A Brief Overview Of Few-Shot Prompting In the Large Language Models

Vladlen Kulikov
Netanya, Israel
vladgkulikoff@gmail.com

Radoslav Neychev
Harbour.Space,
Barcelona, Spain
neychevr@gmail.com

***Abstract* – Working on a larger, more general topic: «Large Language Models (LLMs). Learning and Reasoning at the Inference Stage», among other things, we investigated the following specific questions:**

1. What is more important for the emergent abilities (few-shot prompting and augmented prompting) observed at the inference stage in LLMs – model's size (number of model parameters) or actual training dataset size (number of training tokens)?

2. What is the composition of datasets on which LLMs demonstrating these abilities were trained and are there any correlations with the compositions and sizes of datasets?

3. What are the qualitative data requirements for observing emergent inference abilities, i.e., is there something in the language data that causes these abilities?

To answer these questions, we present analysis of selected theoretical and experimental results focused on LLMs.

I. Introduction

In this analysis, we systematically follow the terminology presented in [1] for the terms in-context learning, few-shot, one-shot, zero-shot, fine-tuning.

Fine-Tuning – involves updating the weights of a pre-trained model by training on a supervised dataset specific to the desired task.

Few-Shot – the setting where the model is given a few demonstrations of the task at inference time as conditioning [17], but no weight updates are allowed.

One-Shot – is the same as few-shot except that only one demonstration is allowed, in addition to a natural language description of the task.

Zero-Shot – is the same as one-shot except that no demonstrations are allowed, and the model is only given a natural language instruction describing the task.

Terminology illustration – Fig. 1.

By *in-context learning* we mean the general description given in [1], which is illustrated in Fig. 2.

But in the form of a short and simplified formulation, you can use the definition given in [6]: "In-context learning is a paradigm that allows language models to learn tasks when only a few examples in the form of demonstrations".

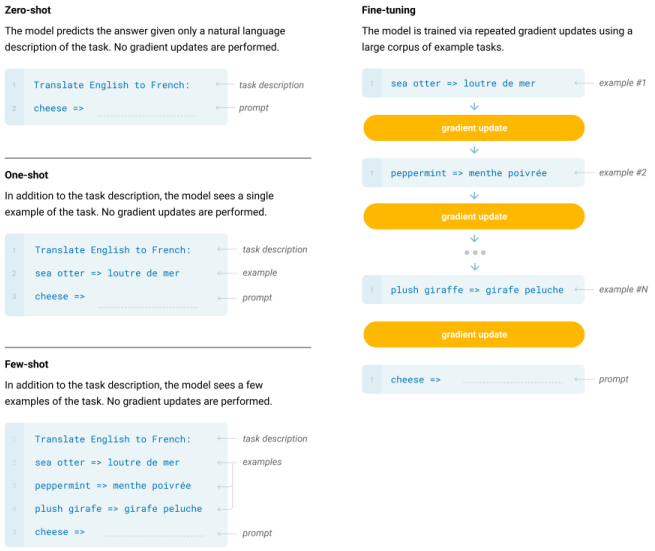

Fig. 1 (Fig. 2.1 from [1]): Zero-shot, one-shot and few-shot, contrasted with traditional fine-tuning.

And today, we are witnessing the emergence of a separate discipline – prompt engineering (the development and optimization of prompts to improve work outcomes), which has emerged as a generalization of the ways we work with prompts.

That is, in the context of our analysis, prompt engineering is a way to manage the results of a few-shot by changing the demonstrations – Few-Shot Prompting, Augmented Prompting – here we follow the terminology [2].

