ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

CENG 463: Introduction to Natural Language Processing
Large Language Models (Training and Inference)

Asst. Prof. Çağrı Toraman
Computer Engineering Department
ctoraman@ceng.metu.edu.tr

23.12.2025

How to train LLMs?

Main Approach

Step #1) Pretrain LLM

data, time, hardware ?

Step #2) Fine-tune it for your task

How to train LLMs?

Main Approach

Encoder-based (Representation) models (BERT) → We discussed it before!

Generative models (GPT)

How to train LLMs?

Main Approach

1. **Auto-regressive Pre-training**

   Train to predict the next token on very large scale corpora (e.g. Llama2 has ~3 trillion tokens)

2. **a. Supervised Fine-tuning (SFT)**

   Fine-tune the pretrained model with having no data modifications but adding new layers

   b. Instruction Fine-tuning

   Fine-tune the pretrained model with pairs of (instruction+input,output)

How to train LLMs?

Main Approach

1. **Auto-regressive Pre-training**

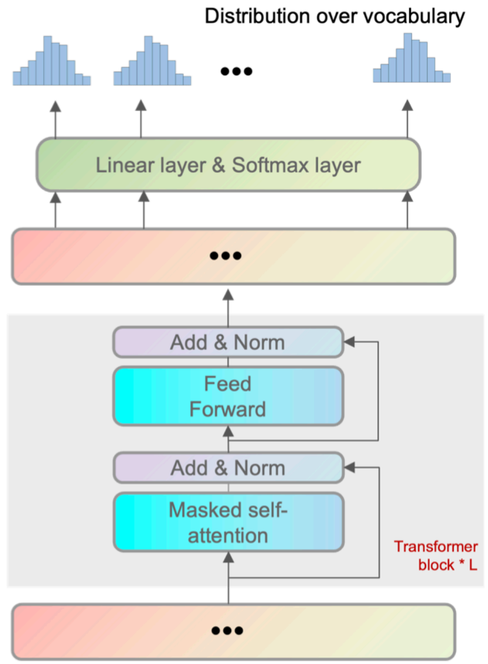
   Loss computed for all tokens
   Raw data (check the Pile data from our last lecture)
   General auto-complete

2. **Supervised Fine-tuning (SFT) / Instruction Fine-tuning**

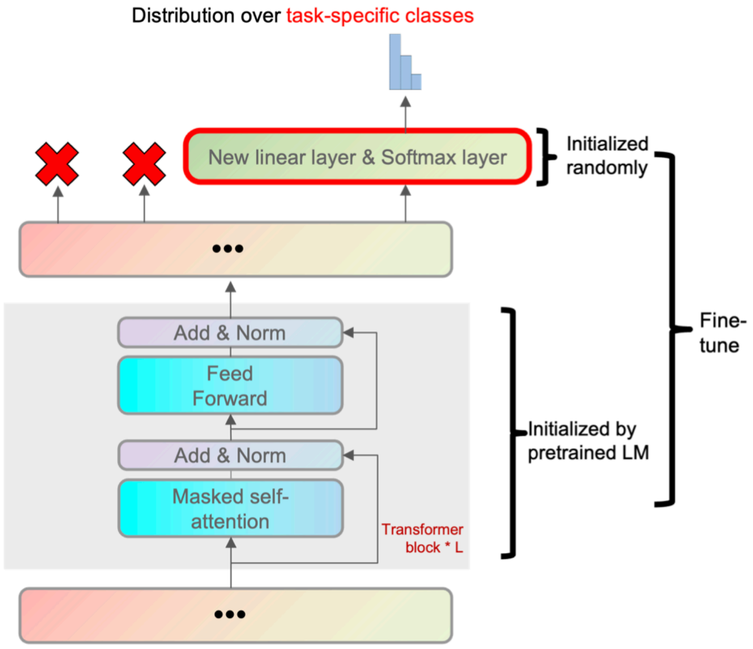
   Loss computed for target tokens
   Instruction data
   Generalize to unseen tasks (and also safety?)

How to train LLMs?

Main Approach (GPT1*)



1. **Pre-training**



2. a. **Supervised Fine-tuning (SFT)**

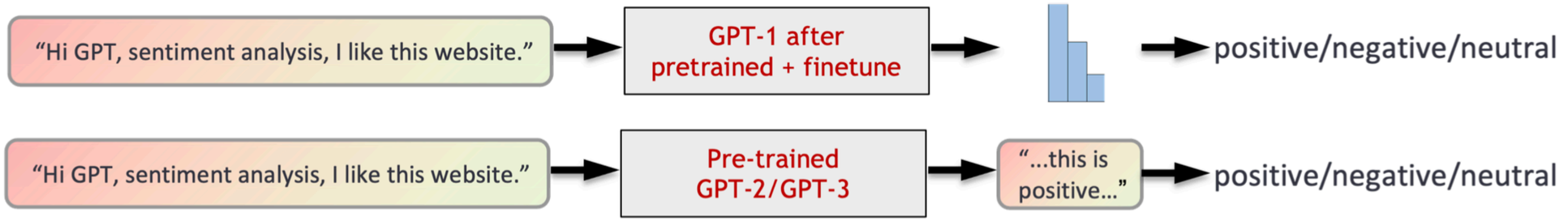
Some slides adapted from C4DT Workshop EPFL

**Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding with unsupervised learning. Technical report, OpenAI, 2018.*

How to train LLMs?

Main Approach (GPT2* and 3**)

Same as GPT-1: we still pre-train the LM on unlabeled corpus.
New: no need to fine-tune anymore. One pre-trained LM for all tasks, achieve SOTA.



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

** Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In Advances in Neural Information Processing Systems, 2020.

How to train LLMs?

Main Approach (GPT1 vs GPT2/3)

- GPT-1: finetune the model on a specific task.

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

- GPT-3: no fine-tuning is fine.

