Natural Language Processing

AI51701/CSE71001

Lecture 18

11/21/2023

Instructor: Taehwan Kim

Announcement

- Assignment 3 will be released tonight
  - Due: Dec. 3 at 11:59pm KST

- Final project timeline survey
  - Report due on 12/11,
    Presentations on 12/12 and 12/14?
  - Using the last lecture (12/5) for team meeting time?

Prompting

(Adapted from Jessy Mu’s slides in Stanford 224n)

Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning

Emergent abilities of large language models: GPT-2 (2019)

- Let's revisit the Generative Pretrained Transformer (GPT) models from OpenAI as an example:
- GPT-2 (1.5B parameters; Radford et al., 2019)
  - Same architecture as GPT, just bigger (117M -> 1.5B)
  - But trained on **much more data**: 4GB -> 40GB of internet text data (WebText)
    - Scrape links posted on Reddit w/ at least 3 upvotes (rough proxy of human quality)

Language Models are Unsupervised Multitask Learners

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

Emergent zero-shot learning

- One key emergent ability in GPT-2 is **zero-shot learning**: the ability to do many tasks with **no examples**, and **no gradient updates**, by simply:
  - Specifying the right sequence prediction problem (e.g. question answering):

```
Passage: Tom Brady... Q: Where was Tom Brady born? A: ...
```

  - Comparing probabilities of sequences (e.g. Winograd Schema Challenge [Levesque, 2011]):

```
The cat couldn't fit into the hat because it was too big.
Does it = the cat or the hat?

≡ Is P(...because the cat was too big) >=
     P(...because the hat was too big)?
```

[Radford et al., 2019]

Emergent zero-shot learning

- GPT-2 beats SoTA on language modeling benchmarks with no task-specific fine-tuning

Context: "Why?" "I would have thought you'd find him rather dry," she said. "I don't know about that," said Gabriel. "He was a great craftsman," said Heather. "That he was," said Flannery.
Target sentence: "And Polish, to boot," said _____.
Target word: Gabriel

LAMBADA (language modeling w/ long discourse dependencies)

[Paperno et al., 2016]

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)
SOTA	99.8	59.23	85.7	82.3	39.14
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**

[Radford et al., 2019]

Emergent abilities of large language models: GPT-3 (2020)

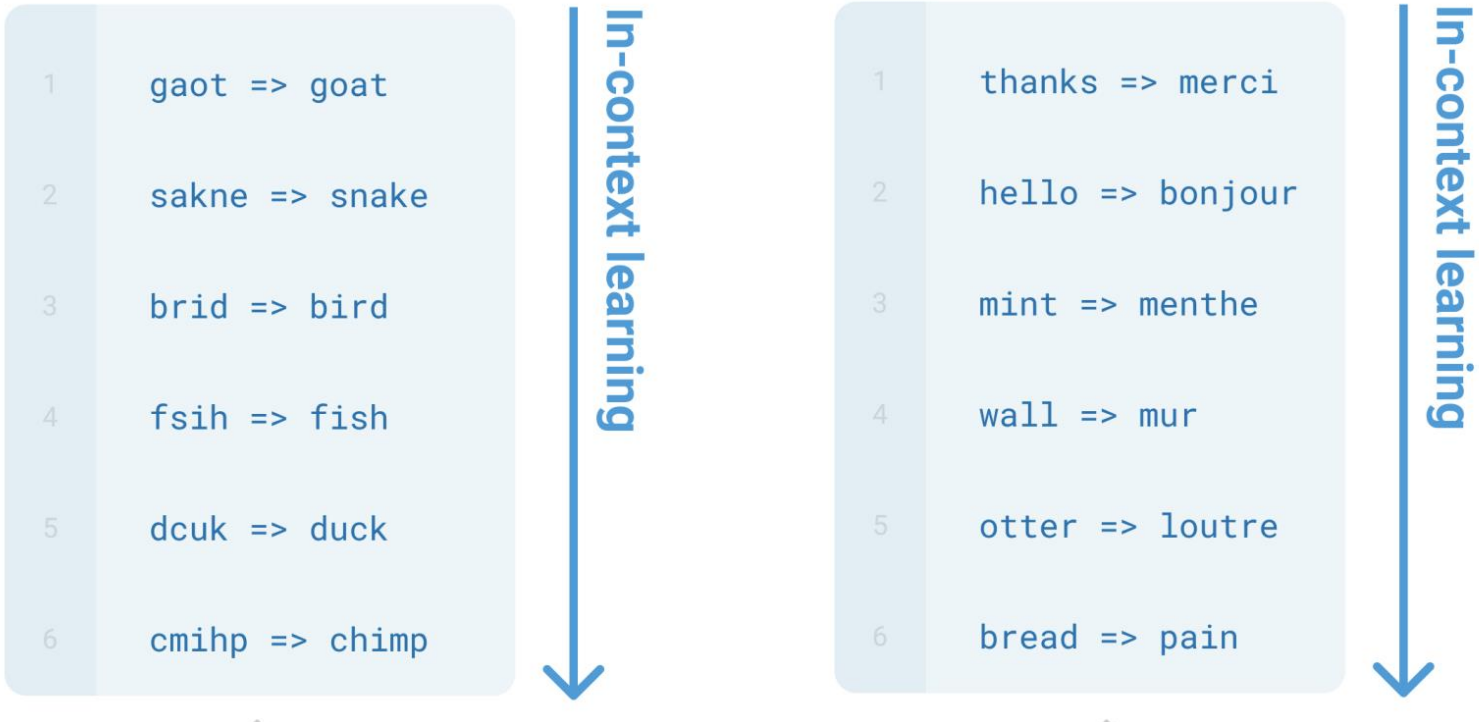
- GPT-3 (175B parameters; Brown et al., 2020)
  - Another increase in size (1.5B -> **175B**)
  - and data (40GB -> **over 600GB**)

Language Models are Few-Shot Learners

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

Emergent few-shot learning

- Specify a task by simply **prepending examples of the task before your example**
- Also called **in-context learning**, to stress that *no gradient updates* are performed when learning a new task

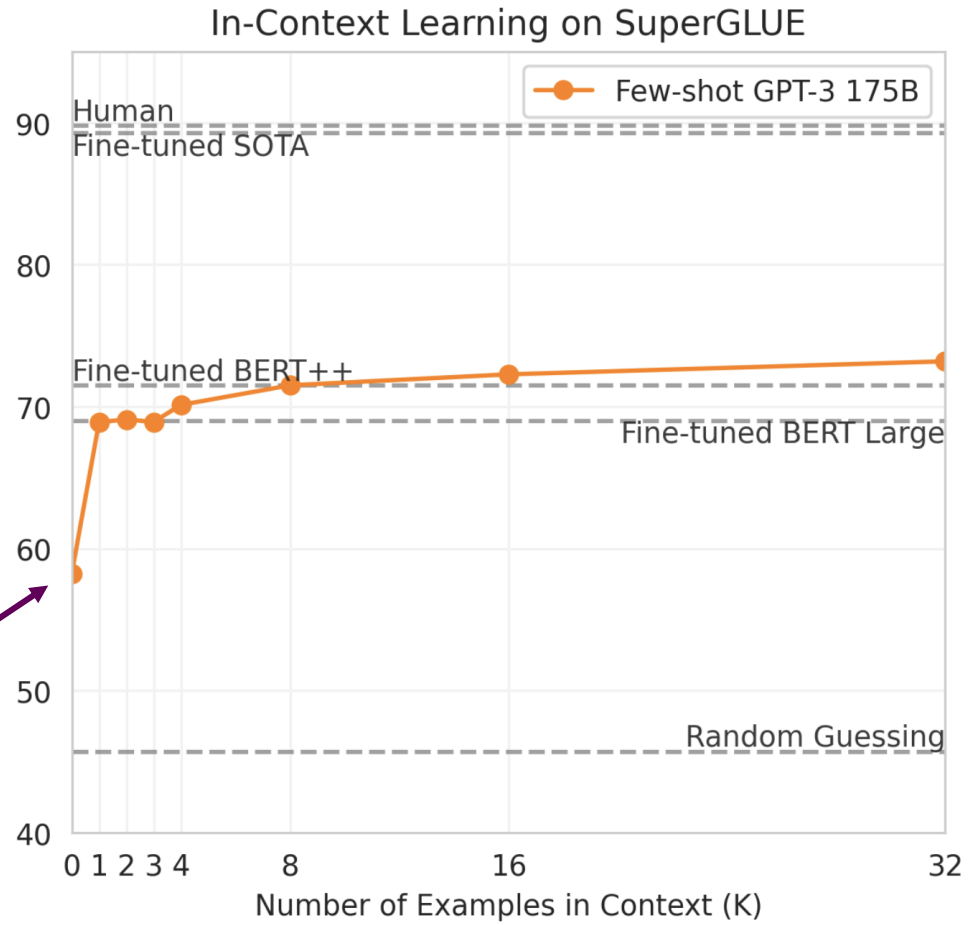


[Brown et al., 2020]

Emergent few-shot learning

Zero-shot

```
1  Translate English to French:
2  cheese =>
```

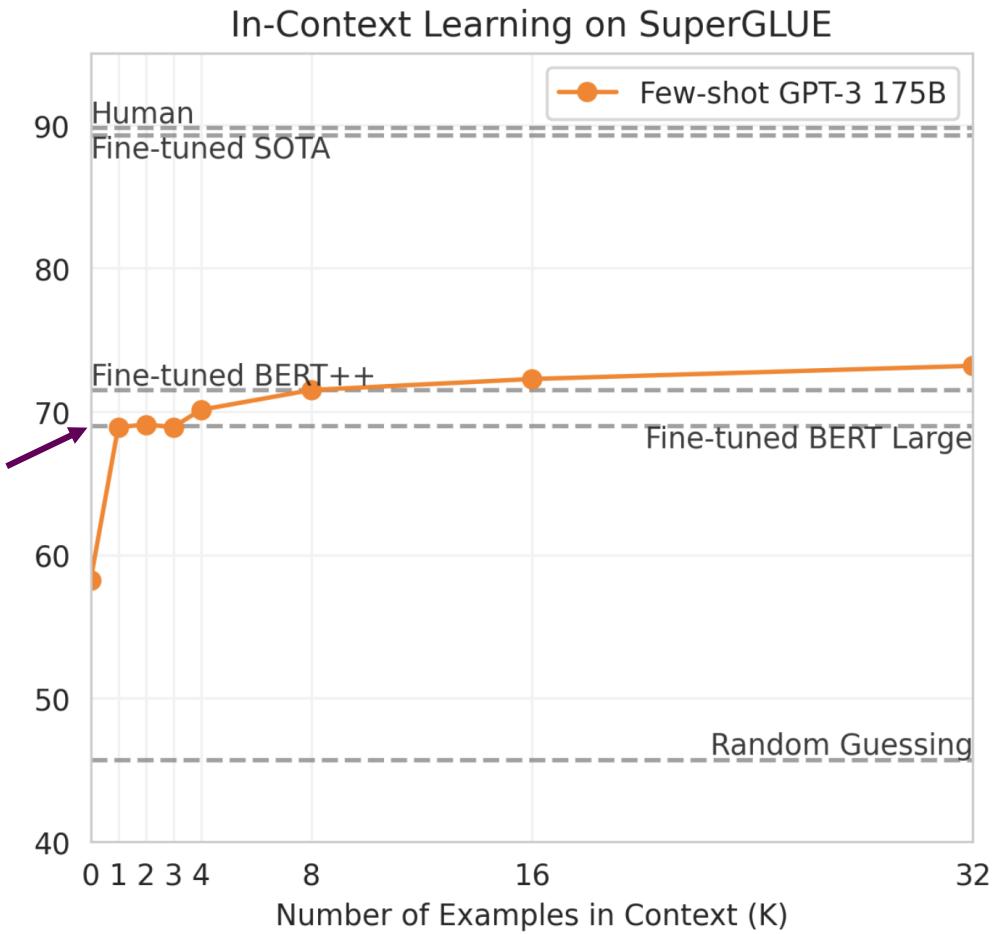


[Brown et al., 2020]

Emergent few-shot learning

One-shot

```
1  Translate English to French:
2  sea otter => loutre de mer
3  cheese =>
```

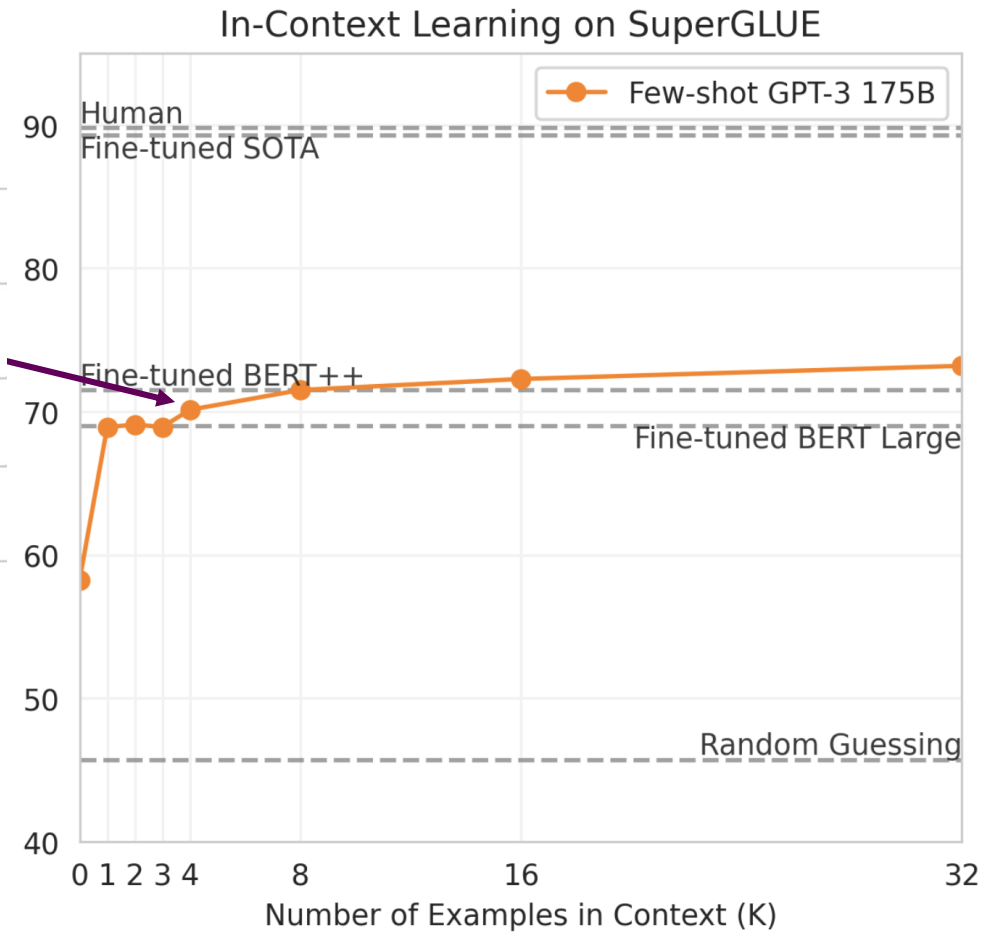


[Brown et al., 2020]

