset might not ever achieve any significant performance. Finally, an ability could emerge and then plateau; in other words, there is no guarantee that scaling enables an ability to reach the desired level.

5.3 Another view of emergence

While scale (e.g., training FLOPs or model parameters) has been highly correlated with language model performance on many downstream metrics so far, scale need not be the only lens to view emergent abilities. For example, the emergence of task-specific abilities can be analyzed as a function of the language model's perplexity on a general text corpus such as WikiText103 (Merity et al., 2016). Figure 4 shows such a plot with WikiText103 perplexity of the language model on the x-axis and performance on the MMLU benchmark on the y-axis, side-by-side with plots of training FLOPs and model parameters on the x-axis.

Because WikiText103 perplexity and training FLOPs happen to be highly correlated for the models considered here (Gopher and Chinchilla), the plots of emergent abilities look similar for both.

However, this correlation between WikiText103 perplexity and scale may not hold in the future as new techniques beyond vanilla dense Transformer models are developed (e.g., retrieval-augmented models may have strong WikiText103 perplexity with less training compute and fewer model parameters (Borgeaud et al., 2021)). Also note that using WikiText103 perplexity to compare across model families can be complicated due to factors such as differences in training data composition. Overall, emergent abilities should probably be viewed as a function of many correlated variables.

5.4 Emergent risks

Importantly, because emergent abilities have been observed in a few-shot prompting setting without explicitly being included in pre-training, risks could also emerge (Bommasani et al., 2021; Steinhardt, 2021; Ganguli et al., 2022). For instance, it is possible to extract training data from language models, with larger models being more likely to memorize training data (Carlini et al., 2021, 2022), though this risk can be mitigated via deduplication methods that have been shown to both reduce memorization and improve performance (Kandpal et al., 2022; Lee et al., 2022a). More generally, approaches involving data filtering, forecasting, governance, and automatically discovering harmful behaviors have also been proposed for discovering and mitigating emergent risks (Bender et al., 2021; Weidinger et al., 2021; Steinhardt, 2021; Ganguli et al., 2022; Perez et al., 2022, *inter alia*). For a more detailed discussion of the risks of large language models, including emergent risks, see Bender et al. (2021); Steinhardt (2021); Bommasani et al. (2021); Ganguli et al. (2022).

5.5 Sociological changes

Finally, the emergent abilities discussed here focus on model behavior and are just one of several types of emergence in NLP (Manning et al., 2020; Teehan et al., 2022). Another notable type of qualitative change is sociological, in which increasing scale has shifted how the community views and uses language models. For instance, NLP has historically focused on task-specific models (Jurafsky and Martin, 2009). Recently, scaling has led to an explosion in research on and development of models that are "general purpose" in that they are single models that aim to perform a range of tasks not explicitly encoded in the training data (e.g., GPT-3, Chinchilla, and PaLM) (Manning, 2022).

One key set of results in the emergent sociological shift towards general-purpose models is when scaling enables a few-shot prompted general-purpose model to outperform prior state of the art held by finetuned task-specific models. As a few examples, GPT-3 175B achieved new state of the art on the TriviaQA and PiQA question-answering benchmarks (Brown et al., 2020); PaLM 540B achieved new state of the art on three arithmetic reasoning benchmarks (Chowdhery et al., 2022); and the multimodal Flamingo 80B model achieved new state of the art on six visual question answering benchmarks (Alayrac et al., 2022). In all of these cases, state-of-the-art performance was achieved by few-shot prompting a language model of unprecendented scale (scaling curves for these examples are shown in Appendix Figure 9). These abilities are not necessarily emergent since they have smooth, predictable scaling curves—however, they do underscore an emergent sociological shift towards general-purpose models in the NLP community.

The ability for general-purpose models to perform unseen tasks given only a few examples has also led to many new applications of language models outside the NLP research community. For instance, language models have been used via prompting to translate natural language instructions into actions executable by robots (Ahn et al., 2022; Huang et al., 2022), interact with users (Coenen et al., 2021; Wu et al., 2021, 2022; Lee et al., 2022b), and facilitate multi-modal reasoning (Zeng et al., 2022; Alayrac et al., 2022). Large language models have also been deployed in the real-world both in products, such as GitHub CoPilot,[5] and directly as services themselves, such as OpenAI's GPT-3 API.[6]

6 Conclusions

We have discussed emergent abilities of language models, for which meaningful performance has only been thus far observed at a certain computational scale. Emergent abilities can span a variety of language models, task types, and experimental scenarios. Such abilities are a recently discovered outcome of scaling up language models, and the questions of how they emerge and whether more scaling will enable further emergent abilities seem to be important future research directions for the field of NLP.

