Learning from Observations of Large Language Model Capabilities

Qinyuan Ye

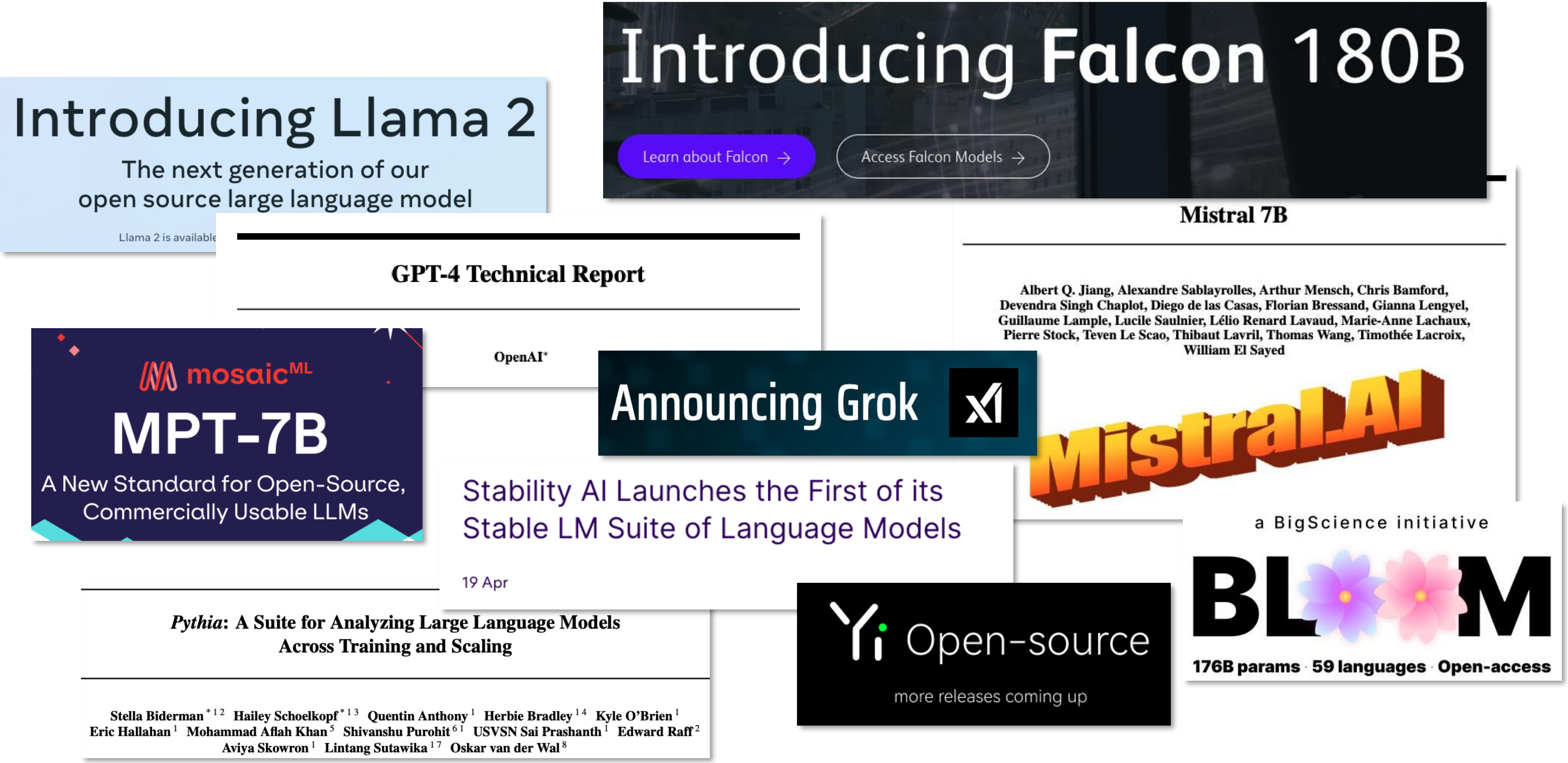
University of Southern California

Nov 28, 2023

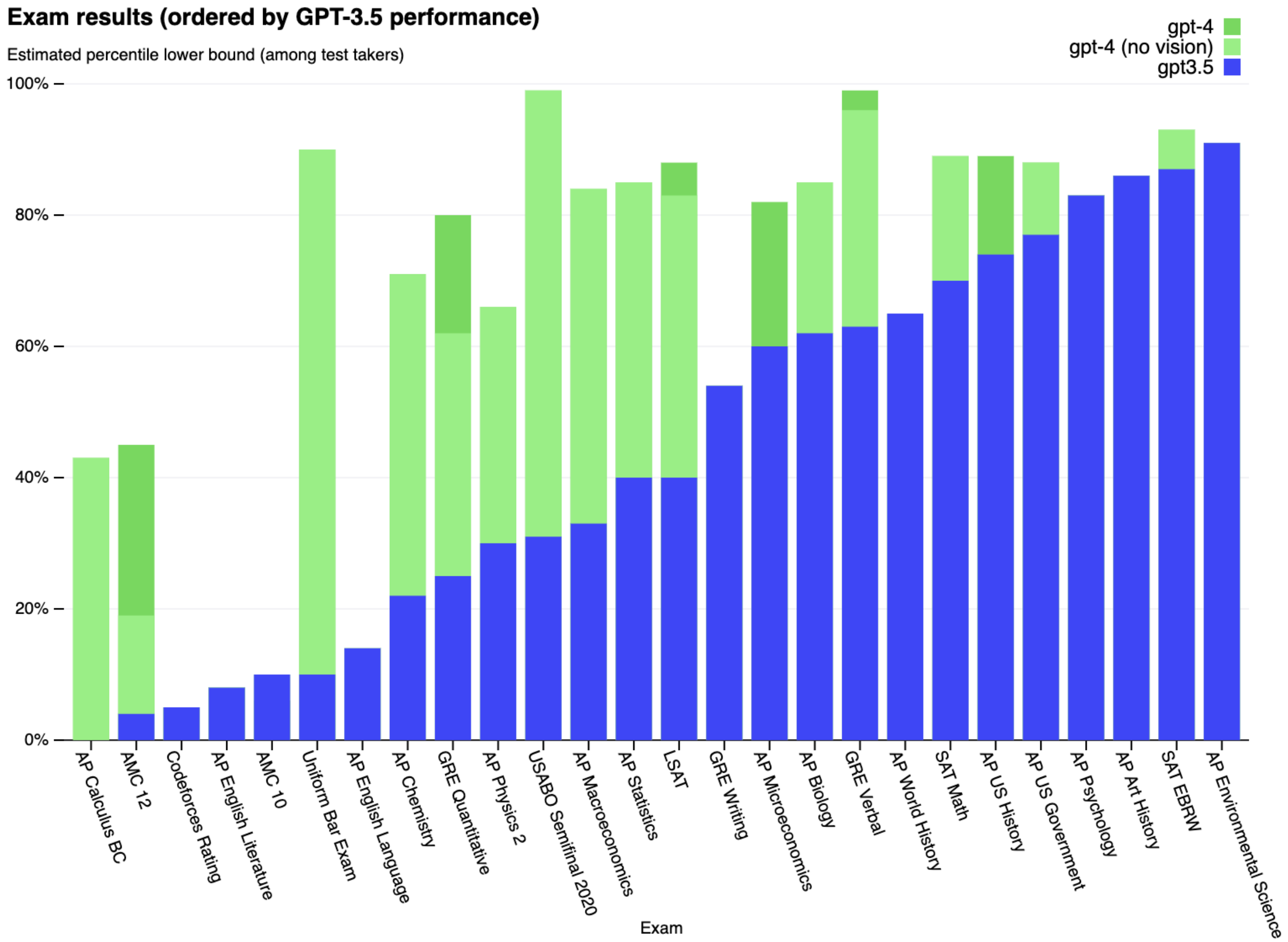
About me

- Qinyuan Ye 叶沁媛
  - Fifth-year PhD student at USC NLP
  - Advised by Professor Xiang Ren
- I want to build intelligent NLP systems that are versatile (adapt quickly) and curious (learn autonomously).
  - Supervision signals: explanations and instructions
  - Learning paradigms: meta-learning and self-improving
- Recently, I'm also interested in understanding large language models capabilities scientifically.
  - **This talk!**

New LLM releases!