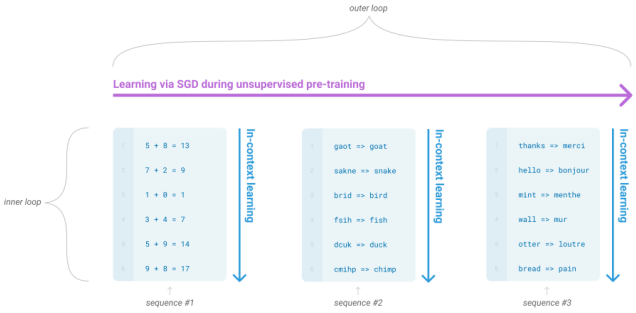

Fig. 2 (Fig. 1.1 from [1]) Learning via SGD (during unsupervised pre-training), contrasted with In-context learning (Meta-learning at Inference stage)

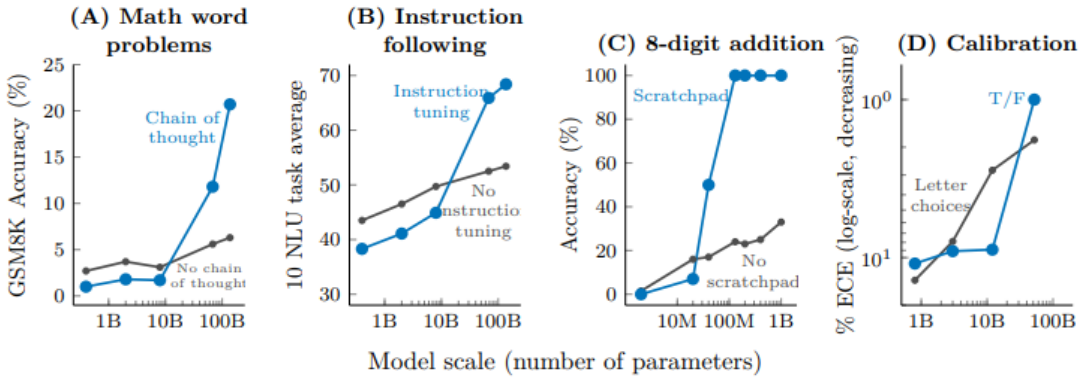

Fig.3 (Fig. 12 from [2]) Specialized prompting may be emergent because it does not have a positive effect up to a certain model scale

Few-Shot and Augmented Prompting are essentially sides of the same phenomenon (as we analyze in our main work – LLMs. Learning and Reasoning at the Inference Stage, which is still being written) – they are emergent in nature (emergence is a qualitative property that arises spontaneously in the system when it reaches a certain threshold of complexity), in which we share the opinion with [2].

Next, we focus on the following questions:

1) What is more important for the emergent abilities – number of parameters or number of training tokens? - Part 1, 2
2) Does the composition of source datasets effect on emergent abilities? - Part 3
3) Is there something in the language data that causes of emergent abilities? - Part 4

In the analysis section we consider all these questions.

II. Analysis

1. Model's size

In [2], the authors purposefully investigated of emergent properties of LLMs, i.e., properties not observed in smaller models. The authors show that specialized prompting can be emergent in that it only has a positive effect at a certain model scale – Fig. 3.

Fig, 4 and 5 [1] also show that an increase in the number of parameters (in this case, GPT-3) directly affects accuracy on in-context tasks.

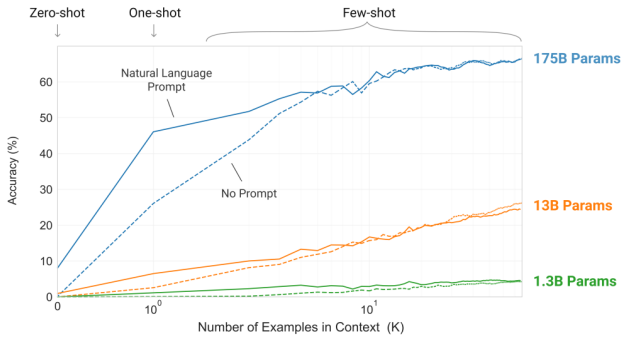

Fig. 4 (Fig. 1.2. from [1] – Larger models make increasingly efficient use of in-context information)

Also note that to achieve accuracy > 50% (aggregate performance across 42 benchmarks), the number of parameters must exceed 13B. However, the nature of the behavior of the model between 13B and 175B is not clear, as intermediate models have not been studied.