[5] https://copilot.github.com/
[6] https://beta.openai.com/docs/introduction

Acknowledgements

We thank Charles Sutton, Slav Petrov, Douglas Eck, Jason Freidenfelds, Jascha Sohl-Dickstein, Ethan Dyer, Dale Schuurmans, and Xavier Garcia for useful discussions and feedback on the manuscript.

References

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, et al. 2022. Do as I can, not as I say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. 2022. Flamingo: A visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*.

Philip W. Anderson. 1972. More is different: Broken symmetry and the nature of the hierarchical structure of science. *Science*, 177(4047):393–396.

Fabrice Bellard. 2021. gpt2tc: Text completion and compression using GPT-2. Accessed Apr. 26, 2022.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*.

BIG-Bench. 2022. Beyond the imitation game: Measuring and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2021. Improving language models by retrieving from trillions of tokens. *arXiv preprint arXiv:2112.04426*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *NeurIPS*.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. *arXiv preprint arXiv:2202.07646*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. 2021. Extracting training data from large language models. *USENIX Security*.

Stephanie C.Y. Chan, Adam Santoro, Andrew K. Lampinen, Jane X. Wang, Aaditya Singh, Pierre H. Richemond, Jay McClelland, and Felix Hill. 2022. Data distributional properties drive emergent few-shot learning in transformers. *arXiv preprint arXiv:2205.05055*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, et al. 2022. PaLM: Scaling language modeling with Pathways. *arXiv preprint arXiv:2204.02311*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Andy Coenen, Luke Davis, Daphne Ippolito, Emily Reif, and Ann Yuan. 2021. Wordcraft: A human-AI collaborative editor for story writing. *arXiv preprint arXiv:2107.07430*.

Antonella Corradini and Timothy O'Connor. 2010. *Emergence in science and philosophy*, volume 6. Routledge.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *NAACL*.

Nan Du, Yanping Huang, Andrew M. Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. 2021. GLaM: Efficient scaling of language models with mixture-of-experts. *arXiv preprint arXiv:2112.06905*.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv preprint arXiv:2101.03961*.

Stephanie Forrest. 1990. Emergent computation: Self-organizing, collective, and cooperative phenomena in natural and artificial computing networks. *Physica D: Nonlinear Phenomena*, 42(1-3):1–11.

Deep Ganguli, Danny Hernandez, Liane Lovitt, Nova DasSarma, Tom Henighan, Andy Jones, Nicholas Joseph, Jackson Kernion, Ben Mann, Amanda Askell, et al. 2022. Predictability and surprise in large generative models. *arXiv preprint arXiv:2202.07785*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. *ACL*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: Retrieval-augmented language model pre-training. *ICML*.

David A. Harper and Paul A. Lewis. 2012. New perspectives on emergence in economics. *New Perspectives on Emergence in Economics*, pages 2–3.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *ICLR*.

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. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Wenlong Huang, Pieter Abbeel, Deepak Pathak, and Igor Mordatch. 2022. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. *arXiv preprint arXiv:2201.07207*.

Bernardo A. Huberman and Tad Hogg. 1987. Phase transitions in artificial intelligence systems. *Artificial Intelligence*, 33(2):155–171.

Harold Y. Hwang, Yoh Iwasa, Masashi Kawasaki, Bernhard Keimer, Naoto Nagaosa, and Yoshinori Tokura. 2012. Emergent phenomena at oxide interfaces. *Nature materials*, 11(2):103–113.

Dan Jurafsky and James H. Martin. 2009. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall series in Artificial Intelligence. Pearson Prentice Hall.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. 2022. Deduplicating training data mitigates privacy risks in language models. *arXiv preprint arXiv:2202.06539*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.