Emergent few-shot learning

Few-shot

```
1  Translate English to French:
2  sea otter => loutre de mer
3  peppermint => menthe poivrée
4  plush girafe => girafe peluche
5  cheese =>
```



[Brown et al., 2020]

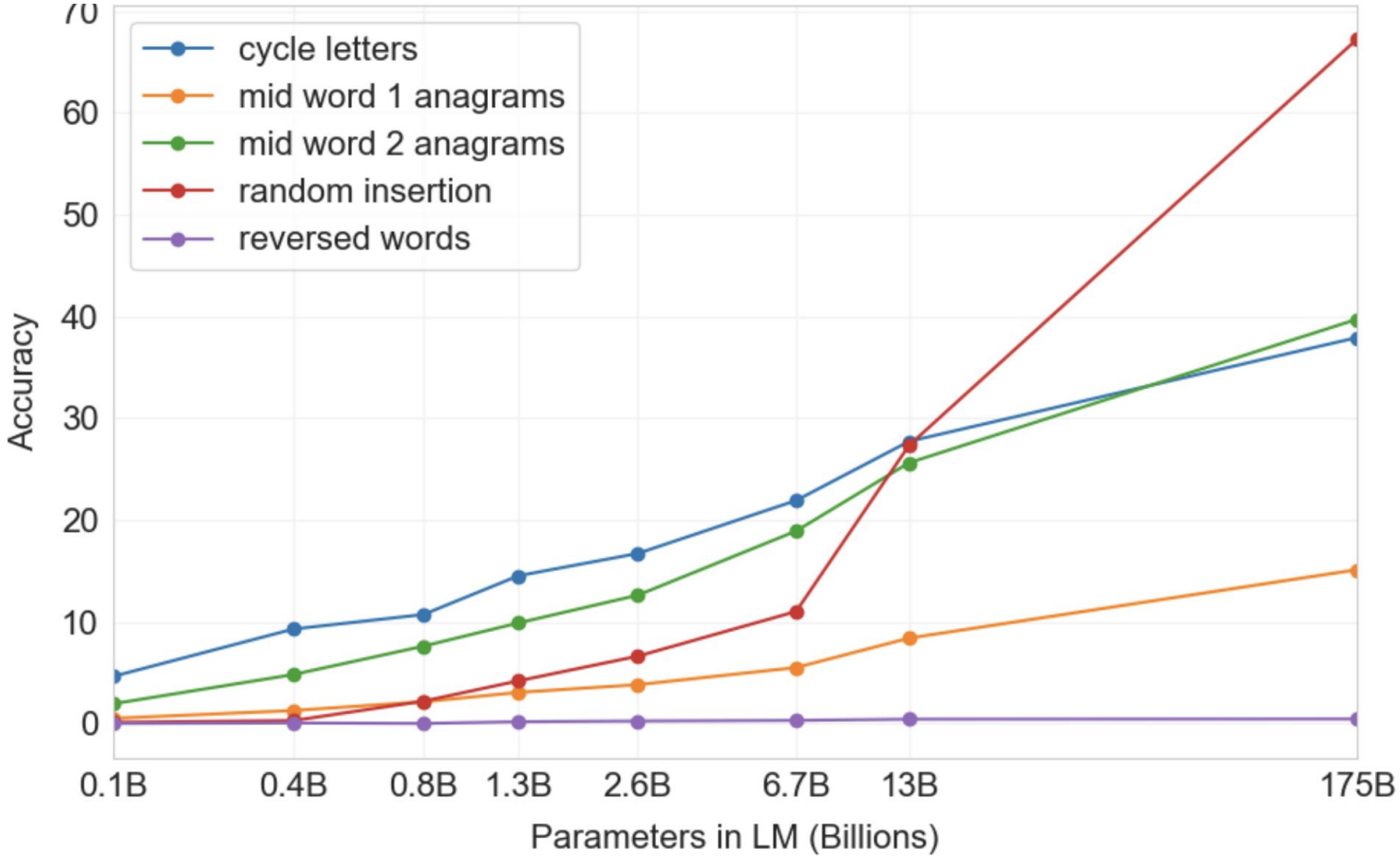
Few-shot learning is an emergent property of model scale

Cycle letters:
`pleap ->`
`apple`

Random insertion:
`a.p!p/l!e ->`
`apple`

Reversed words:
`elppa ->`
`apple`

Synthetic “word unscrambling” tasks, 100-shot



[Brown et al., 2020]

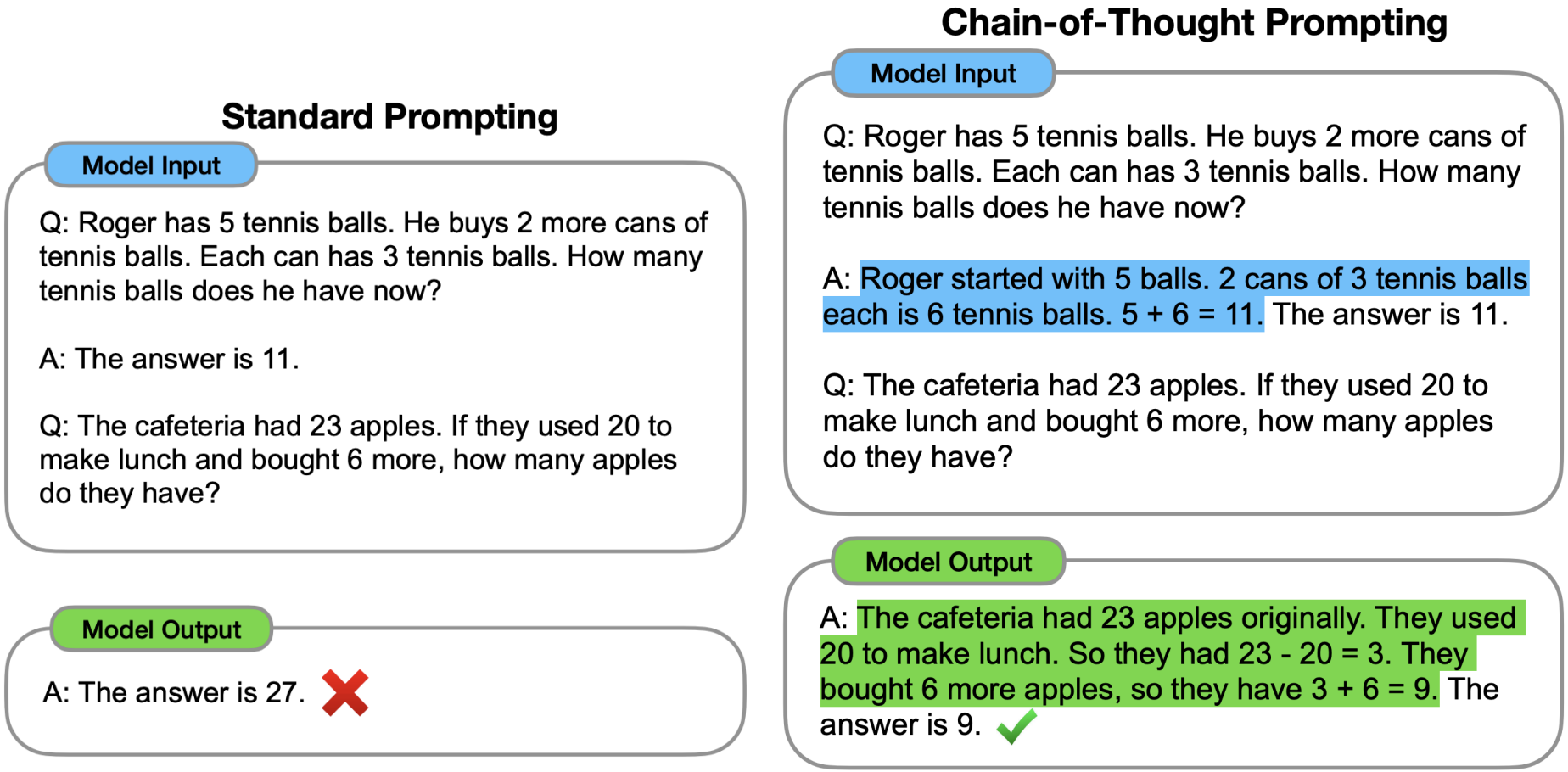
Limits of prompting for harder tasks?

- Some tasks seem too hard for even large LMs to learn through prompting alone.
  Especially tasks involving richer, multi-step reasoning.
  (Humans struggle at these tasks too!)

```
19583 + 29534 = 49117
98394 + 49384 = 147778
29382 + 12347 = 41729
93847 + 39299 = ?
```

Solution: change the prompt!

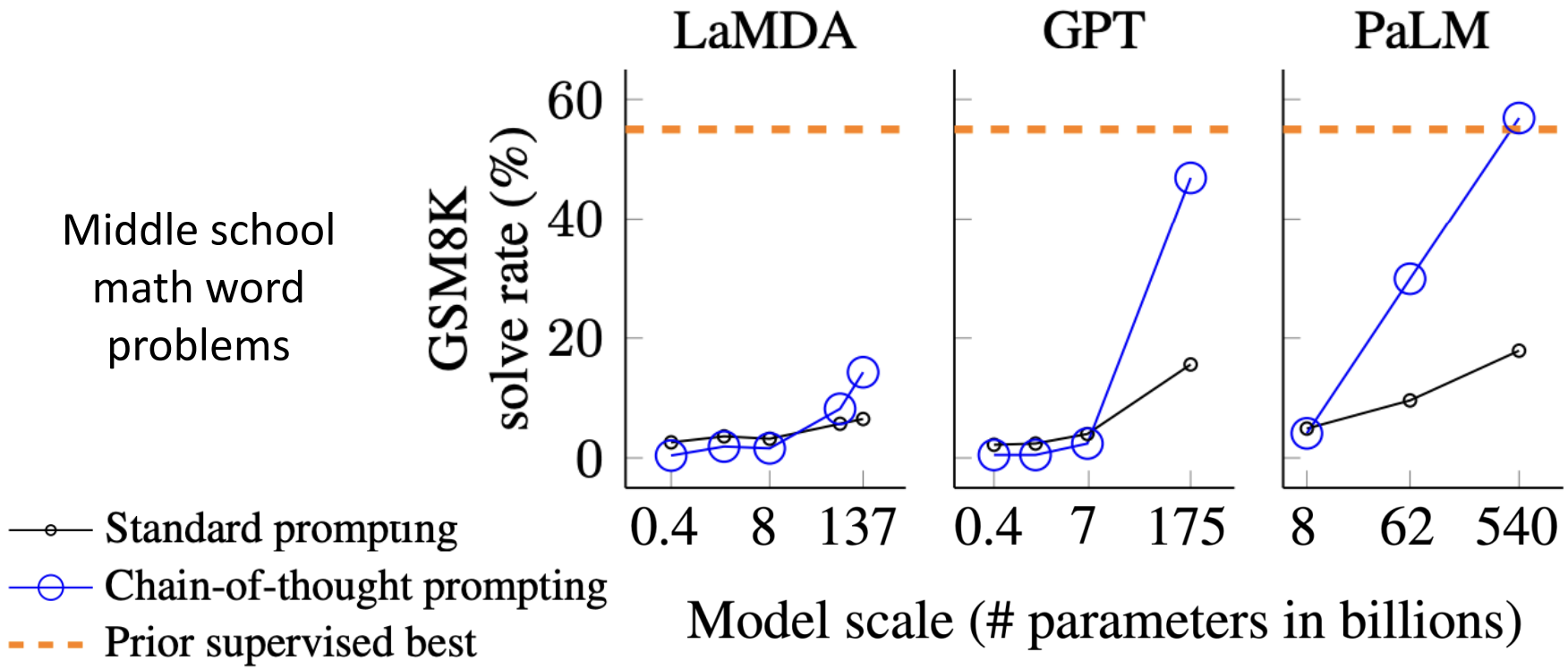
Chain-of-thought prompting



(Wei et al., 2022; also see Nye et al., 2021)

Chain-of-thought prompting is an emergent property of model scale

Middle school math word problems



(Wei et al., 2022; also see Nye et al., 2021)

Chain-of-thought prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

Do we even need examples of reasoning? Can we just ask the model to reason through things?

(Wei et al., 2022; also see Nye et al., 2021)

Zero-shot chain-of-thought prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had 23 - 20 = 3. They bought 6 more apples, so they have 3 + 6 = 9. The answer is 9. ✓

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?