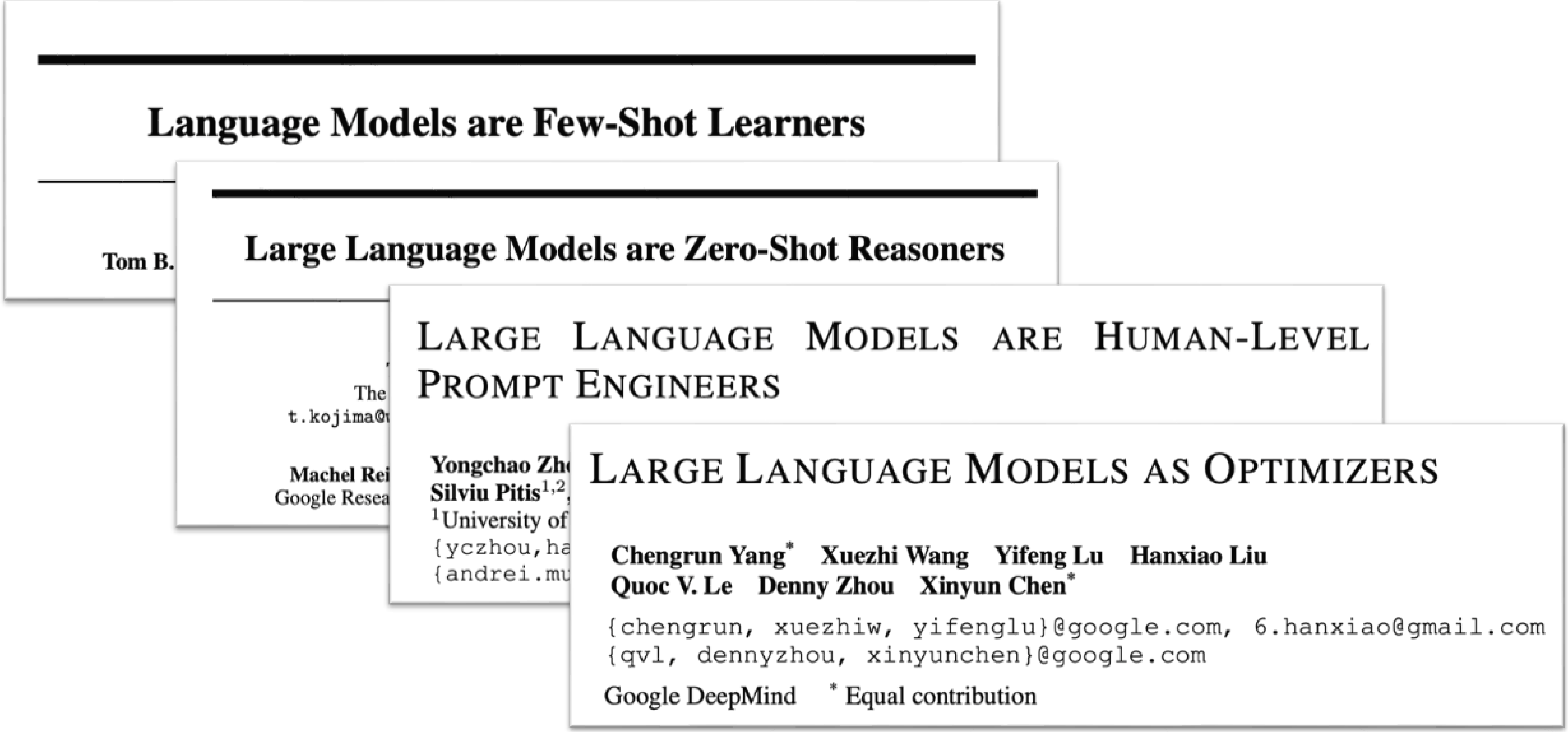


LLMs can do so many things!



GPT-4 Blogpost (https://openai.com/research/gpt-4)

LLMs can do so many things!



Language Models are Few-Shot Learners

Tom B.

Large Language Models are Zero-Shot Reasoners

The
t.kojima@

Machel Rei
Google Resea

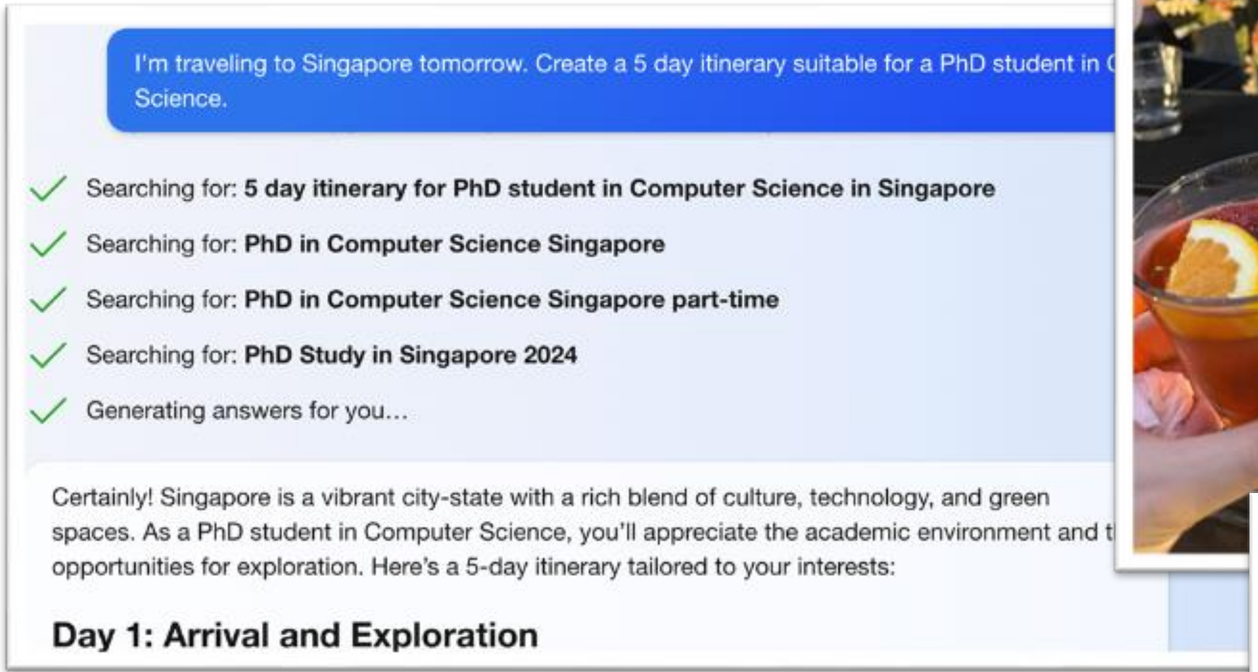
LARGE LANGUAGE MODELS ARE HUMAN-LEVEL PROMPT ENGINEERS

Yongchao Zh
Silviu Pitis[1,2]
[1]University of
{yczhou,ha
{andrei.mu

LARGE LANGUAGE MODELS AS OPTIMIZERS

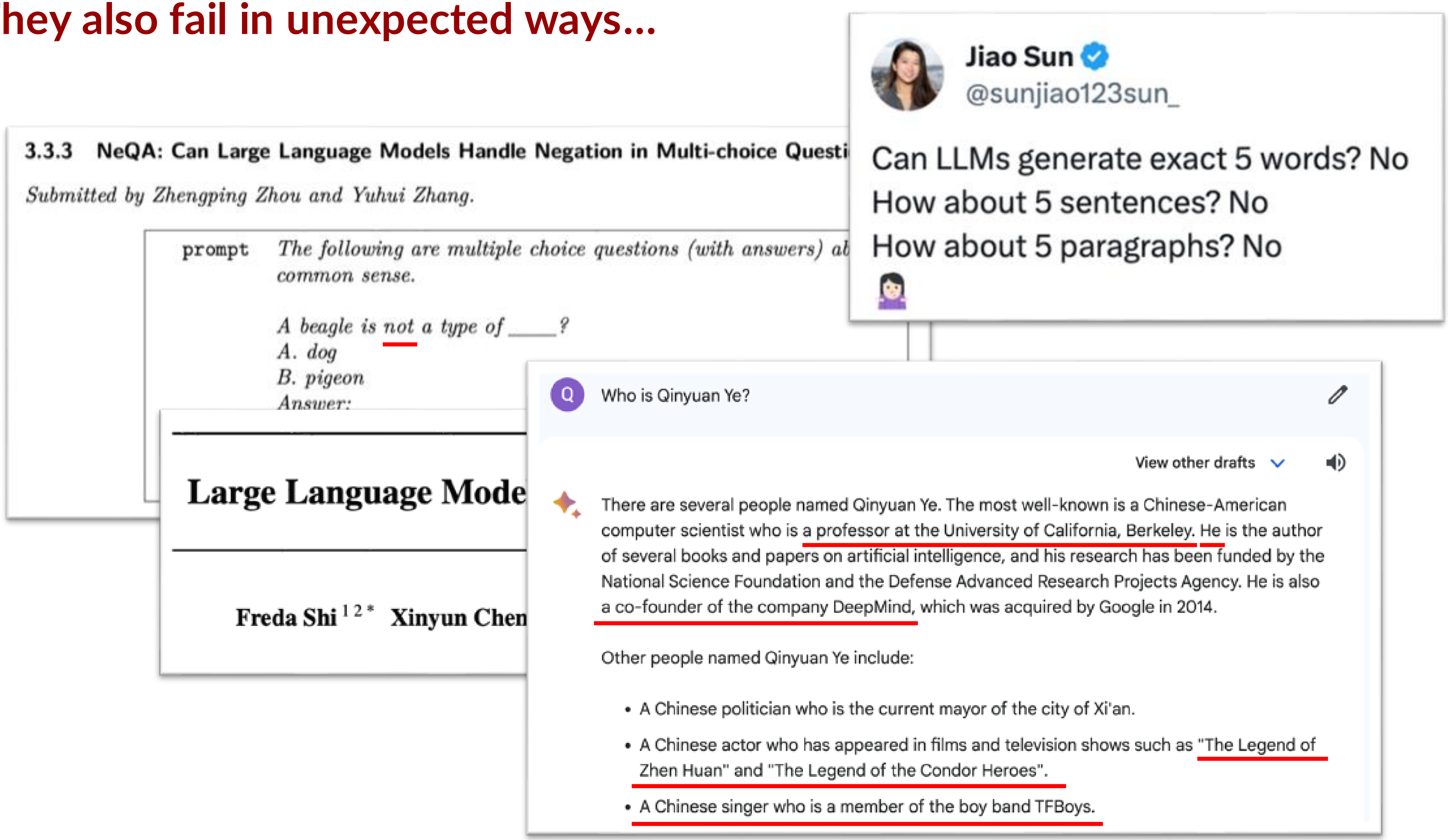
Chengrun Yang* Xuezhi Wang Yifeng Lu Hanxiao Liu
Quoc V. Le Denny Zhou Xinyun Chen*
{chengrun, xuezhiw, yifenglu}@google.com, 6.hanxiao@gmail.com
{qvl, dennyzhou, xinyunchen}@google.com
Google DeepMind * Equal contribution

LLMs can do so many things!



(Source)

They also fail in unexpected ways…



🧑‍💻 ⁉️ 🤖

We have limited understanding of what LLMs are (not) capable of.

🧑‍💻 🔍 🤖

Can we learn from our observations of LLM capabilities?

🧑‍💻 🤝 🤖

And use our findings to assist future LLM research and development?

How Predictable Are Large Language Model Capabilities?

A Case Study on BIG-bench

When new LLMs are released, how are they evaluated?