A similar pattern of previously unobserved properties in LLMs, we see for 4 other LLMs (along with GPT-3): LaMDA [7], Gopher [3], Chinchilla [4], PaLM [5] – on Few-Shot prompted tasks – Fig. 6 [2]:

A more detailed picture of the required model sizes for efficient occurrence of emergent properties can be obtained by analyzing Fig. 6 in conjunction with Table I.

With the rarest exception – Gopher 7.1B on two types of tasks, T5 11B – on one task and GPT-3 13B on one task (and with very poor accuracy on them), the required model size is > 40-50-60B.

That is, emergent properties (properties not observed in models of smaller size) begin to appear significantly (significantly for the accuracy value), in most cases, in models of size larger than 40B-60B.

A notable point is the fact that as the number of parameters of the same model increases and other things being equal, LLMs become able to solve few-shot prompting tasks on the inference stage, which they could not handle with a smaller model size.

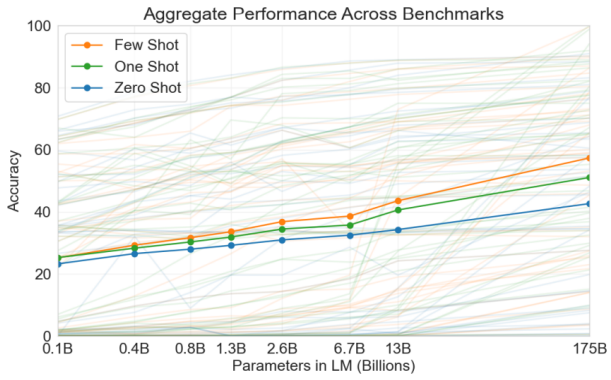

Fig. 5 (Fig. 1.3. from [1] – Aggregate performance for all 42 accuracy-denominated benchmarks)

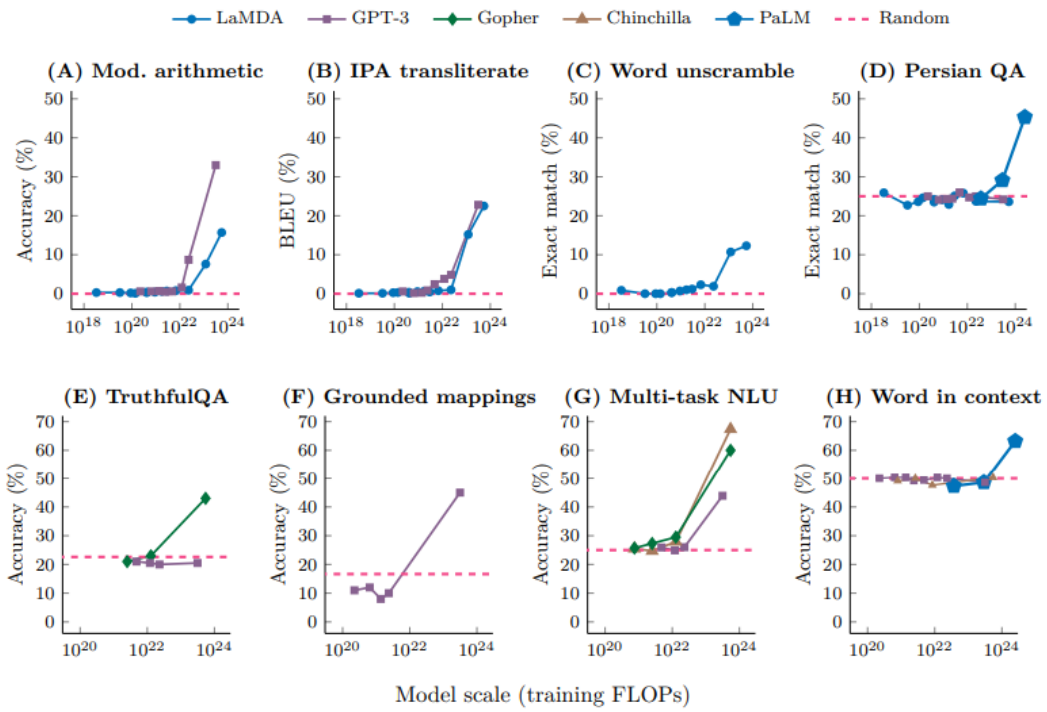

Fig. 6 (Fig. 2 from [2] – Eight examples of emergence in the few-shot prompting setting)