Zero-shot

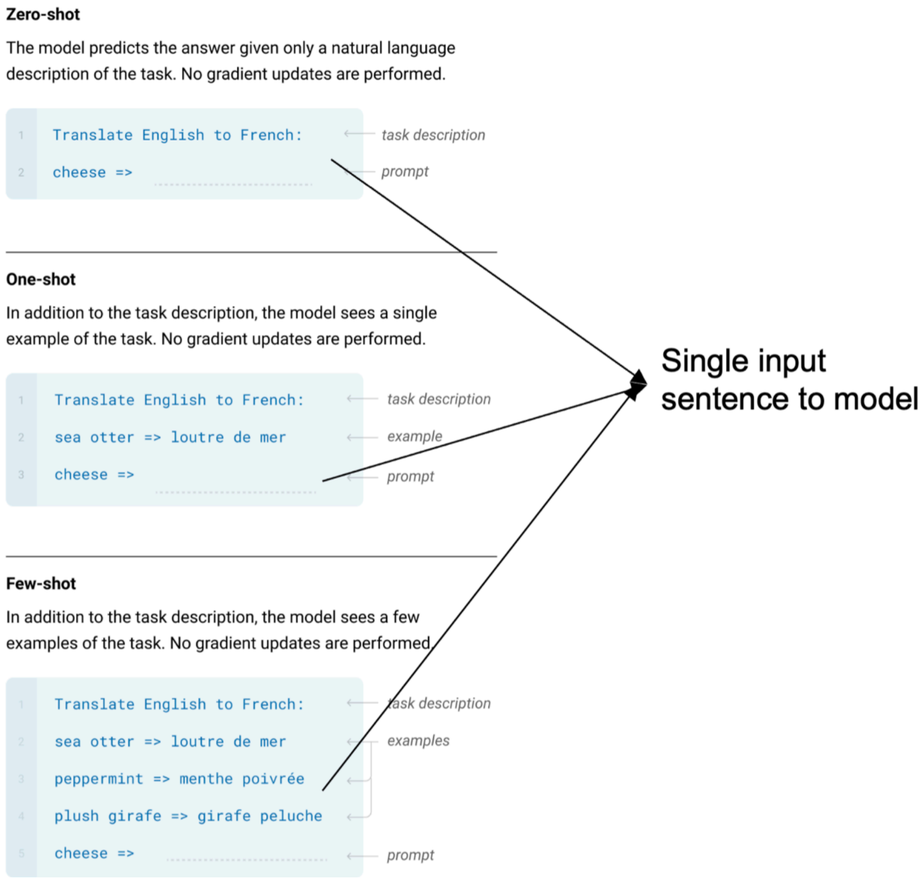
The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

One-shot

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

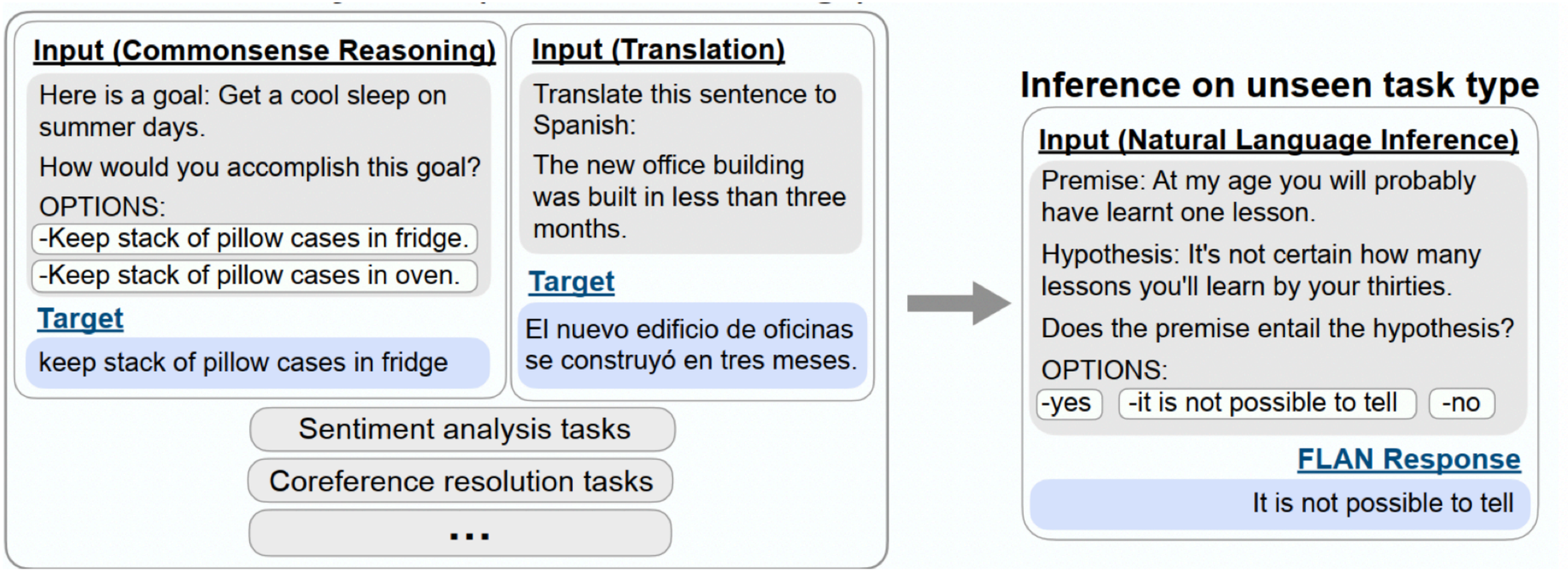
Few-shot

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



How to train LLMs?

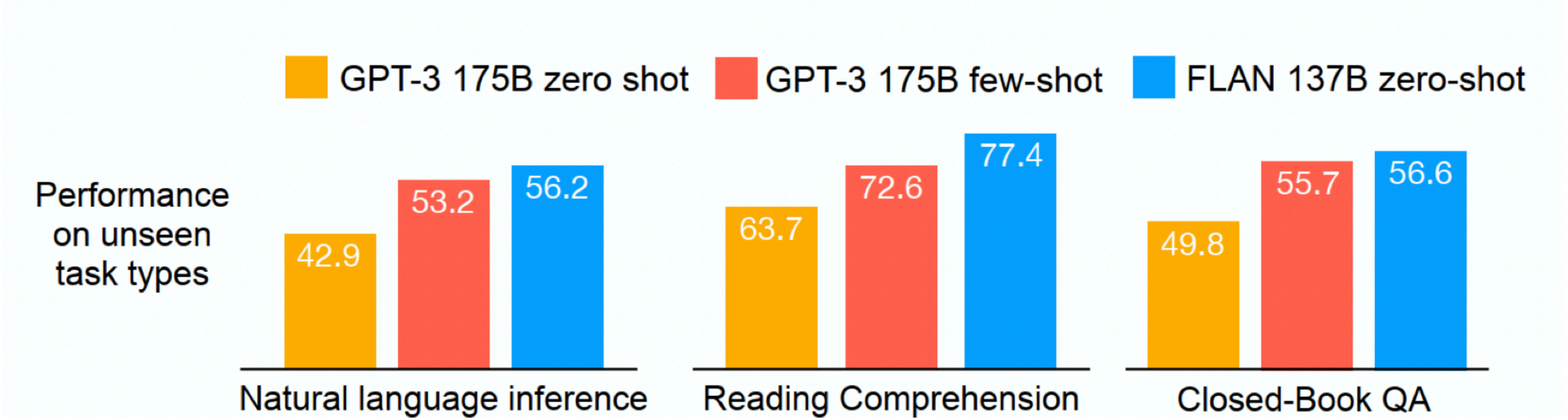
Main Approach (2.b. Instruction Fine-tuning by FLAN*)



Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In International Conference on Learning Representations, 2022.