Model Family | Size | Tasks

Model	Size	Code	Commonsense Reasoning	World Knowledge	Reading Comprehension	Math	MMLU	BBH	AGI Eval
MPT	7B	20.5	57.4	41.0	57.5	4.9	26.8	31.0	23.5
	30B	28.9	64.9	50.0	64.7	9.1	46.9	38.0	33.8
Falcon	7B	5.6	56.1	42.8					
	40B	15.2	69.2	56.7					
LLAMA 1	7B	14.1	60.8	46.2					
	13B	18.9	66.1	52.6					
	33B	26.0	70.0	58.4					
	65B	30.7	70.7	60.5					
LLAMA 2	7B	16.8	63.9	48.9					
	13B	24.5	66.9	55.4					
	34B	27.8	69.9	58.7					
	70B	**37.5**	**71.9**	**63.6**					

\# In-context Examples

		NaturalQuestions				TriviaQA (Wiki)			
		0-shot	1-shot	5-shot	64-shot	0-shot	1-shot	5-shot	64-shot
MPT	7B	11.6	17.8	20.8	22.7	55.7	59.6	61.2	61.6
	30B	15.8	23.0	26.6	29.3	68.0	71.3	73.3	73.6
Falcon	7B	15.7	18.1	21.0	24.0	52.6	56.8	64.6	61.1
	40B	**26.3**	29.5	33.5	35.5	74.6	78.6	79.9	79.6
LLAMA 1	7B	16.8	18.7	22.0	26.1	63.3	67.4	70.4	71.0
	13B	20.1	23.4	28.1	31.9	70.1	74.4	77.1	77.9
	33B	24.9	28.3	32.9	36.0	78.7	80.7	83.8	83.6
	65B	23.8	31.0	35.0	39.9	81.7	84.5	85.9	86.0
LLAMA 2	7B	16.4	22.7	25.7	29.5	65.8	68.9	72.1	73.7
	13B	16.1	28.0	31.2	34.6	73.1	77.2	79.6	79.4
	34B	25.1	30.0	32.8	39.9	81.0	83.3	84.5	84.6
	70B	25.3	**33.0**	**39.5**	**44.3**	**82.4**	**85.0**	**87.6**	**87.5**

🤯 So many experiment configurations!

Llama 2: Open Foundation and Fine-Tuned Chat Models (Touvron et al., 2023)

How predictable are large language model capabilities?

LLM User

What model scale should I use?

LLM Developer

What tasks should I prioritize in evaluation?

LLM Researcher

Which capabilities are hard to predict?

Part 1: Performance Prediction on BIG-bench

- Problem Definition

* limitations apply

\# Parameters

\# In-context Examples

Normalized Performance

$$\hat{y} = f(l, n_{param}, t, n_{shot})$$

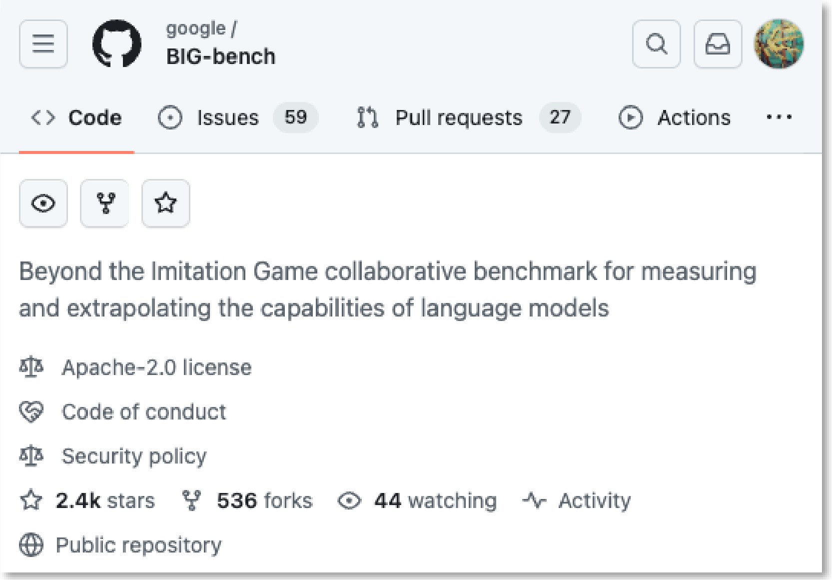
Model Family

Tasks

Regression Problem. Evaluated with **RMSE** and **R^2 score**.

Part 1: Performance Prediction on BIG-bench

- Data



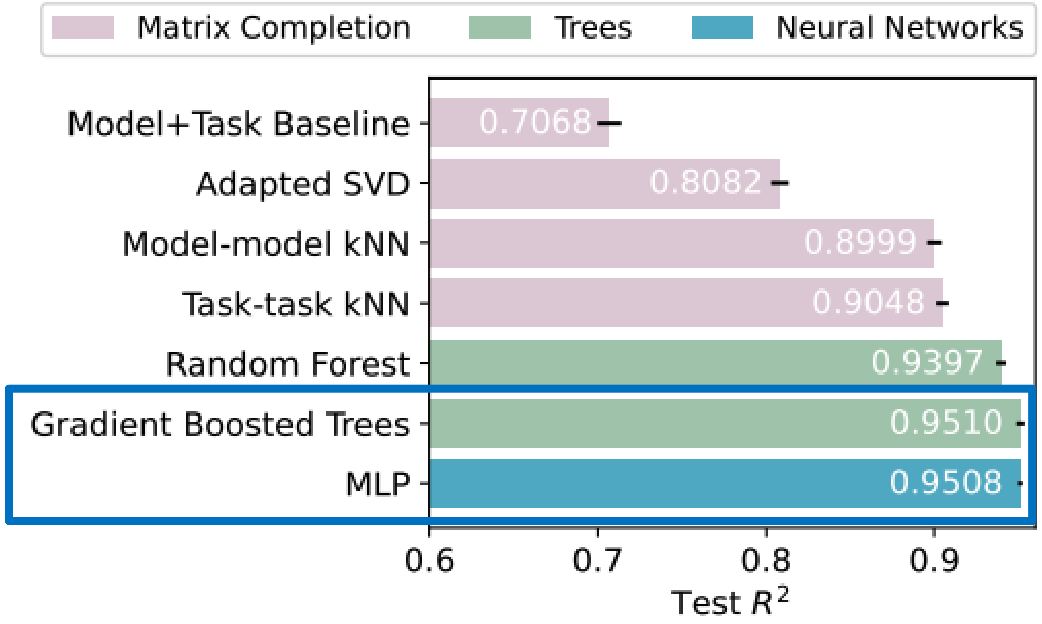
We gathered and filtered the records in **BIG-bench**.

# Experiment Records	56,143
# Model Families	6
	BIG-G T=0, BIG-G T=1, BIG-G Sparse, PaLM GPT-3, Gopher
# Models†	51
# BIG-bench Tasks	134
# BIG-bench Subtasks‡	313
$\{n_{shot}\}$	$\{0, 1, 2, 3, 5\}$

We got **56k records** covering diverse models and tasks.

Part 1: Performance Prediction on BIG-bench

- Results (Random Train-Test Split)

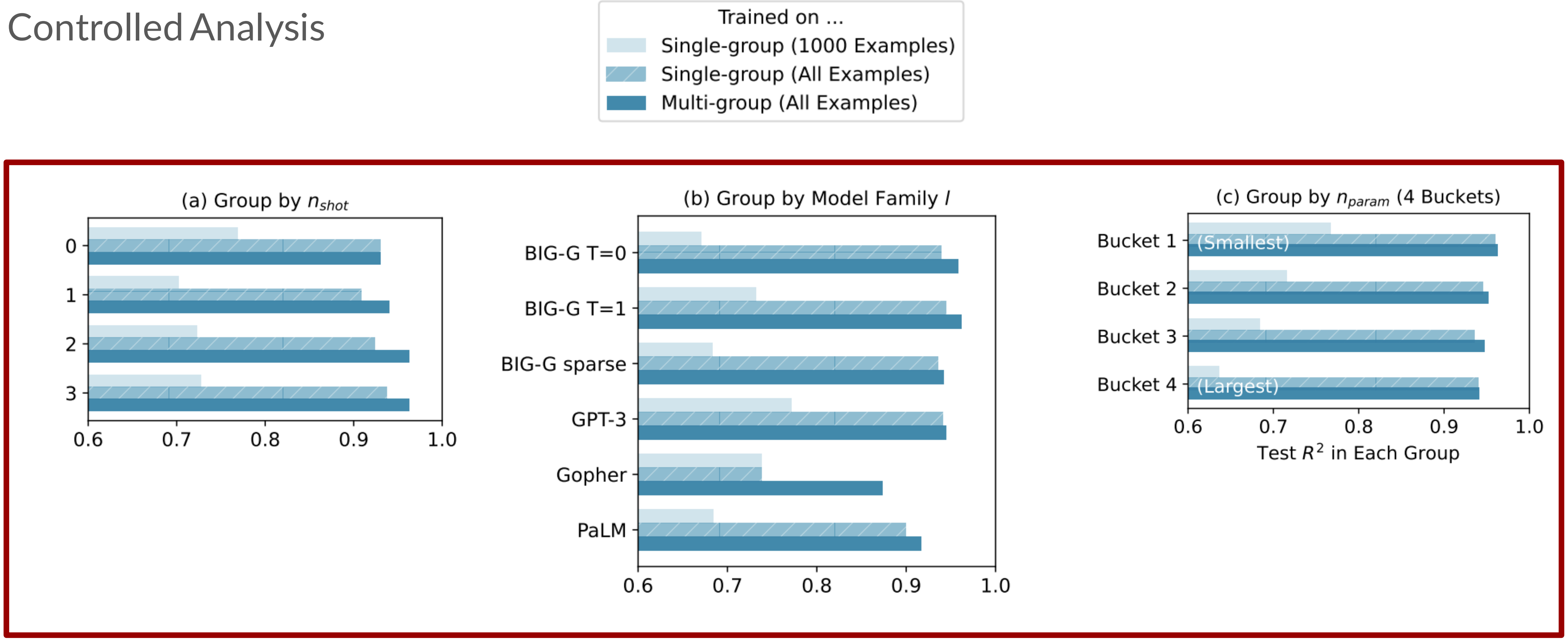


RMSE < 0.05
on average mis-predict by <0.05 when the range is [0,1]