We, like other researchers, have not considered in our research the influence of the nuances of the architecture of each of the models. All existing LLMs are Transformers, and at this stage of the development of research in this area, it seems to us that there is no mechanism to adequately take this into account.

Therefore, we evaluated the models by the number of parameters, that allows us to draw some general conclusions

TABLE I. (TABLE 1 FROM [2]):
LIST OF EMERGENT ABILITIES OF LARGE LANGUAGE MODELS AND THE SCALE
(BOTH TRAINING FLOPS AND NUMBER OF MODEL PARAMETERS) AT WHICH THE ABILITIES EMERGE

	Emergent scale		
	Train. FLOPs	Params.	Model
Few-shot prompting abilities			
• Addition/subtraction (3 digit)	2.3E+22	13B	GPT-3
• Addition/subtraction (4-5 digit)	3.1E+23	175B	
• MMLU Benchmark (57 topic avg.)	3.1E+23	175B	GPT-3
• Toxicity classification (CivilComments)	1.3E+22	7.1B	Gopher
• Truthfulness (Truthful QA)	5.0E+23	280B	
• MMLU Benchmark (26 topics)	5.0E+23	280B	
• Grounded conceptual mappings	3.1E+23	175B	GPT-3
• MMLU Benchmark (30 topics)	5.0E+23	70B	Chinchilla
• Word in Context (WiC) benchmark	2.5E+24	540B	PaLM
• Many BIG-Bench tasks (see Appendix E)	Many	Many	Many
Augmented prompting abilities			
• Instruction following (finetuning)	1.3E+23	68B	FLAN
• Scratchpad: 8-digit addition (finetuning)	8.9E+19	40M	LaMDA
• Using open-book knowledge for fact checking	1.3E+22	7.1B	Gopher
• Chain-of-thought: Math word problems	1.3E+23	68B	LaMDA
• Chain-of-thought: StrategyQA	2.9E+23	62B	PaLM
• Differentiable search index	3.3E+22	11B	T5
• Self-consistency decoding	1.3E+23	68B	LaMDA
• Leveraging explanations in prompting	5.0E+23	280B	Gopher
• Least-to-most prompting	3.1E+23	175B	GPT-3
• Zero-shot chain-of-thought reasoning	3.1E+23	175B	GPT-3
• Calibration via P(True)	2.6E+23	52B	Anthropic
• Multilingual chain-of-thought reasoning	2.9E+23	62B	PaLM
• Ask me anything prompting	1.4E+22	6B	EleutherAI

about model's global behavior on inference stage in few-shot and augmented prompting tasks.

2. Dataset's size

We have not come across any direct studies on this issue, namely the influence of the size of data sets on the manifestation of emergence.

However, we can estimate the order of the required data based on the datasets on which the LLMs discussed in the previous part 1 were trained.

In the works on models themselves, it is not always clear from the available data on total dataset size what is the number of training tokens and in this part, we will follow [4] – Table II:

Table II. (TABLE 1 FROM [4])

Model	Size (# Parameters)	Training Tokens
LaMDA	137 Billion	168 Billion
GPT-3	175 Billion	300 Billion
Jurassic	178 Billion	300 Billion
Gopher	280 Billion	300 Billion
MT-NLG 530B	530 Billion	270 Billion
Chinchilla	70 Billion	1.4 Trillion

Comparing the data of Table I and Table II, we notice that the larger models 137B-530B, with the number of training tokens from 168B to 300B, appear more often in Table I – displaying the manifestation of emergent properties.