How to train LLMs?

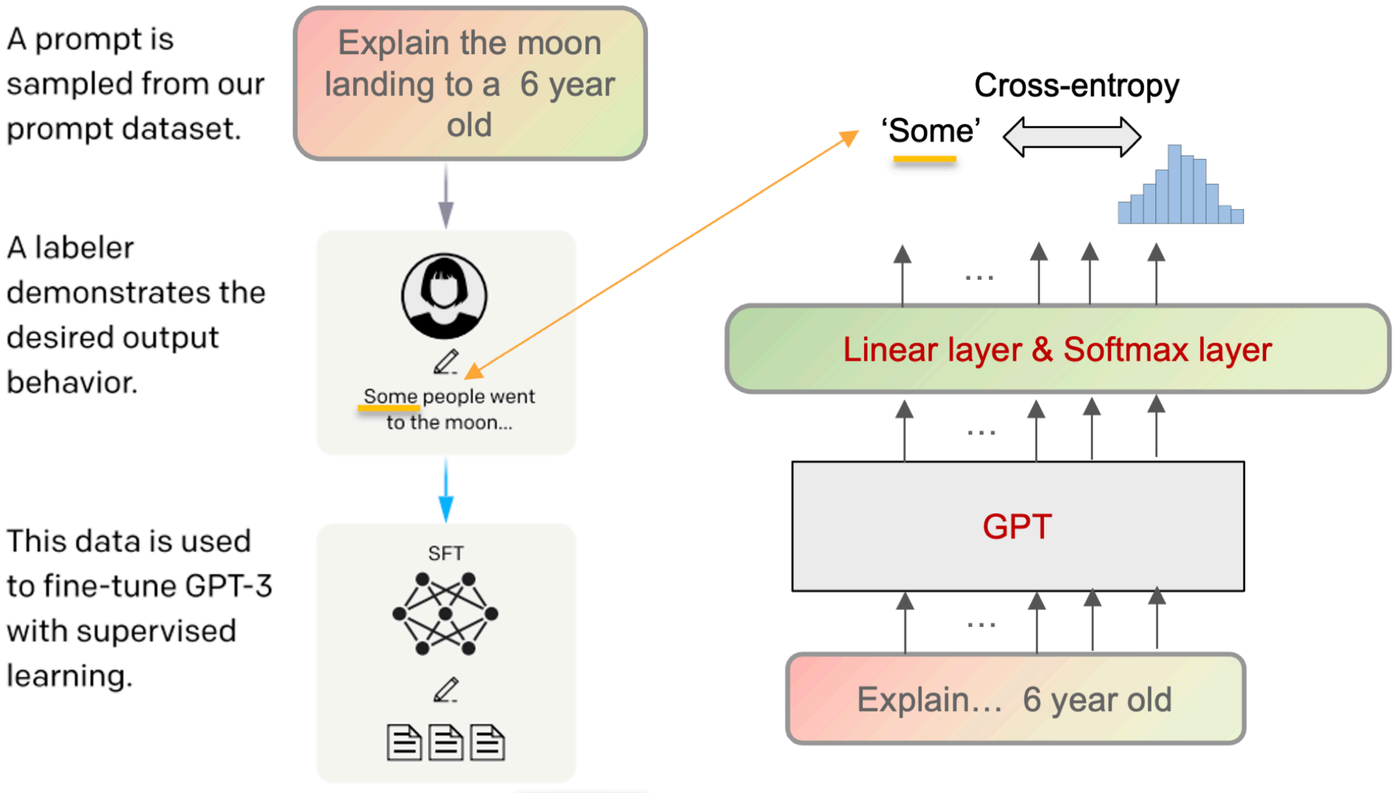
Main Approach (2.b. Instruction Fine-tuning by FLAN*)



Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In International Conference on Learning Representations, 2022.

How to train LLMs?

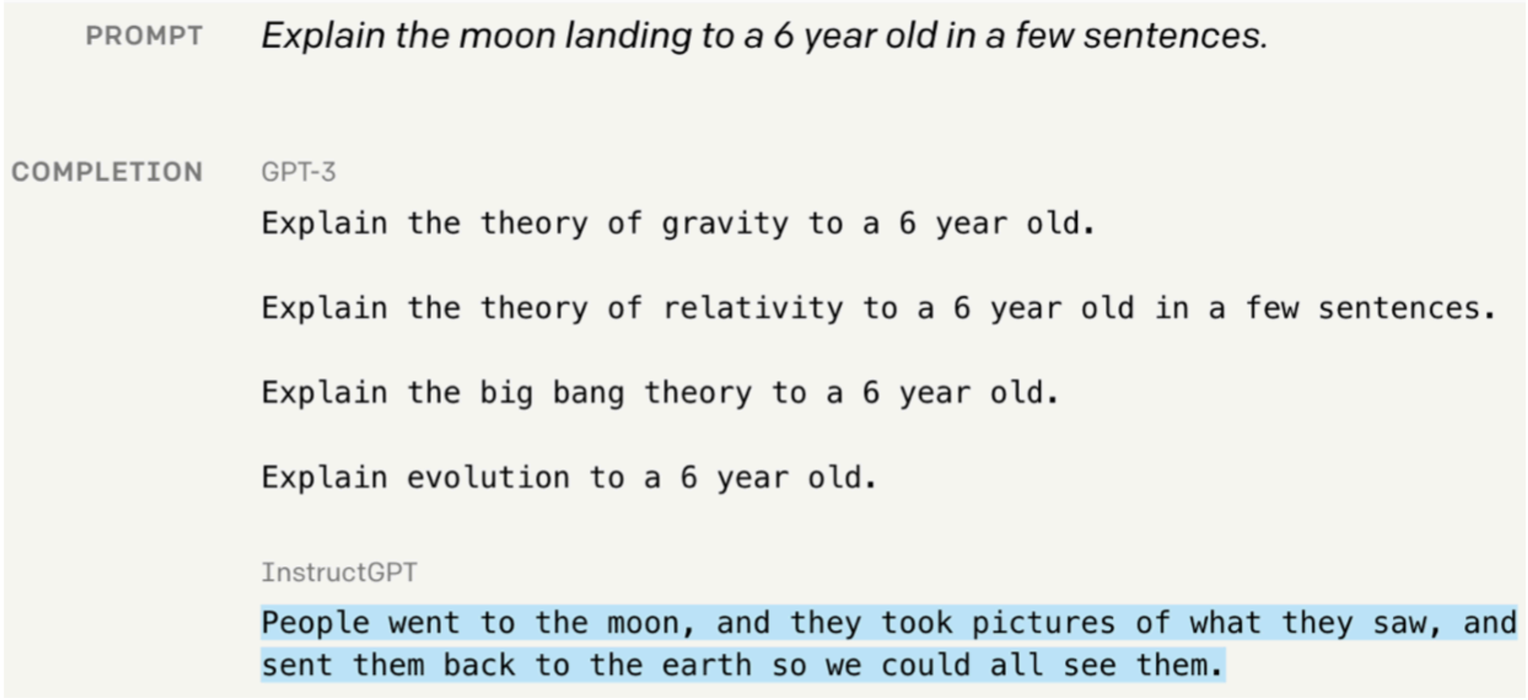
Main Approach (2.b. Instruction Fine-tuning by InstructGPT/GPT3.5*)



*Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.

How to train LLMs?

Main Approach (2.b. Instruction Fine-tuning by InstructGPT*)



*Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.

How to train LLMs?

Main Approach (2.b. Instruction Fine-tuning by InstructGPT*)

1. **Auto-regressive Pre-training**

   Train to predict the next token on very large scale corpora (e.g. Llama2 has ~3 trillion tokens)

2. **b. Instruction Fine-tuning**

   Fine-tune the pretrained model with pairs of (instruction+input,output)

3. **Human Alignment**

**Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.*

How to train LLMs?

Main Approach (2.b. Instruction Fine-tuning by InstructGPT*)

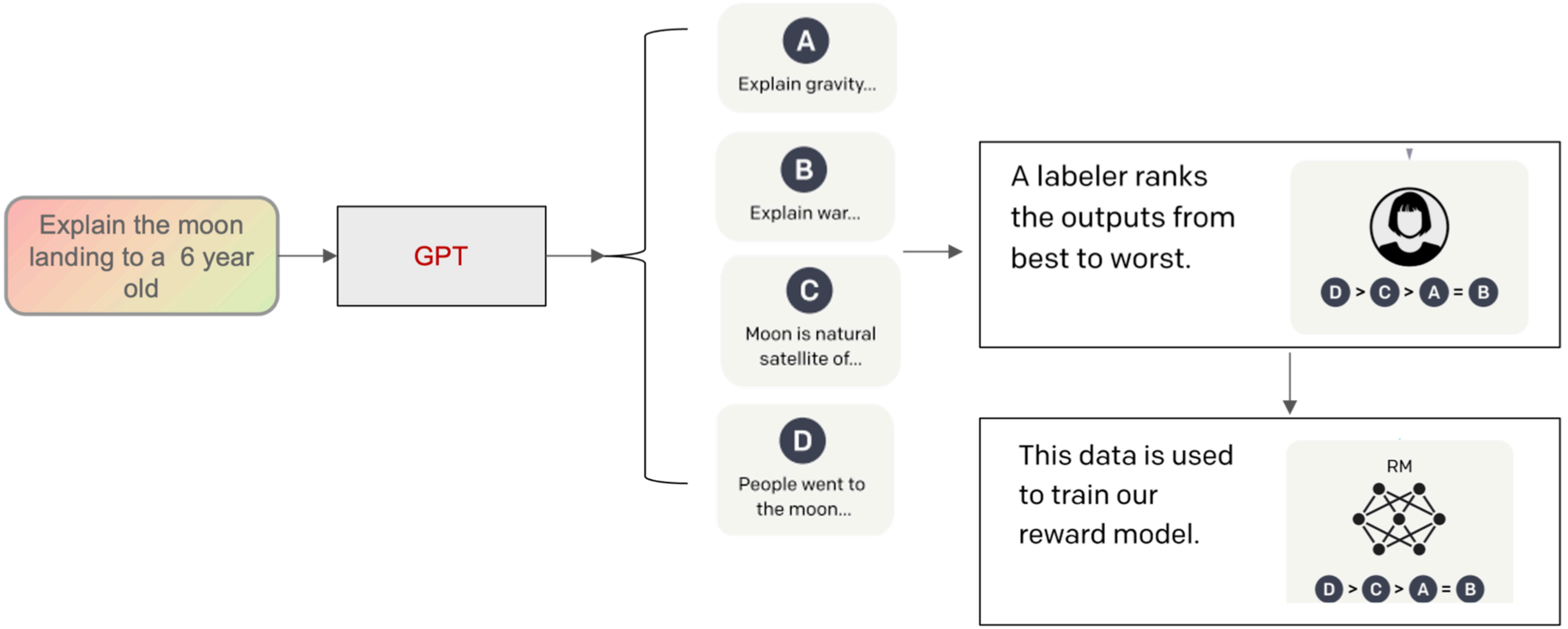
Human Alignment

*Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.

How to train LLMs?

Main Approach (2.b. Instruction Fine-tuning by InstructGPT*)

Human Alignment



*Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.

How to train LLMs?

Main Approach (2.b. Instruction Fine-tuning by InstructGPT*)

Human Alignment

Train a reward model $r_{\mathbf{x}}(S_{\text{prompt}}, S_{\text{response}})$ with parameters $\mathbf{x}$.

Loss:

$$L_{\mathbf{x}} = -\frac{1}{\binom{K}{2}} E_{(S_{\text{prompt}}, S_{\text{response1}}, S_{\text{response2}}) \sim D} \left[\log \left(\sigma \left(r_{\mathbf{x}} \left(S_{\text{prompt}}, S_{\text{response1}} \right) - r_{\mathbf{x}} \left(S_{\text{prompt}}, S_{\text{response2}} \right) \right) \right) \right],$$

*Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.

How to train LLMs?

Main Approach (2.b. Instruction Fine-tuning by InstructGPT*)

Human Alignment

Using this reward model to fine-tune GPT via Proximal Policy Optimization (PPO)

(state, action): $(S_{\text{prompt}}, S_{\text{response}})$.
Initialize a policy to be the fine-tuned GPT in step 2, i.e., π^{SFT}.
Initialize a copy of the above policy with parameters ϕ that we want to optimize, i.e., π_{ϕ}^{RL}.
Use PPO to optimize ϕ in order to maximize the following objective.