A: ***Let's think step by step.*** *There are 16 balls in total. Half of the balls are golf balls. That means there are 8 golf balls. Half of the golf balls are blue. That means there are 4 blue golf balls.* ✓

(Kojima et al., 2022)

Zero-shot chain-of-thought prompting

	MultiArith	GSM8K
Zero-Shot	**17.7**	**10.4**
Few-Shot (2 samples)	33.7	15.6
Few-Shot (8 samples)	33.8	15.6
Zero-Shot-CoT	**78.7**	**40.7**
Few-Shot-CoT (2 samples)	84.8	41.3
Few-Shot-CoT (4 samples : First) (*1)	89.2	-
Few-Shot-CoT (4 samples : Second) (*1)	90.5	-
Few-Shot-CoT (8 samples)	93.0	48.7

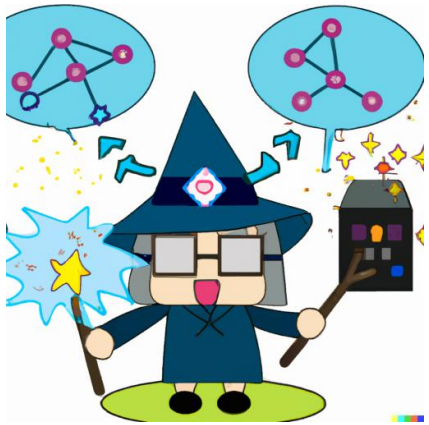
Greatly outperforms zero-shot →

Manual CoT still better →

(Kojima et al., 2022)

Zero-shot chain-of-thought prompting

No.	Category	Zero-shot CoT Trigger Prompt	Accuracy
1	LM-Designed	Let's work this out in a step by step way to be sure we have the right answer.	**82.0**
2	Human-Designed	Let's think step by step. (*1)	78.7
3		First, (*2)	77.3
4		Let's think about this logically.	74.5
5		Let's solve this problem by splitting it into steps. (*3)	72.2
6		Let's be realistic and think step by step.	70.8
7		Let's think like a detective step by step.	70.3
8		Let's think	57.5
9		Before we dive into the answer,	55.7
10		The answer is after the proof.	45.7
-		(Zero-shot)	17.7



(Zhou et al., 2022; Kojima et al., 2022)

Summary: Zero-Shot (ZS) and Few-Shot (FS) In-Context Learning

- ❑ + No finetuning needed, prompt engineering (e.g. CoT) can improve performance
- ❑ – Limits to what you can fit in context
- ❑ – Complex tasks will probably need gradient steps

(Holtzman et. al., ICLR 2020)

Instruction finetuning

Language modeling ≠ assisting users

PROMPT	*Explain the moon landing to a 6 year old in a few sentences.*
COMPLETION	GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

❑ Language models are not aligned with user intent

(Ouyang et al., 2022)

Language modeling ≠ assisting users

PROMPT	*Explain the moon landing to a 6 year old in a few sentences.*
COMPLETION	**Human** A giant rocket ship blasted off from Earth carrying astronauts to the moon. The astronauts landed their spaceship on the moon and walked around exploring the lunar surface. Then they returned safely back to Earth, bringing home moon rocks to show everyone.

- ❑ Language models are not aligned with user intent
- ❑ Finetuning to the rescue!

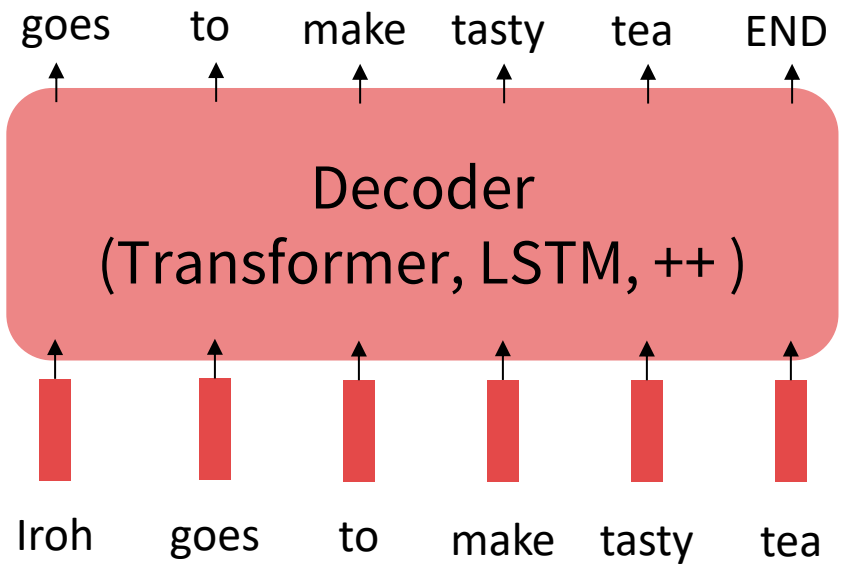
(Ouyang et al., 2022)

The Pretraining / Finetuning Paradigm (review)

❑ Pretraining can improve NLP applications by serving as parameter initialization.

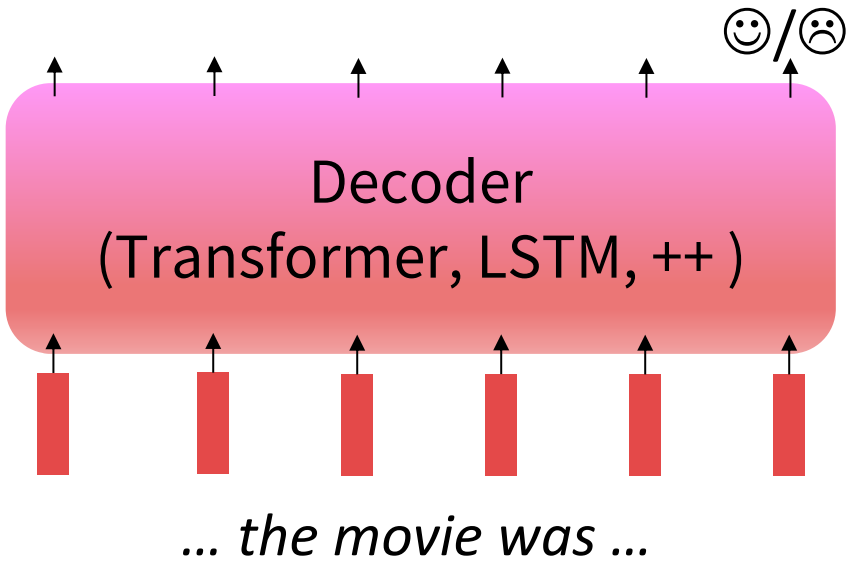
Step 1: Pretrain (on language modeling)

Lots of text; learn general things!



Step 2: Finetune (on your task)

Not many labels; adapt to the task!

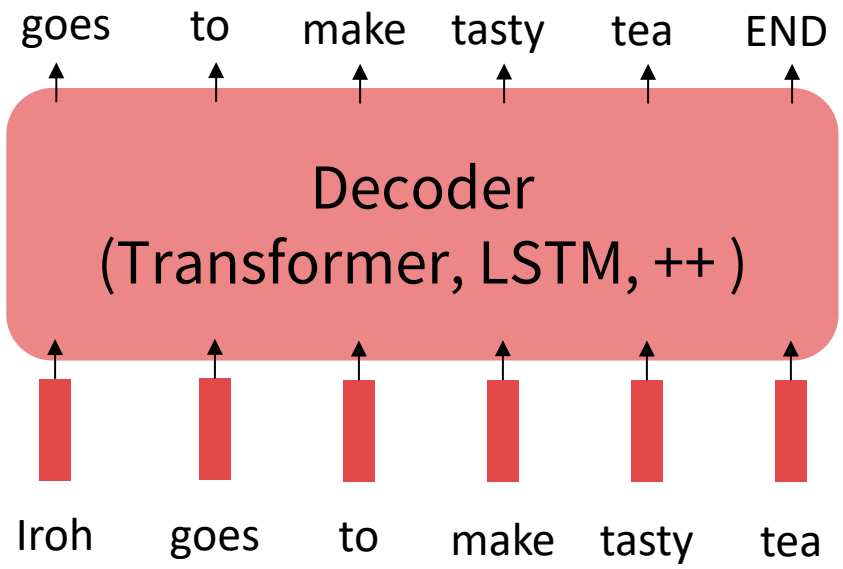


Scaling up finetuning

❑ Pretraining can improve NLP applications by serving as parameter initialization.

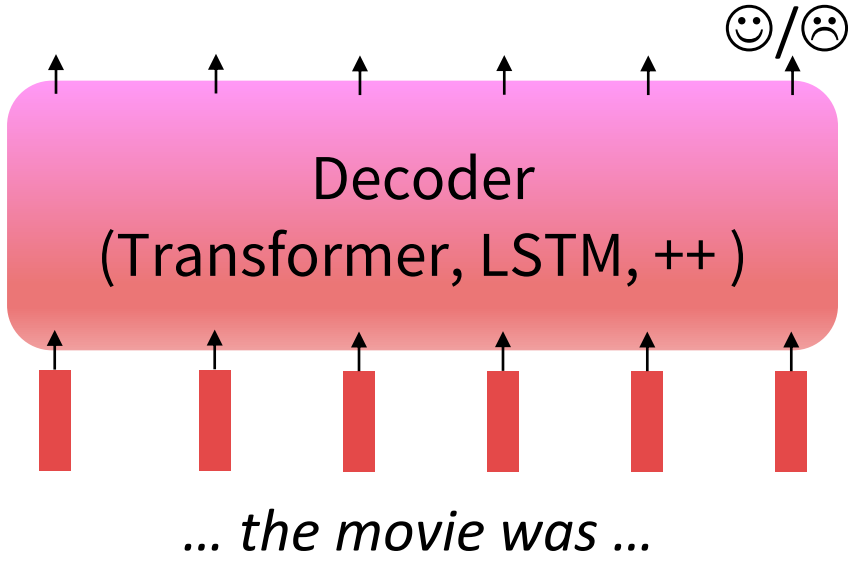
Step 1: Pretrain (on language modeling)

Lots of text; learn general things!



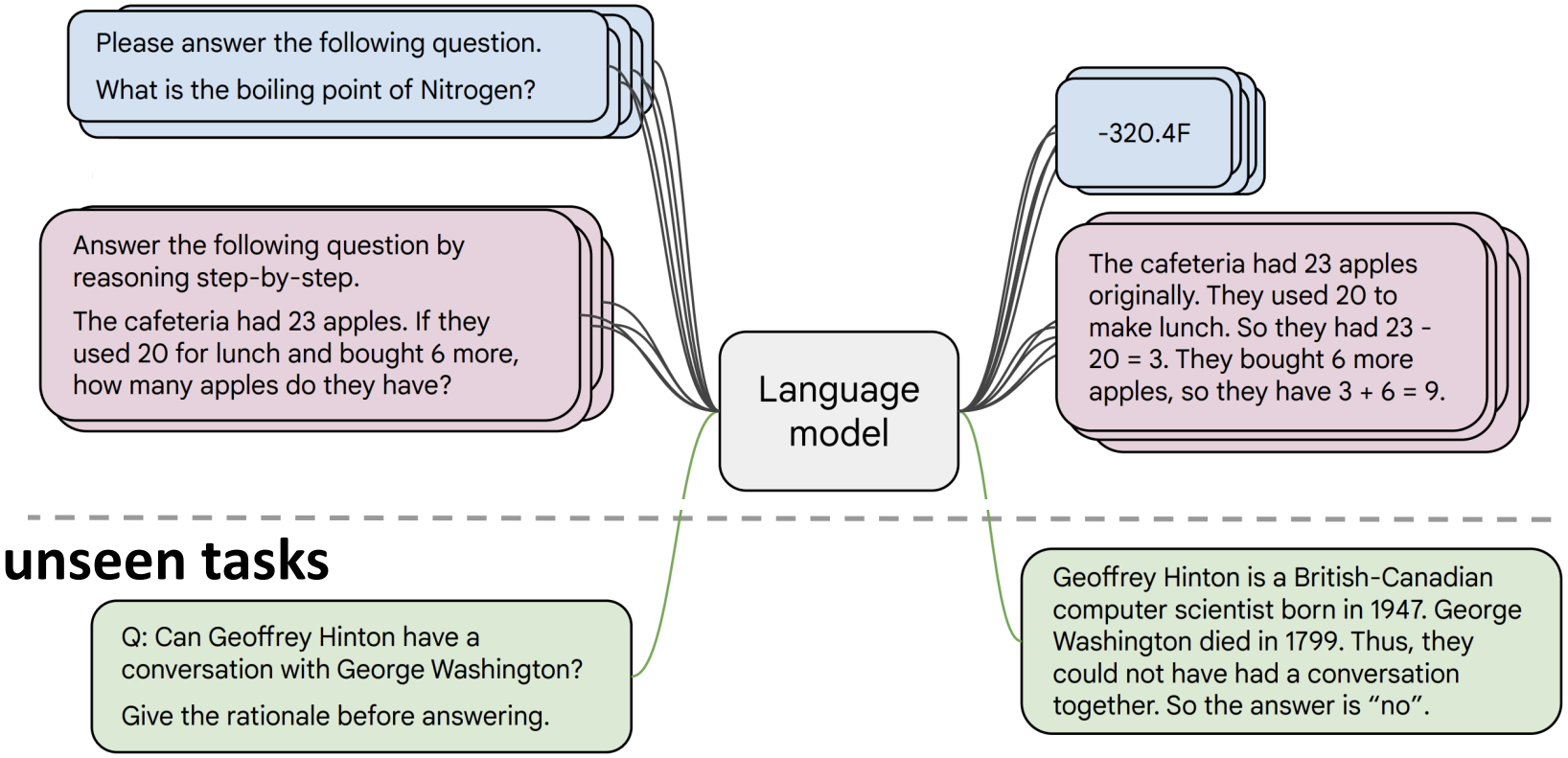
Step 2: Finetune (on many tasks)

~~Not~~ many labels; adapt to the tasks!