R^2 > 95%
explain more than 95% variance in the target variable

Part 1: Performance Prediction on BIG-bench

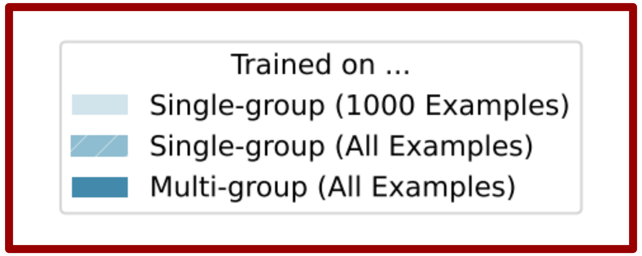
- Controlled Analysis



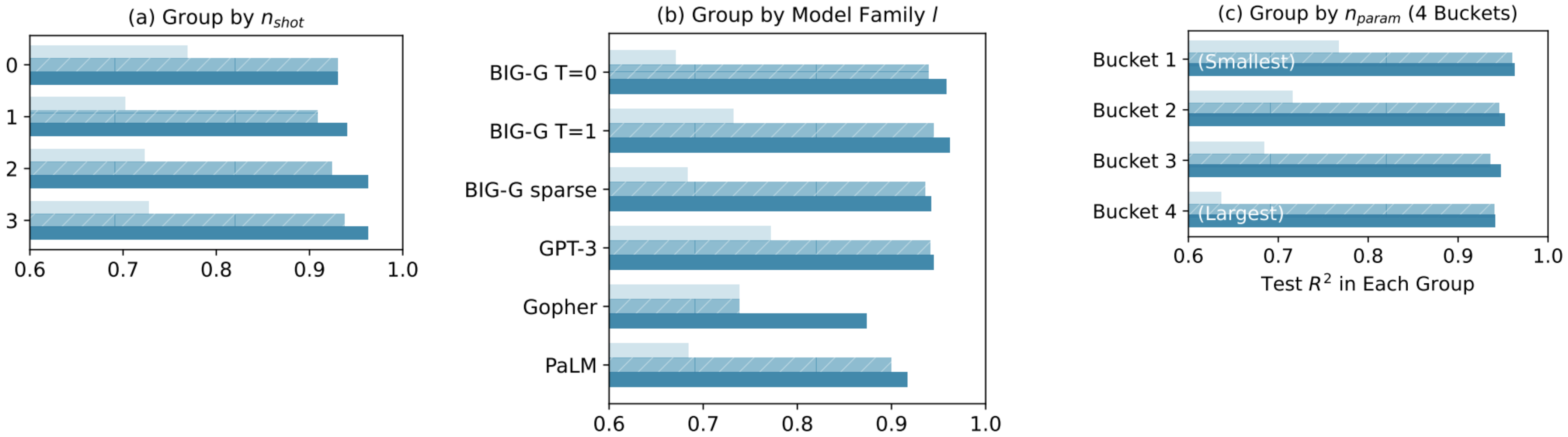
We split test set into groups and report the performance in each group.

Part 1: Performance Prediction on BIG-bench

- Controlled Analysis

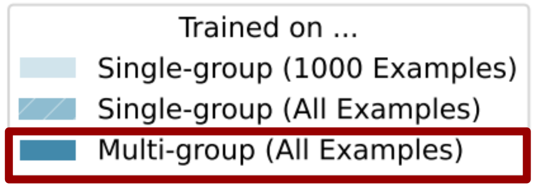


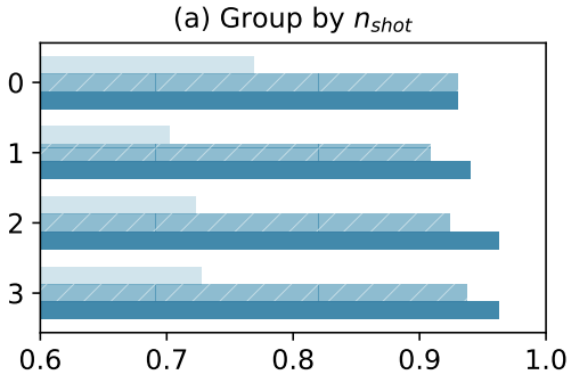
We control what the training set contains.



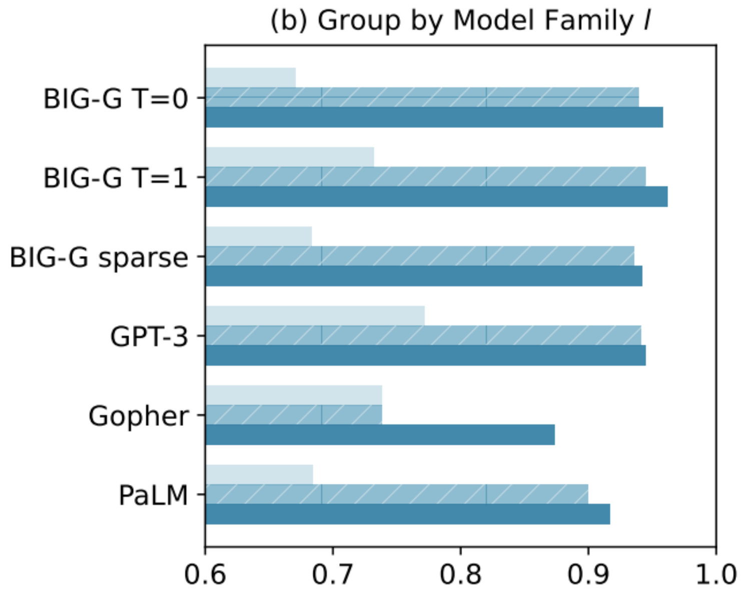
Part 1: Performance Prediction on BIG-bench

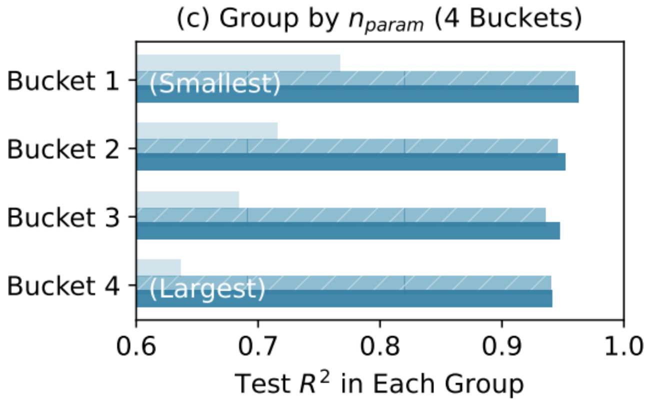
- Controlled Analysis





Zero-shot performance is harder to predict.

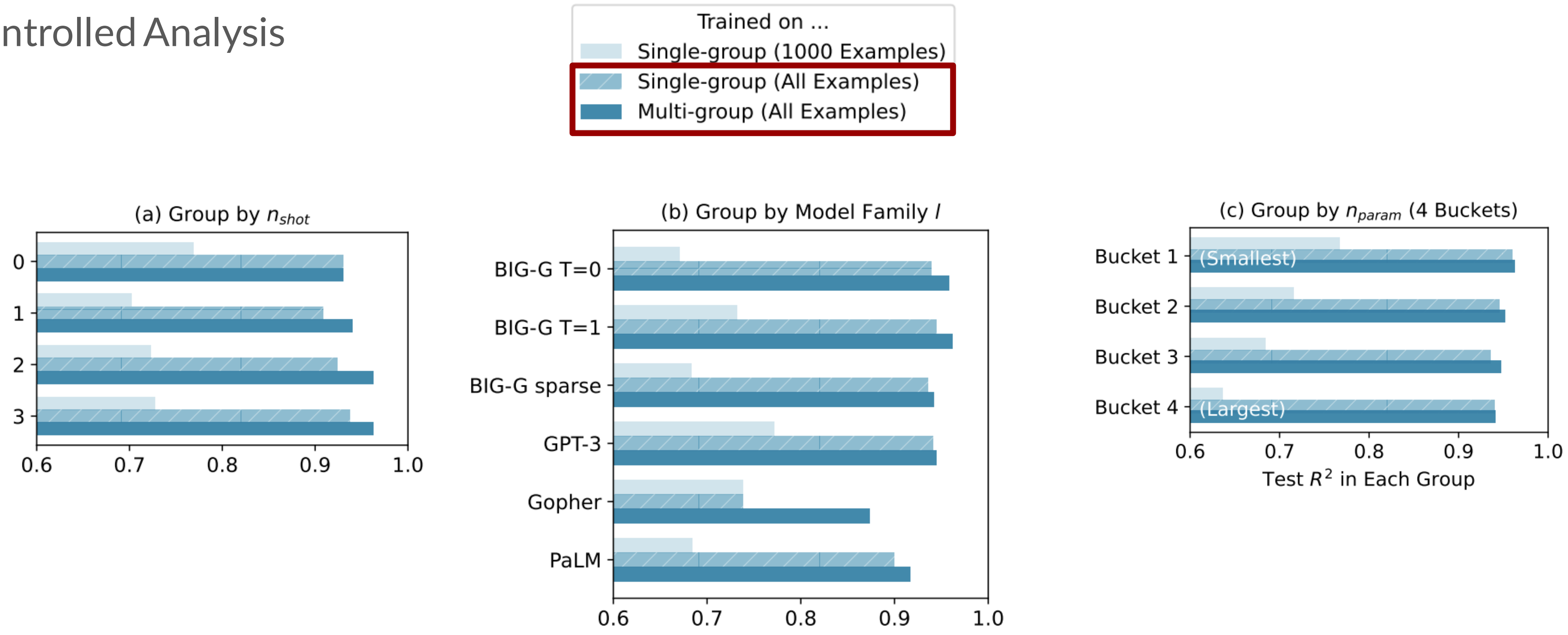




Performance of larger models is harder to predict.