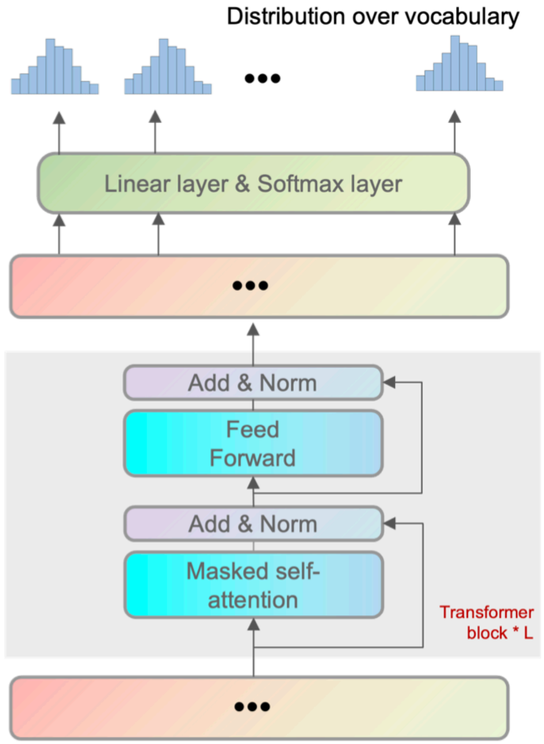
$$L_{\phi}(S_{\text{prompt}}, S_{\text{response}}) = r_{\mathbf{x}}(S_{\text{prompt}}, S_{\text{response}}) - \underbrace{\beta \log[\pi_{\phi}^{\text{RL}}(S_{\text{response}}|S_{\text{prompt}})/\pi^{\text{SFT}}(S_{\text{response}}|S_{\text{prompt}})]}_{\text{penalty term}}$$

**Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35, 27730-27744.*

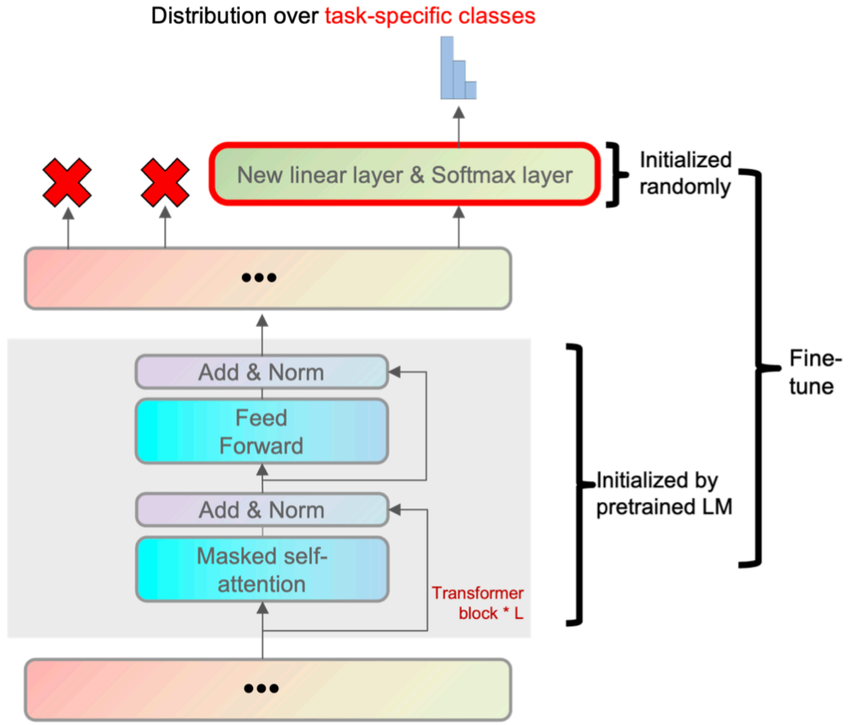
How to train LLMs?

Main Approach (Llama1*)

- Pre-normalization
- SwiGLU instead of ReLU
- Rotary position embeddings



1. **Pre-training**



2. a. **Supervised Fine-tuning (SFT)**

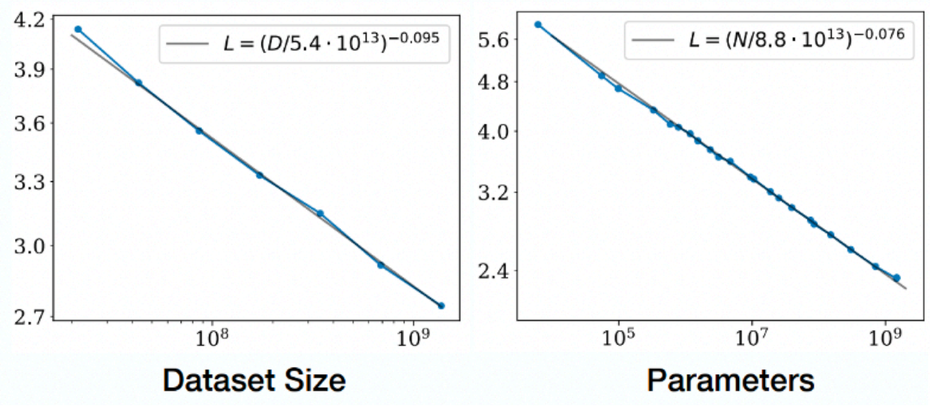
**Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.*

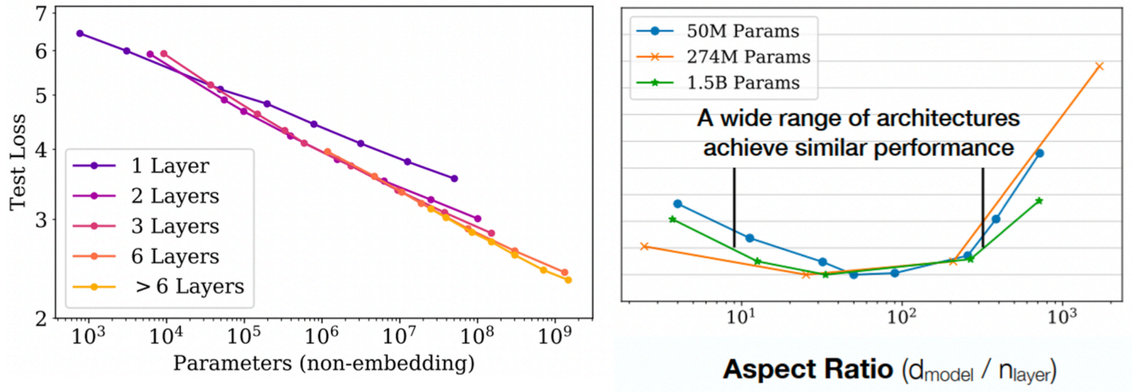
How to train LLMs?

Main Approach (Llama1*)

LLaMA-13B outperforms GPT-3 (175B) on many benchmarks!
GPT-3: 300B tokens
LLaMA: 1T tokens
Why 13B is better than 175B in this case?