Andrew K. Lampinen, Ishita Dasgupta, Stephanie CY Chan, Kory Matthewson, Michael Henry Tessler, Antonia Creswell, James L. McClelland, Jane X. Wang, and Felix Hill. 2022. Can language models learn from explanations in context? *arXiv preprint arXiv:2204.02329*.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022a. Deduplicating training data makes language models better. *ACL*.

Mina Lee, Percy Liang, and Qian Yang. 2022b. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. *CHI*.

Percy Liang. 2005. *Semi-supervised learning for natural language*. Ph.D. thesis, Massachusetts Institute of Technology.

Percy Liang, Rishi Bommasani, Kathleen A. Creel, and Rob Reich. 2022. The time is now to develop community norms for the release of foundation models.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2021. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*.

Christopher D. Manning. 2022. Human language understanding & reasoning. *Daedalus*, 151(2):127–138.

Christopher D. Manning, Kevin Clark, John Hewitt, Urvashi Khandelwal, and Omer Levy. 2020. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences*, 117(48):30046–30054.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models.

Scott Miller, Jethran Guinness, and Alex Zamanian. 2004. Name tagging with word clusters and discriminative training. In *NAACL*.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2021. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL*.

Roma Patel and Ellie Pavlick. 2022. Mapping language models to grounded conceptual spaces. *ICLR*.

Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. 2022. Red teaming language models with language models. *arXiv preprint arXiv:2202.03286*.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. *NAACL*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8).

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training Gopher. *arXiv preprint arXiv:2112.11446*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*.

Victor Sanh, Albert Webson, Colin Raffel, Stephen Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2022. Multitask prompted training enables zero-shot task generalization. *ICLR*.

Timo Schick and Hinrich Schütze. 2021. It's not just size that matters: Small language models are also few-shot learners. *NAACL*.

Jacob Steinhardt. 2021. On the risks of emergent behavior in foundation models. Accessed Apr 13, 2022.

Jacob Steinhardt. 2022. Future ml systems will be qualitatively different. Accessed May 20, 2022.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Dara Bahri, Tal Schuster, Huaixiu Steven Zheng, Neil Houlsby, and Donald Metzler. 2022a. Unifying language learning paradigms. *arXiv preprint arXiv:2205.05131*.

Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022b. Transformer memory as a differentiable search index. *arXiv preprint arXiv:2202.06991*.

Ryan Teehan, Miruna Clinciu, Oleg Serikov, Eliza Szczechla, Natasha Seelam, Shachar Mirkin, and Aaron Gokaslan. 2022. Emergent structures and training dynamics in large language models. In *ACL Big Science Workshop*.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. LaMDA: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning. *arXiv preprint arXiv:1806.02847*.

Thomas Wang, Adam Roberts, Daniel Hesslow, Teven Le Scao, Hyung Won Chung, Iz Beltagy, Julien Launay, and Colin Raffel. 2022a. What language model architecture and pretraining objective work best for zero-shot generalization? *arXiv preprint arXiv:2204.05832*.

Xuezhi Wang, Jason Wei, Xuezhi Wang, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. 2022b. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners. *ICLR*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022b. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. 2021. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*.

Tongshuang Wu, Ellen Jiang, Aaron Donsbach, Jeff Gray, Alejandra Molina, Michael Terry, and Carrie J. Cai. 2022. PromptChainer: Chaining large language model prompts through visual programming. *arXiv preprint arXiv:2203.06566*.

Tongshuang Wu, Michael Terry, and Carrie J. Cai. 2021. AI chains: Transparent and controllable human-ai interaction by chaining large language model prompts. *arXiv preprint arXiv:2110.01691*.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. 2022. An explanation of in-context learning as implicit bayesian inference. *ICLR*.

Andy Zeng, Adrian Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, et al. 2022. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Chi. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

Appendix

A BIG-Bench analysis

A.1 Cross-entropy loss analysis

Here we study how scaling curves may appear differently depending on evaluation metric used to measure performance. We will focus on the six few-shot prompted BIG-Bench tasks that we consider emergent. Three of these tasks are generative and use Exact Match (EM) or BLEU (Papineni et al., 2002) as the evaluation metric. The other three tasks are classification and use accuracy (acc) as the evaluation metric.

