SFU NatLangLab

CMPT 413/713: Natural Language Processing

Adapting LLMs for tasks
Few-shot learning with LLMs

Spring 2024

2024-02-28

Slides adapted from Anoop Sarkar

"Language provides a natural domain for the study of artificial intelligence, as the vast majority of reasoning tasks can be efficiently expressed and evaluated in language, and the world's text provides a wealth of data for unsupervised learning via generative modeling."

- OpenAI

GPT models

OpenAI

GPT

- *Improving language understanding by generative pre-training* [Radford et al, 2018]
- Large language model with transformers with supervised fine-tuning
  - different model for each task
- Trained on BooksCorpus (800M words), 117M parameters (12 layers)

GPT-2

- *Language Models are Unsupervised Multitask Learner* [Radford et al, 2019]
- Model all tasks as sequence completion with special tokens indicating task
- Trained on WebText (40B words), 1.5B parameters (48 layers)
- No fine-tuning, demonstrated few-shot learning

GPT-3

- *Language Models are Few-Shot Learners* [Brown et al, 2020]
- Trained on Web+Books+Wikipedia (300B words), 175B parameters (96 layers)
- Demonstrated zero-shot and few-shot prompting abilities

GPT models (after GPT-3)

OpenAI

InstructGPT and GPT-3.5 [2022]

- Align responses to human feedback
- Instruction fine-tuning
- Reinforcement learning from human feedback
- Used in initial ChatGPT

GPT-4 [March 2023]

- Multimodal with images and text (GPT-4V)
- Larger, better model

Improving Language Understanding by Generative Pre-Training

GPT1

Alec Radford
OpenAI
alec@openai.com

Karthik Narasimhan
OpenAI
karthikn@openai.com

Tim Salimans
OpenAI
tim@openai.com

Ilya Sutskever
OpenAI
ilyasu@openai.com

https://openai.com/research/language-unsupervised **Jun 2018**

GPT1

Pre-training an autoregressive language model

BooksCorpus: 7K unpublished books (1B words)

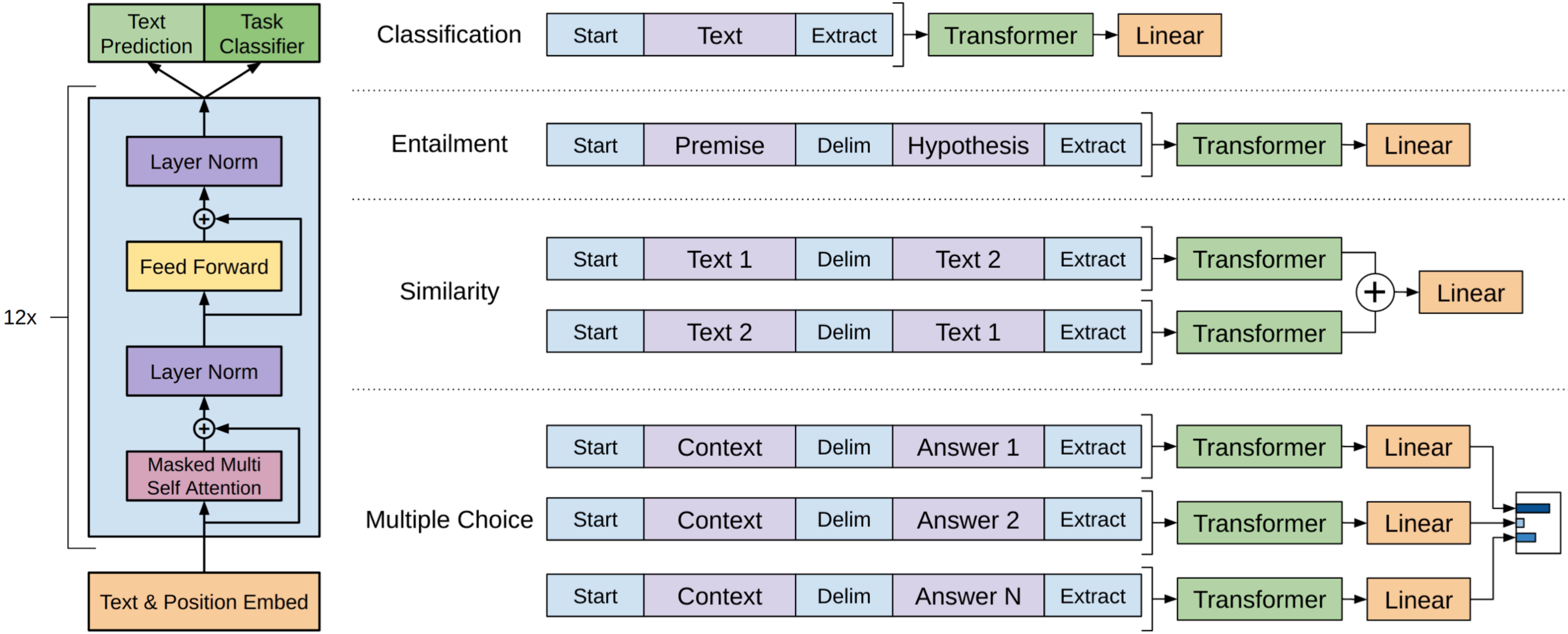
- Start with a large amount of unlabeled data $\mathscr{U} = \{u_1, \ldots, u_n\}$
- Pre-training objective: Maximize the likelihood of predicting the next token
- $L_i(\mathscr{U}) = \sum_i \log P(u_i \mid u_{i-k}, \ldots, u_{i-1}; \Theta)$
- This is equivalent to training a Transformer decoder
    - $h_0 = U\boxed{W_e} + W_p$
    - $h_\ell = \text{transformer\_block}(h_{\ell-1}) \forall \ell \in [1,n]$
    - $P(u) = \text{softmax}(h_n \boxed{W_e^T})$
- Directionality is needed to generate a well-formed probability distribution

$U = (u_{-k}, \ldots, u_{-1})$ is the context vector of tokens

n is the number of Transformer layers

W_e is the token embedding matrix

W_p is the position embedding matrix



This setup was for fine-tuning GPT1 but also works for in-context learning in GPT2 and GPT3.

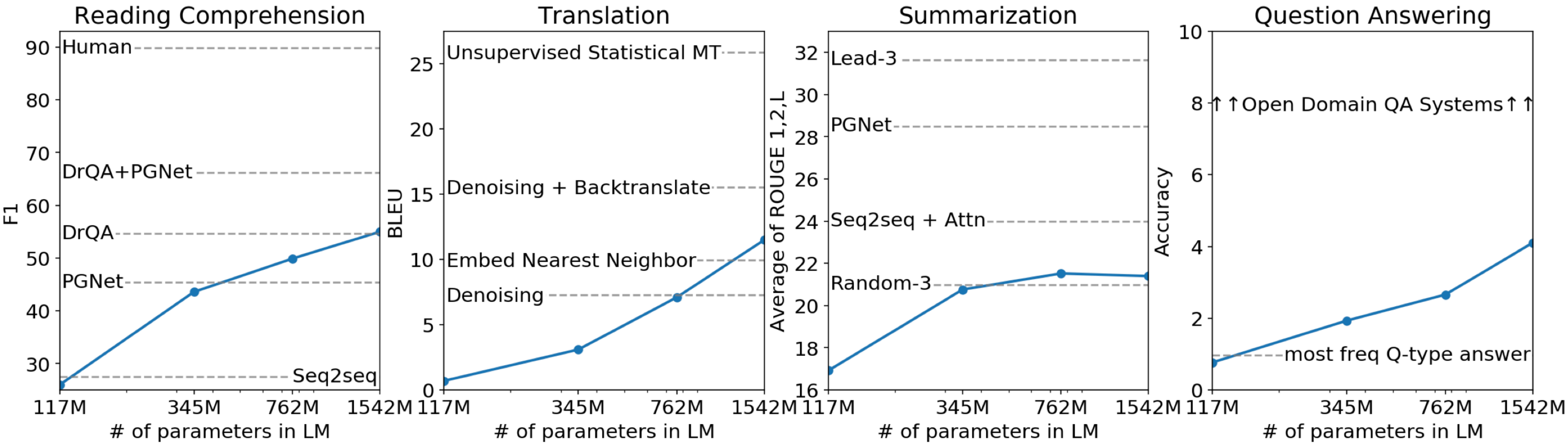
The GPT2 paper

Language Models are Unsupervised Multitask Learners

Alec Radford [*1] **Jeffrey Wu** [*1] **Rewon Child** [1] **David Luan** [1] **Dario Amodei** [**1] **Ilya Sutskever** [**1]

https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

Feb 2019


Reading Comprehension
Human
DrQA+PGNet
DrQA
PGNet
Seq2seq
F1
90
80
70
60
50
40
30
117M
345M
762M
1542M
# of parameters in LM
Translation
Unsupervised Statistical MT
Denoising + Backtranslate
Embed Nearest Neighbor
Denoising
BLEU
25
20
15
10
5
0
117M
345M
762M
1542M
# of parameters in LM
Summarization
Lead-3
PGNet
Seq2seq + Attn
Random-3
Average of ROUGE 1,2,L
32
30
28
26
24
22
20
18
16
117M
345M
762M
1542M
# of parameters in LM
Question Answering
↑↑Open Domain QA Systems↑↑
most freq Q-type answer
Accuracy
10
8
6
4
2
0
117M
345M
762M
1542M
# of parameters in LM

WebText corpus

- Train on web scale corpus but with more reliable data compared to the CommonCrawl.
- English-only, so language detection is used
- Outgoing links from reddit (with at least 3 karma)
- No reddit data was used, instead use the content of the web sites linked on reddit discussions
- 8M documents with 40GB of text

Language detection: https://github.com/CLD2Owners/cld2

News site scraping: https://github.com/codelucas/newspaper

”I’m not the cleverest man in the world, but like they say in French: **Je ne suis pas un imbecile [I’m not a fool].**
In a now-deleted post from Aug. 16, Soheil Eid, Tory candidate in the riding of Joliette, wrote in French: ”**Mentez mentez, il en restera toujours quelque chose**,” which translates as, ”**Lie lie and something will always remain.**”
“I hate the word ‘**perfume**,”’ Burr says. ‘It’s somewhat better in French: ‘**parfum**.’
If listened carefully at 29:55, a conversation can be heard between two guys in French: “**-Comment on fait pour aller de l’autre coté? -Quel autre coté?**”, which means “**- How do you get to the other side? - What side?**”.
If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?
“**Brevet Sans Garantie Du Gouvernement**”, translated to English: “**Patented without government warranty**”.

Table 1. Examples of naturally occurring demonstrations of English to French and French to English translation found throughout the WebText training set.

Parameters	Layers	d_{model}
117M	12	768
345M	24	1024
762M	36	1280
1542M	48	1600

Table 2. Architecture hyperparameters for the 4 model sizes.

Perplexity Results

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

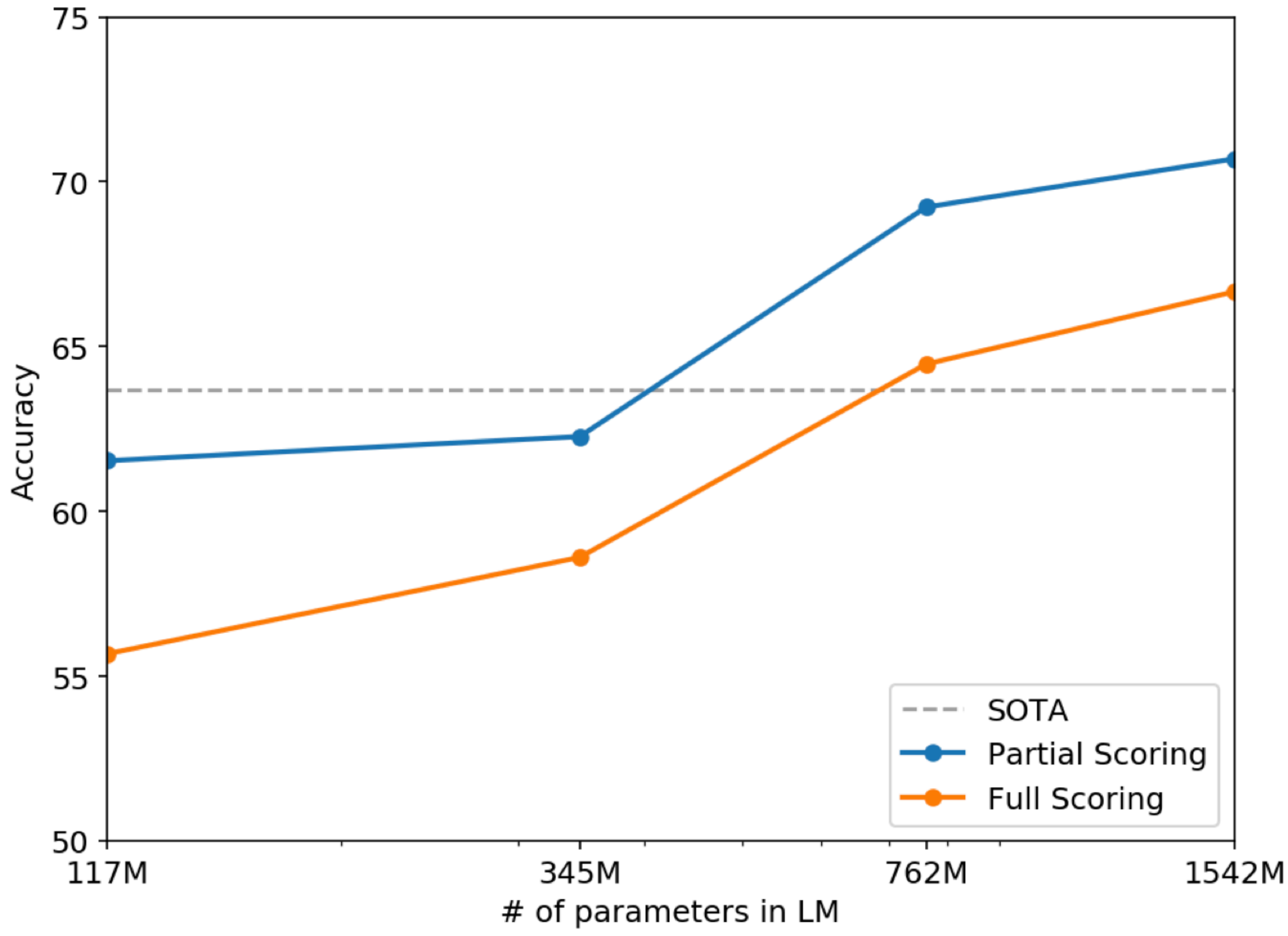


Figure 3. Performance on the Winograd Schema Challenge as a function of model capacity.