Which one has impact? (model size), (data-token size), or (architecture size)?**





*Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.
**Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. arXiv preprint arXiv:2001.08361, 2020.

How to train LLMs?

Main Approach (Llama2*)

Llama2: updated version of pretrained Llama1

Llama2-Chat: Similar to InstructGPT (GPT3.5)
(Pretraining + Instruction-tuning + RLHF)

LLaMA-2 Params	Time (GPU hours)	Power Consumption (W)
7B	184320	400
13B	368640	400
34B	1038336	350
70B	1720320	400

Table: GPU time (Nvidia A100 80GB) and power consumption for pre-training each model. If you have 1000 Nvidia A100, you can finish the pre-training in one week.

*Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., ... & Scialom, T. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288.

How to train LLMs?

Main Approach (Llama3*)

Similar to InstructGPT (GPT3.5)
(Pretraining + Instruction-tuning + RLHF)

	Pre-training Data	Params	Context Length	Grouped-query attention	Tokens
LLaMA-1	*See previous slide*	7B	2k	✗	1T
		13B	2k	✗	1T
		33B	2k	✗	1.4T
		65B	2k	✗	1.4T
LLaMA-2	*A new mix of publicly available online data*	7B	4k	✗	2T
		13B	4k	✗	2T
		34B	4k	✓	2T
		70B	4k	✓	2T
LLaMA-3	*7 times larger than that of* LLaMA-2	8B	8k	✓	15T
		70B	8k	✓	15T

*https://ai.meta.com/blog/meta-llama-3

How to train LLMs?

Main Approach (Llama3.1*, 3.2** and 3.3***)

Similar to InstructGPT (GPT3.5)
(Pretraining + Instruction-tuning + RLHF)

Llama3.1: updated version of Llama3

- 3.1-405B outperforms 3 in many benchmarks, particularly in mathematical reasoning
- 3.1 provides extensive context (128k vs. 8k), long-form content generation (4096 vs. 2048)

Llama3.2: light-weight (1B and 3B) and multimodal versions of Llama3.1

Llama3.3: better version of its predecessors (multilingual, 70B, 15T tokens, cheaper than OpenAI)

*https://ai.meta.com/blog/meta-llama-3-1/
**https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/
***https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_3/#introduction

What is prompt engineering?

GPT-1: finetune the model on a specific task.

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

- GPT-3: no fine-tuning is fine.

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:
cheese =>
```

task description, 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:
sea otter => loutre de mer
cheese =>
```

task description, example, 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:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>
```

task description, examples, prompt

Single input sentence to model

What is prompt engineering?

Prompt:

Tell the model what to do in natural language.
Short or long as required.

Prompt Engineering:

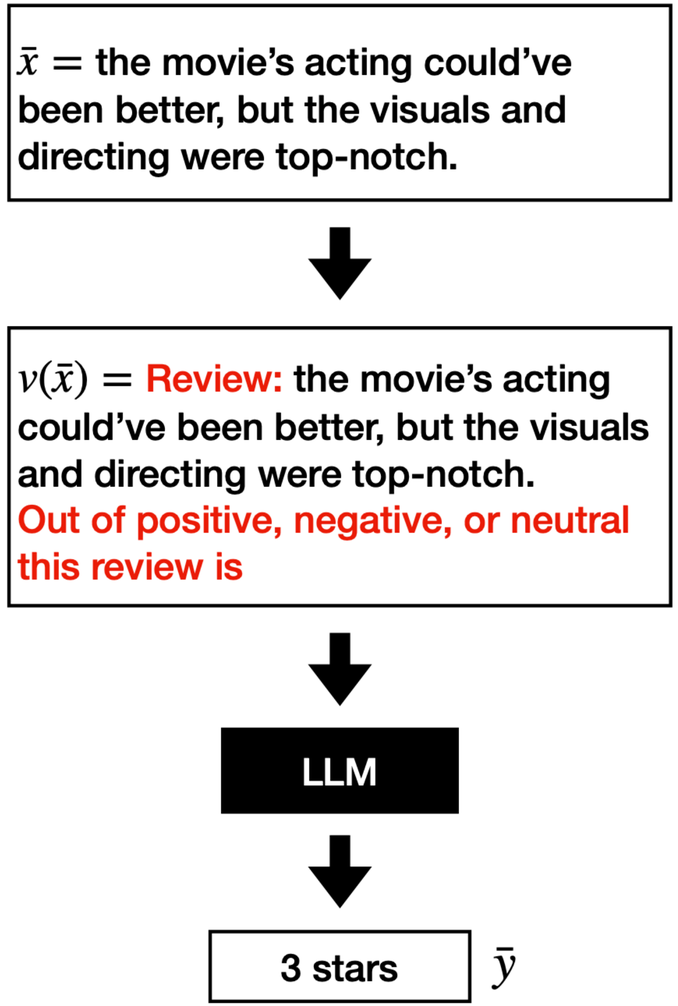
Identifying the correct prompt needed to perform a task.
Specific and descriptive as possible.
Manual or automatic.

What is prompt engineering?

Zero-shot prompting:

Only instruction

Any text-to-text task



**Some slides adapted from Cornell CS 5740*

What is prompt engineering?

Zero-shot prompting:

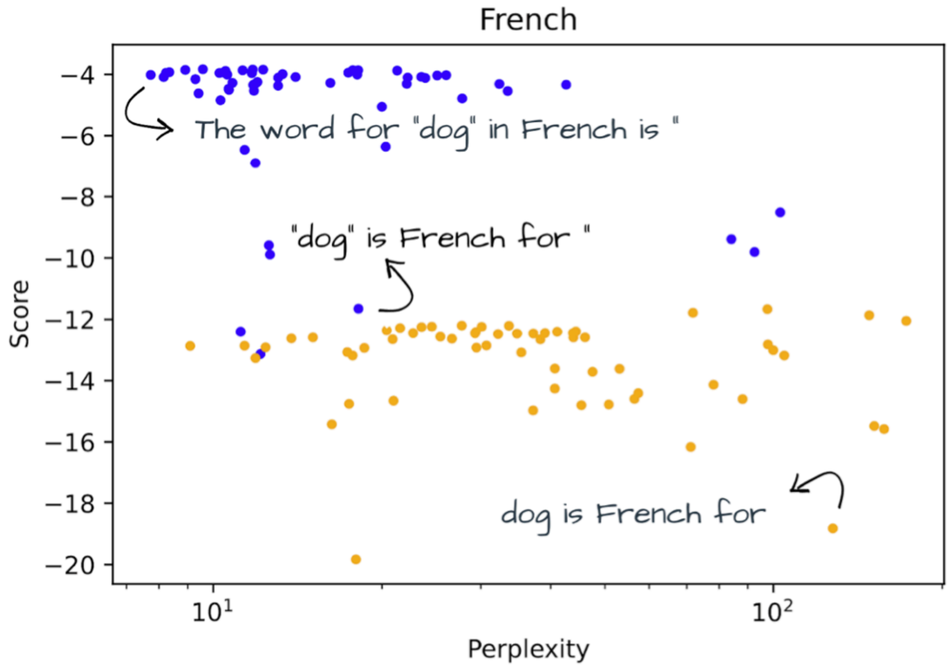
+ No need to train/fine-tune

- What if model outputs correct answer in a different shape/syntax? (e.g. 8 or eight)

What is prompt engineering?

Zero-shot prompting:

- Many ways to write prompts for the same task*

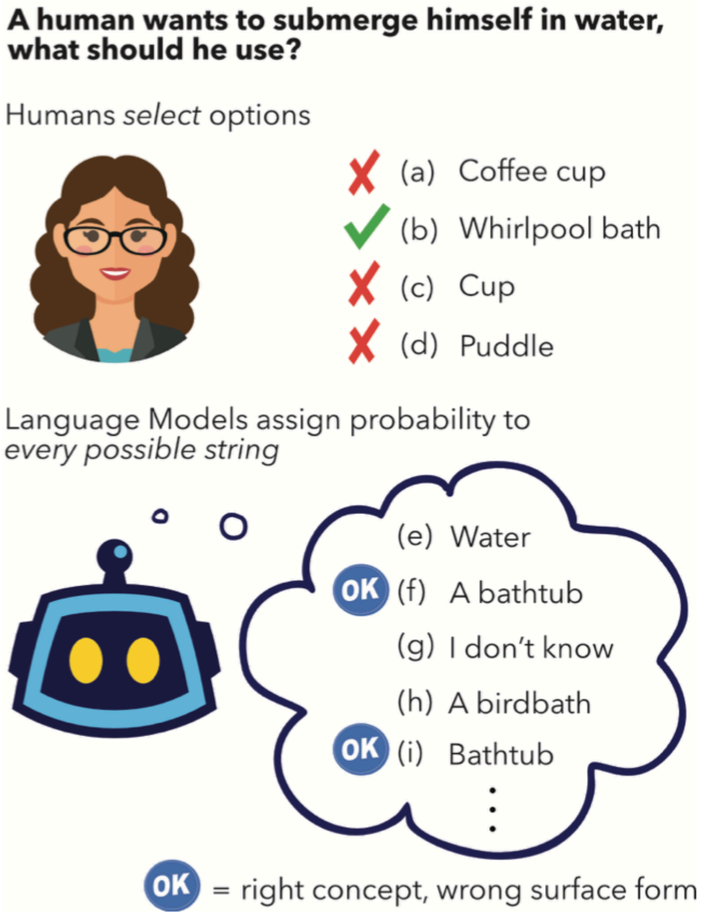


*Gonen, H., Iyer, S., Blevins, T., Smith, N. A., & Zettlemoyer, L. (2022). Demystifying prompts in language models via perplexity estimation. arXiv preprint arXiv:2212.04037.

What is prompt engineering?

Zero-shot prompting:

- Humans restrict their choice, while AI does not*



*Holtzman, A., West, P., Shwartz, V., Choi, Y., & Zettlemoyer, L. (2021). Surface form competition: Why the highest probability answer isn't always right. arXiv preprint arXiv:2104.08315.

What is prompt engineering?

Prompt engineering

Similar to hyperparameter search

Find optimal prompts for your task

Manual engineering vs. automated searching (using gradients or black-box search)

What is prompt engineering?

Zero-shot prompting:

Only instruction

Few-shot prompting (In-context learning)*:

With >1 demonstrations/examples

Instruction