At the same time, Chinchilla [4] – a smaller model – 70B, but with a significantly larger number of training tokens 1.4T, does not show overwhelming superiority in Table I (that is, on the few-shot prompting and augmented prompting abilities we are interested in).

It shows that the influence of the model size the emergent effect is more significant than the number of training tokens,

Curiously, on many tests that do not include emergent abilities for inference, Chinchilla performs better than models with more parameters but fewer training tokens (Hoffmann et al., 2022). Although it is outside our scope of study.

And to show emergence, the number of training tokens should probably not be less than a certain limit, which in the considered cases averages about 300B on GPT-3 – the model most often featured in Table 1 (but these data are not enough for more accurate estimates and question requires a separate study).

3. Dataset composition

Let's analyze the composition of the dataset for some of the models that appear in both Table I and Table II.

GPT-3:

The model that most often shows the best results in Table I, was trained on a large unlabeled text corpus – Table II. To what extent this factor can determine the manifestation of emergence is the material for a separate large study.

TABLE II. GPT-3 DATASET (TABLE 2.2 FROM [1])

Dataset	Quantity (tokens)	Weight in training mix	Epochs elapsed when training for 300B tokens
Common Crawl (filtered)	410 billion	60%	0.44
WebText2	19 billion	22%	2.9
Books1	12 billion	8%	1.9
Books2	55 billion	8%	0.43
Wikipedia	3 billion	3%	3.4

PaLM:

PaLM train dataset include large multilingual corpus – text from more than 100 languages. – Table III.

TABLE III. PaLM DATASET (TABLE 2 FROM [5])

Total dataset size = 780 billion tokens	
Data source	Proportion of data
Social media conversations (multilingual)	50%
Filtered webpages (multilingual)	27%
Books (English)	13%
GitHub (code)	5%
Wikipedia (multilingual)	4%
News (English)	1%

LaMDA:

It is initially focused on dialogue and therefore has a specific composition of dataset – Appendix E from [7]:

«Pre-training data composition of LaMDA: The pre-training data, called Infiniset, is a combination of dialog data from public dialog data and other public web documents. It consists of 2.97B documents and 1.12B dialogs with 13.39B utterances.

The composition of the data is as follows: 50% dialogs data from public forums; 12.5% C4 data [11]; 12.5% code documents from sites related to programming like Q&A sites, tutorials, etc.; 12.5% Wikipedia (English); 6.25% English web documents; and 6.25% non-English web documents. The total number of words in the dataset is 1.56T.»

According to Table I – LaMDA as well as GPT-3, it often demonstrates good results for augmented prompting abilities.

Gopher:

TABLE IV. GOPHER DATASET (TABLE 2 FROM [3])

	Disk Size	Documents	Tokens	Sampling proportion
MassiveWeb	1.9 TB	604M	506B	48%
Books	2.1 TB	4M	560B	27%
C4	0.75 TB	361M	182B	10%
News	2.7 TB	1.1B	676B	10%
GitHub	3.1 TB	142M	422B	3%
Wikipedia	0.001 TB	6M	4B	2%

Gopher MassiveText data makeup. For each subset of MassiveText, authors list its total disk size, its number of documents, and its number of SentencePiece tokens. During training authors sample from MassiveText non-uniformly, using the sampling proportion shown in the right-most column.

Chinchilla:

TABLE V. CHINCHILA DATASET (TABLE A1 FROM [2])

	Disk Size	Documents	Sampling proportion	Epochs in 1.4T tokens
MassiveWeb	1.9 TB	604M	45% (48%)	1.24
Books	2.1 TB	4M	30% (27%)	0.75
C4	0.75 TB	361M	10% (10%)	0.77
News	2.7 TB	1.1B	10% (10%)	0.21
GitHub	3.1 TB	142M	4% (3%)	0.13
Wikipedia	0.001 TB	6M	1% (2%)	3.40

Chinchilla MassiveText data makeup. For each subset of MassiveText, authors list its total disk size, the number of documents and the sampling proportion used during training– authors use a slightly different distribution than in Gopher: (Rae et al. (2021)). In the rightmost column show the number of epochs that are used in 1.4 trillion tokens.