In the scaling curves for these tasks, peformance in EM/BLEU/acc is close to random for small models ($\leq 10^{22}$ FLOPs / $\leq$27B params). We will compare these scaling curves against alternative plots that have a different y-axis measured by cross-entropy loss. Cross-entropy loss differs from EM/BLEU/acc in that it captures improvements in performance (the predicted distribution getting closer to ground truth) even when the EM/BLEU/acc is random. For example, if two examples are both wrong as measured by EM/BLEU/acc, one example may be closer to the ground truth in terms of probabilities, and this information is captured by the cross-entropy loss.

These plots are expected to look like one of the following:

- Outcome 1: For the model scales where EM/BLEU/acc is random, cross-entropy loss also does not improve as scale increases. This outcome implies that for these scales, the model truly does not get any better at the tasks.

- Outcome 2: For the model scales where EM/BLEU/acc is random, cross-entropy loss does improve. This outcome implies that the models do get better at the task, but these improvements are not reflected in the downstream metric of interest. The broader implication is that scaling small models improves the models in a way that is not reflected in EM/BLEU/Acc, and that there is some critical model scale where these improvements enable the downstream metric to increase to above random as an emergent ability.

We find that all six BIG-Bench tasks fall under Outcome 2, and detail this analysis below. Overall, the conclusion from this analysis is that small models do improve in some ways that downstream metrics that EM/BLEU/Acc do not capture. However, these tasks are still considered emergent, and this analysis does not provide any straightforward indicators of how to predict such emergent behaviors.

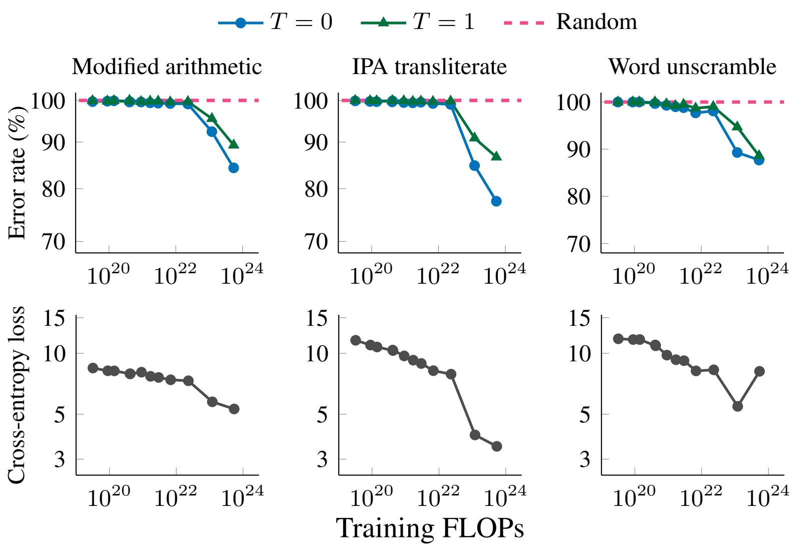

Figure 5: Adjacent plots for error rate and cross-entropy loss on three emergent generative tasks in BIG-Bench for LaMDA. We show error rate fot both greedy decoding ($T = 0$) as well as random sampling ($T = 1$). Error rate is (1 - exact match score) for modified arithmetic and word unscramble, and (1 - BLEU score) for IPA transliterate.

A.1.1 Generative tasks

Figure 5 shows the cross-entropy loss on the three generative BIG-Bench tasks (modified arithmetic, IPA transliterate, and word unscramble) alongside the downstream evaluation metrics used in Figure 2. For all three tasks, notice that while the error rate is nearly 100% for small models ($\leq 10^{22}$ FLOPs / $\leq$27B params), the cross-entropy loss does actually improve for these model sizes. At the point of emergence as measured by error rate, we also see an "elbow" in performance improvement for cross-entropy loss.

A.1.2 Classification tasks

Figure 6 (middle row) shows the cross-entropy loss of the three classification BIG-Bench tasks. Similar to the generative tasks, when the error rate is close to random, cross-entropy loss consistently still improves for models trained with more compute. This again shows that performance as computed by accuracy can mask consistent improvements in the likelihood of the target sequences.