Part 1: Performance Prediction on BIG-bench

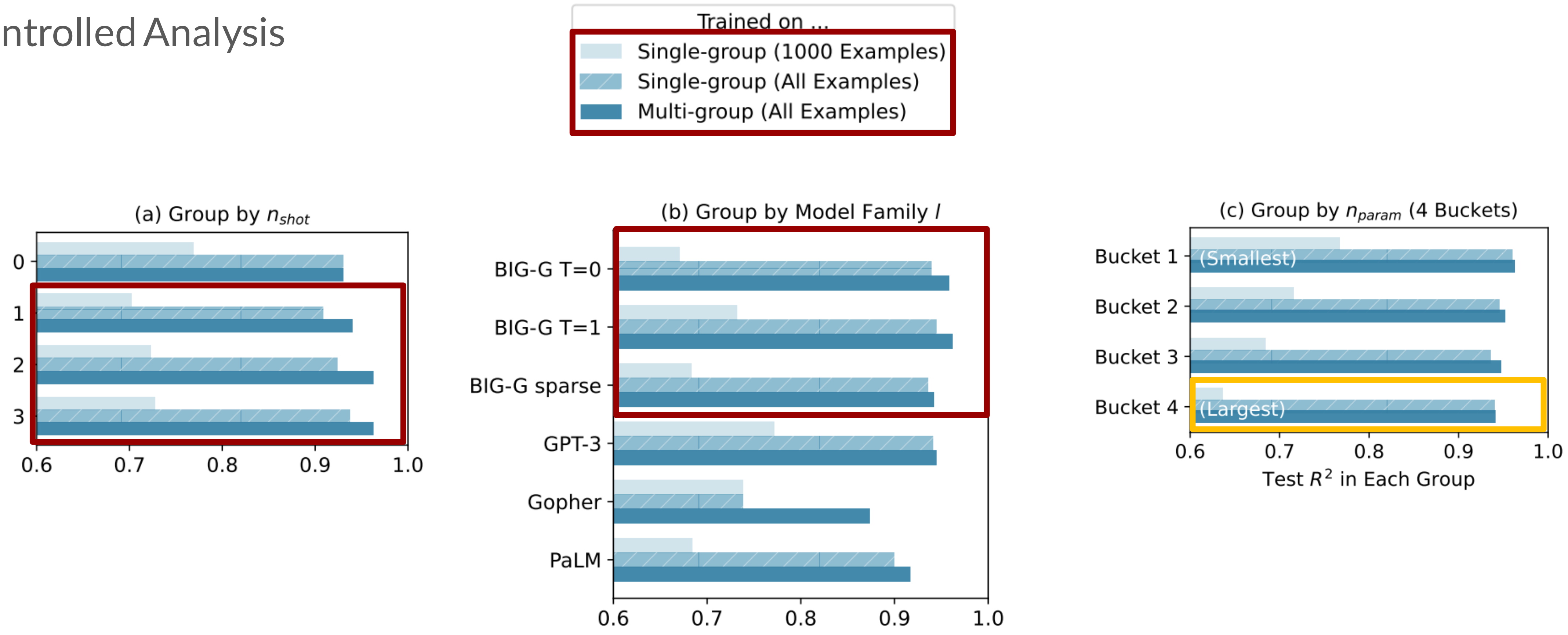
- Controlled Analysis



Multi-group training is always helpful.

Part 1: Performance Prediction on BIG-bench

- Controlled Analysis

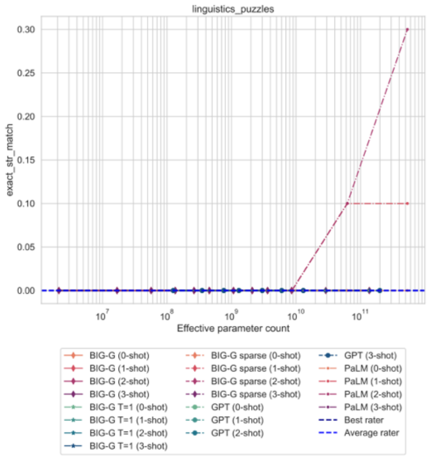


Some groups benefit more from multi-group training.

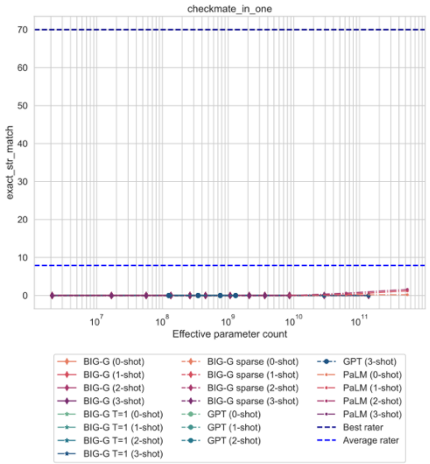
Some groups are intrinsically harder to predict.

Part 1: Performance Prediction on BIG-bench

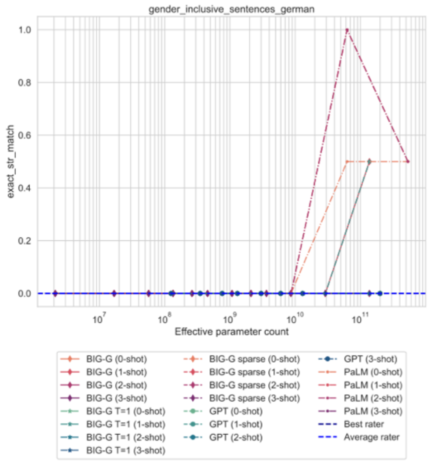
- Least Predictable Tasks



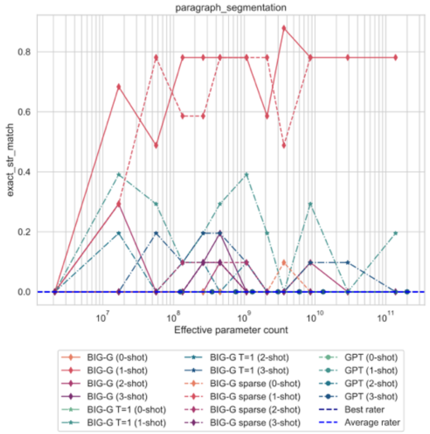
(a) linguistics_puzzles



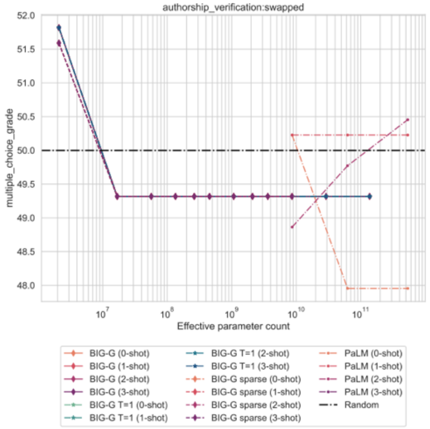
(b) checkmate_in_one



(c) gender_inclusive _sentences_german



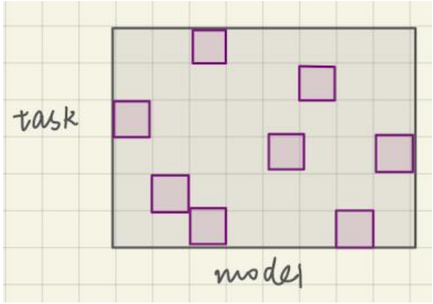
(d) paragraph_segmentation



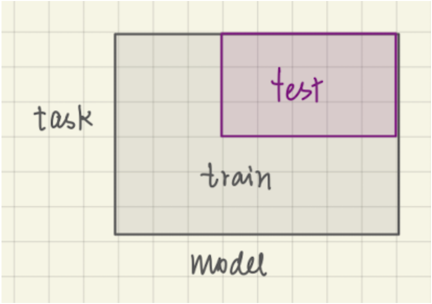
(e) authorship_verification: swapped

Part 1: Performance Prediction on BIG-bench

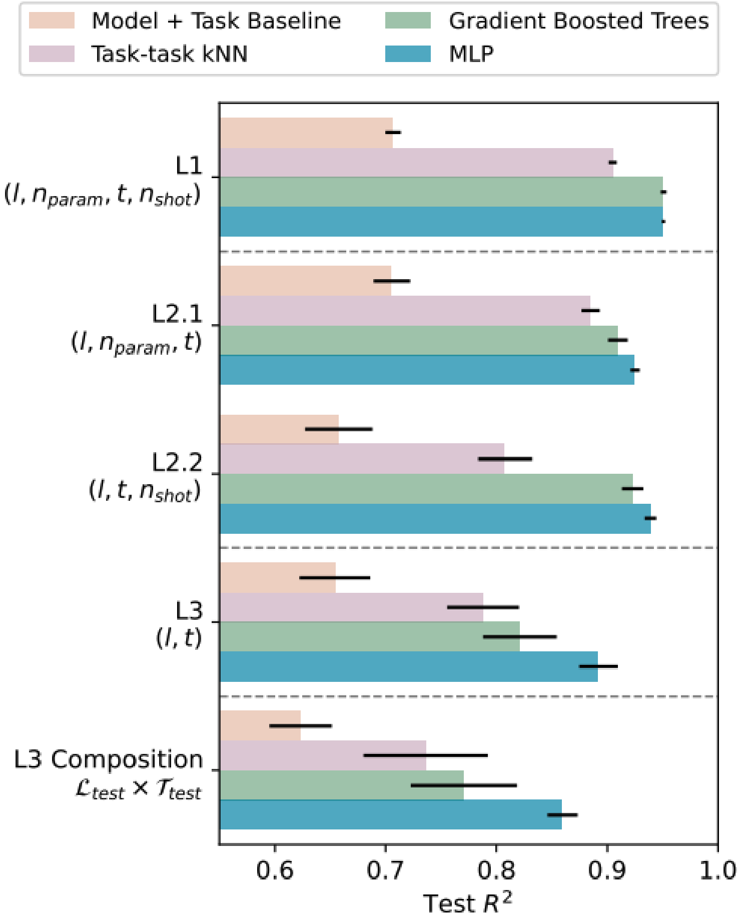
- Results (Challenging Train-Test Split)



Easier →



Harder →



Prediction accuracy decreases when the train-test split becomes more challenging!