Inference Tasks

WNLI, Winograd Schema Challenge

- Reading comprehension task to identify referent of a pronoun using entailment between two sentences (one has pronoun reference explicit)
- Predict 1 (entailment) or 0 (not_entailment)
- Designed to fool simple statistical techniques.
- Test set is imbalanced (65% not entailment) and dev set is adversarial (memorization will hurt performance)

"I stuck a pin through a carrot. When I pulled the pin out, it had a hole."	"The carrot had a hole."	1
"George got free tickets to the play, but he gave them to Eric, because he was particularly eager to see it."	"George was particularly eager to see it."	0

The GPT3 paper

Language Models are Few-Shot Learners

Tom B. Brown* **Benjamin Mann*** **Nick Ryder*** **Melanie Subbiah***

Jared Kaplan† **Prafulla Dhariwal** **Arvind Neelakantan** **Pranav Shyam**

Girish Sastry **Amanda Askell** **Sandhini Agarwal** **Ariel Herbert-Voss**

Gretchen Krueger **Tom Henighan** **Rewon Child** **Aditya Ramesh**

Daniel M. Ziegler **Jeffrey Wu** **Clemens Winter**

Christopher Hesse **Mark Chen** **Eric Sigler** **Mateusz Litwin** **Scott Gray**

Benjamin Chess **Jack Clark** **Christopher Berner**

Sam McCandlish **Alec Radford** **Ilya Sutskever** **Dario Amodei**

https://arxiv.org/abs/2005.14165

NeurIPS 2020, Vancouver, BC

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

`Translate English to French:`	*task description*
`cheese =>`	*prompt*

One-shot

In addition to the task description, the model sees a single example of the task. No gradient updates are performed.

`Translate English to French:`	*task description*
`sea otter => loutre de mer`	*example*
`cheese =>`	*prompt*

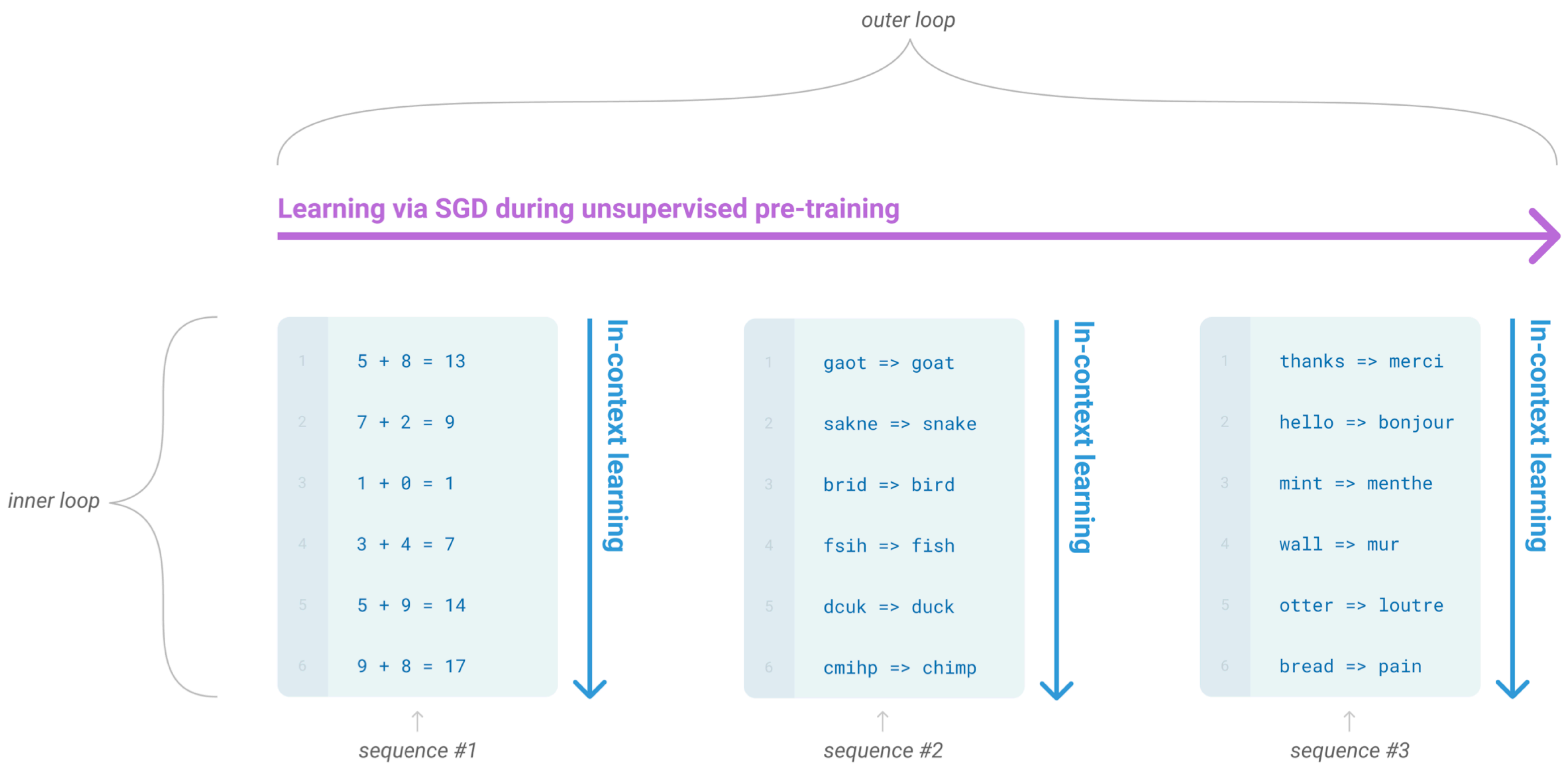
Few-shot

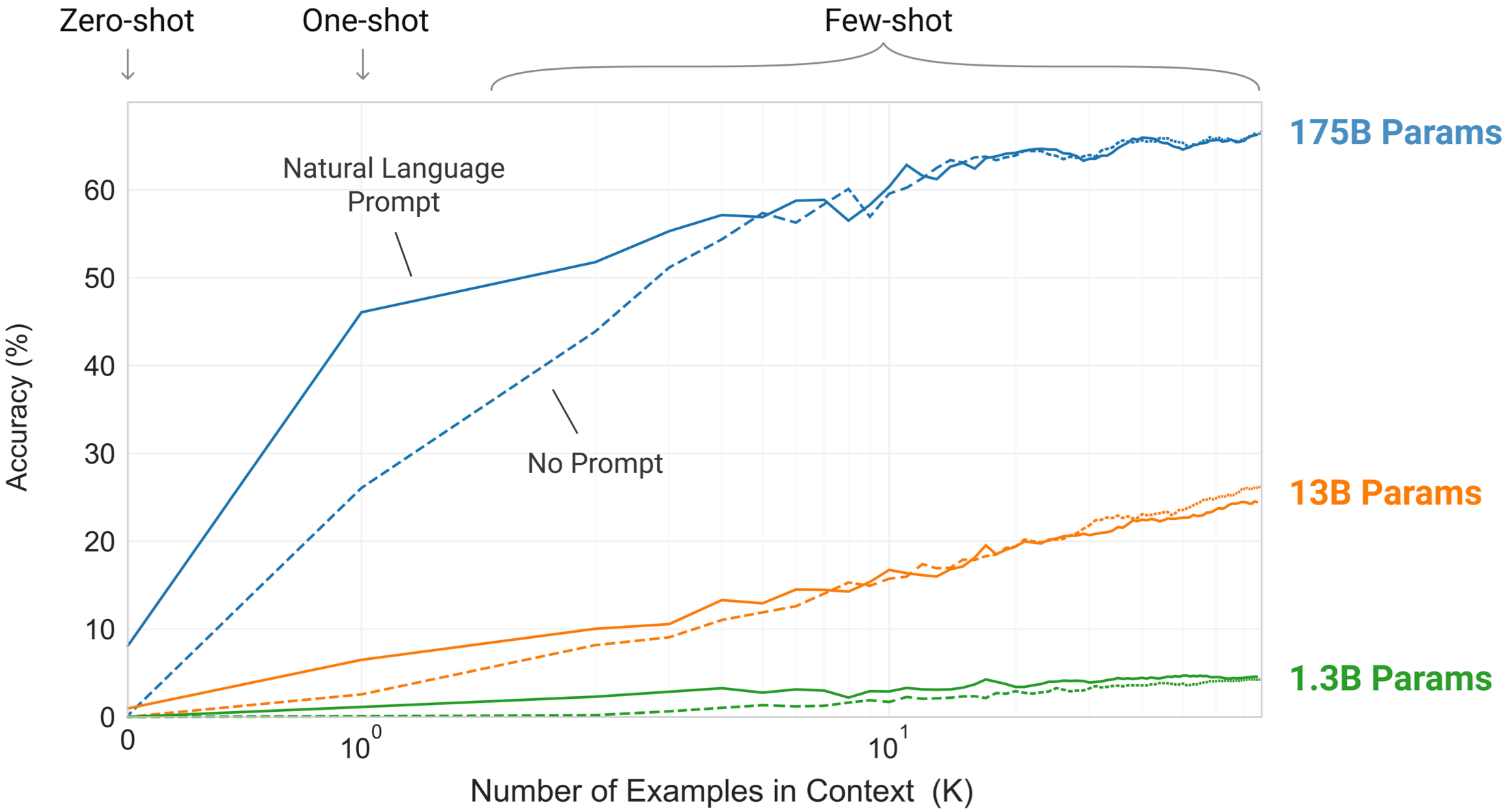
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

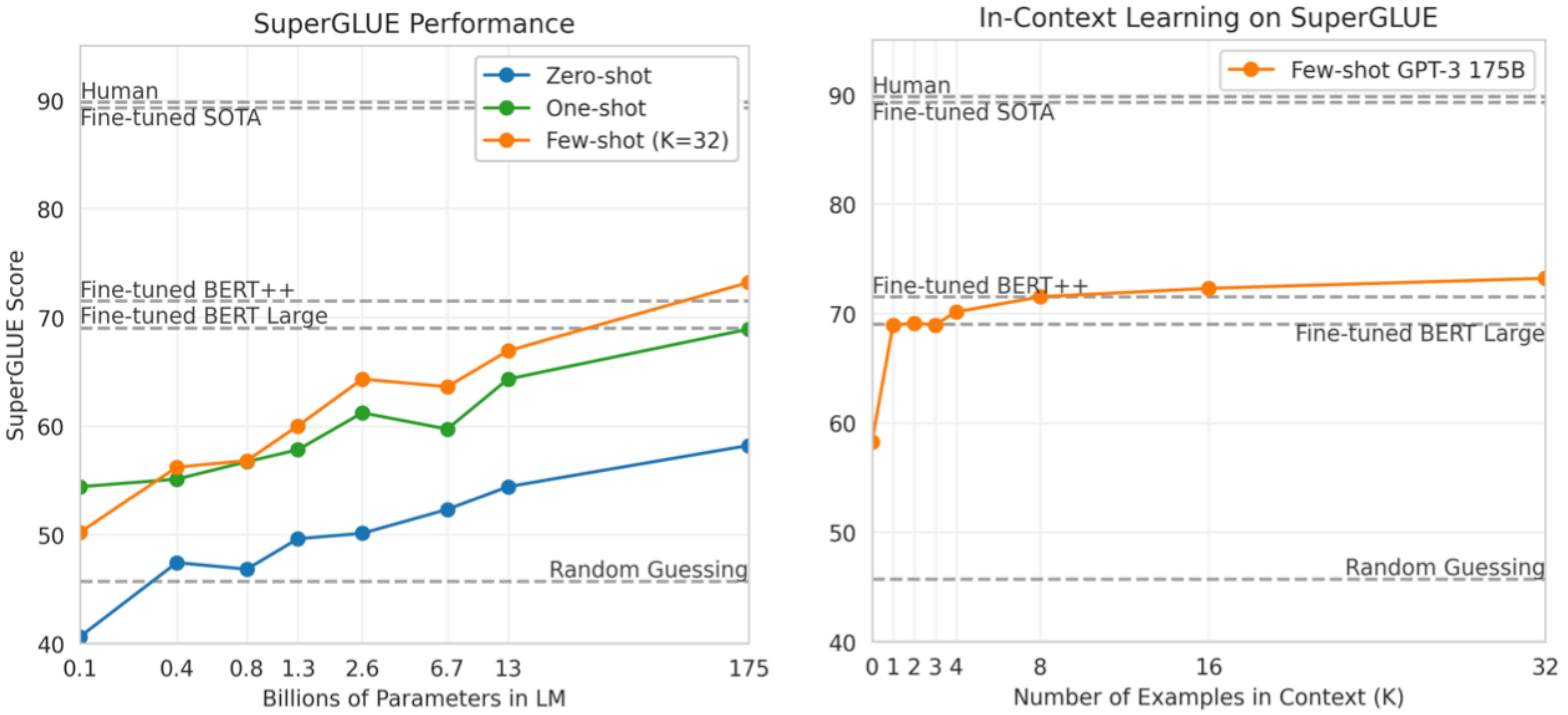
`Translate English to French:`	*task description*
`sea otter => loutre de mer`	*examples*
`peppermint => menthe poivrée`	
`plush girafe => girafe peluche`	
`cheese =>`	*prompt*

Fine-tuning fails at scale

- LLMs >10B parameters are very difficult to fine-tune and requires a big compute budget
- So in-context learning using a long prompt or prefix is needed to coax the answer from a "predict the next token" approach to solving multiple tasks
- Pre-training on web-scale text can observe many different tasks in-context during training in the inner loop (per batch)
- Gradient descent improves the model representations based on next token prediction over many batch updates in the outer loop


outer loop
Learning via SGD during unsupervised pre-training
inner loop
1 5 + 8 = 13
2 7 + 2 = 9
3 1 + 0 = 1
4 3 + 4 = 7
5 5 + 9 = 14
6 9 + 8 = 17
In-context learning
sequence #1
1 gaot => goat
2 sakne => snake
3 brid => bird
4 fsih => fish
5 dcuk => duck
6 cmihp => chimp
In-context learning
sequence #2
1 thanks => merci
2 hello => bonjour
3 mint => menthe
4 wall => mur
5 otter => loutre
6 bread => pain
In-context learning
sequence #3


Zero-shot
One-shot
Few-shot
175B Params
13B Params
1.3B Params
Natural Language
Prompt
No Prompt
Accuracy (%)
0
10
20
30
40
50
60
0
10^0
10^1
Number of Examples in Context (K)



Performance on SuperGLUE increases with number of examples in context. We find the difference in performance between the BERT-Large and BERT++ to be roughly equivalent to the difference between GPT-3 with one example per context versus eight examples per context.