In general, the question of the influence of the dataset composition on the emergent abilities remains open. We have not seen unambiguous evidence, and although some comments can be made, which is done in this section, but there is an obvious need for more in-depth research.

However, we can see that the Gopher – Table IV and the Chinchilla – Table V were trained on almost identical datasets.

The difference in the number of training tokens: Gopher – 300B, Chinchilla – 1.4T.

Fig. 6 – Graph G – MultiTask NLU at few-shot settings is the only task in which Chinchilla (70B parameters) is better than other models, but it is ahead of Gopher (280B parameters) by no more than 6-7%.

That is, with a dataset almost identical in composition, almost 5 (4.67) times the number of training tokens – 1.4T training tokens in Chinchilla 70B parameters, against 300B training tokens in Gopher 280B parameters (with the same training costs of about 10 in 24 degree of FLOPs) does not give a significant advantage (5 times more tokens and the gain in accuracy is only 0.06-0.07 – a difference of 2 orders of magnitude).

Whereas Gopher has a model size of 280B prs (but with fewer tokens – 300B training tokens), which is 4 times more in the number of parameters than in Chinchilla, allows you to achieve almost the same results in few-shot prompting tasks.

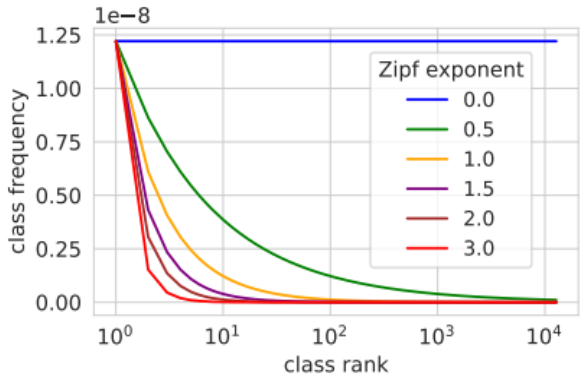

Fig. 7a. Examples of Zipfian distributions, Fig. 6a from [8])

Thus, larger number of parameters compensates smaller number of training tokens – it matches even the order of difference 4.67 times in training tokens versus 4 times in parameters.

4. Language as a training set.

Intuitively, the fact that the training of a language model should be done on a language dataset is obvious, since the data on which we train, the model must correspond to the problem being solved.

But what should be the properties of the original dataset in order to observe the effect of in-context learning at the inference stage? Are they related at all?

As shown in (Chan et al., 2022), in-context learning was observed under certain distribution properties of the training data themselves (and was not observed in their absence), namely when the model was trained:

1. On data following a skewed Zipfian distribution – which is a common property of naturalistic data, including language – Fig. 7a, Fig 7b.

2. When the data exhibits a property such as burstiness – items appear in clusters rather than being uniformly distributed over time.

That is, a given entity (word, person, object, etc.) may have a distribution that is not uniform across time, instead tending to appear in clusters.

3. When the data has a large number of rare classes.

4. In-context learning also emerges more strongly when item meanings or interpretations are dynamic rather than fixed,

That is, the "meaning" of entities in data, such as words in a language, can have many possible interpretations-polysemy, and synonymy-when a single value can correspond to a set of entities.

All these 4 properties are present in languages (but are also inherent in other natural data, which opens a field for thinking about what other types of data it is possible to train LLMs on).