We also perform an additional analysis of the multiple choice emergent tasks in Figure 6 (bottom row), which shows the log probabilities of the correct response and incorrect response(s). We find that the cross-entropy loss decreases for both the correct and incorrect responses in the three emergent multiple choice tasks. Counterintuitively, both log-probabilities can decrease in tandem even when the probability across all available multiple choice responses is normalized. The reason is that larger models produce less-extreme probabilities (i.e., values approaching 0 or 1) and therefore the average log-probabilities have fewer extremely small values. However, we note that for each of these three tasks, that the average log-probability of the correct and incorrect responses eventually deviates at a certain scale, during which performance on the task increases substantially.

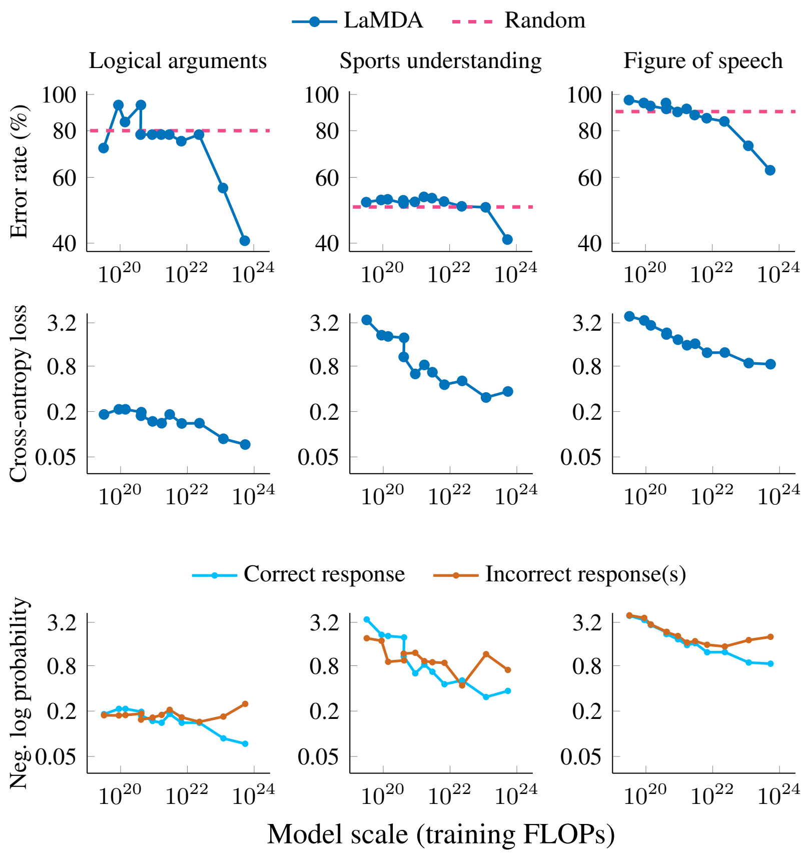

Figure 6: Adjacent plots for error rate, cross-entropy loss, and log probabilities of correct and incorrect responses on three classification tasks on BIG-Bench that we consider to demonstrate emergent abilities. Logical arguments only has 32 samples, which may contribute to noise. Error rate is (1 - accuracy).

B All Model Details

Table 2 below summarizes the parameter count, number of training tokens, and the training FLOPs for the models highlighted in our work. The models span from the smallest LaMDA model with 2.1M parameters to the largest PaLM model with 540B parameters and 2.5E+24 training FLOPs—roughly 8x the computational budget of GPT-3.

Model	Parameters	Train tokens	Train FLOPs
GPT-3	125M	300B	2.25E+20
	350M	300B	6.41E+20
	760M	300B	1.37E+21
	1.3B	300B	2.38E+21
	2.7B	300B	4.77E+21
	6.7B	300B	1.20E+22
	13B	300B	2.31E+22
	175B	300B	3.14E+23
LaMDA	2.1M	262B	3.30E+18
	17M	313B	3.16E+19
	57M	262B	8.90E+19
	134M	170B	1.37E+20
	262M	264B	4.16E+20
	453M	150B	4.08E+20
	1.1B	142B	9.11E+20
	2.1B	137B	1.72E+21
	3.6B	136B	2.96E+21
	8.6B	132B	6.78E+21
	29B	132B	2.30E+22
	69B	292B	1.20E+23
	137B	674B	5.54E+23
Gopher	417M	300B	7.51E+20
	1.4B	300B	2.52E+21
	7.1B	300B	1.28E+22
	280B	325B	5.46E+23
Chinchilla	417M	314B	7.86E+20
	1.4B	314B	2.63E+21
	7.1B	[sic] 199B	8.47E+21
	70B	1.34T	5.63E+23
PaLM	8B	780B	3.74E+22
	62B	780B	2.90E+23
	540B	780B	2.53E+24

Table 2: Parameters, training examples, and training FLOPs of large language models.