	SuperGLUE Average	BoolQ Accuracy	CB Accuracy	CB F1	COPA Accuracy	RTE Accuracy
Fine-tuned SOTA	**89.0**	**91.0**	**96.9**	**93.9**	**94.8**	**92.5**
Fine-tuned BERT-Large	69.0	77.4	83.6	75.7	70.6	71.7
GPT-3 Few-Shot	71.8	76.4	75.6	52.0	92.0	69.0

	WiC Accuracy	WSC Accuracy	MultiRC Accuracy	MultiRC F1a	ReCoRD Accuracy	ReCoRD F1
Fine-tuned SOTA	**76.1**	**93.8**	**62.3**	**88.2**	**92.5**	**93.3**
Fine-tuned BERT-Large	69.6	64.6	24.1	70.0	71.3	72.0
GPT-3 Few-Shot	49.4	80.1	30.5	75.4	90.2	91.1

Table 3.5: Performance of GPT-3 on SuperGLUE compared to fine-tuned baselines and SOTA. All results are reported on the test set. GPT-3 few-shot is given a total of 32 examples within the context of each task and performs no gradient updates.

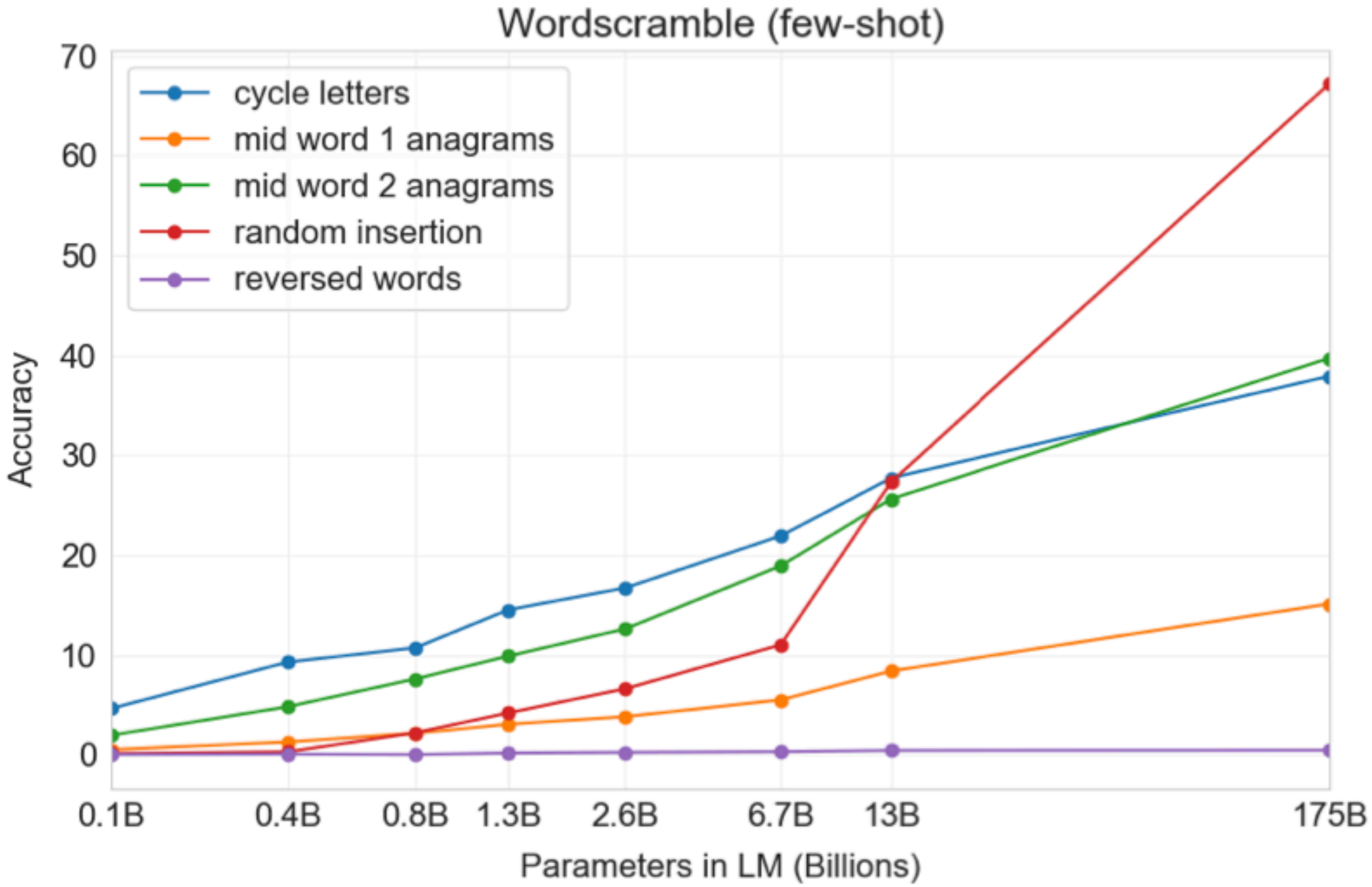
Setting	LAMBADA (acc)	LAMBADA (ppl)	StoryCloze (acc)	HellaSwag (acc)
SOTA	68.0[a]	8.63[b]	**91.8**[c]	**85.6**[d]
GPT-3 Zero-Shot	**76.2**	**3.00**	83.2	78.9
GPT-3 One-Shot	**72.5**	**3.35**	84.7	78.1
GPT-3 Few-Shot	**86.4**	**1.92**	87.7	79.3

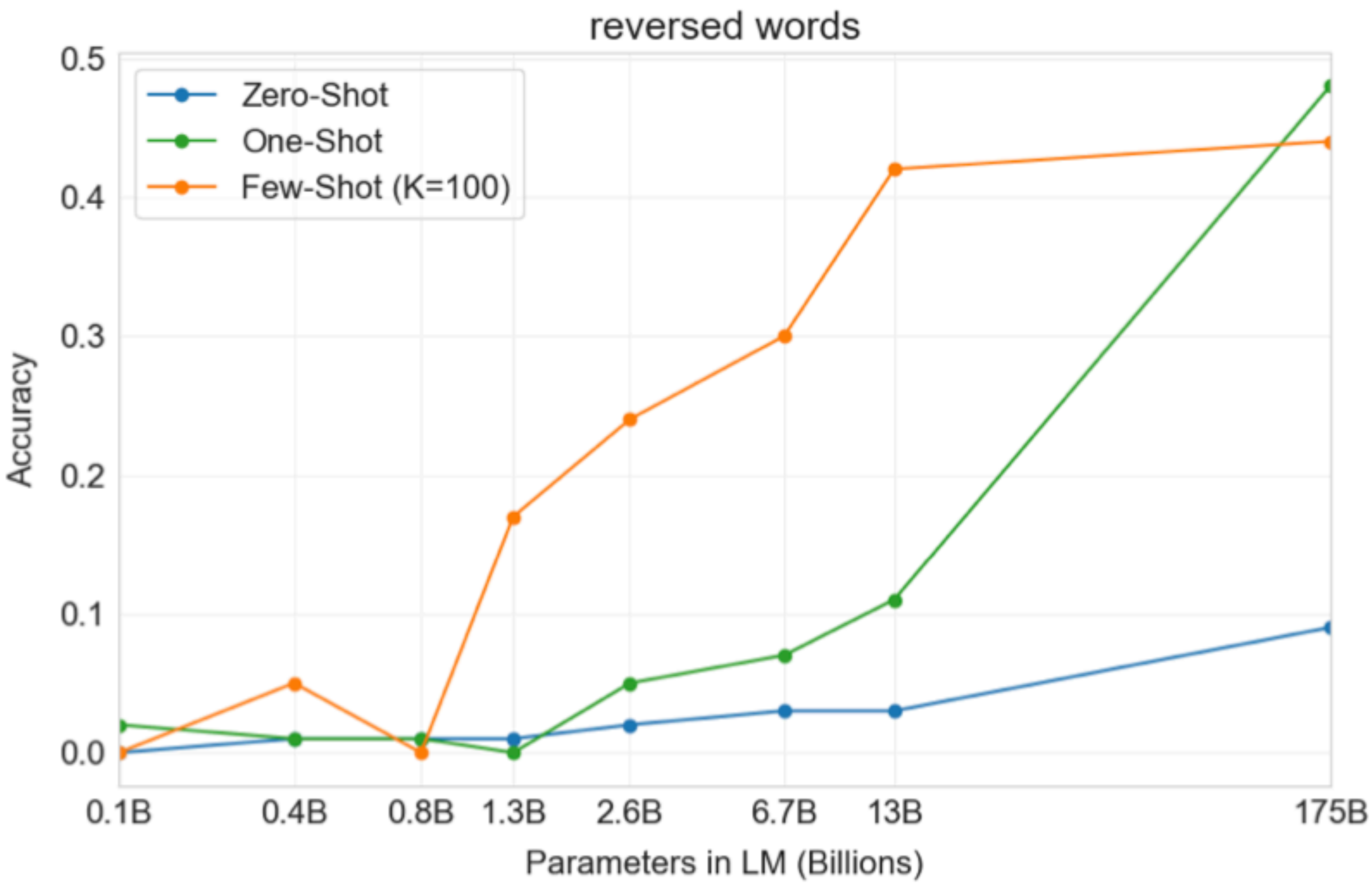
Setting	NaturalQS	WebQS	TriviaQA
RAG (Fine-tuned, Open-Domain) [LPP+20]	**44.5**	**45.5**	**68.0**
T5-11B+SSM (Fine-tuned, Closed-Book) [RRS20]	36.6	44.7	60.5
T5-11B (Fine-tuned, Closed-Book)	34.5	37.4	50.1
GPT-3 Zero-Shot	14.6	14.4	64.3
GPT-3 One-Shot	23.0	25.3	**68.0**
GPT-3 Few-Shot	29.9	41.5	**71.2**

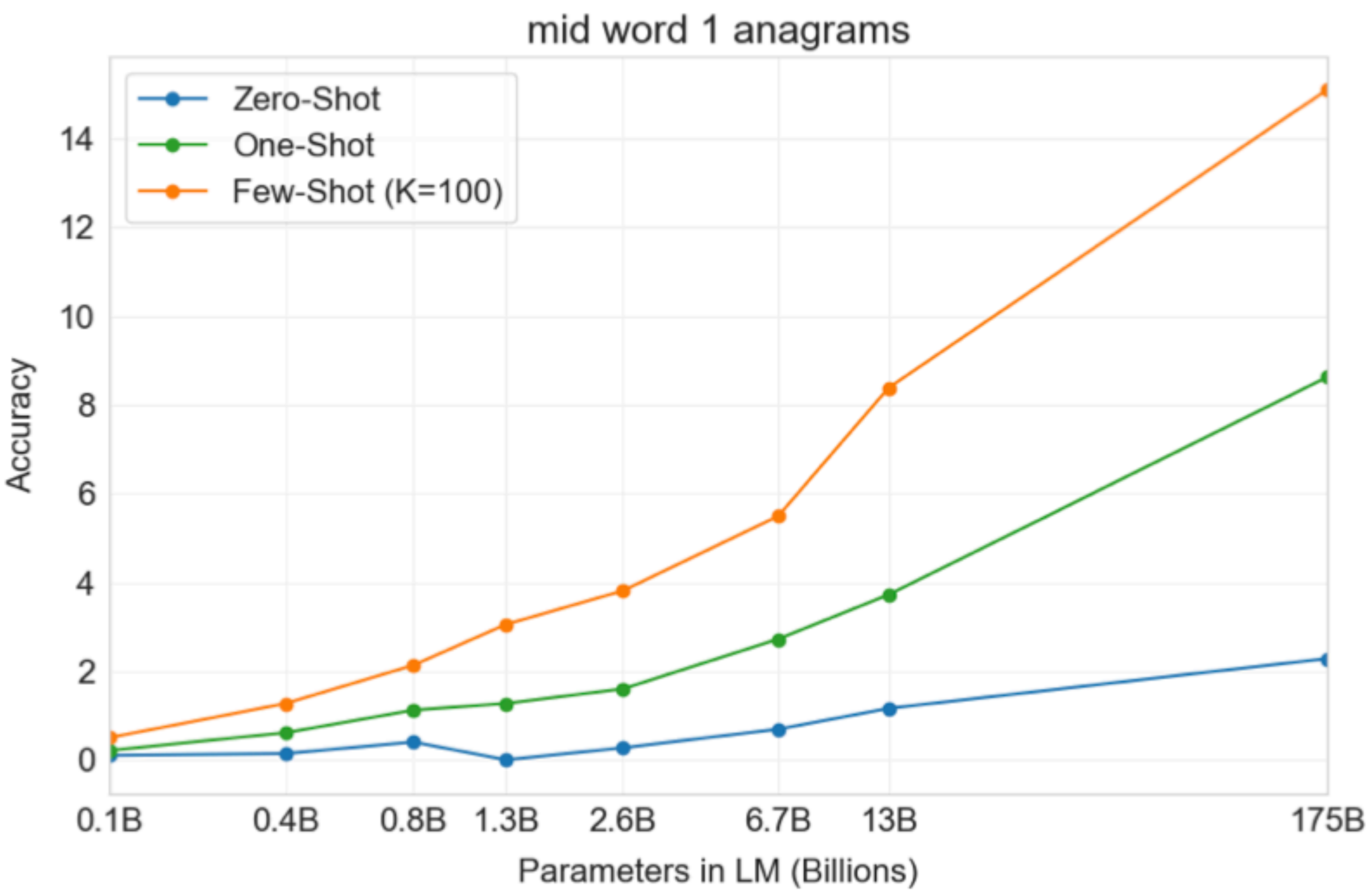
Setting	ARC (Easy)	ARC (Challenge)	CoQA	DROP
Fine-tuned SOTA	**92.0**[a]	**78.5**[b]	**90.7**[c]	**89.1**[d]
GPT-3 Zero-Shot	68.8	51.4	81.5	23.6
GPT-3 One-Shot	71.2	53.2	84.0	34.3
GPT-3 Few-Shot	70.1	51.5	85.0	36.5