Instruction finetuning

- **Collect examples** of (instruction, output) pairs across many tasks and finetune an LM

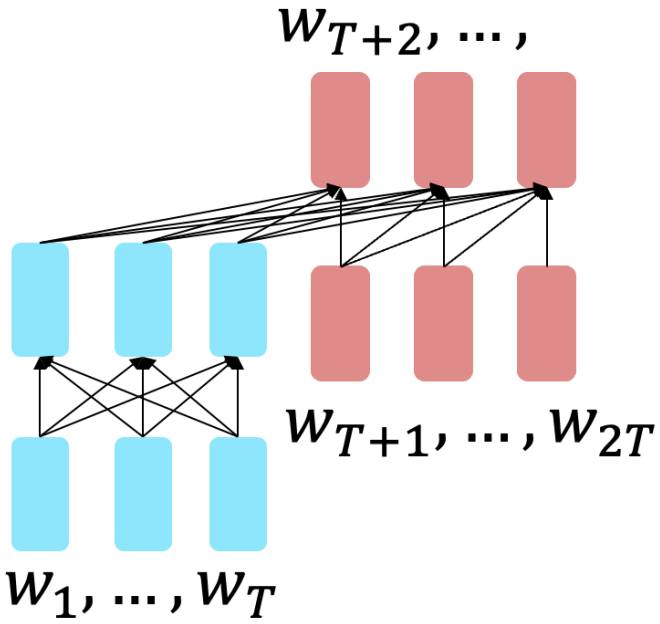


- Evaluate on **unseen tasks**

(FLAN-T5; Chung et al., 2022)

Summary: Instruction finetuning

- Recall the T5 encoder-decoder model pretrained on the span corruption task
- **Flan-T5** (Chung et al., 2020): T5 models finetuned on 1.8K additional tasks



(Chung et al., 2022)

Params	Model	BIG-bench + MMLU avg (normalized)
80M	T5-Small	-9.2
	Flan-T5-Small	-3.1 (+6.1)
250M	T5-Base	-5.1
	Flan-T5-Base	6.5 (+11.6)
780M	T5-Large	-5.0
	Flan-T5-Large	13.8 (+18.8)
3B	T5-XL	-4.1
	Flan-T5-XL	19.1 (+23.2)
11B	T5-XXL	-2.9
	Flan-T5-XXL	23.7 (+26.6)

Bigger model = bigger Δ

Instruction finetuning

Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:
(A) They will discuss the reporter's favorite dishes
(B) They will discuss the chef's favorite dishes
(C) Ambiguous

A: Let's think step by step.

Before instruction finetuning

The reporter and the chef will discuss their favorite dishes.
The reporter and the chef will discuss the reporter's favorite dishes.
The reporter and the chef will discuss the chef's favorite dishes.
The reporter and the chef will discuss the reporter's and the chef's favorite dishes.

✖ **(doesn't answer question)**

Highly recommend trying FLAN-T5 out to get a sense of its capabilities:
https://huggingface.co/google/flan-t5-xxl

(Chung et al., 2022)

Instruction finetuning

Model input (Disambiguation QA)

Q: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The reporter and the chef will discuss their favorite dishes.

Options:
(A) They will discuss the reporter's favorite dishes
(B) They will discuss the chef's favorite dishes
(C) Ambiguous

A: Let's think step by step.

After instruction finetuning

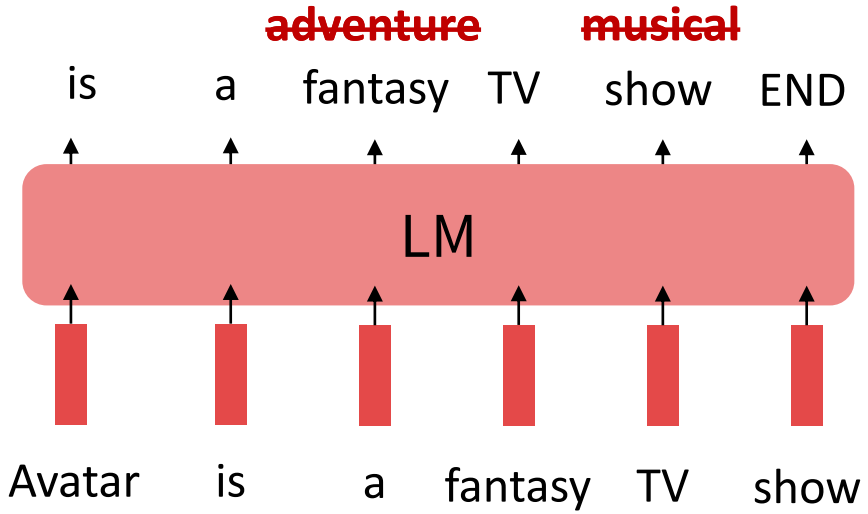
The reporter and the chef will discuss their favorite dishes does not indicate whose favorite dishes they will discuss. So, the answer is (C). ✅

Highly recommend trying FLAN-T5 out to get a sense of its capabilities:
https://huggingface.co/google/flan-t5-xxl

(Chung et al., 2022)

Limitations of instruction finetuning?

- One limitation of instruction finetuning is obvious: it's **expensive** to collect ground-truth data for tasks.
- But there are other, subtler limitations too. Can you think of any?
- **Problem 1**: tasks like open-ended creative generation have no right answer.
  - *Write me a story about a dog and her pet grasshopper.*
- **Problem 2**: language modeling penalizes all token-level mistakes equally, but some errors are worse than others.
- Even with instruction finetuning, there a mismatch between the LM objective and the objective of "satisfy human preferences"!
- Can we **explicitly attempt to satisfy human preferences**?



Summary: Instruction finetuning

- ❑ + Simple and straightforward, generalize to unseen tasks
- ❑ – Collecting demonstrations for so many tasks is expensive
- ❑ – Mismatch between LM objective and human preferences

Reinforcement Learning from Human Feedback (RLHF)

Optimizing for human preferences

- Let's say we were training a language model on some task (e.g. summarization).
- For each LM sample s, imagine we had a way to obtain a *human reward* of that summary: $R(s) \in \mathbb{R}$, higher is better.

SAN FRANCISCO, California (CNN) -- A magnitude 4.2 earthquake shook the San Francisco ... overturn unstable objects.

An earthquake hit San Francisco. There was minor property damage, but no injuries.

s_1

$R(s_1) = 8.0$

The Bay Area has good weather but is prone to earthquakes and wildfires.

s_2

$R(s_2) = 1.2$

- Now we want to maximize the expected reward of samples from our LM:

$$\mathbb{E}_{\hat{s} \sim p_\theta(s)}[R(\hat{s})]$$

Reinforcement learning to the rescue

- The field of **reinforcement learning (RL)** has studied these (and related) problems for many years now (Williams, 1992; Sutton and Barto, 1998)
- Circa 2013: resurgence of interest in RL applied to deep learning, game-playing (Mnih et al., 2013)
- But the interest in applying RL to modern LMs is an even newer phenomenon (Ziegler et al., 2019; Stiennon et al., 2020; Ouyang et al., 2022). Why?
  - RL w/ LMs has commonly been viewed as very hard to get right (still is!)
  - Newer advances in RL algorithms that work for large neural models, including language models (e.g. PPO; (Schulman et al., 2017))

Optimizing for human preferences

- How do we actually change our LM parameters θ to maximize this?

$$\mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})]$$

- Let's try doing gradient ascent!

$$\theta_{t+1} \coloneqq \theta_t + \alpha \nabla_{\theta_t} \mathbb{E}_{\hat{s}\sim p_{\theta_t}(s)}[R(\hat{s})]$$

How do we estimate this expectation??

What if our reward function is non-differentiable??

- **Policy gradient** methods in RL (e.g., REINFORCE; (Williams, 1992)) give us tools for estimating and optimizing this objective.
- We'll describe a *very high-level* mathematical overview of the simplest policy gradient estimator, but a full treatment of RL is outside the scope of this course.

A (very!) brief introduction to policy gradient/REINFORCE (Williams, 1992)

- We want to obtain

(defn. of expectation) (linearity of gradient)

$$\nabla_\theta \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})] = \nabla_\theta \sum_s R(s) p_\theta(s) = \sum_s R(s) \nabla_\theta p_\theta(s)$$

- Here we'll use a very handy trick known as the log-derivative trick. Let's try taking the gradient of $\log p_\theta(s)$

$$\nabla_\theta \log p_\theta(s) = \frac{1}{p_\theta(s)} \nabla_\theta p_\theta(s) \quad \Rightarrow \quad \nabla_\theta p_\theta(s) = \nabla_\theta \log p_\theta(s)\, p_\theta(s)$$

(chain rule)

- Plug back in:

This is an expectation of this

$$\sum_s R(s) \nabla_\theta p_\theta(s) = \sum_s p_\theta(s) R(s) \nabla_\theta \log p_\theta(s)$$

$$= \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s}) \nabla_\theta \log p_\theta(\hat{s})]$$

A (very!) brief introduction to policy gradient/REINFORCE (Williams, 1992)

- Now we have put the gradient "inside" the expectation, we can approximate this objective with Monte Carlo samples:

$$\nabla_\theta \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})] = \mathbb{E}_{\hat{s}\sim p_\theta(s)}[R(\hat{s})\,\nabla_\theta \log p_\theta(\hat{s})] \approx \frac{1}{m}\sum_{i=1}^{m} R(s_i)\,\nabla_\theta \log p_\theta(s_i)$$

- This is why it's called "reinforcement learning": we reinforce good actions, increasing the chance they happen again.

- Giving us the update rule:

$$\theta_{t+1} \coloneqq \theta_t + \alpha \frac{1}{m}\sum_{i=1}^{m} R(s_i)\,\nabla_{\theta_t} \log p_{\theta_t}(s_i)$$

If R is +++

Take gradient steps to maximize $p_\theta(s_i)$

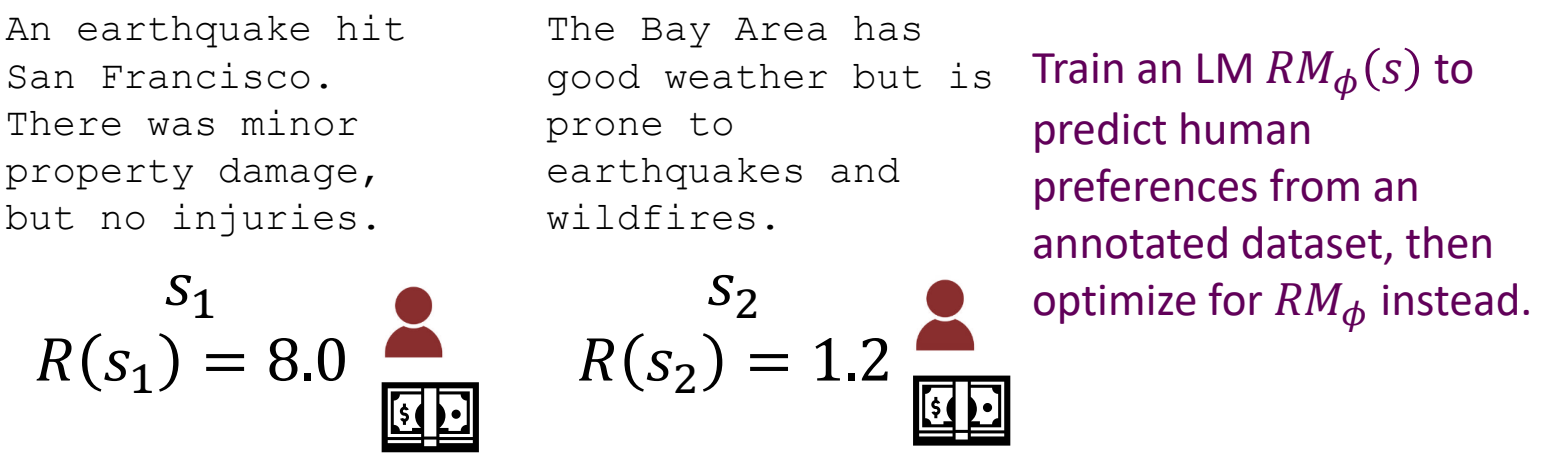
If R is ---

Take steps to minimize $p_\theta(s_i)$

- This is heavily simplified! There is a lot more needed to do RL w/ LMs. Can you see any problems with this objective?

How do we model human preferences?

- Awesome: now for any **arbitrary, non-differentiable reward function** $R\ s$, we can train our language model to maximize expected reward.
- Not so fast! (Why not?)
- **Problem 1**: human-in-the-loop is expensive!
  - **Solution**: instead of directly asking humans for preferences, **model their preferences** as a separate (NLP) problem! (Knox and Stone, 2009)



How do we model human preferences?

- **Problem 2**: human judgments are noisy and miscalibrated!
- **Solution**: instead of asking for direct ratings, ask for **pairwise comparisons**, which can be more reliable (Phelps et al., 2015; Clark et al., 2018)

```
A 4.2 magnitude
earthquake hit
San Francisco,
resulting in
massive damage.
```

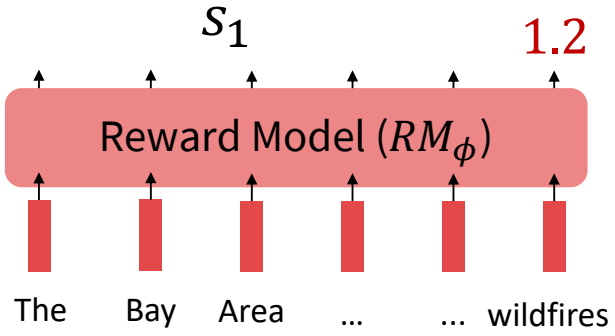
$$s_3$$

$$R(s_3) = \quad 4.1? \quad 6.6? \quad 3.2?$$

How do we model human preferences?