C Scaling with Parameter Count

Figures 7, 8, and 9 shows emergent abilities with an x-axis of number of model parameters.

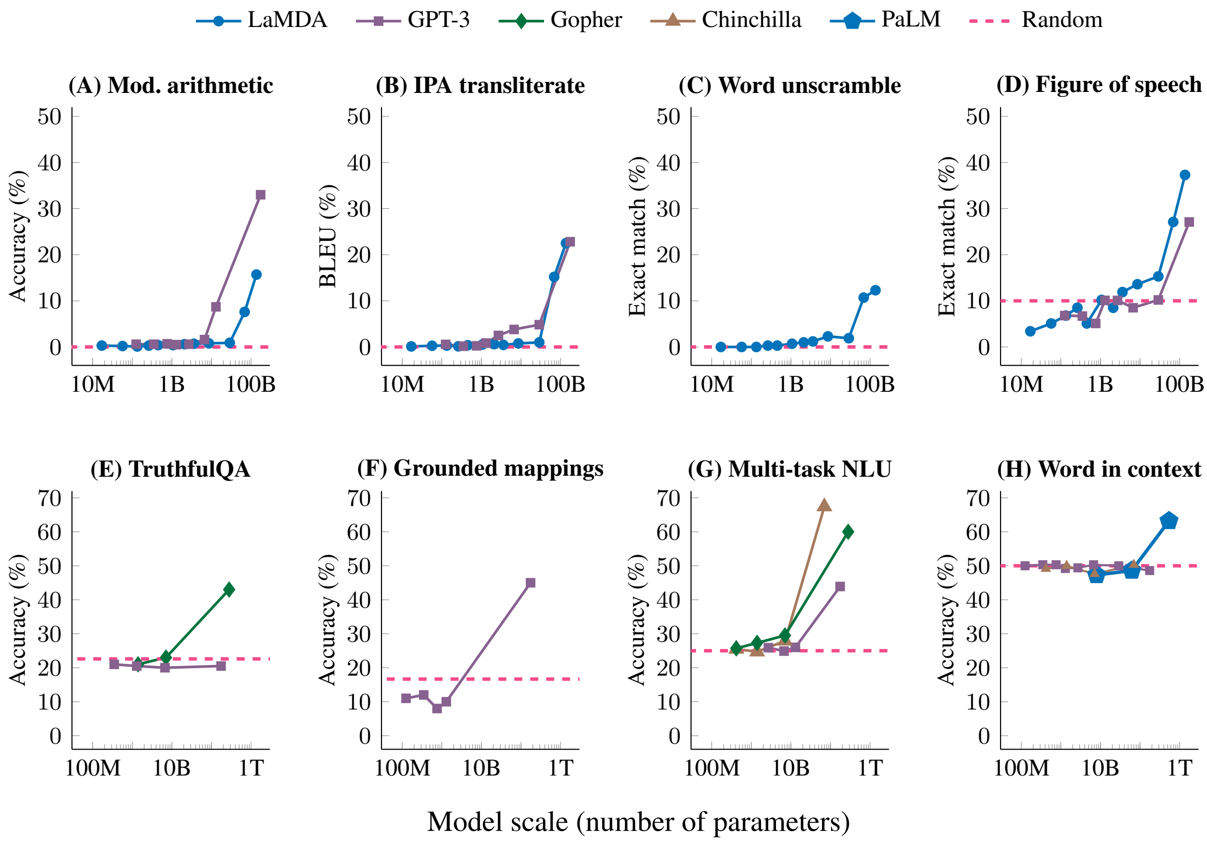

Figure 7: Eight examples of emergence in the few-shot prompting setting. Each point is a separate model. The ability to perform a task via few-shot prompting is emergent when a language model achieves random performance until a certain scale, after which performance significantly increases to well-above random. Note that models with more parameters also typically use more training compute—hence, we show an analogous figure with training FLOPs instead of number of model parameters as the x-axis in Figure 2. A–D: BIG-Bench (2022), 2-shot. E: Lin et al. (2021) and Rae et al. (2021). F: Patel and Pavlick (2022). G: Hendrycks et