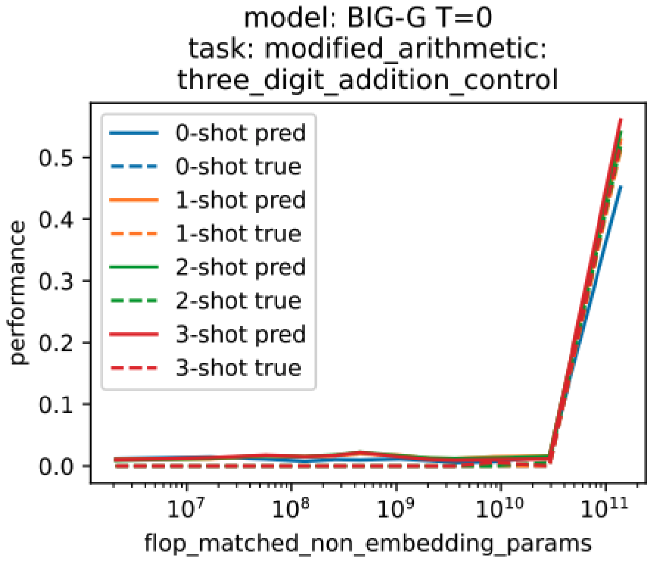
Part 1: Performance Prediction on BIG-bench

Emergent abilities (Wei et al., 2022)

... are *in general* harder to predict

	RMSE (↓)	R^2 (↑)
Emergent Tasks	0.0541	93.86%
Non-emergent Tasks	0.0496	95.16%
All	0.0499	95.07%

... can be predicted accurately *in certain cases*

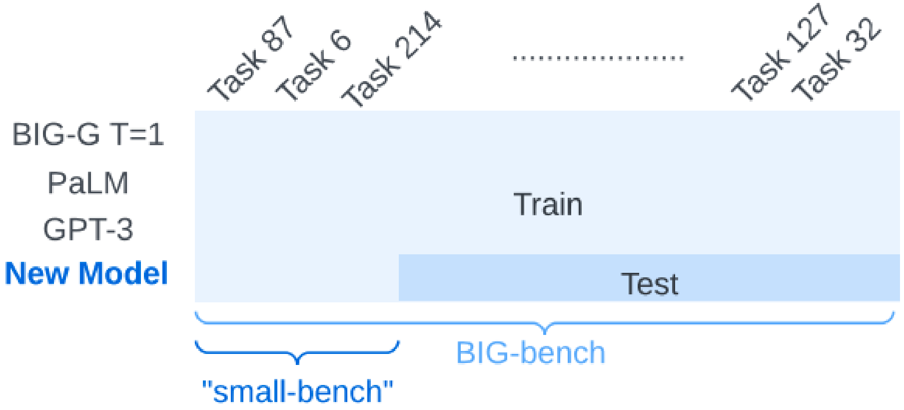


Potential Reason

A similar task is emergent and is in the training set.

Part 2: Searching for "small-bench"

- Problem Definition



Performance on remaining tasks are *maximally* recovered

$$\underset{\mathcal{T}_{train}}{\arg\max}\ \boxed{R^2(\mathcal{T}_{test} \times \mathcal{L}_{test})}$$

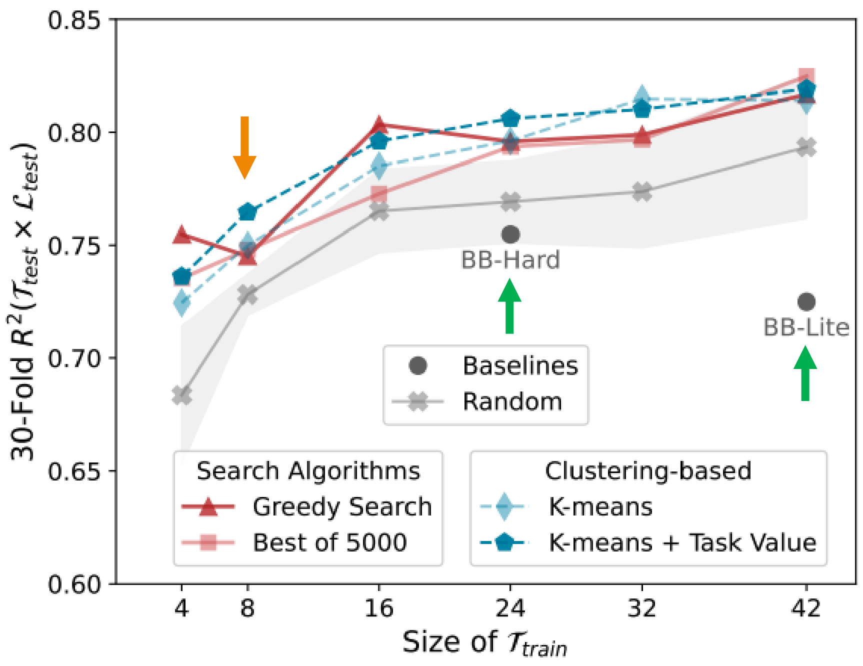
$$\text{s.t.}\ \boxed{\mathcal{T}_{train} \subseteq \mathcal{T},}\ \boxed{|\mathcal{T}_{train}| = b}$$

Select *b* tasks

Given an evaluation budget of *b*

Part 2: Searching for “small-bench”

- Results

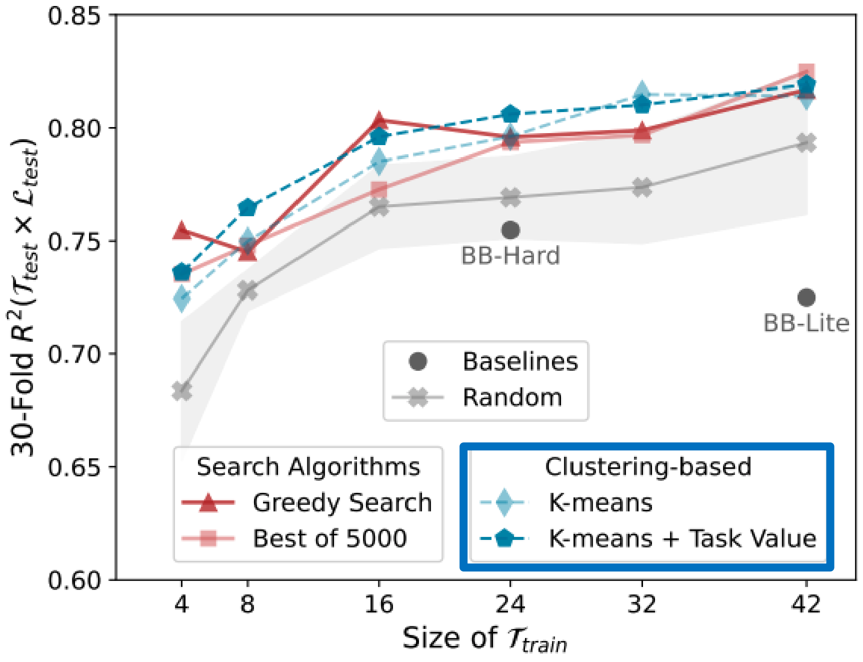


BIG-bench Lite and BIG-bench Hard are suboptimal if the goal is to recover the performance on remaining tasks.

We are able to find subsets that are as informative as BIG-bench Hard while being 3x smaller.

Part 2: Searching for “small-bench”

- Results



K-means
Clustering task representations learned by the MLP predictors in Part 1;
Then select tasks close to cluster centroids.

Task Value
Estimated from “Best of 5000”.

Task diversity and task value are important factors in constructing “small-bench.”

Summary

- We gathered **56k LLM experiment records** in BIG-bench.
- We trained models to **predict LLM performance on unseen experiment configurations**.
  - An MLP predictor can achieve RMSE < 5%, R^2 > 95% on the random train-test split.
  - Prediction performance changes when train-test distribution changes.
  - Emergent abilities are harder to predict in general, but can be predicted accurately in some cases.
- We searched for **"small-bench,"** a subset of BIG-bench, from which the full BIG-bench performance can be maximally recovered.
  - BIG-bench Lite and BIG-bench Hard are sub-optimal for this purpose.
  - Task diversity and task value are important factors for constructing "small-bench."

Looking Ahead

- Rethinking LLM evaluation

Previous work

Models

Scenarios	J1-Jumbo v1	J1-Grande v1	J1-Large v1	Anthropic-LM v4-s3	BLOOM	T0++	Cohere Xlarge v20220609	Cohere Large v20220720	Cohere Medium v20220720	Cohere Small v20220720	GPT-NeoX	GPT-J	T5	UL2	OPT (175B)	OPT (66B)	TNLGv2 (530B)	TNLGv2 (7B)	davinci	curie	babbage	ada	text-davinci-002	text-curie-001	text-babbage-001	text-ada-001	GLM	YaLM
NaturalQuestions (open)																												
NaturalQuestions (closed)																			✔	✔	✔	✔						
BoolQ	✔		✔		✔								✔	✔	✔	✔	✔		✔	✔	✔	✔						
NarrativeQA																												
QuAC																			✔	✔	✔	✔	✔	✔	✔	✔		
HellaSwag	✔		✔	✔	✔	✔					✔	✔	✔	✔	✔	✔	✔		✔	✔	✔	✔	✔	✔	✔			
OpenBookQA					✔						✔	✔		✔	✔	✔			✔	✔	✔	✔						
TruthfulQA				✔															✔	✔	✔	✔	✔	✔	✔	✔		
MMLU											✔	✔							✔	✔							✔	
MS MARCO																												
TREC																												
XSUM													✔	✔														
CNN/DM													✔	✔					✔	✔	✔		✔	✔	✔			
IMDB														✔														
CivilComments														✔														
RAFT																			✔									

Holistic Evaluation of Language Models (Liang et al., 2023)

Task selection is often heuristic, following past practices, or done arbitrarily.