III. CONCLUSION

As a result of this analysis, two potential areas of possible further in-depth research were identified:

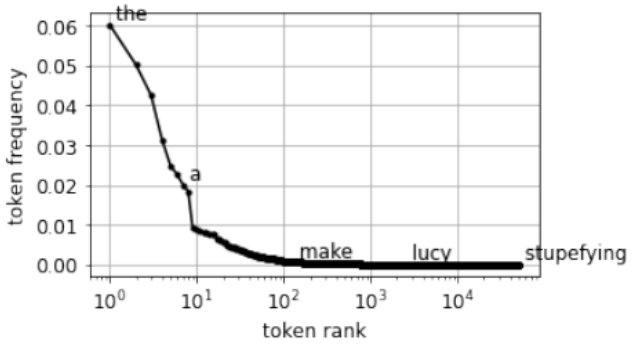

Fig. 7b. Distribution of tokens in a natural language corpus, (Fig. 6b from [8])

a) The influence of the composition of the language dataset on the occurrence and effectiveness of emergent abilities in LLMs (the available evidence does not allow unambiguous conclusions, while only to compile a series of observations, point 4 in this section).

b) Determination of the lower bound on the number of training tokens required for observation of emergent abilities in LLMs at the inference stage (point 5 in this section).

This analysis also allows us to make the following conclusions and observations:

1. Learning from data comparable in complexity to languages is a critical moment for the appearance of emergent abilities in LLMs in the inference stage.

2. The size of LLMs, namely, the number of the parameters, is more important than the number of training tokens on which LLMs were trained, which is observed both with the same (and different) datasets in general composition.

3. Generally, model size should be greater than 40-50B for stable few-shot and augmented prompt effect, with high accuracy.

However, in certain types of tasks this behavior can be observed from 7-13B sized models.

As the number of parameters of the same model increases, and all other things being equal, LLMs become able to solve problems in inference (for few-shot, augmented prompt tasks) that they could not handle with a smaller model size.

4. In general, the question of the influence of the composition of the language dataset on the manifestation of emergent abilities remains open. We have not observed unequivocal evidence, and the need for more in-depth research is clear, but although individual comments can be made:

GPT-3 – the model most often showing the best results (for few-shot and augmented prompting), trained on a large body of unlabeled text

LaMDA – initially dialogue-oriented and therefore has a specific dataset composition, as well as GPT-3, it often shows good results for augmented prompting abilities.

5. Obviously, there is a lower bound on the required number of training tokens for observing emergence, and this bound should be, among other things, a function of the number of model parameters, but this issue requires a separate detailed study. Rough estimate of the average value ~ 300B training tokens for models in the parameter range from 175 to 530 B.

ACKNOWLEDGMENT

We would like to express our deep gratitude to all the authors whose works allowed us to make this analysis.

REFERENCES

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. “Language models are few-shot learners”, NeurIPS, 2020. WEB: https://arxiv.org/abs/2005.14165

[2] 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, William Fedus. “Emergent Abilities of Large Language Models”, Published in Transactions on Machine Learning Research (08/2022). WEB: https://arxiv.org/abs/2206.07682

[3] Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. “Scaling language models: Methods, analysis & insights from training Gopher”, arXiv preprint arXiv:2112.11446, 2021 WEB: https://arxiv.org/abs/2112.11446

[4] 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”, NeurIPS, 2022. WEB: https://arxiv.org/abs/2203.15556

[5] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, et al. “PaLM: Scaling language modeling with Pathways”. arXiv preprint arXiv:2204.02311, 2022 WEB: https://arxiv.org/abs/2204.02311

[6] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, Zhifang Sui. “A Survey on In-context Learning”, arXiv preprint arXiv:2301.00234v2 WEB: https://arxiv.org/abs/2301.00234

[7] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. “LaMDA: Language models for dialog applications”. arXiv preprint arXiv:2201.08239, 2022. WEB: https://arxiv.org/abs/2201.08239.

[8] (Chan et al., 2022) Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland. Felix Hill. “Data Distributional Properties Drive Emergent In-Context Learning in Transformers”. arXiv preprint arXiv:2205.05055v6 WEB: https://arxiv.org/abs/2205.05055

[9] Roma Patel and Ellie Pavlick. “Mapping language models to grounded conceptual spaces”. ICLR, 2022. WEB: https://openreview.net/forum?id=gJcEM8sxHK.