al. (2021), Rae et al. (2021), and Hoffmann et al. (2022). H: Brown et al. (2020), Hoffmann et al. (2022), and Chowdhery et al. (2022) on the WiC benchmark (Pilehvar and Camacho-Collados, 2019).

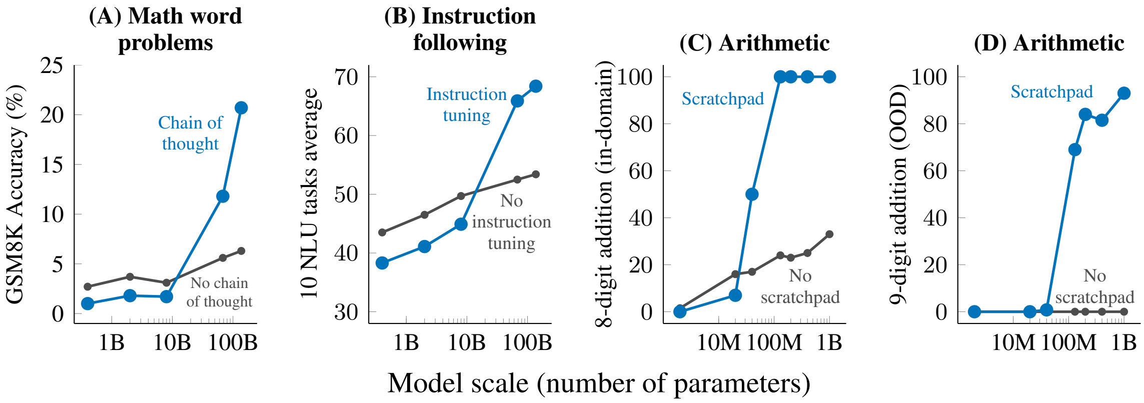

Figure 8: Specialized prompting or finetuning methods are emergent in that they do not have a positive effect until a certain model scale. A: Wei et al. (2022b). B: Wei et al. (2022a). C & D: Nye et al. (2021). The model shown here is LaMDA (Thoppilan et al., 2022).

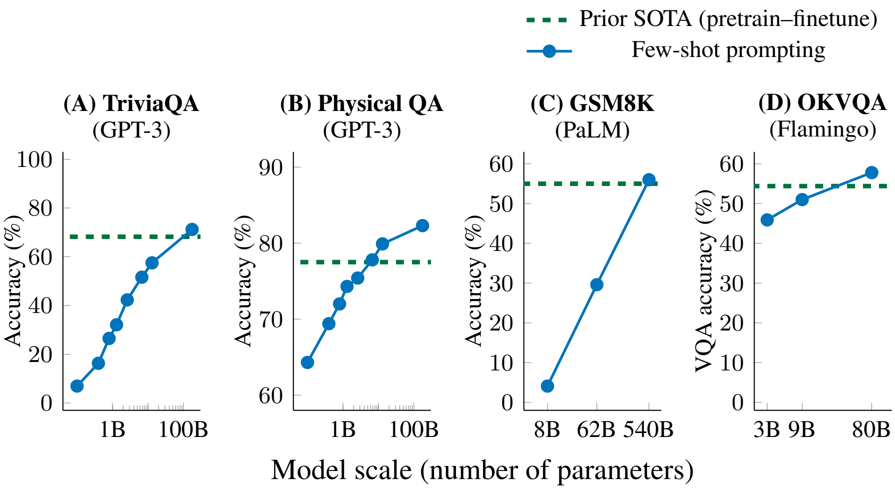

Figure 9: On some benchmarks, task-general models (not explicitly trained to perform a task) surpass prior state-of-the-art performance held by a task-specific model. A & B: Brown et al. (2020). C: Chowdhery et al. (2022). D: Alayrac et al. (2022).