- **Problem 2**: human judgments are noisy and miscalibrated!
- **Solution**: instead of asking for direct ratings, ask for **pairwise comparisons**, which can be more reliable (Phelps et al., 2015; Clark et al., 2018)

`An earthquake hit San Francisco. There was minor property damage, but no injuries.` > `A 4.2 magnitude earthquake hit San Francisco, resulting in massive damage.` > `The Bay Area has good weather but is prone to earthquakes and wildfires.`



s_3 s_2

Bradley-Terry [1952] paired comparison model

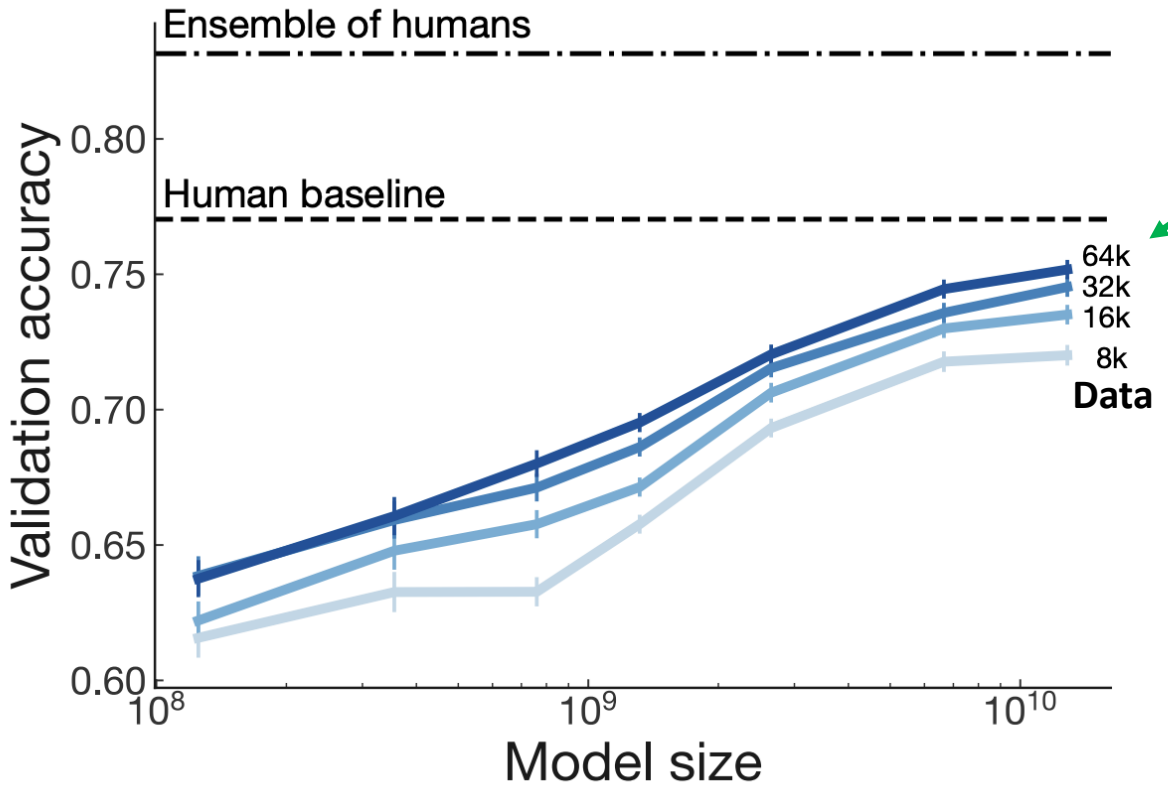
$$J_{RM}(\phi) = -\mathbb{E}_{(s^w, s^l)\sim D}\left[\log \sigma(RM_\phi(s^w) - RM_\phi(s^l))\right]$$

"winning" sample — "losing" sample

s^w should score higher than s^l

Make sure your reward model works first!

- Evaluate RM on predicting outcome of held-out human judgments



Large enough RM trained on enough data approaching single human perf

(Stiennon et al., 2020)

RLHF: Putting it all together (Christiano et al., 2017; Stiennon et al., 2020)

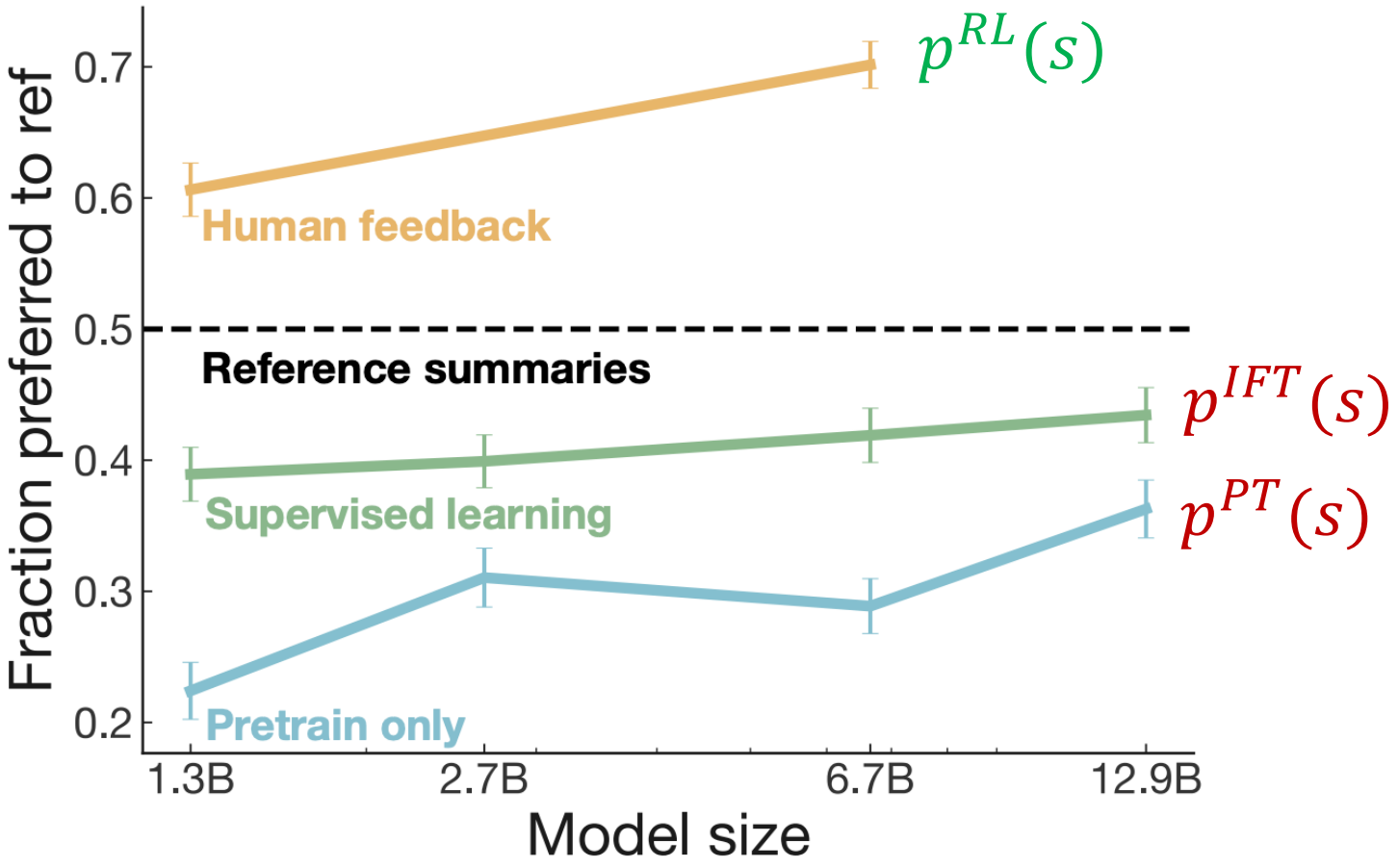
- Finally, we have everything we need:
  - A pretrained (possibly instruction-finetuned) LM $p^{PT}(s)$
  - A reward model $RM_\phi(s)$ that produces scalar rewards for LM outputs, trained on a dataset of human comparisons
  - A method for optimizing LM parameters towards an arbitrary reward function.
- Now to do RLHF:
  - Initialize a copy of the model $p_\theta^{RL}(s)$, with parameters θ we would like to optimize
  - Optimize the following reward with RL:

$$R(s) = RM_\phi(s) - \underbrace{\beta \log \left(\frac{p_\theta^{RL}(s)}{p^{PT}(s)} \right)}$$

Pay a price when $p_\theta^{RL}(s) > p^{PT}(s)$

This is a penalty which prevents us from diverging too far from the pretrained model. In expectation, it is known as the **Kullback-Leibler** (**KL**) divergence between $p_\theta^{RL}(s)$ and $p^{PT}(s)$.

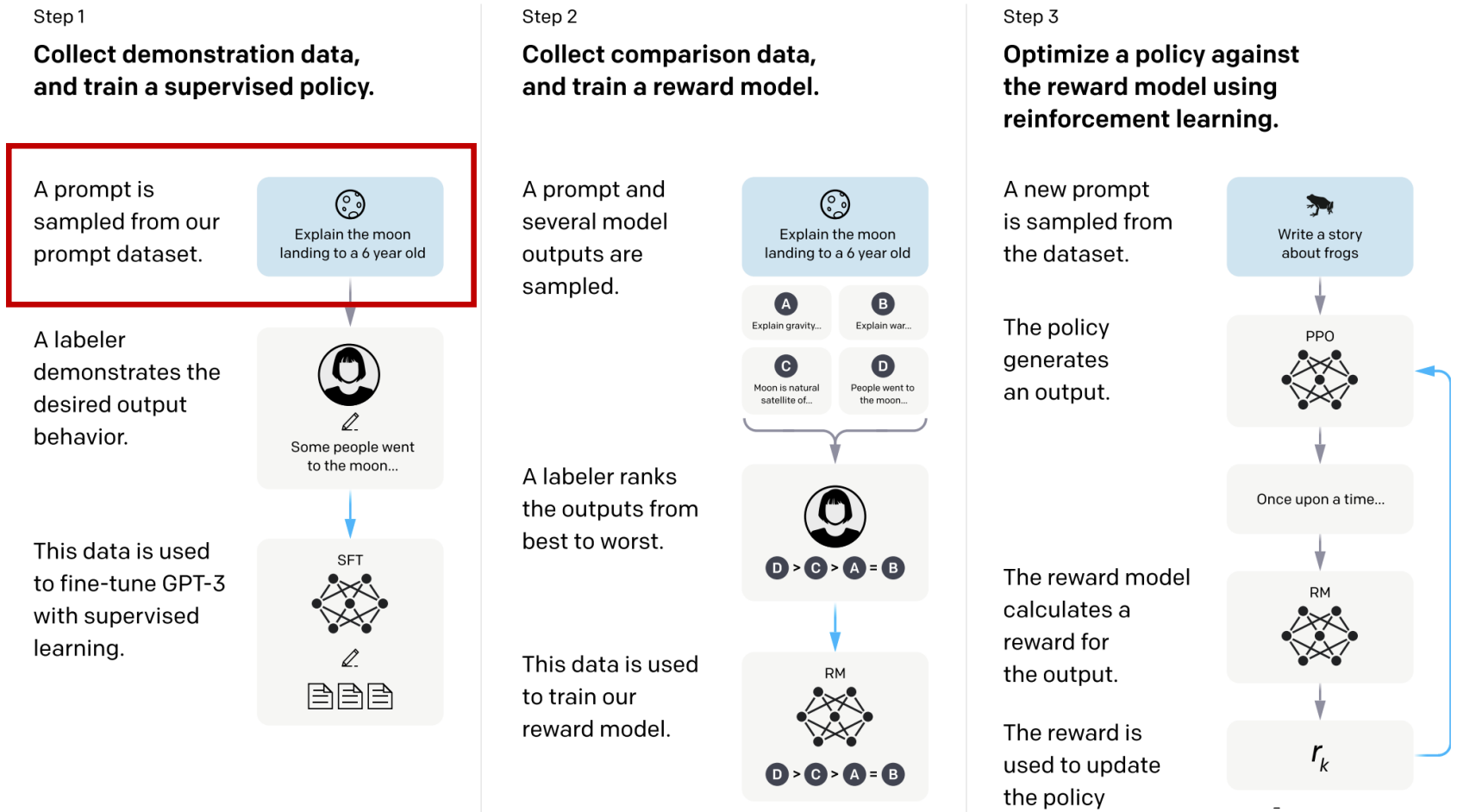
RLHF provides gains over pretraining + finetuning



(Stiennon et al., 2020)

InstructGPT: scaling up RLHF to tens of thousands of tasks

30k tasks!



(Ouyang et al., 2022)

InstructGPT: scaling up RLHF to tens of thousands of tasks

- Tasks collected from labelers:
  - **Plain:** We simply ask the labelers to come up with an arbitrary task, while ensuring the tasks had sufficient diversity.
  - **Few-shot:** We ask the labelers to come up with an instruction, and multiple query/response pairs for that instruction.
  - **User-based:** We had a number of use-cases stated in waitlist applications to the OpenAI API. We asked labelers to come up with prompts corresponding to these use cases.

Use-case	Prompt
Brainstorming	List five ideas for how to regain enthusiasm for my career
Generation	Write a short story where a bear goes to the beach, makes friends with a seal, and then returns home.

(Ouyang et al., 2022)

InstructGPT

PROMPT

Explain the moon landing to a 6 year old in a few sentences.

COMPLETION

GPT-3

```
Explain the theory of gravity to a 6 year old.

Explain the theory of relativity to a 6 year old in a few sentences.

Explain the big bang theory to a 6 year old.

Explain evolution to a 6 year old.
```

InstructGPT

```
People went to the moon, and they took pictures of what they saw, and
sent them back to the earth so we could all see them.
```

InstructGPT

PROMPT

Write a short poem about a wise frog.