Looking Ahead

- Broadening observations on LLM capability landscape

Currently

google / BIG-bench

Normalized Performance

Parameters

In-context Examples

$$\hat{y} = f(l, n_{param}, t, n_{shot})$$

Model Family

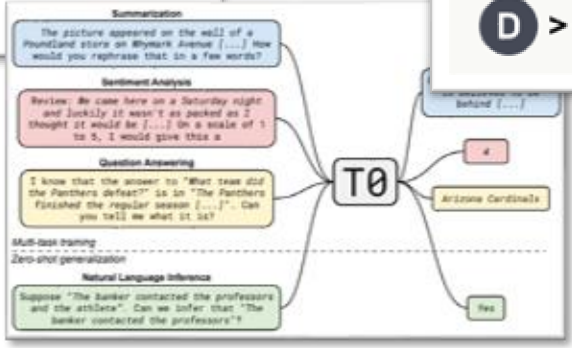
Tasks

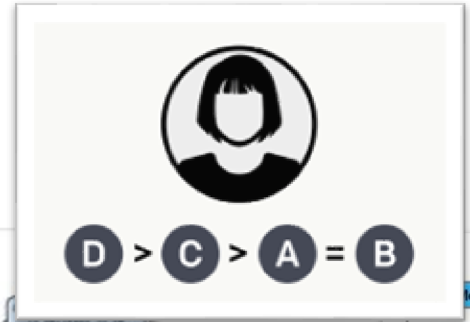
HELM

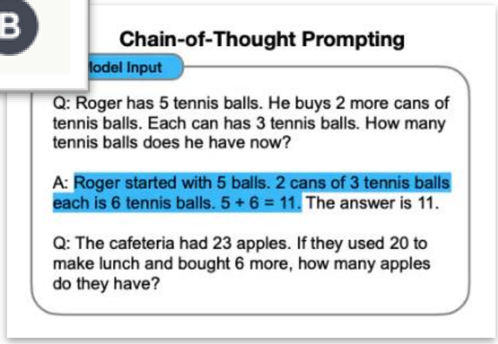
Spaces HuggingFaceH4 / open_ll

like 6.04k Running

Open LLM Leaderboard







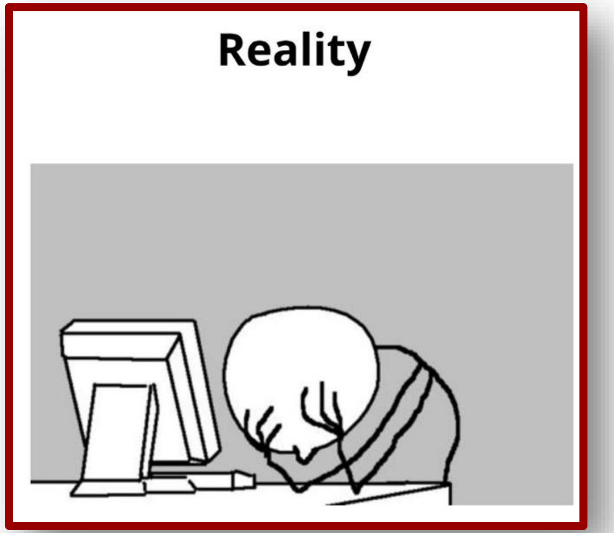
Liang et al., 2023; Sahn et al., 2022; Ouyang et al., 2022; Wei et al., 2022.

Will be nice to take these into account!

Estimating Large Language Model Capabilities without Labeled Test Data

The Promises of Few-shot Learning

I only need 16 annotated examples!



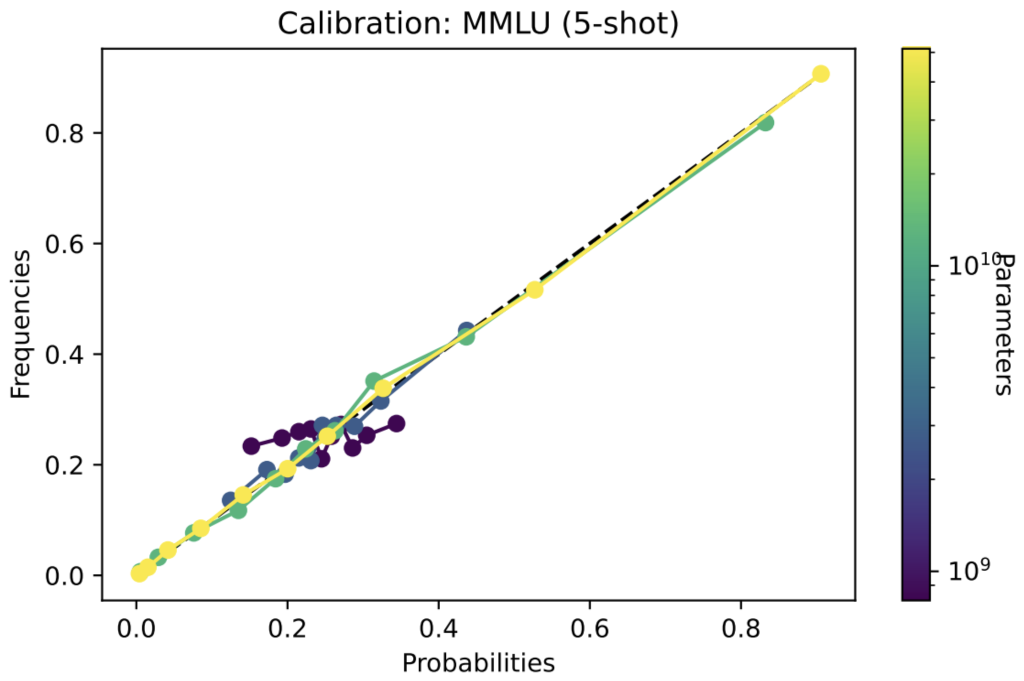
To know whether the LLM works well for my task,
I need to annotate 1000 examples for testing...

Can we estimate LLM capabilities without labeled test data?

https://javascript.plainenglish.io/15-things-all-programmers-can-relate-to-7db1ce811b8

Estimating Large Language Model Capabilities without Labeled Test Data

- We can gain insights from model confidence / calibration.



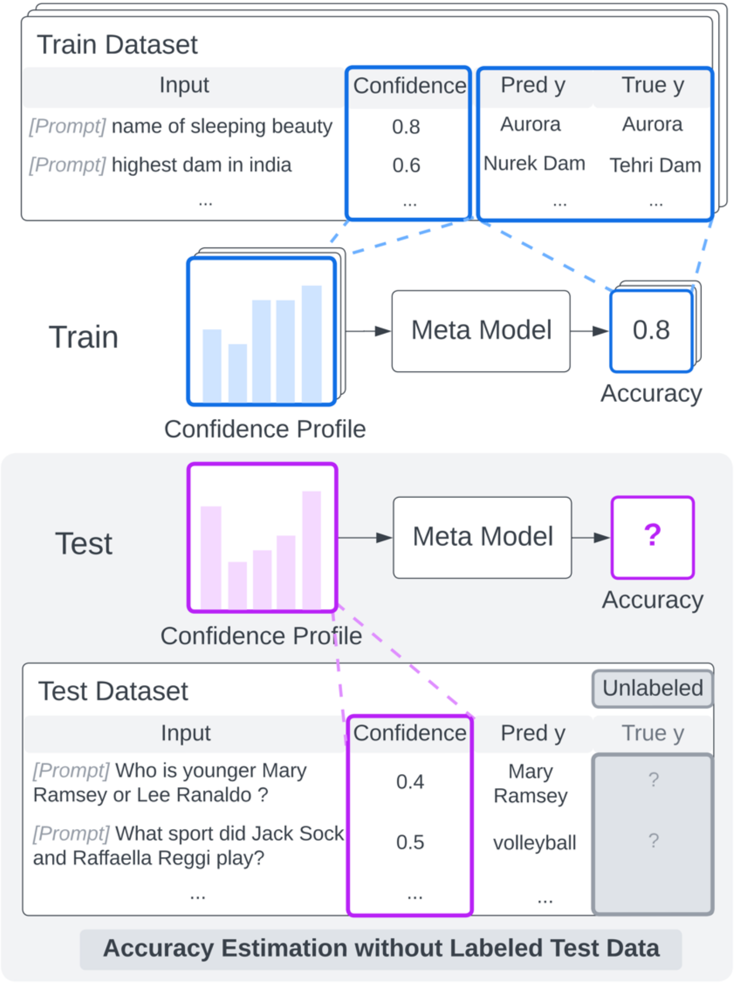
Language Models (Mostly) Know What They Know (Kadavath et al., 2022)

Naïve Solution:
Take the average of confidence on all test examples.

Limitations:
Perfect calibration is not guaranteed to happen.
Limited to classification / multiple-choice QA.

Estimating Large Language Model Capabilities without Labeled Test Data

- **Our proposed method**
- Training a meta-model that estimates performance based on **confidence profiles**.



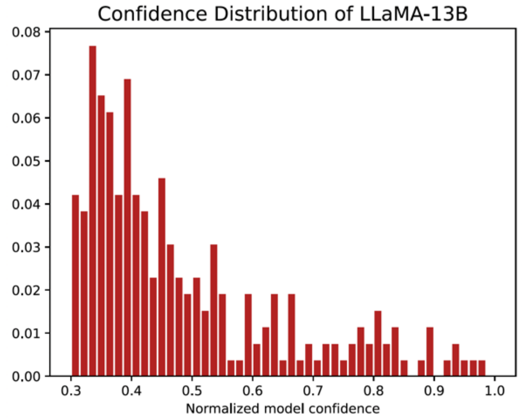
Confidence Profiles

- Compute the confidence score for each test example

Closed-set Generation $$s^{M,c}(x) = \frac{p_{\hat{y}}}{\sum_{\tilde{y} \in \mathcal{Y}} p_{\tilde{y}}}$$

Open-ended Generation $$s^{M,c}(x) = -\sum_{t=1}^{|\hat{y}|} \log p_t(\hat{y}_t).$$

- Collect confidence distributions and extract the percentile vector



1. Get model confidence at 5%, 10%, ..., 95% percentile of the distribution.

2. Use this vector as the input feature to the meta-model.

- Train a meta-model to output dataset-level accuracy based on the percentile vector