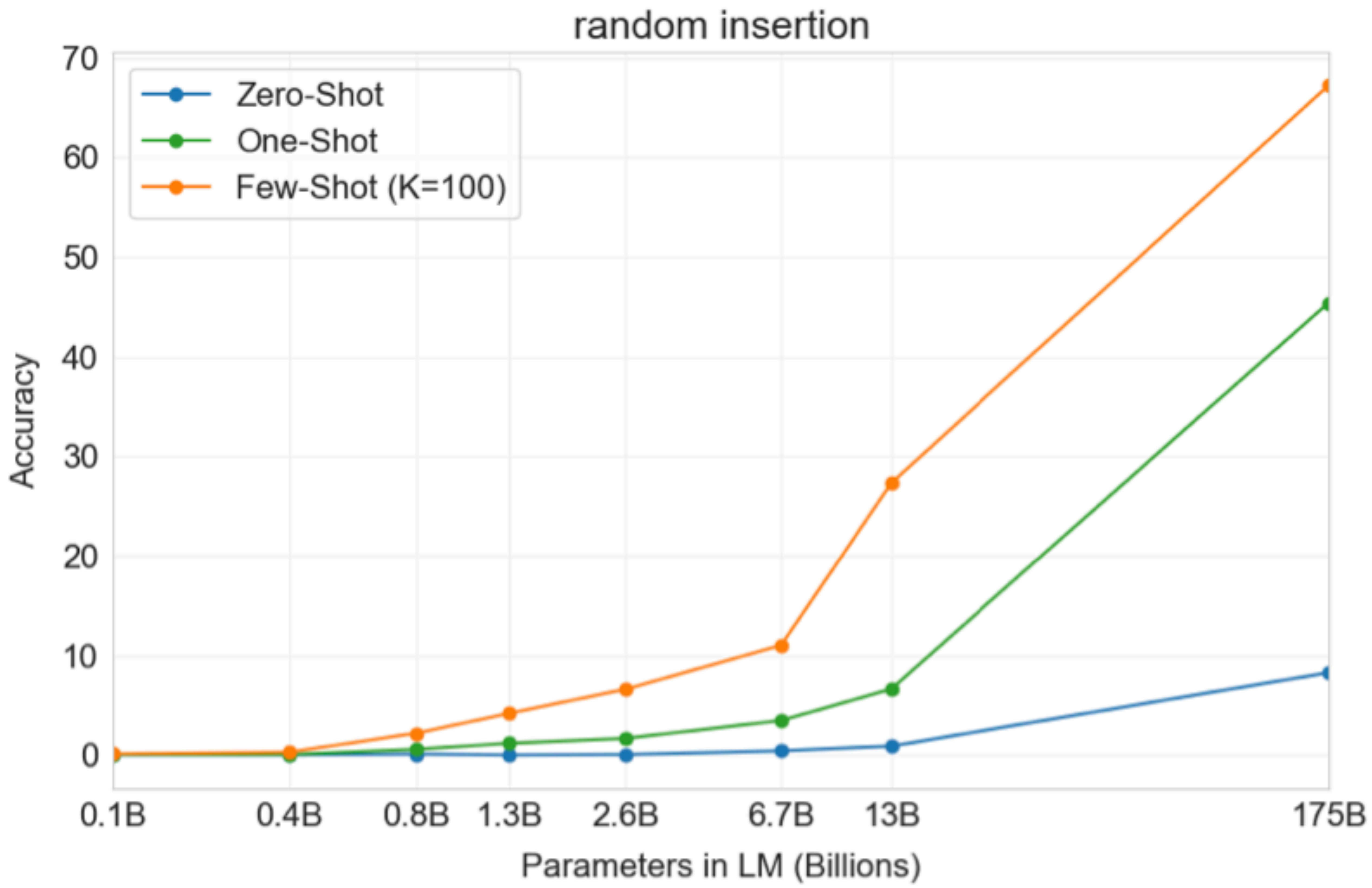
WMT 2014

Setting	En→Fr	Fr→En	En→De	De→En	En→Ro	Ro→En
SOTA (Supervised)	**45.6**[a]	35.0 [b]	**41.2**[c]	40.2[d]	**38.5**[e]	**39.9**[e]
XLM [LC19]	33.4	33.3	26.4	34.3	33.3	31.8
MASS [STQ+19]	37.5	34.9	28.3	35.2	35.2	33.1
mBART [LGG+20]	-	-	29.8	34.0	35.0	30.5
GPT-3 Zero-Shot	25.2	21.2	24.6	27.2	14.1	19.9
GPT-3 One-Shot	28.3	33.7	26.2	30.4	20.6	38.6
GPT-3 Few-Shot	32.6	39.2	29.7	40.6	21.0	39.5


Wordscramble (few-shot)
cycle letters
mid word 1 anagrams
mid word 2 anagrams
random insertion
reversed words
Accuracy
0
10
20
30
40
50
60
70
0.1B
0.4B
0.8B
1.3B
2.6B
6.7B
13B
175B
Parameters in LM (Billions)


reversed words
Zero-Shot
One-Shot
Few-Shot (K=100)
Accuracy
0.0
0.1
0.2
0.3
0.4
0.5
0.1B
0.4B
0.8B
1.3B
2.6B
6.7B
13B
175B
Parameters in LM (Billions)

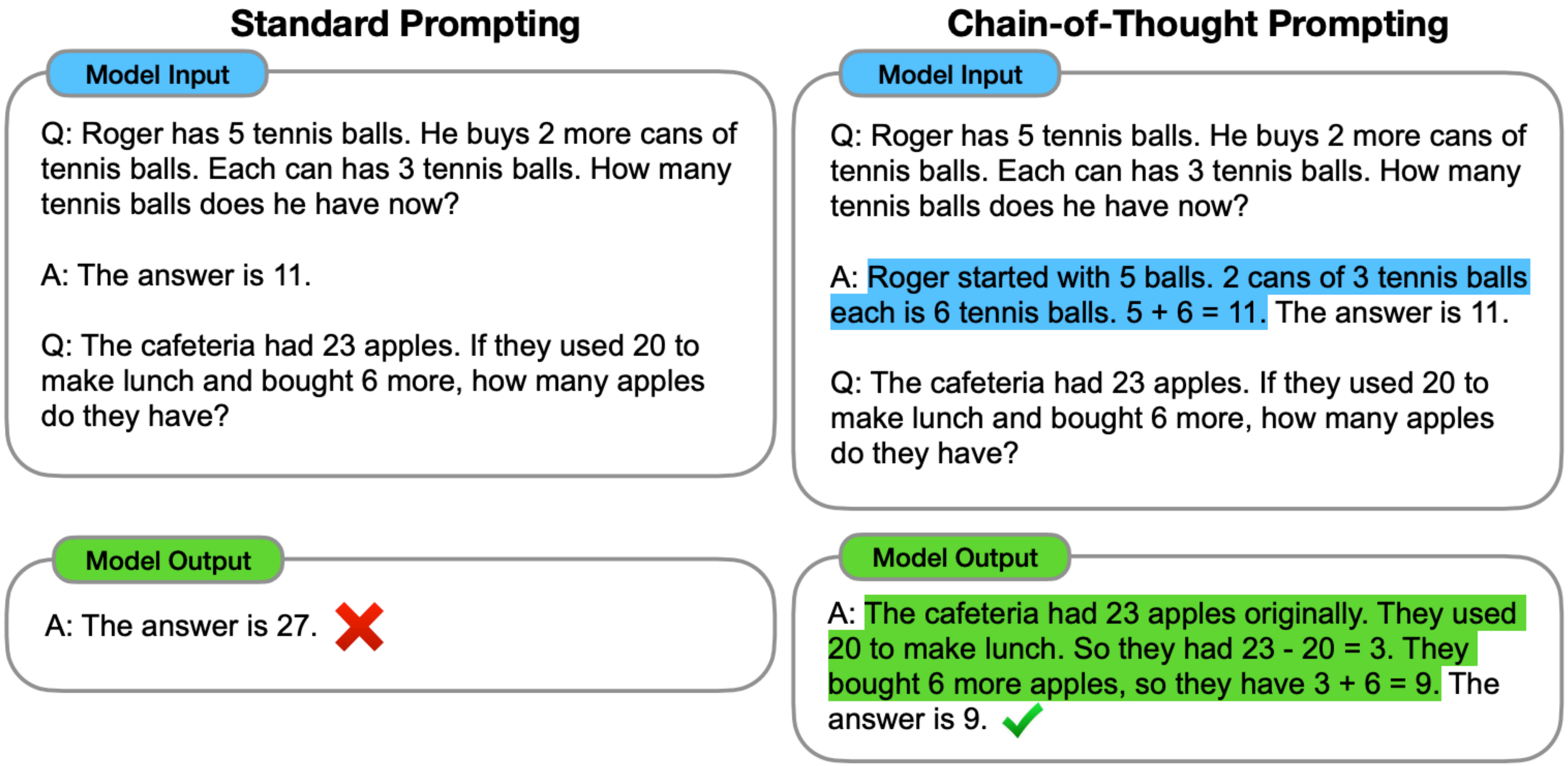

mid word 1 anagrams
Zero-Shot
One-Shot
Few-Shot (K=100)
Accuracy
0
2
4
6
8
10
12
14
0.1B
0.4B
0.8B
1.3B
2.6B
6.7B
13B
175B
Parameters in LM (Billions)


random insertion
Zero-Shot
One-Shot
Few-Shot (K=100)
Accuracy
0
10
20
30
40
50
60
70
0.1B
0.4B
0.8B
1.3B
2.6B
6.7B
13B
175B
Parameters in LM (Billions)

Chain of thought prompting

Chain-of-Thought Prompting Elicits Reasoning in Large Language Models [Wei et al, 2022]

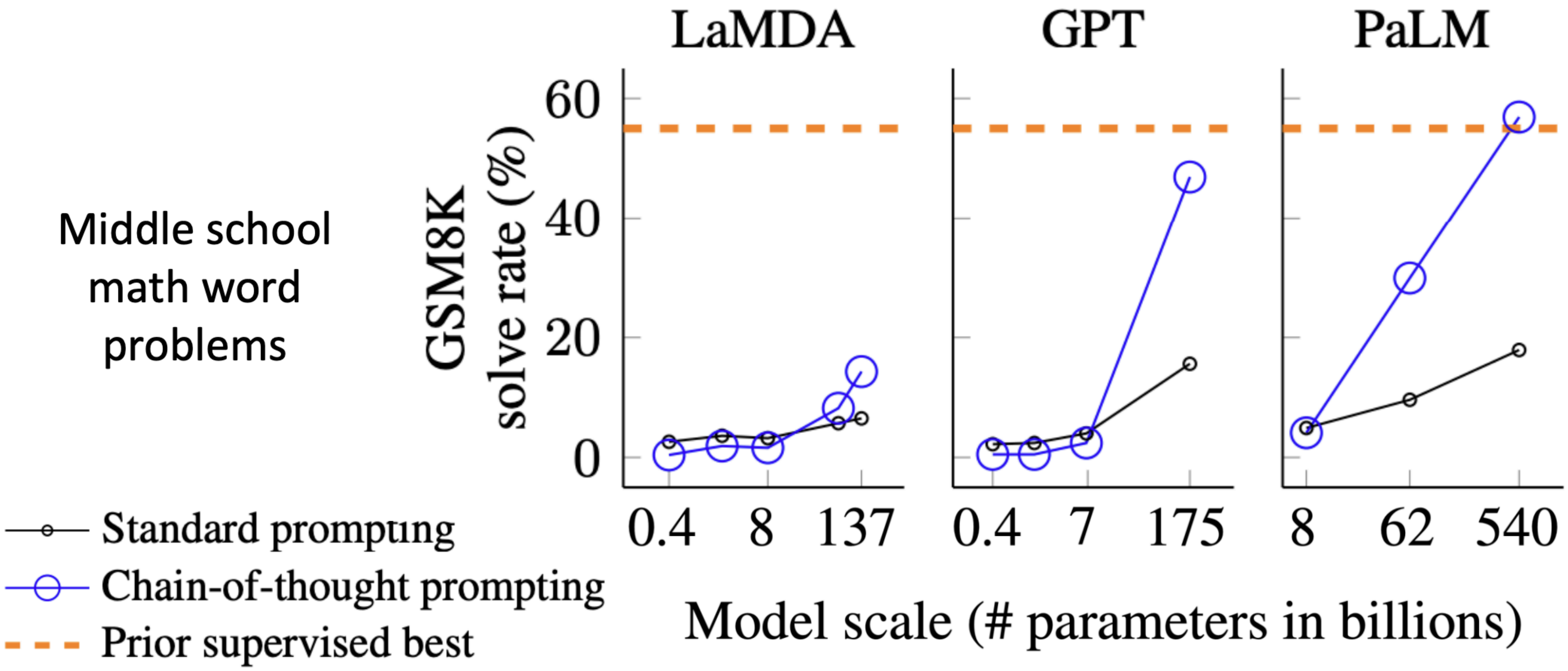
- How you prompt matters.
- For more complex problems, may need to provide prompts that illustrate the reasoning you expect



Chain of thought prompting

Chain-of-Thought Prompting Elicits Reasoning in Large Language Models [Wei et al, 2022]

Middle school math word problems

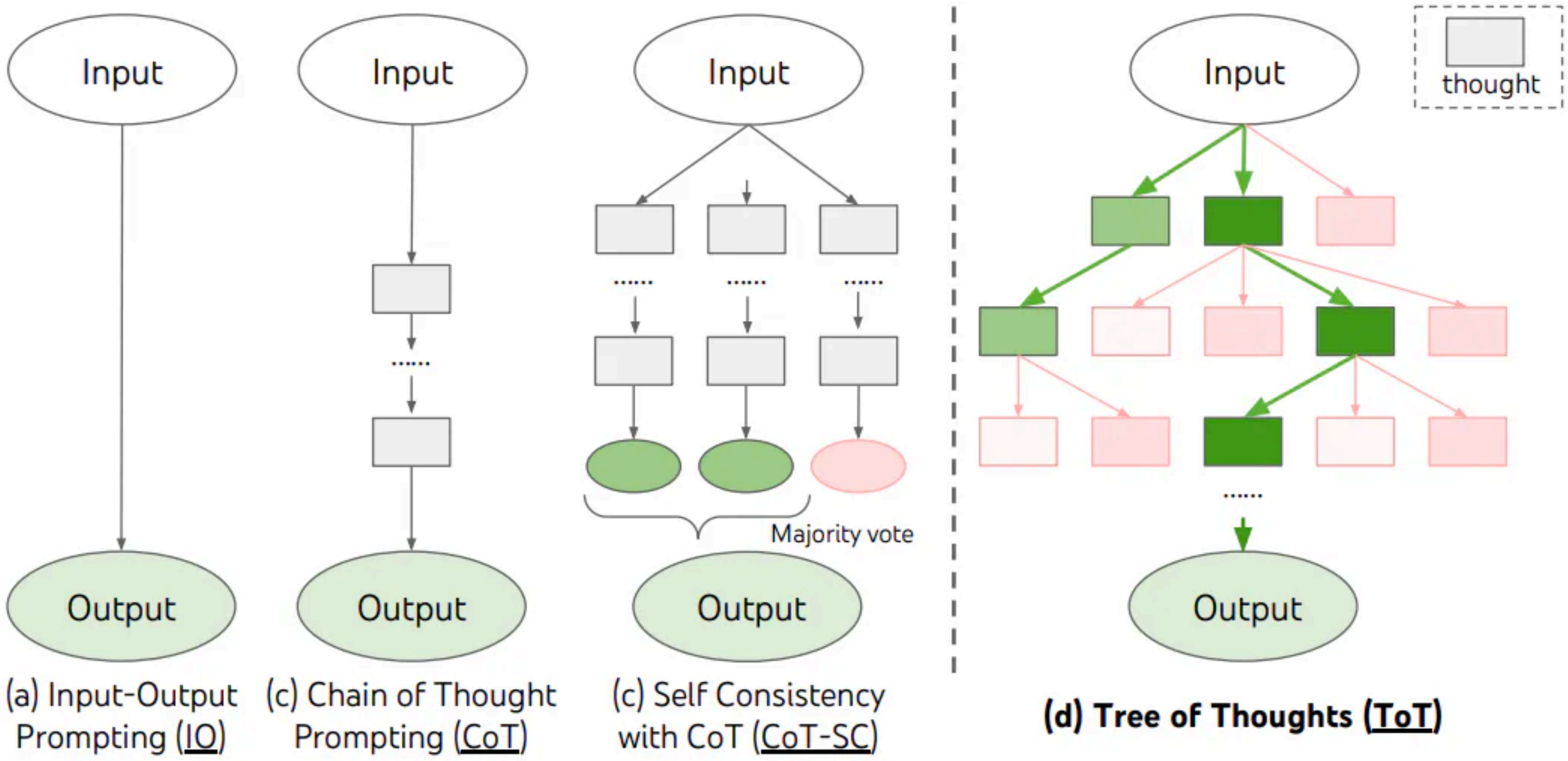


Slide Credit: Stanford CS224n, Jesse Mu

Tree of thought prompting

Tree of Thoughts: Deliberate Problem Solving with Large Language Models [Yao et al, 2023]
Large Language Model Guided Tree-of-Thought [Long 2023]