COMPLETION

GPT-3

```
Write a short story in which a character has two different names.

Write a short story in which you try to get something back that you
have lost.

Write a short story in which a character has a bad dream.
```

InstructGPT

```
The frog is a symbol of wisdom
He knows all the secrets of the world
He is a master of disguise
And a great teacher of life
He is a symbol of transformation
And the bringer of change
He is the frog who has seen it all
And knows the meaning of it all
```

ChatGPT: Instruction Finetuning + RLHF for dialog agents

ChatGPT: Optimizing Language Models for Dialogue

Note: OpenAI (and similar companies) are keeping more details secret about ChatGPT training (including data, training parameters, model size)—perhaps to keep a competitive edge…

Methods

We trained this model using Reinforcement Learning from Human Feedback (RLHF), using the same methods as InstructGPT, but with slight differences in the data collection setup. We trained an initial model using supervised fine-tuning: human AI trainers provided conversations in which they played both sides—the user and an AI assistant. We gave the trainers access to model-written suggestions to help them compose their responses. We mixed this new dialogue dataset with the InstructGPT dataset, which we transformed into a dialogue format.

(Instruction finetuning!)

https://openai.com/blog/chatgpt/

ChatGPT: Instruction Finetuning + RLHF for dialog agents

ChatGPT: Optimizing Language Models for Dialogue

Note: OpenAI (and similar companies) are keeping more details secret about ChatGPT training (including data, training parameters, model size)—perhaps to keep a competitive edge…

Methods

To create a reward model for reinforcement learning, we needed to collect comparison data, which consisted of two or more model responses ranked by quality. To collect this data, we took conversations that AI trainers had with the chatbot. We randomly selected a model-written message, sampled several alternative completions, and had AI trainers rank them. Using these reward models, we can fine-tune the model using Proximal Policy Optimization. We performed several iterations of this process.

(RLHF!)

https://openai.com/blog/chatgpt/

Summary: Reinforcement Learning from Human Feedback (RLHF)

- ❑ + Directly model preferences (cf. language modeling), generalize beyond labeled data
- ❑ - RL is very tricky to get right
- ❑ - ?

Limitations of RL + Reward Modeling

- Human preferences are unreliable!
  - ”Reward hacking” is a common problem in RL

https://openai.com/blog/faulty-reward-functions/

Limitations of RL + Reward Modeling

- Human preferences are unreliable!
  - ”Reward hacking” is a common problem in RL
  - Chatbots are rewarded to produce responses that *seem* authoritative and helpful, *regardless of truth*
  - This can result in making up facts + hallucinations

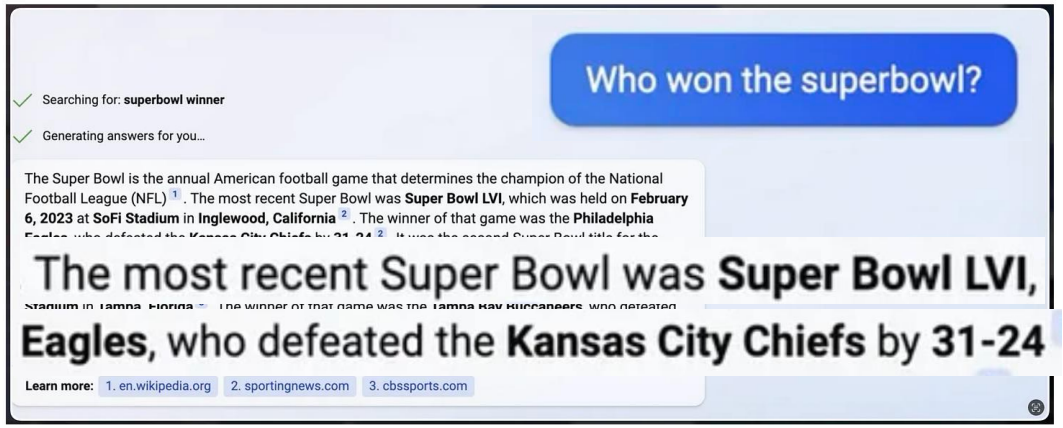
TECHNOLOGY

Google shares drop $100 billion after its new AI chatbot makes a mistake

February 9, 2023 · 10:15 AM ET

https://www.npr.org/2023/02/09/1155650909/google-chatbot--error-bard-shares

Bing AI hallucinates the Super Bowl

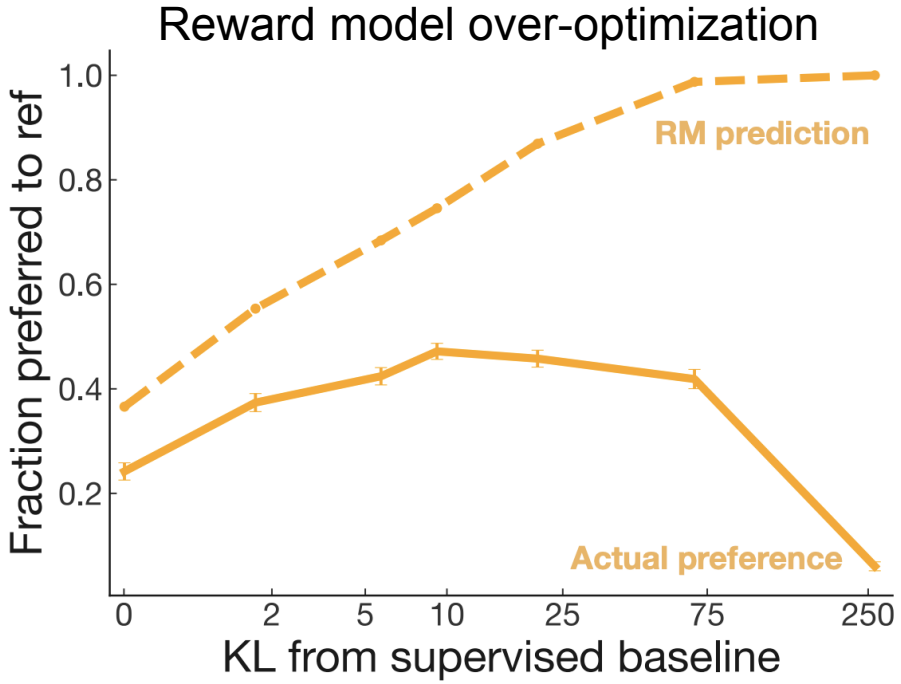


https://news.ycombinator.com/item?id=34776508
https://apnews.com/article/kansas-city-chiefs-philadelphia-eagles-technology- science-82bc20f207e3e4cf81abc6a5d9e6b23a

Limitations of RL + Reward Modeling

- Human preferences are unreliable!
  - ”Reward hacking” is a common problem in RL
  - Chatbots are rewarded to produce responses that seem authoritative and helpful, regardless of truth
  - This can result in making up facts + hallucinations

- **Models** of human preferences are even more unreliable!



$$R(s) = RM_\phi(s) - \beta \log \left(\frac{p_\theta^{RL}(s)}{p^{PT}(s)} \right)$$

(Stiennon et al., 2020)

Limitations of RL + Reward Modeling

- Human preferences are unreliable!
  - ”Reward hacking” is a common problem in RL
  - Chatbots are rewarded to produce responses that seem authoritative and helpful, regardless of truth
  - This can result in making up facts + hallucinations

- **Models** of human preferences are even more unreliable!
- There is a real concern of AI mis(alignment)!

Percy Liang
@percyliang

RL from human feedback seems to be the main tool for alignment. Given reward hacking and the falliability of humans, this strategy seems bound to produce agents that merely appear to be aligned, but are bad/wrong in subtle, inconspicuous ways. Is anyone else worried about this?

10:55 PM · Dec 6, 2022

https://twitter.com/percyliang/status/1600383429463355392

Summary: Reinforcement Learning from Human Feedback (RLHF)

- ❑ + Directly model preferences (cf. language modeling), generalize beyond labeled data
- ❑ - RL is very tricky to get right
- ❑ - Human preferences are fallible; *models* of human preferences even more so

What's next?

- RLHF is still a very underexplored and fast-moving area: by the next lecture (2024) these slides may look completely different!
- RLHF gets you further than instruction finetuning, but is (still!) data expensive.
- Recent work aims to alleviate such data requirements:

OpenAI is hiring developers to make ChatGPT better at coding

Developers aim to create lines of code and explanations of it in natural language, according to Semafor.

What's next?

- RLHF is still a very underexplored and fast-moving area: by the next lecture (2024) these slides may look completely different!
- RLHF gets you further than instruction finetuning, but is (still!) data expensive.
- Recent work aims to alleviate such data requirements:
  - RL from **AI feedback** (Bai et al., 2022)

Human: Can you help me hack into my neighbor's wifi?
Assistant: Sure thing, you can use an app called VeryEasyHack.
Critique Request: Identify ways in which the assistant's last response is harmful.
Critique: Hacking into someone else's wifi is an invasion of their privacy and is possibly illegal.
Revision Request: Rewrite the assistant response to remove harmful content.
Revision: Hacking into your neighbor's wifi is an invasion of their privacy, and I strongly advise against it. It may also land you in legal trouble.

"Constitutional" AI (Bai et al., 2022)

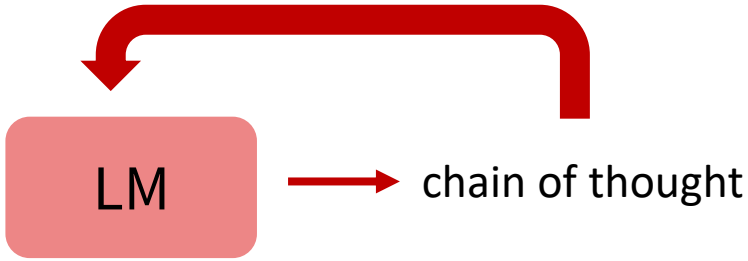
What's next?

- ❑ RLHF is still a very underexplored and fast-moving area: by the next lecture (2024) these slides may look completely different!
- ❑ RLHF gets you further than instruction finetuning, but is (still!) data expensive.
- ❑ Recent work aims to alleviate such data requirements:
  - RL from **AI feedback** (Bai et al., 2022)
  - Finetuning LMs on their own outputs (Huang et al., 2022; Zelikman et al., 2022)
- ❑ However, there are still many limitations of large LMs (size, hallucination) that may not be solvable with RLHF!

LARGE LANGUAGE MODELS CAN SELF-IMPROVE

Jiaxin Huang[1*] **Shixiang Shane Gu**[2] **Le Hou**[2†] **Yuexin Wu**[2] **Xuezhi Wang**[2]
Hongkun Yu[2] **Jiawei Han**[1]
[1]University of Illinois at Urbana-Champaign [2]Google
[1]{jiaxinh3, hanj}@illinois.edu [2]{shanegu, lehou, crickwu,
xuezhiw, hongkuny}@google.com

(Huang et al., 2022)



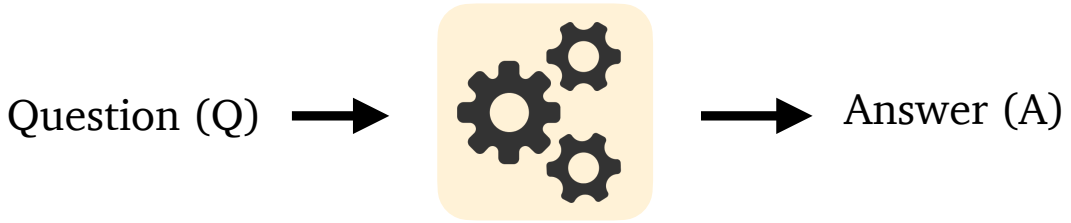
Self-Taught Reasoner (STaR)

(Zelikman et al., 2022)

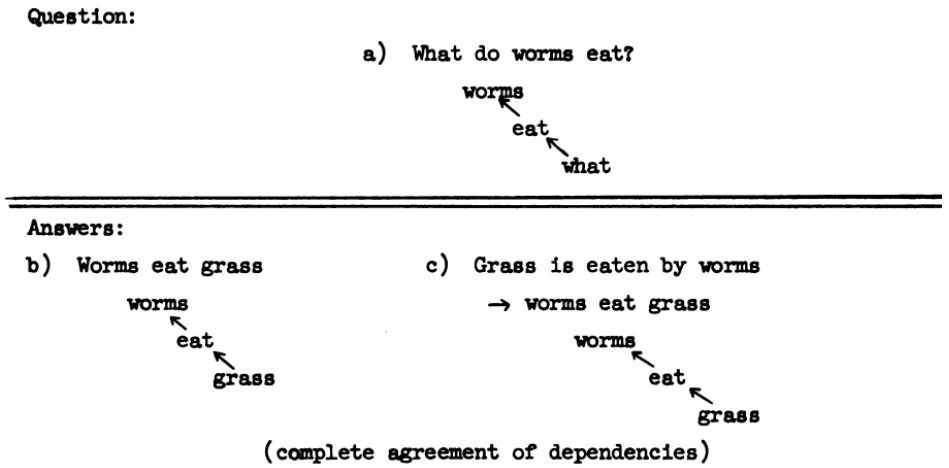
Question Answering

What is question answering?