Experiment Settings

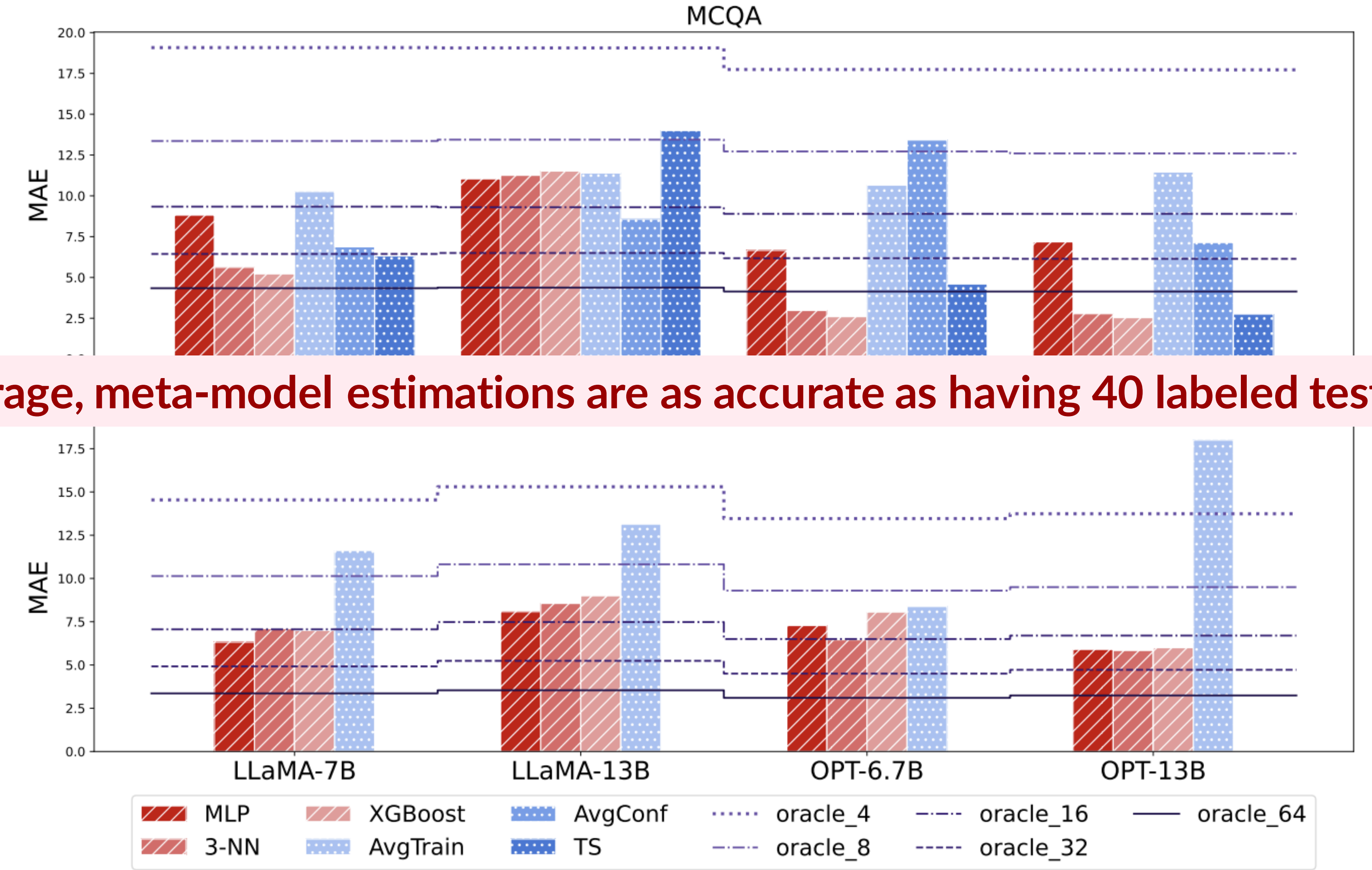
- Getting the observations of LLM performance
  - **3 Tasks**: MMLU (57 subtasks), MCQA (21 subtasks), CBQA (13 subtasks)
  - **4 LLMs**: OPT 6.7B, OPT 13B, LLaMA 7B, LLaMA 13B
  - **12 settings** in total (3 tasks x 4 LLMs), **42,360 ICL performance observations** in total
- For each setting, run cross-validation on the observations
  - Training the meta-model on some subtasks
  - Estimate the accuracy on unseen subtasks

Compared Methods

- Meta-models (ours)
  - We consider **k-NN**, **MLP**, **XGBoost** models for regression
- Baselines
  - **Train Avg**: use average accuracy on seen subtasks as estimated accuracy.
  - **Avg Confidence**: use average confidence on test examples as estimated accuracy.
  - **Temperature Scaling**: scale the confidence with an extra temperature parameter; fit it on the training set.
- Oracles
  - **Oracle *k***: assuming we have *k* annotated test examples, use model accuracy on these *k* examples as estimation of accuracy on the full test set.
  - *k*=4, 8, 16, 32, 64

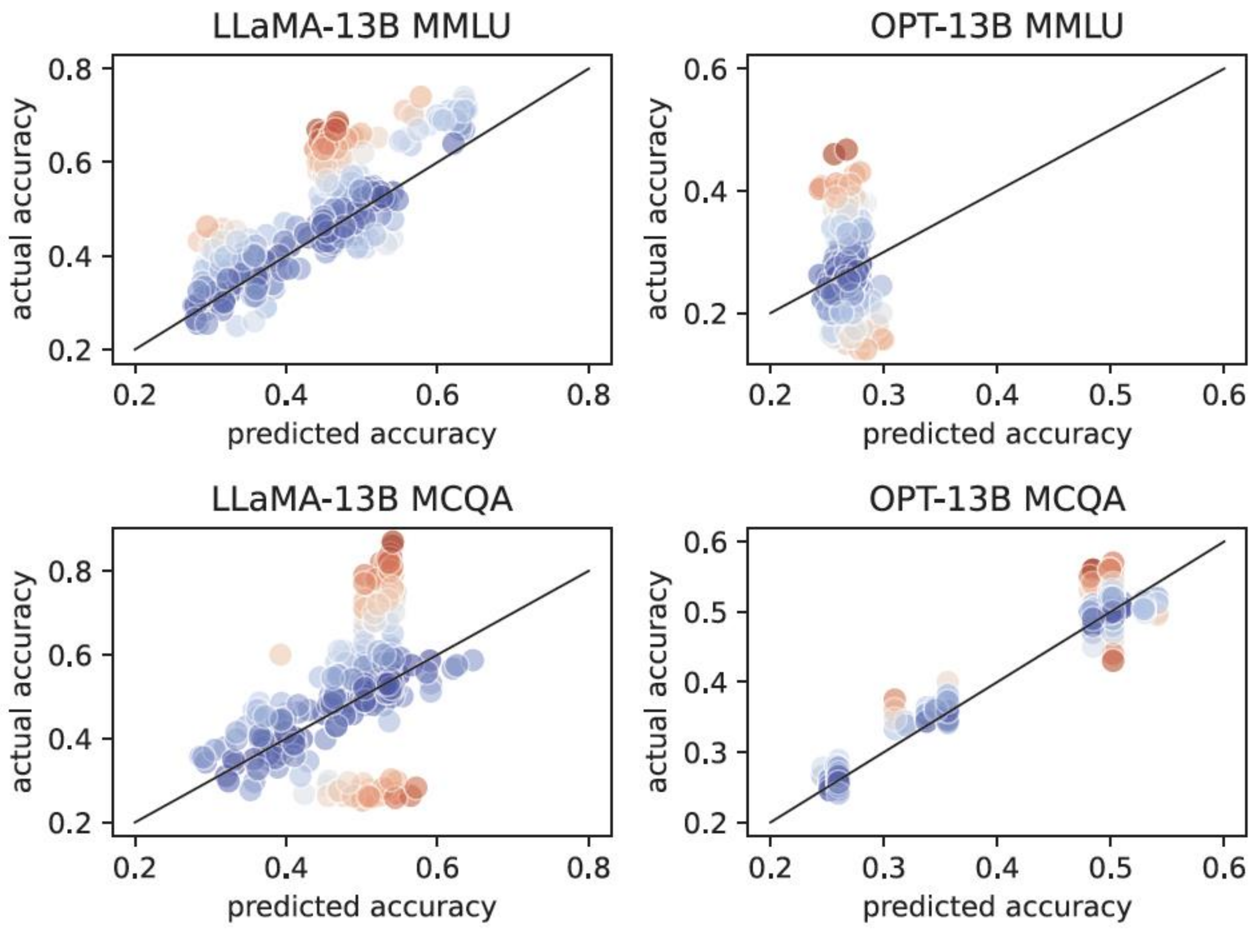
Results

Meta Models (ours) | Baselines | Oracles (lines)



On average, meta-model estimations are as accurate as having 40 labeled test examples.

Results



LLaMA-13B MMLU
OPT-13B MMLU
LLaMA-13B MCQA
OPT-13B MCQA
actual accuracy
predicted accuracy
0.2
0.3
0.4
0.5
0.6
0.8

Summary

- We formalize the problem of **few-shot ICL accuracy estimation**.
    - Given a handful of labeled in-context examples and **a set of *unlabeled* test examples**, our goal is to estimate the overall accuracy of ICL on these test examples

- We propose to address this problem by **training a "meta-model,"** which takes in the LLM's **confidence profile** as input and outputs the task accuracy.

- We benchmark our meta-model method and various baselines on 12 settings (4 LLMs x 3 tasks).
    - On average, **meta-model estimations are as accurate as having 40 labeled test examples**.

Looking Ahead

- Providing a channel for LLMs to say “sorry, I’m not confident about completing this task.”
- Ensuring safety and reliability in LLM-powered applications.

Thank you!

Also, thanks to my awesome collaborators and mentors at USC NLP!

Harvey Yiyun Fu

Albert Xu

Xiang Ren

Robin Jia

USC University of Southern California

https://nlp.usc.edu/

Links

- How Predictable Are Large Language Model Capabilities? A Case Study on BIG-bench

- Paper: https://arxiv.org/abs/2305.14947
- Code: https://github.com/INK-USC/predicting-big-bench

- Estimating Large Language Model Capabilities without Labeled Test Data

- Paper: https://arxiv.org/abs/2305.14802
- Code: https://github.com/harvey-fin/icl-estimate