- Decision making by considering multiple paths of reasoning
- Consider different "thoughts" expressed in language
- Use to solve different types of problems



Tree of thought prompting

Tree of Thoughts: Deliberate Problem Solving with Large Language Models [Yao et al, 2023]
Large Language Model Guided Tree-of-Thought [Long 2023]

- Problem solving by considering multiple reasoning path over "thoughts"
- Decompose problem solving process into "thought" steps and states (partial solution)

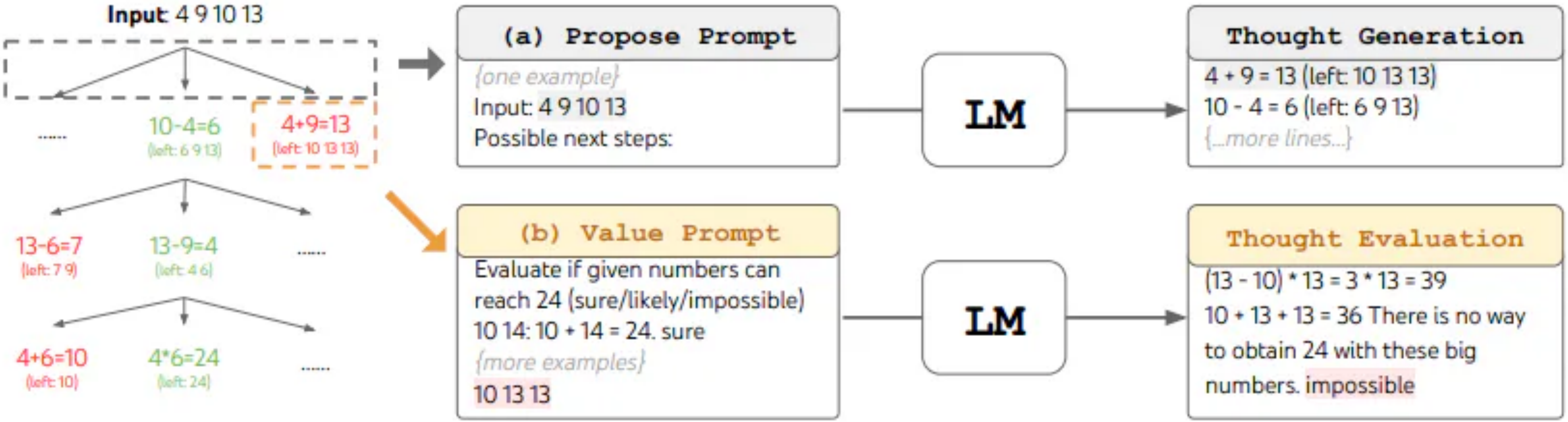
	Game of 24	Creative Writing	5x5 Crosswords
Input	4 numbers (4 9 10 13)	4 random sentences	10 clues (h1. presented;..)
Output	An equation to reach 24 (13-9)*(10-4)=24	A passage of 4 paragraphs ending in the 4 sentences	5x5 letters: SHOWN; WIRRA; AVAIL; ...
Thoughts	3 intermediate equations (13-9=4 (left 4,4,10); 10-4=6 (left 4,6); 4*6=24)	A short writing plan (1. Introduce a book that connects...)	Words to fill in for clues: (h1. shown; v5. naled; ...)
#ToT steps	3	1	5-10 (variable)

Table 1: Task overview. Input, output, thought examples are in blue.

Tree of thought prompting

Tree of Thoughts: Deliberate Problem Solving with Large Language Models [Yao et al, 2023]

Example: want to get to number 24 from four input numbers



Use LLM to propose next steps and evaluate how good the state is

Tree of thought prompting

Tree of Thoughts: Deliberate Problem Solving with Large Language Models [Yao et al, 2023]

Method	Success
IO prompt	7.3%
CoT prompt	4.0%
CoT-SC (k=100)	9.0%
ToT (ours) (b=1)	45%
ToT (ours) (b=5)	**74%**
IO + Refine (k=10)	27%
IO (best of 100)	33%
CoT (best of 100)	49%

Table 2: Game of 24 Results.

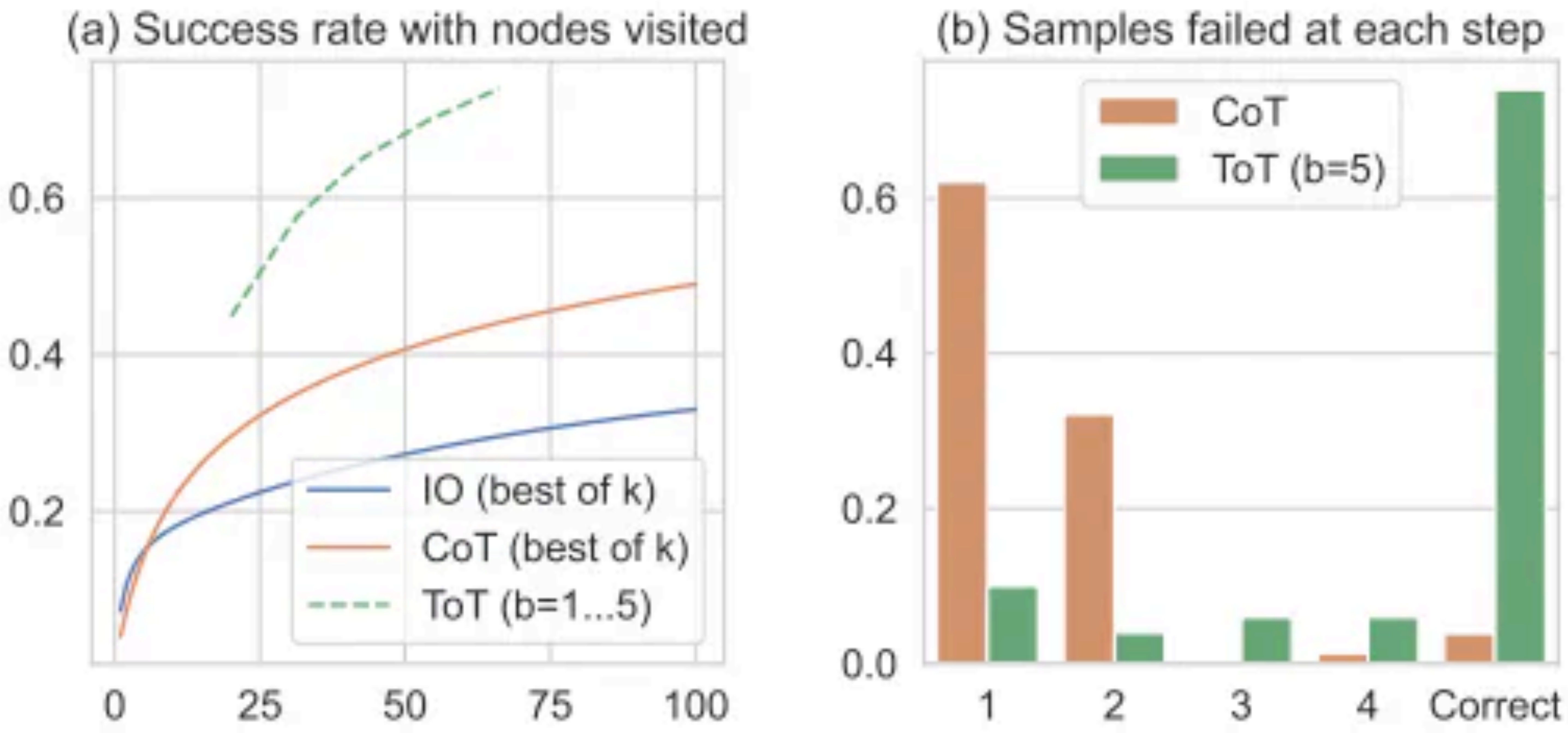


Figure 3: Game of 24 (a) scale analysis & (b) error analysis.

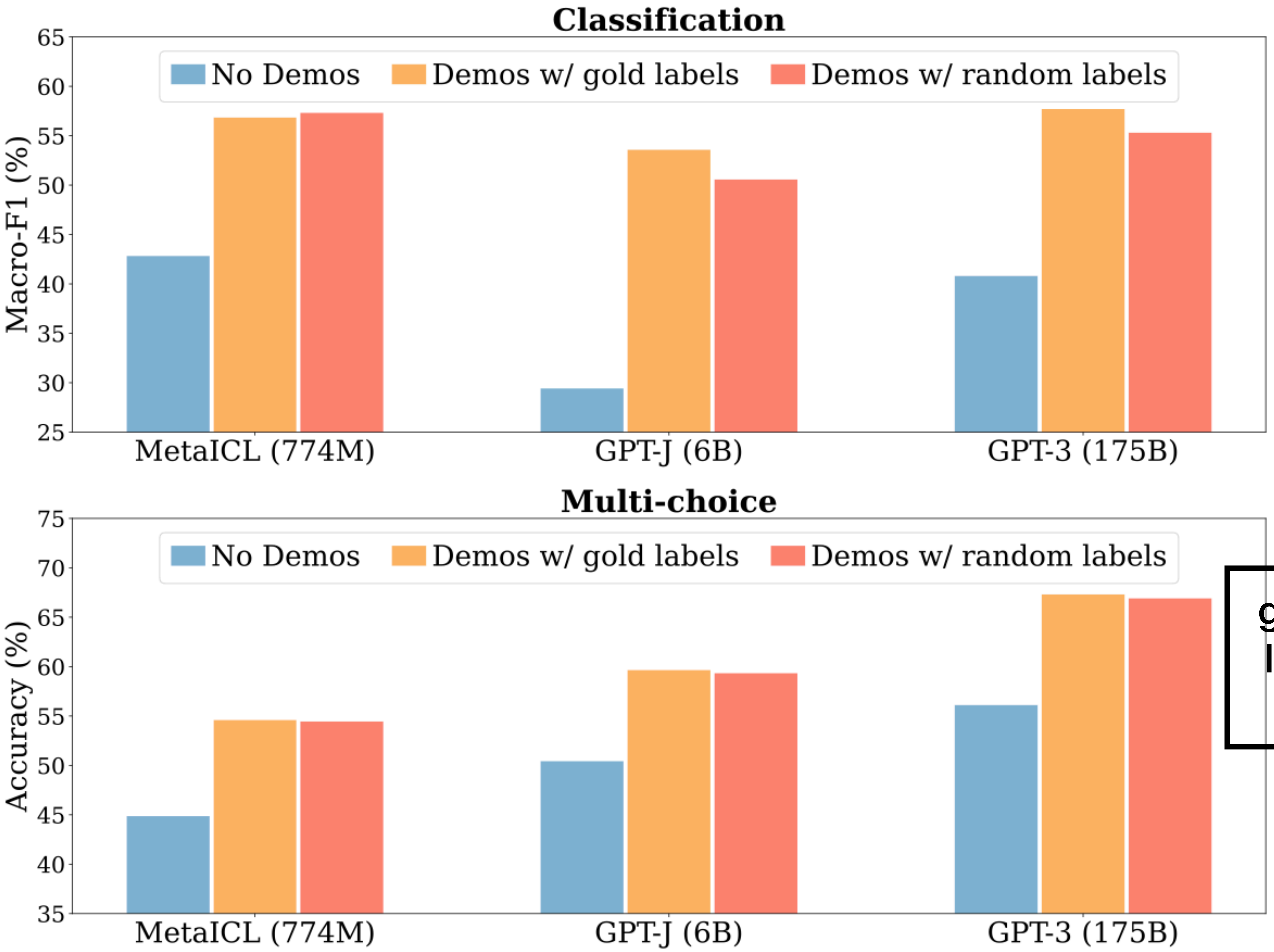
Different types of prompting: https://www.promptingguide.ai/

Why does in-context learning work?

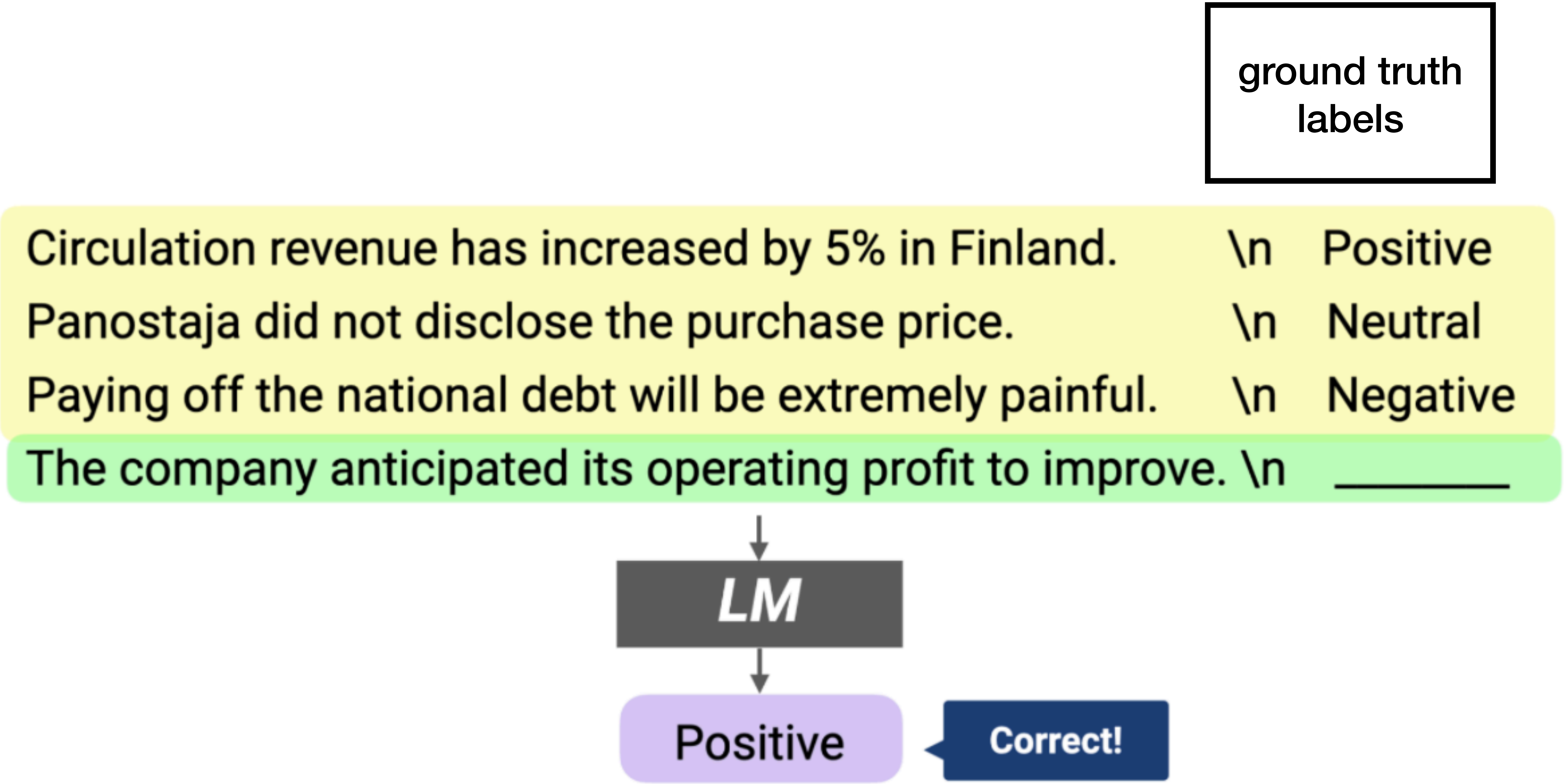
Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?