[10] BIG-Bench. “Beyond the imitation game: Measuring and extrapolating the capabilities of language models”. arXiv preprint arXiv:2206.04615, 2022 WEB: https://arxiv.org/abs/2206.04615

[11] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. “Finetuned language models are zero-shot learners”. ICLR, 2022. WEB: https: //openreview.net/forum?id=gEZrGCozdqR

[12] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. “Chain of thought prompting elicits reasoning in large language models.” NeurIPS, 2022. WEB: https://arxiv. org/abs/2201.11903

[13] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. “Show your work: Scratchpads for intermediate computation with language models”. arXiv preprint arXiv:2112.00114, 2021. WEB: https://openreview.net/forum?id=iedYJm92o0a.

[14] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield Dodds, Nova DasSarma, Eli Tran-Johnson, et al. “Language models (mostly) know what they know”. arXiv preprint arXiv:2207.05221, 2022. WEB: https://arxiv.org/abs/2207.05221.

[15] Stephanie Lin, Jacob Hilton, and Owain Evans. “TruthfulQA: Measuring how models mimic human falsehoods”. arXiv preprint arXiv:2109.07958, 2021. WEB: https://arxiv.org/abs/2109.07958.

[16] Mohammad Taher Pilehvar and Jose Camacho-Collados. “WiC: the word-in-context dataset for evaluating context-sensitive meaning representations”. NAACL, 2019. WEB: https://arxiv.org/abs/1808.09121

[17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. “Language models are unsupervised multitask learners”, 2019WEB:https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[18] Yi Tay, Vinh Q. Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al.

"Transformer memory as a differentiable search index". arXiv preprint arXiv:2202.06991, 2022. WEB: https://arxiv.org/abs/2202.06991

[19] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, and Denny Zhou. "Self-consistency improves chain of thought reasoning in language models". arXiv preprint arXiv:2203.11171, 2022 WEB: https: //arxiv.org/abs/2203.11171

[20] Andrew K. Lampinen, Ishita Dasgupta, Stephanie C.Y. Chan, Kory Matthewson, Michael Henry Tessler, Antonia Creswell, James L. McClelland, Jane X. Wang, and Felix Hill. "Can language models learn from explanations in context?". Findings of EMNLP, 2022. WEB: https://arxiv.org/abs/2204.02329.

[21] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Chi. "Least-to-most prompting enables complex reasoning in large language models". arXiv preprint arXiv:2205.10625, 2022. WEB: https://arxiv.org/abs/2205.10625

[22] Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, Dipanjan Das, and Jason Wei. "Language models are multilingual chain-of-thought reasoners". arXiv preprint arXiv:2210.03057, 2022. WEB: https://arxiv.org/abs/ 2210.03057.

[23] Simran Arora, Avanika Narayan, Mayee F Chen, Laurel J Orr, Neel Guha, Kush Bhatia, Ines Chami, Frederic Sala, and Christopher Ré. "Ask me anything: A simple strategy for prompting language models". arXiv preprint arXiv:2210.02441, 2022. URL https://arxiv.org/abs/2210.02441

[24] Lieber, O. Sharir, B. Lenz, and Y. Shoham. "Jurassic-1: Technical details and evaluation". White Paper. AI21 Labs, 2021. WEB:https://uploads-ssl.webflow.com/60fd4503684b466578c0d307/61138924626a6981ee09caf6_jurassic_tech_paper.pdf

[25] Smith, M. Patwary, B. Norick, P. LeGresley, S. Rajbhandari, J. Casper, Z. Liu, S. Prabhumoye, G. Zerveas, V. Korthikanti, E. Zhang, R. Child, R. Y. Aminabadi, J. Bernauer, X. Song, M. Shoeybi, Y. He, M. Houston, S. Tiwary, and B. Catanzaro. "Using Deepspeed and Megatron to Train Megatronturing NLG 530b, A Large-Scale Generative Language Model". arXiv preprint arXiv:2201.11990, 2022. WEB: https://arxiv.org/abs/2201.11990