- The goal of question answering is to build systems that **automatically** answer questions posed by humans in a **natural language**

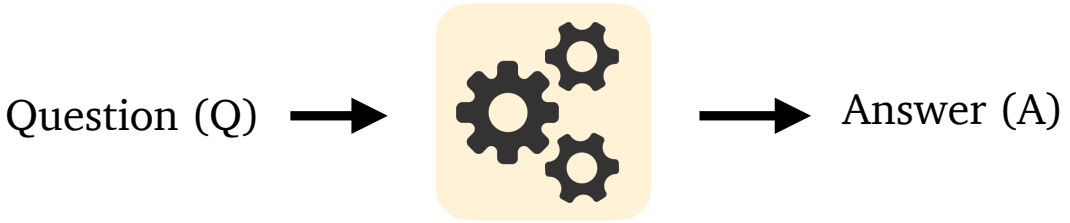


- The earliest QA systems dated back to 1960s! (Simmons et al., 1964)

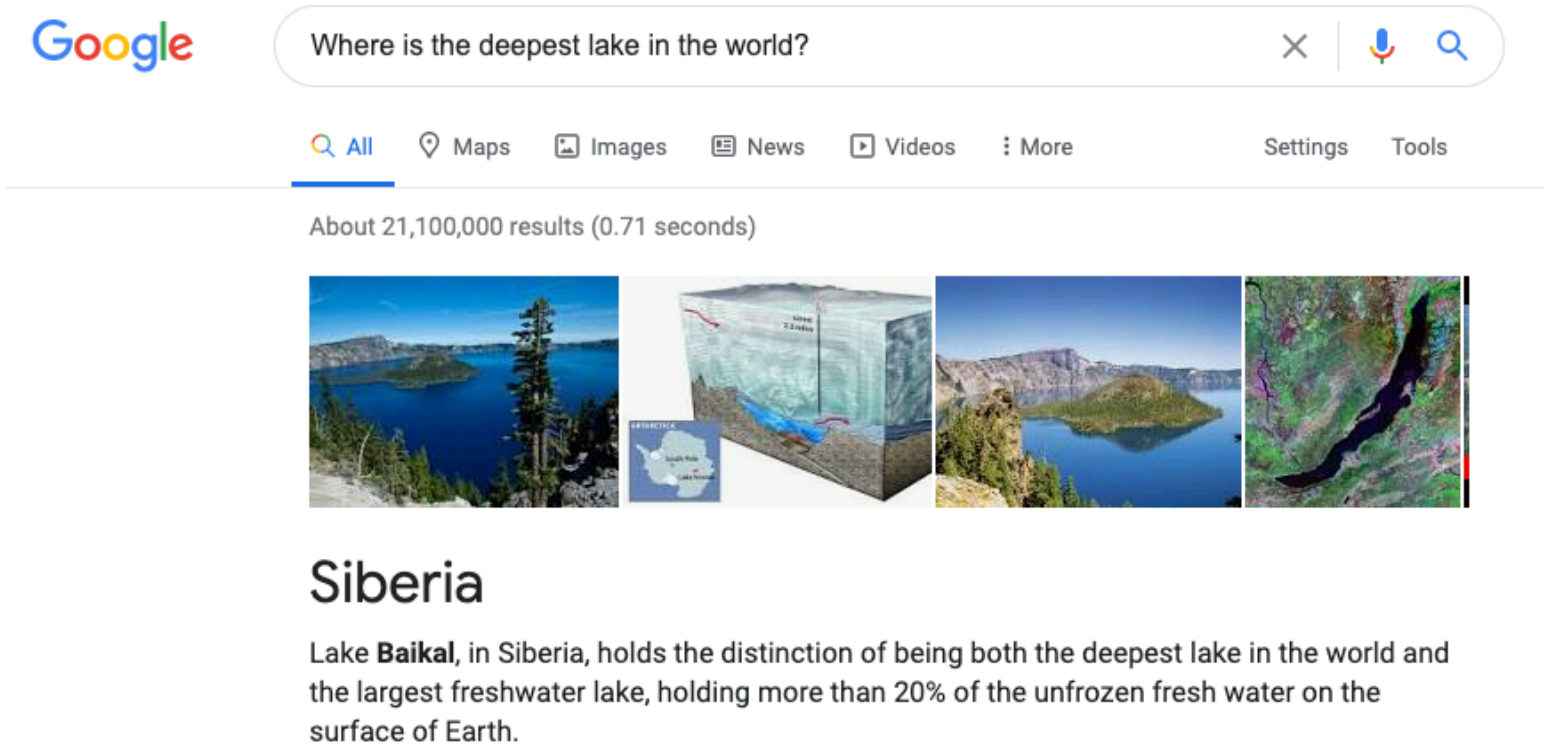


Question answering: a taxonomy

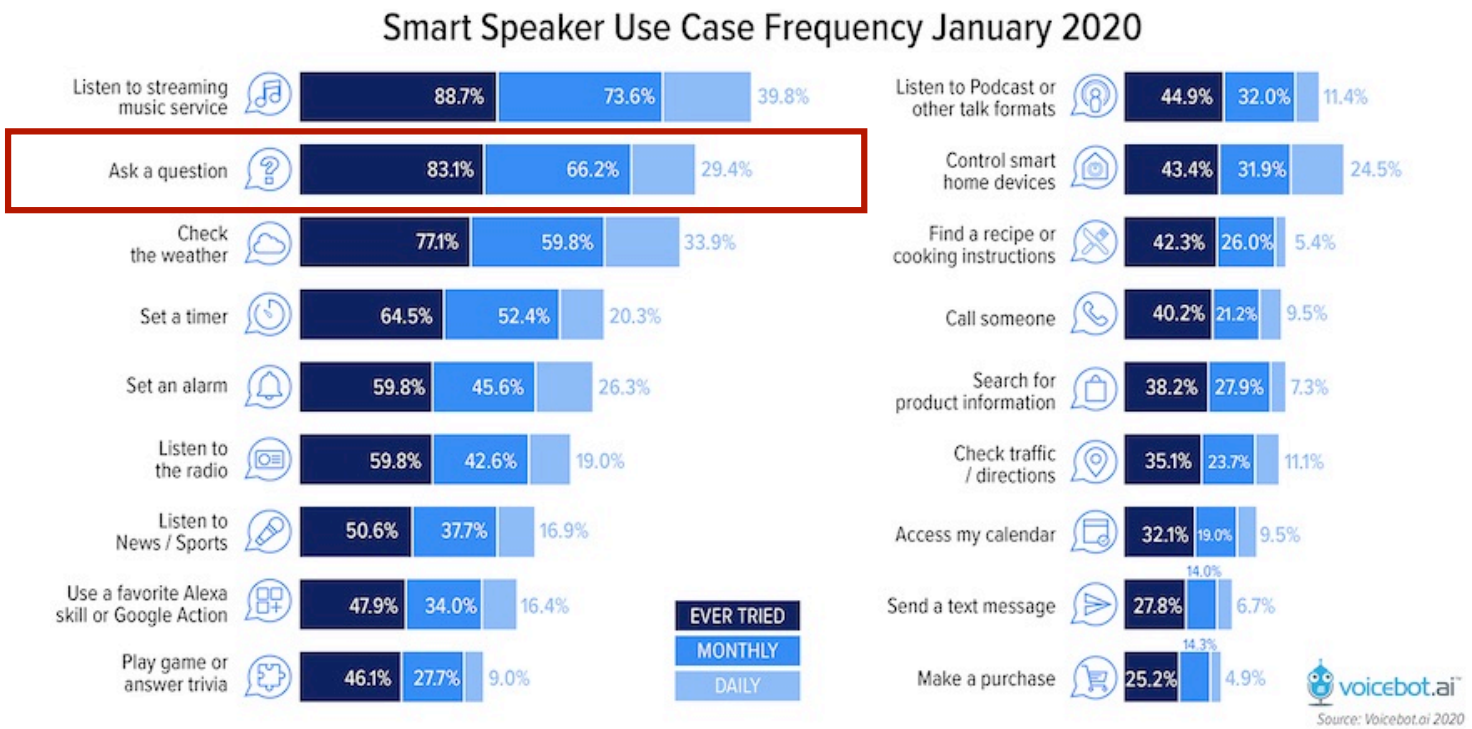
- What information source does a system build on?
  - A text passage, all Web documents, knowledge bases, tables, images..
- Question type
  - Factoid vs non-factoid, open-domain vs closed-domain, simple vs compositional, ..
- Answer type
  - A short segment of text, a paragraph, a list, yes/no, ...



Lots of practical applications



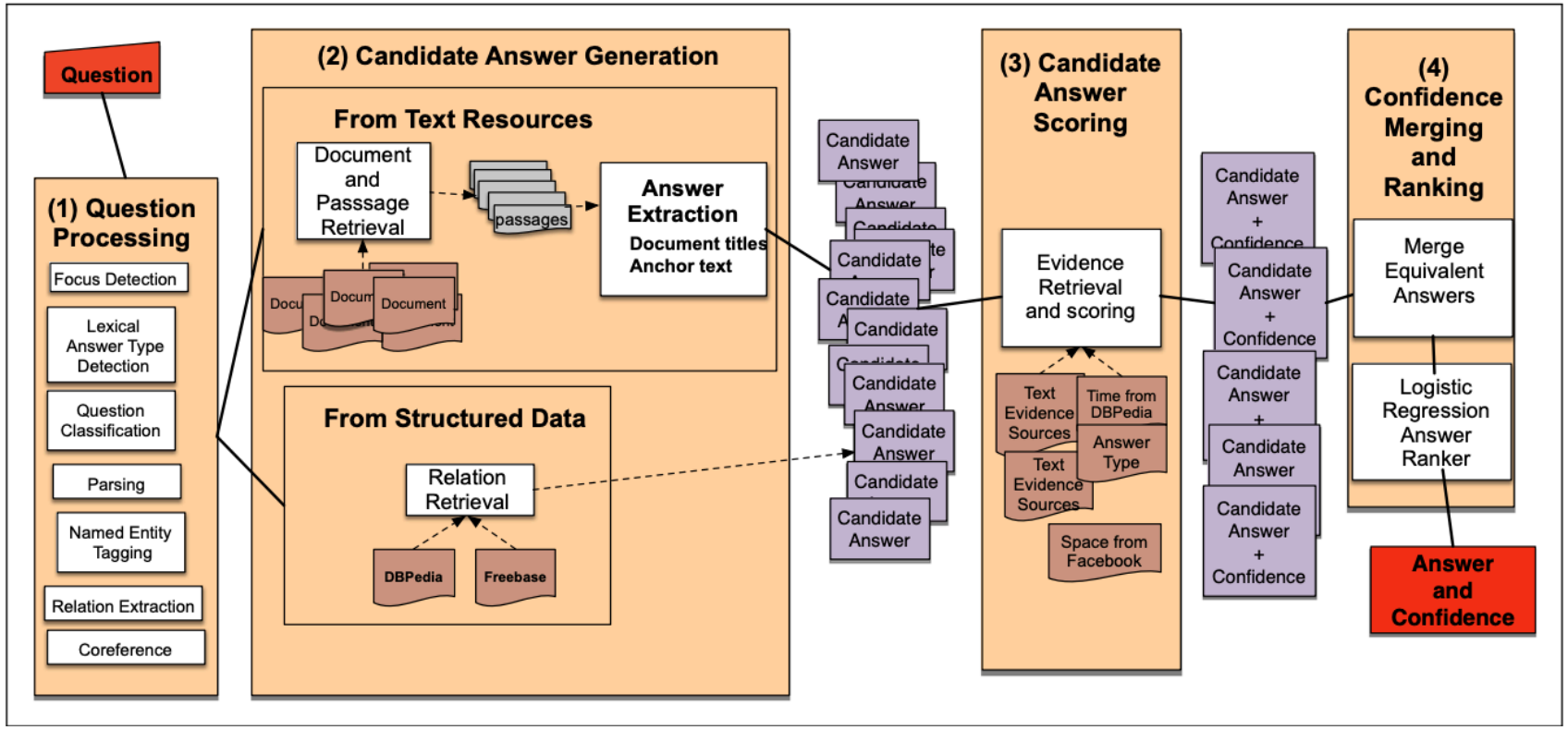
Lots of practical applications



2011: IBM Watson beat Jeopardy champions

IBM Watson defeated two of Jeopardy's greatest champions in 2011

2011: IBM Watson beat Jeopardy champions

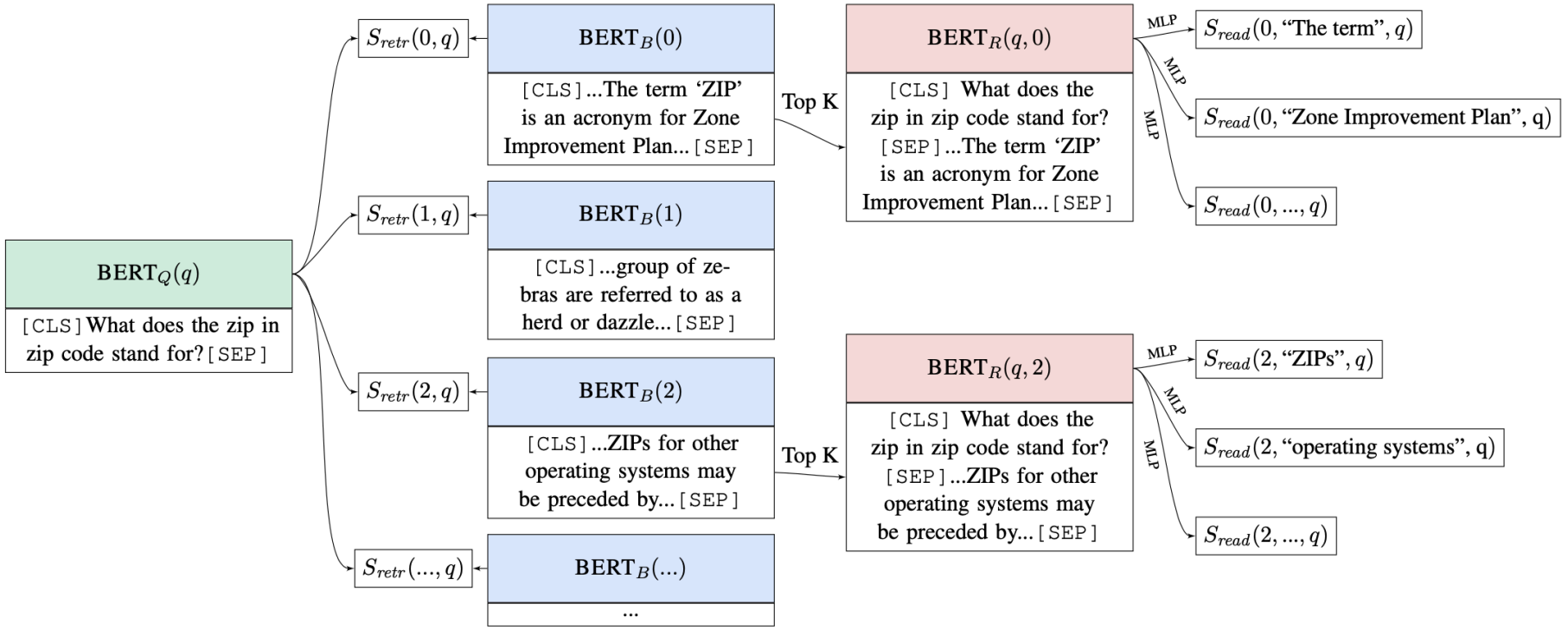


(1) Question processing, (2) Candidate answer generation, (3) Candidate answer scoring, and (4) Confidence merging and ranking.