Sewon Min[1,2] **Xinxi Lyu**[1] **Ari Holtzman**[1] **Mikel Artetxe**[2]
Mike Lewis[2] **Hannaneh Hajishirzi**[1,3] **Luke Zettlemoyer**[1,2]
[1]University of Washington [2]Meta AI [3]Allen Institute for AI

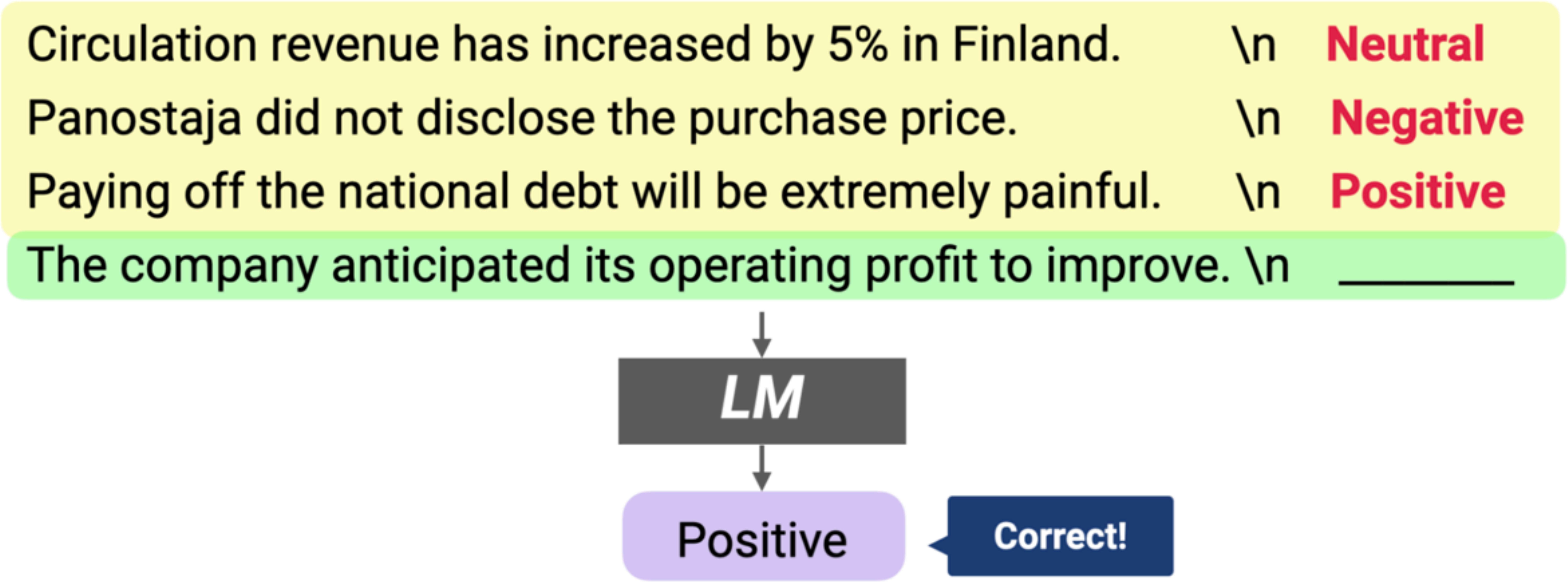
https://arxiv.org/abs/2202.12837



ground truth labels don't matter!


ground truth labels
Circulation revenue has increased by 5% in Finland. \n Positive
Panostaja did not disclose the purchase price. \n Neutral
Paying off the national debt will be extremely painful. \n Negative
The company anticipated its operating profit to improve. \n ______
LM
Positive
Correct!

replace true labels with random labels



Why does in-context learning work?

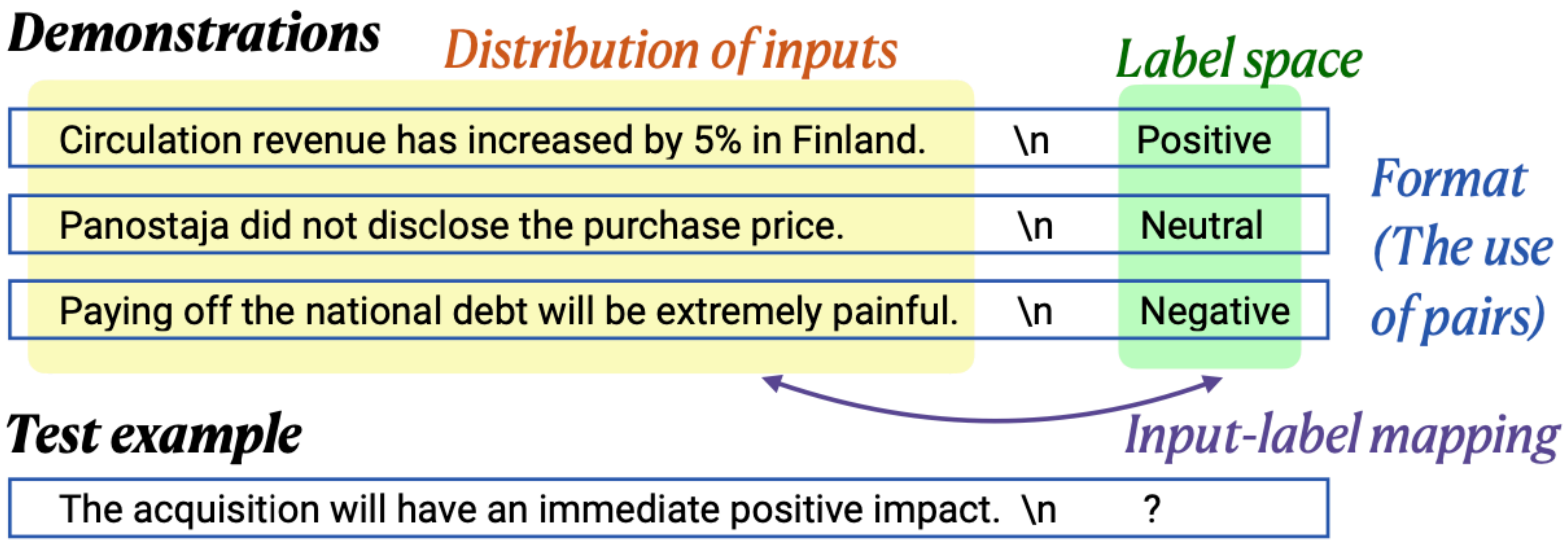
Four hypotheses

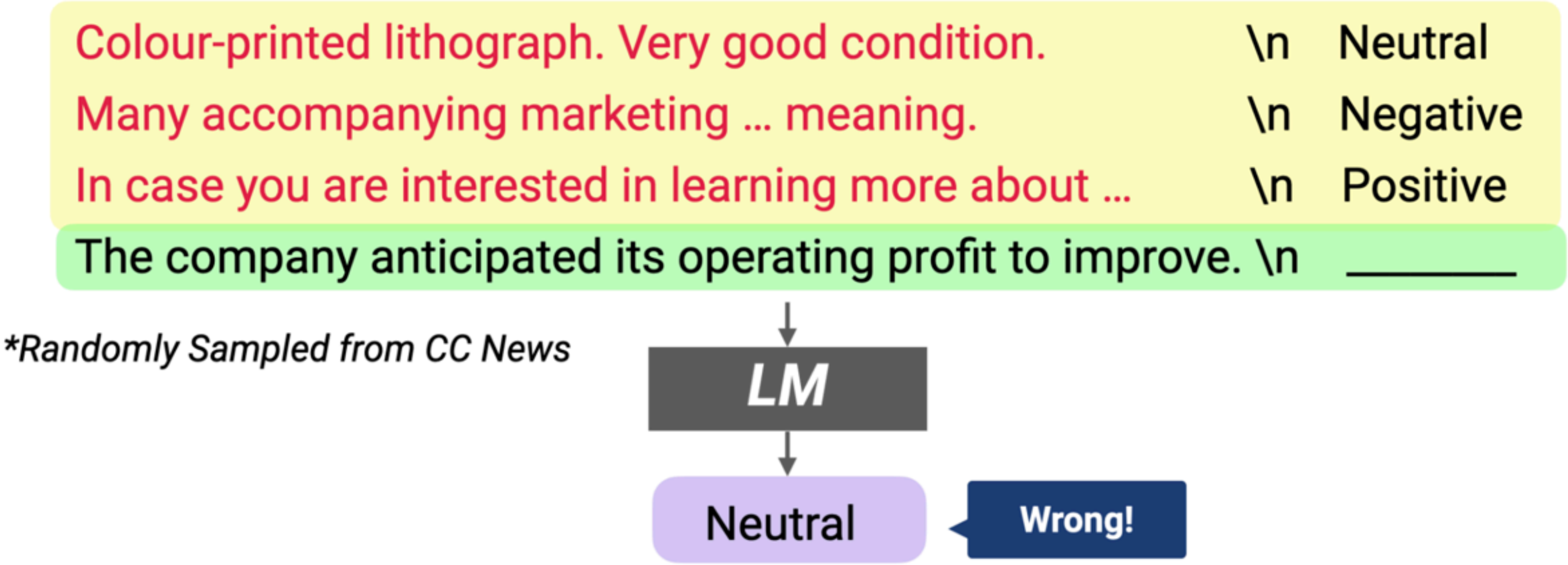
1. The **input-label mapping**, whether each input x_i is paired with the correct label y_i (not true)

2. The **distribution** that the input $x_1, \ldots, x_k$ are from (is it from a sports article, or business news?)

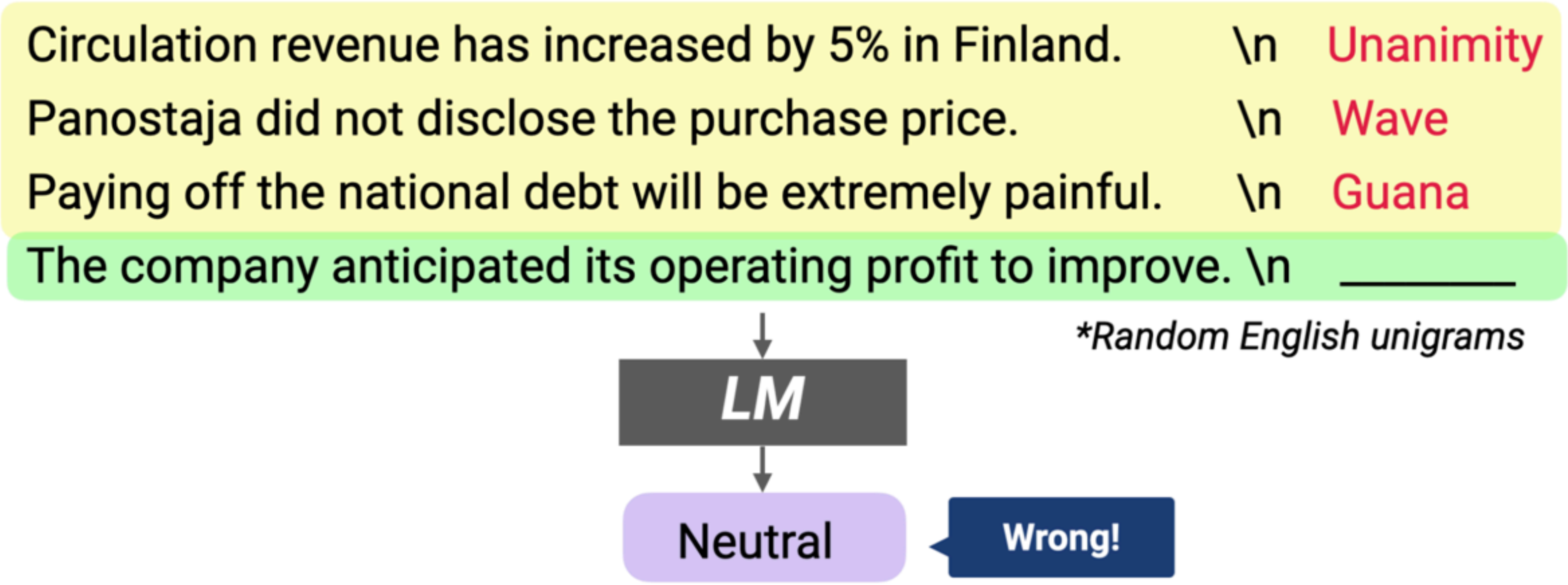
3. The **output label space** $y_1, \ldots, y_k$