```
Please classify movie reviews as 'positive' or 'negative'
```

Examples

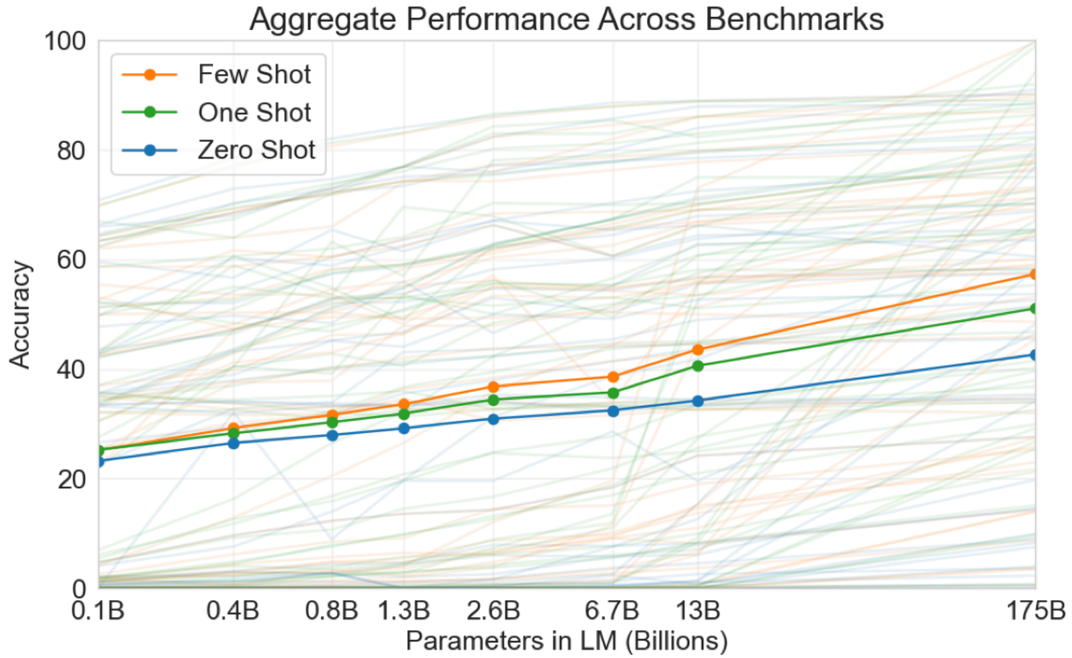
```
Input: I really don't like this movie.
Output: negative