4. The **format of the demonstration**, e.g. `x // y`; `Input: x Output: y`; etc.

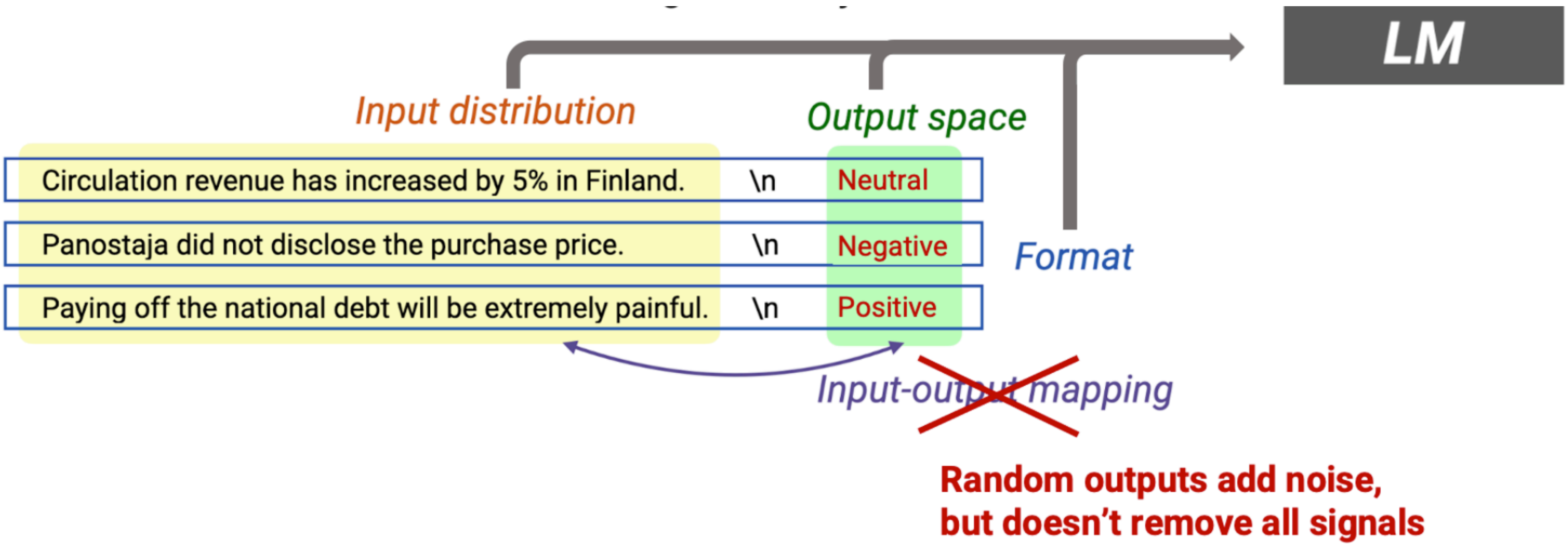

Demonstrations
Distribution of inputs
Label space
Circulation revenue has increased by 5% in Finland. \n Positive
Panostaja did not disclose the purchase price. \n Neutral
Paying off the national debt will be extremely painful. \n Negative
Format (The use of pairs)
Test example
Input-label mapping
The acquisition will have an immediate positive impact. \n ?

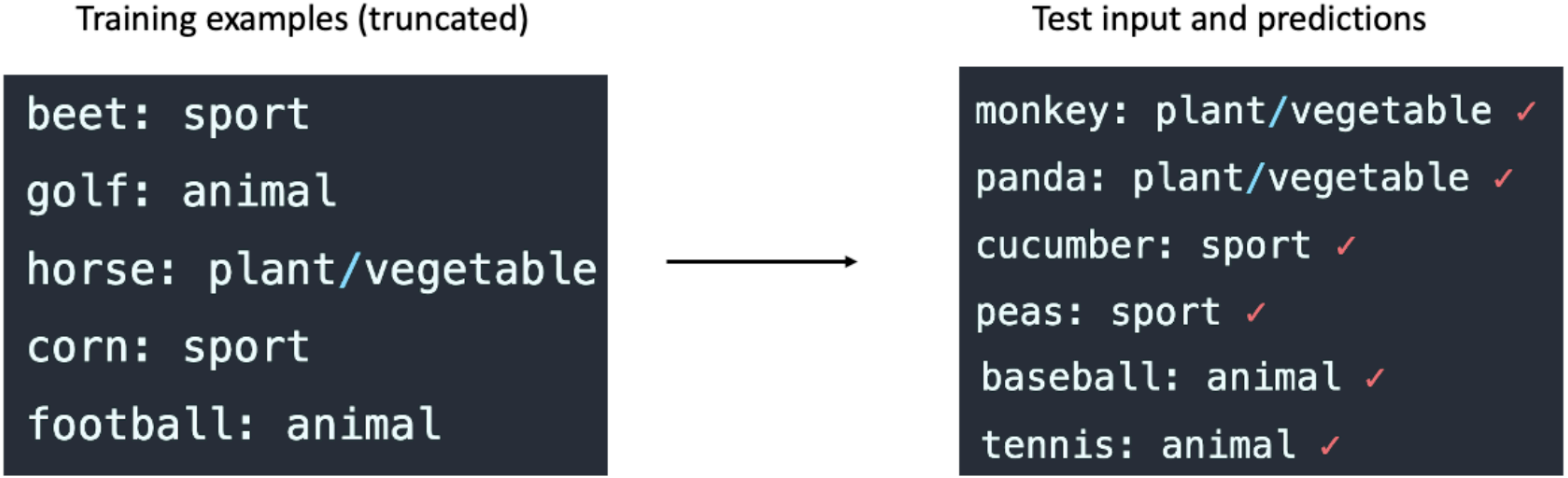


The **input distribution matters**: using inputs from an out of domain corpus causes a large performance drop



The **output distribution matters**: using labels that are random English unigrams causes a large performance drop


LM
Input distribution
Output space
Circulation revenue has increased by 5% in Finland.
\n
Neutral
Panostaja did not disclose the purchase price.
\n
Negative
Paying off the national debt will be extremely painful.
\n
Positive
Format
Input-output mapping
Random outputs add noise,
but doesn't remove all signals



An example synthetic task with unusual semantics that GPT-3 can successfully learn. A modified figure from Rong.

In-Context Learning Learns Label Relationships but Is Not Conventional Learning

Jannik Kossen[1 ∇] **Yarin Gal**[1 ∆] **Tom Rainforth**[2 ∆]

[1] OATML, Department of Computer Science, University of Oxford
[2] Department of Statistics, University of Oxford

In-Context Learning (ICL)

- How does the **conditional label distribution** of ICL examples affect **accuracy**?
- ICL does incorporate in-context label information and can even learn truly novel tasks in-context.
- Analogies between ICL and conventional learning algorithms fall short in a variety of ways
  - Label relationships inferred from pre-training have a lasting effect that cannot be surmounted by in-context observations
  - Additional prompting can improve but likely not overcome this deficiency
  - ICL does not treat all information provided in-context equally and preferentially makes use of label information that appears closer to the query

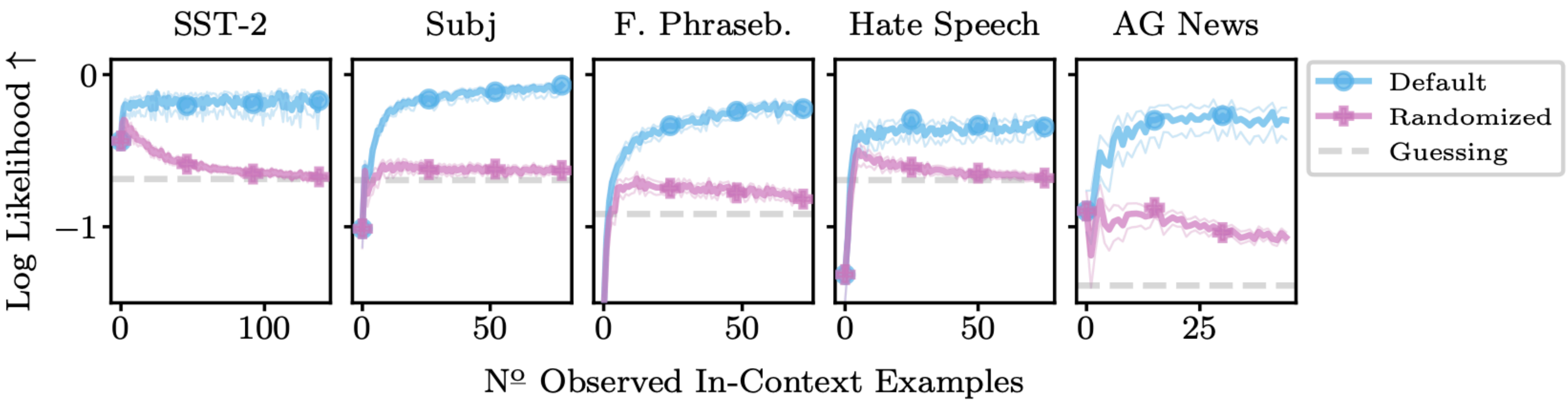


Figure 1: ICL predictions generally depend on the conditional label distribution of in-context examples: when in-context labels are randomized, average log likelihoods of label predictions decrease compared to ICL with default labels for LLaMa-2-70B across a variety of tasks. Results averaged over 500 in-context datasets and thin lines are 99 % confidence intervals. See §5 for details.

Table 1: Average differences between ICL log likelihoods for default and randomized labels. Bold entries indicate differences are statistically significant. We can disregard lightgray entries: for them, default ICL performance is not significantly better than a random guessing baseline. Whenever default ICL outperforms the baseline, ICL almost always performs significantly worse (positive differences) for random labels. Averages over 500 runs at max. context size, standard errors in Table F.1.

Δ Log Likelihood	SST-2	Subj	FP	HS	AGN	MQP	MRPC	RTE	WNLI
LLaMa-2 7B	**0.42**	**0.39**	**0.57**	**0.18**	**0.53**	0.03	0.02	**0.03**	0.02
LLaMa-2 13B	**0.41**	**0.62**	**0.49**	**0.24**	**0.81**	**0.04**	0.01	**0.06**	0.02
LLaMa-2 70B	**0.51**	**0.53**	**0.57**	**0.34**	**0.80**	**0.29**	**0.04**	**0.22**	**0.18**
Falcon 7B	**0.20**	**0.19**	**0.25**	**0.06**	**0.31**	0.01	0.01	−0.01	0.01
Falcon 7B Instr.	**0.13**	**0.08**	**0.11**	**0.03**	**0.15**	0.03	0.02	−0.00	0.00
Falcon 40B	**0.34**	**0.35**	**0.31**	**0.18**	**0.90**	**0.06**	0.01	0.01	0.02
Falcon 40B Instr.	**0.25**	**0.37**	**0.27**	0.02	**0.77**	**0.06**	0.02	0.02	**0.04**

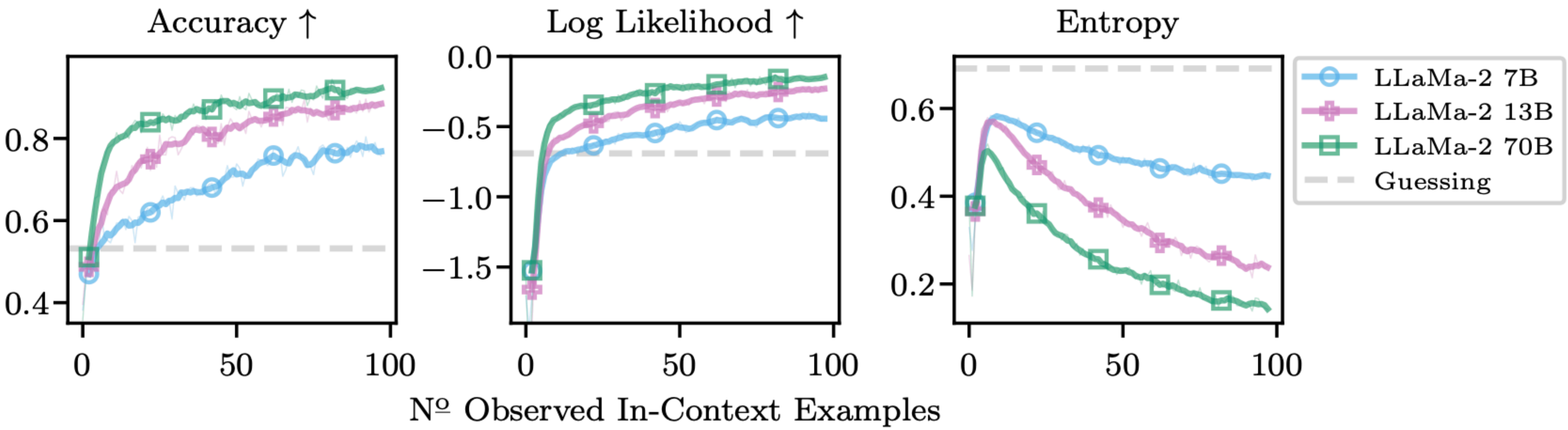
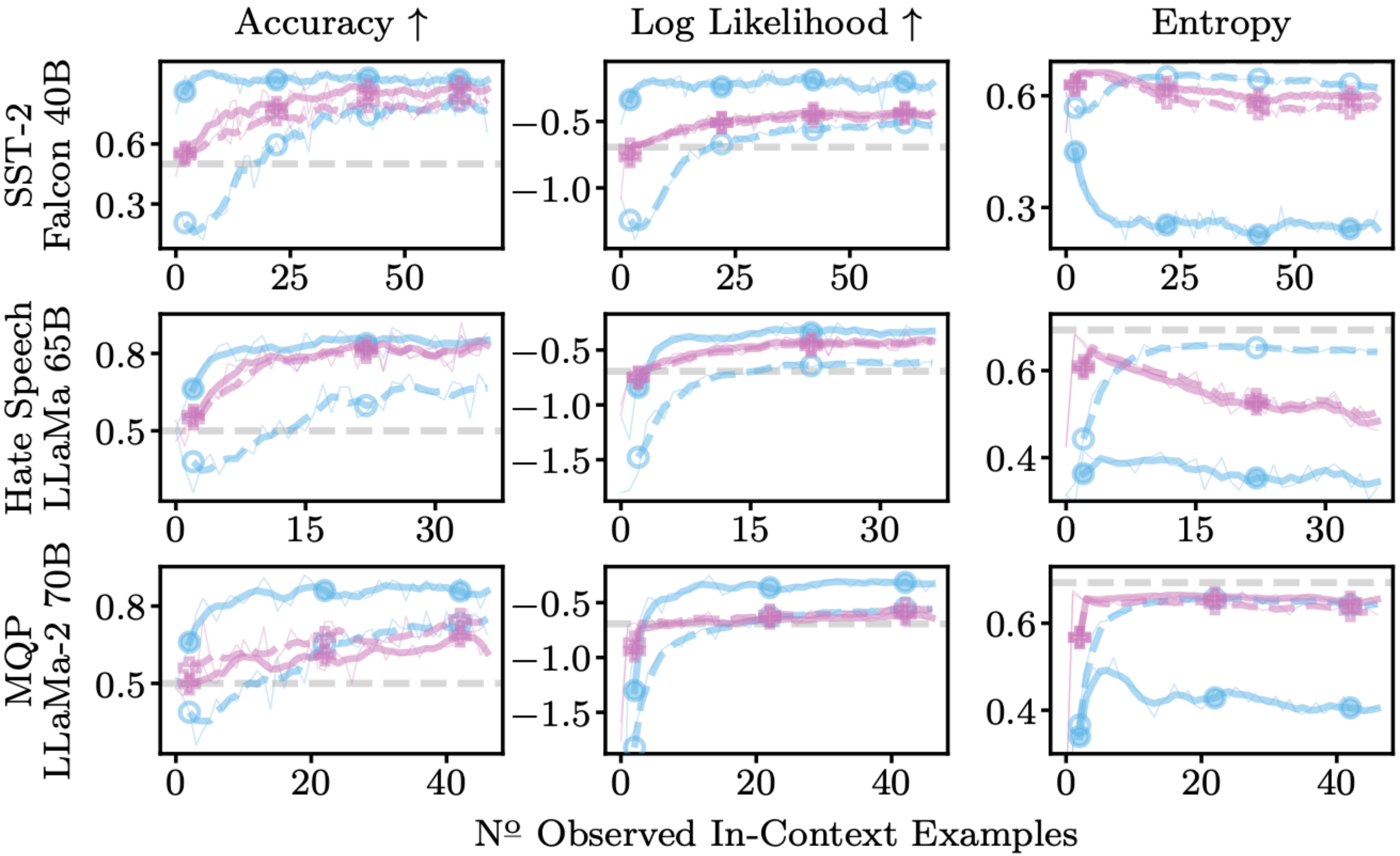


Figure 4: Few-shot ICL achieves accuracies significantly better than random guessing on our **novel author identification** task. Thus, LLMs can learn novel label relationships entirely in-context. Averages over 500 runs, thick lines with additional moving average (window size 5) for clarity.



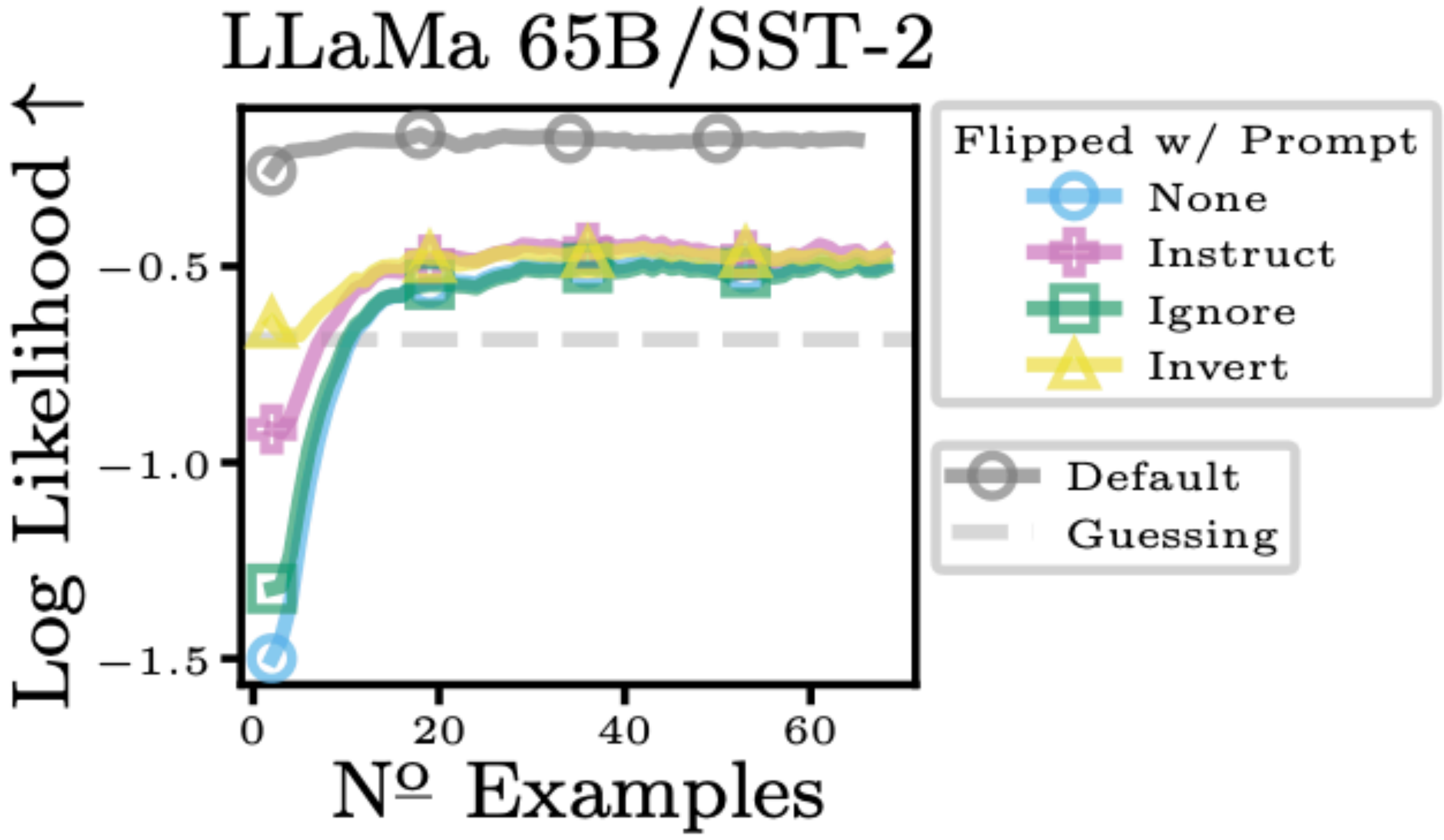


Fig. 6: Prompts yield limited gains.

Figure 5: Few-shot ICL with **replacement labels** for Falcon-40B on SST-2, LLaMa-2-65B on Hate Speech, and LLaMa-2-70B on MQP. Table 2 and §F contain results for all other models and tasks. ICL achieves better than guessing performance for all label relations and models. However, predictions for flipped labels (dashed blue) plateau at a higher entropies and lower likelihoods than those for the default label relation (solid blue). For arbitrary labels (pink), the model performs similarly for both label directions. Averages over 100 runs and thick lines with moving average (window size 5).

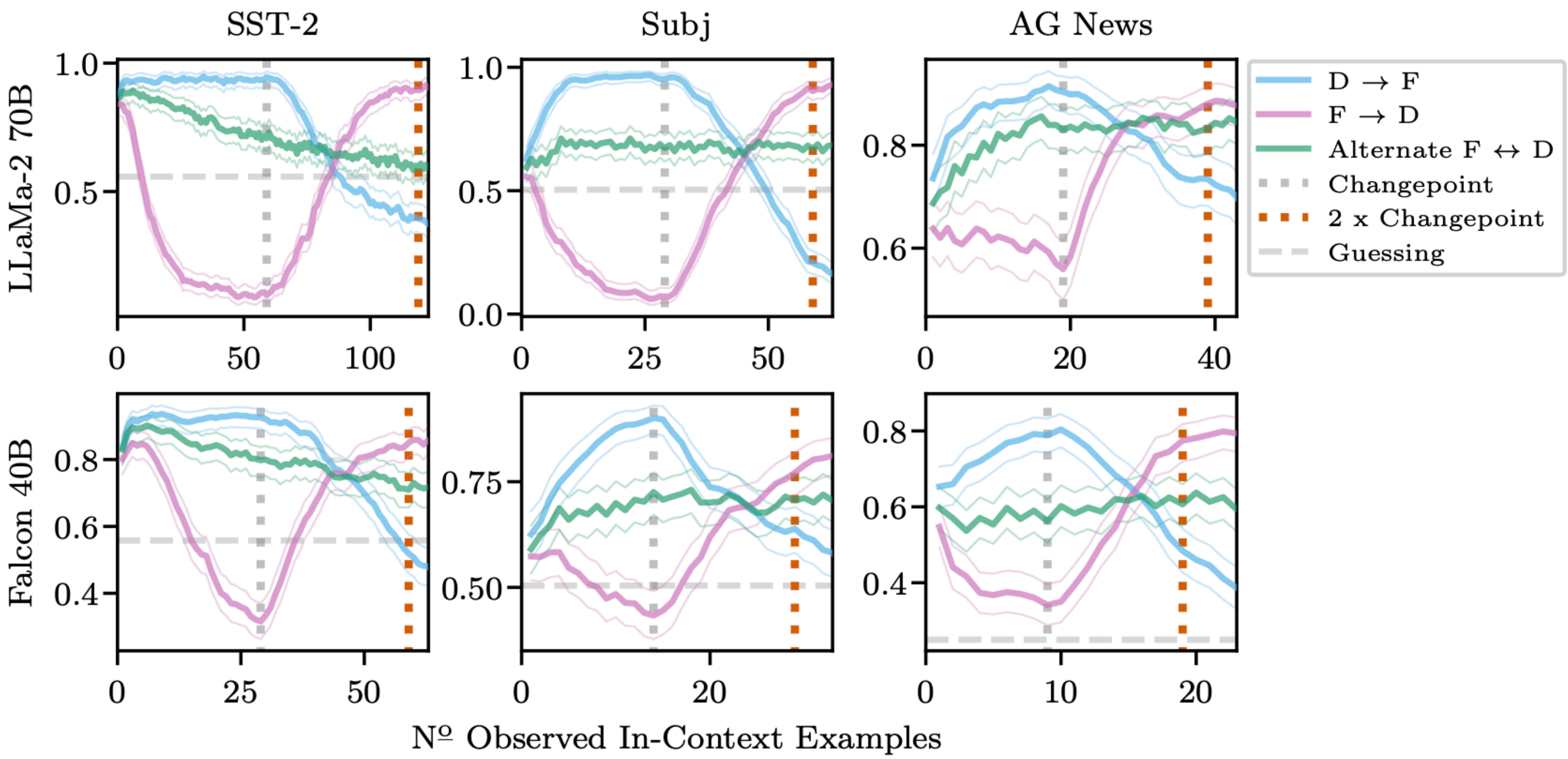


Figure 7: Few-shot ICL accuracies when the **label relationship changes throughout ICL**. For (D → F), we start with default labels and change to flipped labels at the changepoint, for (F → D) we change from flipped to the default labels at the changepoint, and for (Alternating F ↔ D) we alternate between the two label relationships after every observation. For all setups, at '2 x Changepoint', the LLMs have observed the same number of examples for both label relations. If, according to NH3, ICL treats all in-context information equally, predictions should be equal at that point—but they are not. Bootstrapped 99 % confidence intervals, moving averages (size 3), and 500 repetitions.

Efficient few-shot learning

Prompt tuning: few-shot with smaller LMs

iPet: better pre-training for each task improves accuracy for small LMs

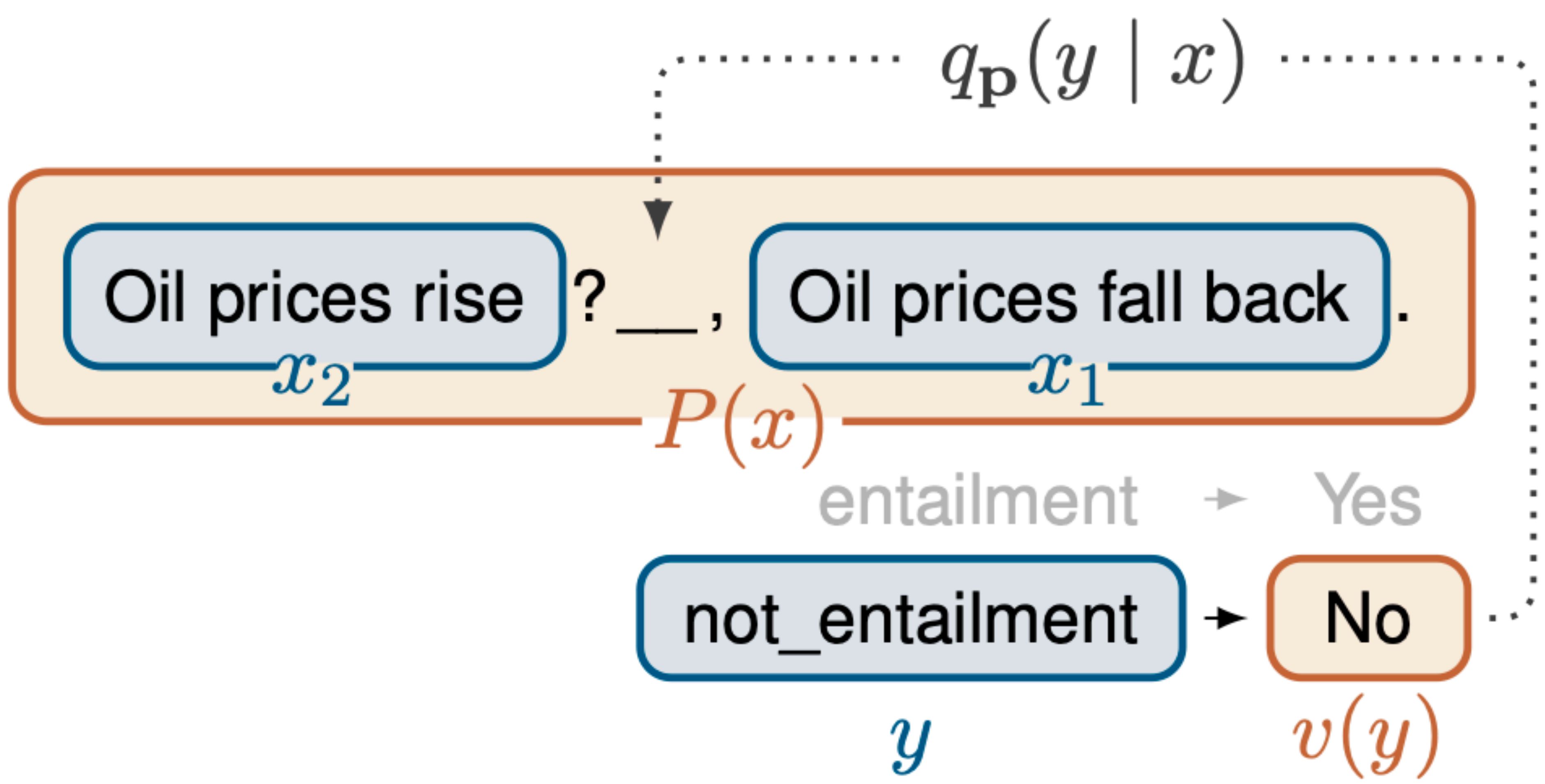


Figure 2: Application of a PVP $\mathbf{p} = (P, v)$ for recognizing textual entailment: An input $x = (x_1, x_2)$ is converted into a cloze question $P(x)$; $q_{\mathbf{p}}(y \mid x)$ for each y is derived from the probability of $v(y)$ being a plausible choice for the masked position.

test	GPT-3	175,000	71.8	prompt
	PET	223	74.0	prompt FT
	iPET	223	**75.4**	prompt FT
	SotA	11,000	*89.3*	full FT

https://arxiv.org/abs/2001.07676