Input: This movie is great!
Output: positive
```

*Brown, T. B. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.

What is prompt engineering?

Few-shot prompting (In-context learning):

Compared to zero-shot setting?



*Brown, T. B. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.

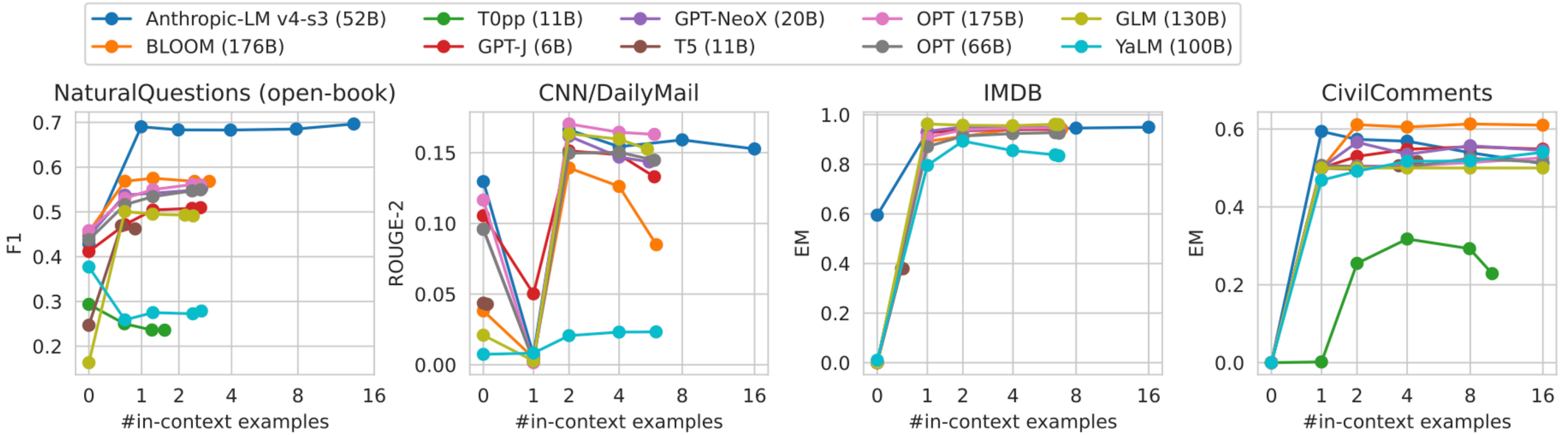
Model	0-shot	1-shot	2-shot	3-shot
Llama-7b	0.00	0.50	0.53	0.50
LlamaTurk-7b-c	0.00	0.47	0.54	0.51
LlamaTurk-7b-i	0.06	0.48	0.48	0.56
LlamaTurk-7b-t	0.90●	0.84○	0.61	0.78
LlamaTurk-7b-c-i	0.10	0.52	0.50	0.54
LlamaTurk-7b-i-t	0.83○	0.90●	0.93●	0.89●
LlamaTurk-7b-c-t	0.82	0.60	0.62	0.86○
LlamaTurk-7b-c-i-t	0.62	0.52	0.56	0.51
LlamaTurk-7b-v-i	0.35	0.44	0.49	0.53
LlamaTurk-7b-v-t	0.44	0.50	0.53	0.53

**Toraman, C. (2024). Adapting Open-Source Generative Large Language Models for Low-Resource Languages: A Case Study for Turkish. In Proceedings of the Fourth Workshop on Multilingual Representation Learning (MRL 2024) (pp. 30-44).

What is prompt engineering?

Few-shot prompting (In-context learning):

How many demonstrations needed?

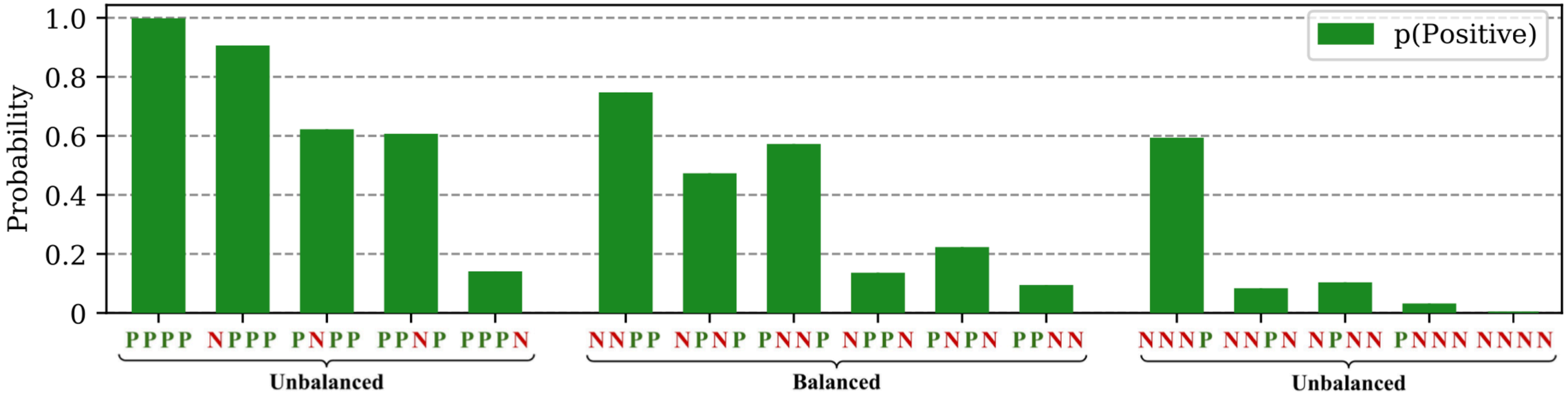


*Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., ... & Koreeda, Y. (2022). Holistic evaluation of language models. arXiv preprint arXiv:2211.09110.

What is prompt engineering?

Few-shot prompting (In-context learning):

Majority label bias and recency bias



*Zhao, Z., Wallace, E., Feng, S., Klein, D., & Singh, S. (2021, July). Calibrate before use: Improving few-shot performance of language models. In International conference on machine learning (pp. 12697-12706).

What is prompt engineering?

Few-shot prompting (In-context learning):

What if we have random examples?*

* *Rethinking the Role of Demonstrations: What Makes In-Context Learning Work? OpenReview https://openreview.net/forum?id=cnRGMv-Ak7u*

What is prompt engineering?

Full Fine-tuning (FT)

a. +Strongest performance
b. - Need curated and labeled dataset for each new task (typically 1k-100k+ ex.)
c. - Poor generalization, spurious feature exploitation

Few-shot (FS)

a. +Much less task-specific data needed
b. +No spurious feature exploitation
c. - Challenging

One-shot (1S)

a. +"Most natural," e.g. giving humans instructions
b. - Challenging

Zero-shot (0S)

a. +Most convenient
b. - Challenging, can be ambiguous

More data, task-specific

Less data, general

ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

Thanks for your participation!

Çağrı Toraman
23.12.2025