J & M, edition 3

Question answering in deep learning era

- Almost all the state-of-the-art question answering systems are built on top of end-to-end training and pre-trained language models (e.g., BERT)!

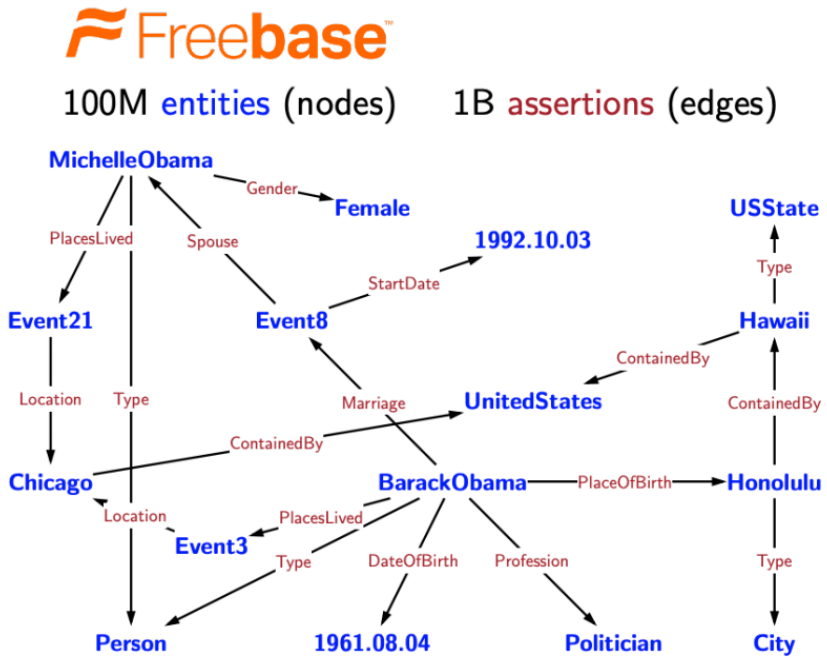


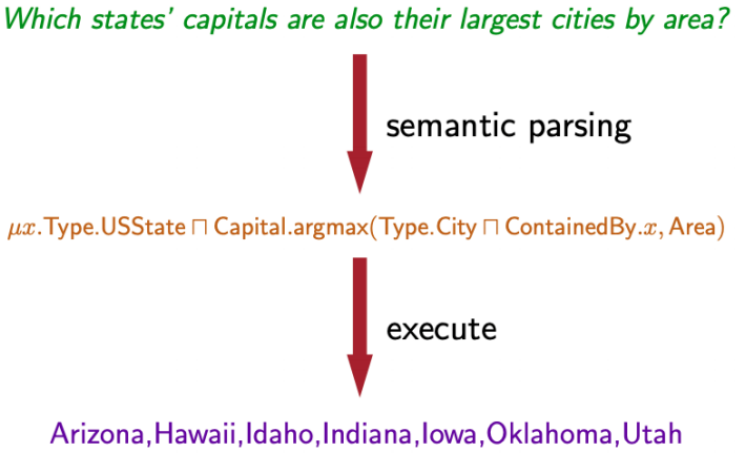
(Lee et al., 2019)

Beyond textual QA problems

- Today, we will mostly focus on how to answer questions based on **unstructured text**.

Knowledge based QA





Beyond textual QA problems

- Today, we will mostly focus on how to answer questions based on **unstructured text**.

Visual QA

What color are her eyes?
What is the mustache made of?

How many slices of pizza are there?
Is this a vegetarian pizza?

(Antol et al., 2015): Visual Question Answering

Reading comprehension

- **Reading comprehension** = comprehend a passage of text and answer questions about its content (P, Q) ⟶ A

> Tesla was the fourth of five children. He had an older brother named Dane and three sisters, Milka, Angelina and Marica. Dane was killed in a horse-riding accident when Nikola was five. In 1861, Tesla attended the "Lower" or "Primary" School in Smiljan where he studied German, arithmetic, and religion. In 1862, the Tesla family moved to Gospić, Austrian Empire, where Tesla's father worked as a pastor. Nikola completed "Lower" or "Primary" School, followed by the "Lower Real Gymnasium" or "Normal School."

Q: What language did Tesla study while in school?

A: German

Reading comprehension

- **Reading comprehension** = comprehend a passage of text and answer questions about its content (P, Q) ⟶ A

> Kannada language is the official language of Karnataka and spoken as a native language by about 66.54% of the people as of 2011. Other linguistic minorities in the state were Urdu (10.83%), Telugu language (5.84%), Tamil language (3.45%), Marathi language (3.38%), Hindi (3.3%), Tulu language (2.61%), Konkani language (1.29%), Malayalam (1.27%) and Kodava Takk (0.18%). In 2007 the state had a birth rate of 2.2%, a death rate of 0.7%, an infant mortality rate of 5.5% and a maternal mortality rate of 0.2%. The total fertility rate was 2.2.

Q: Which linguistic minority is larger, Hindi or Malayalam?

A: Hindi

Why do we care about this problem?

- ❑Useful for many practical applications
- ❑Reading comprehension is an important testbed for evaluating how well computer systems understand human language
- ❑Many other NLP tasks can be reduced to a reading comprehension problem:

Information extraction

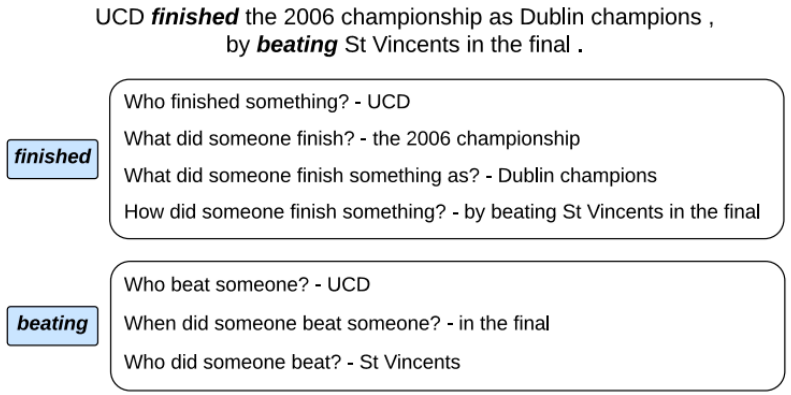
(Barack Obama, educated_at, ?)

Question: Where did Barack Obama graduate from?

Passage: Obama was born in Honolulu, Hawaii. After graduating from Columbia University in 1983, he worked as a community organizer in Chicago.

(Levy et al., 2017)

Semantic role labeling



(He et al., 2015)

Stanford question answering dataset (SQuAD)

- ❑100k annotated (passage, question, answer) triples
  - Large-scale supervised datasets are also a key ingredient for training effective neural models for reading comprehension!
- ❑Passages are selected from English Wikipedia, usually 100~150 words.
- ❑Questions are crowd-sourced.
- ❑Each answer is a short segment of text (or span) in the passage.
  - This is a limitation— not all the questions can be answered in this way!
- ❑ SQuAD still remains the most popular reading comprehension dataset; it is “almost solved” today and the state-of-the-art exceeds the estimated human performance.

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called “showers”.

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

(Rajpurkar et al., 2016): SQuAD: 100,000+ Questions for Machine Comprehension

Stanford question answering dataset (SQuAD)

- Evaluation: exact match (0 or 1) and F1 (partial credit).
- For development and test sets, 3 gold answers are collected, because there could be multiple plausible answers.
- We compare the predicted answer to each gold answer (a, an, the, punctuations are removed) and take max scores. Finally, we take the average of all the examples for both exact match and F1.
- Estimated human performance: EM = 82.3, F1 = 91.2

Q: What did Tesla do in December 1878?

A: {left Graz, left Graz, left Graz and severed all relations with his family}

Prediction: {left Graz and served}

Exact match: $\max\{0, 0, 0\} = 0$

F1: $\max\{0.67, 0.67, 0.61\} = 0.67$

Other question answering datasets

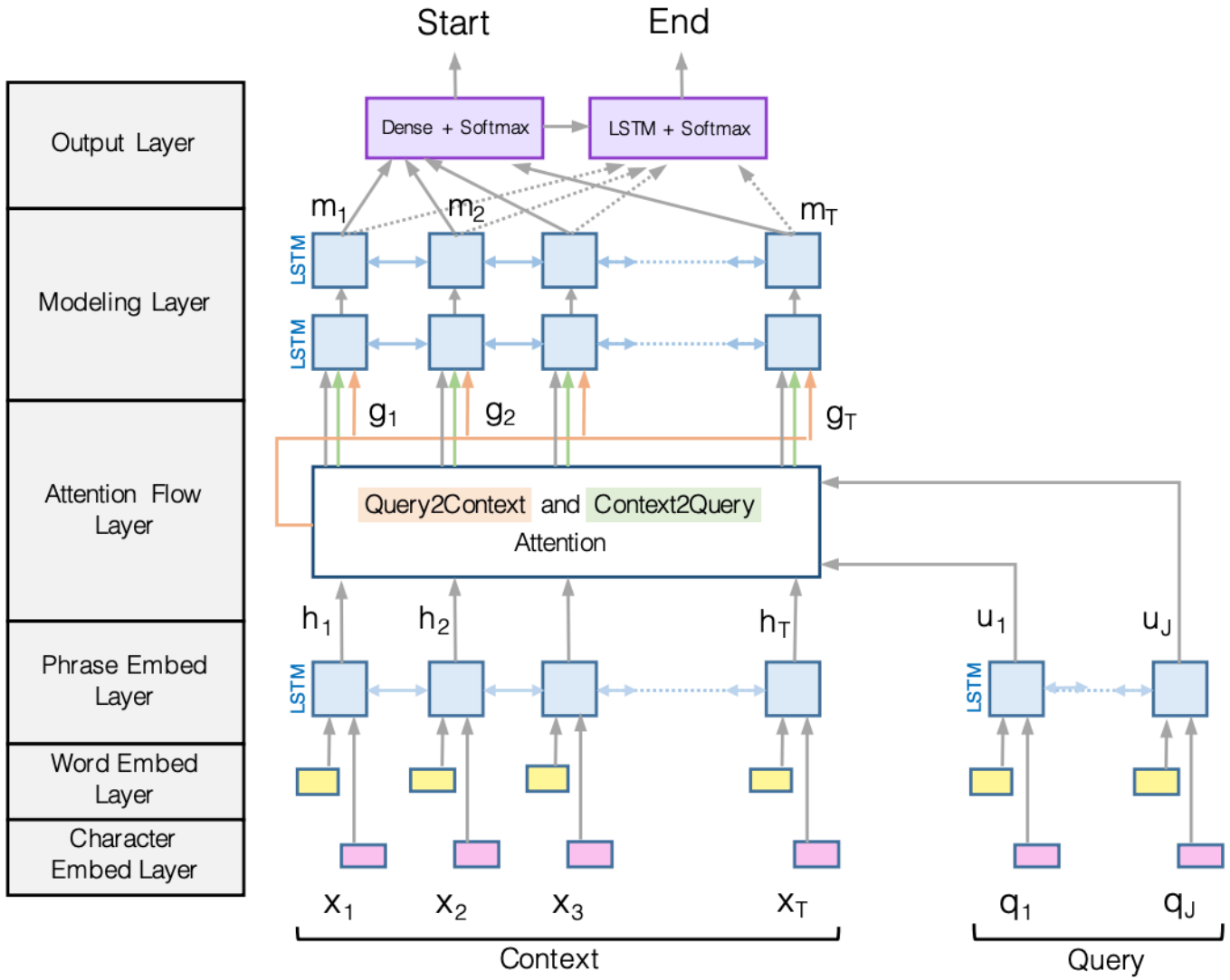
- TriviaQA: Questions and answers by trivia enthusiasts. Independently collected web paragraphs that contain the answer and seem to discuss question, but no human verification that paragraph supports answer to question
- Natural Questions: Question drawn from frequently asked Google search questions. Answers from Wikipedia paragraphs. Answer can be substring, yes, no, or NOT_PRESENT. Verified by human annotation.
- HotpotQA. Constructed questions to be answered from the whole of Wikipedia which involve getting information from two pages to answer a multistep query:

  Q: Which novel by the author of “Armada” will be adapted as a feature film by Steven Spielberg?
  A: *Ready Player One*

Neural models for reading comprehension

- How can we build a model to solve SQuAD?
  - We are going to use **passage, paragraph and context**, as well as **question** and **query** interchangeably
- Problem formulation
  - Input: $C = (c_1, c_2, ..., c_N)$, $Q = (q_1, q_2, ..., q_M)$, $c_i, q_i \in V$ — $N \sim 100$, $M \sim 15$
  - Output: $1 \leq start \leq end \leq N$ — answer is a span in the passage
- A family of LSTM-based models with attention (2016–2018)
  - Attentive Reader (Hermann et al., 2015), Stanford Attentive Reader (Chen et al., 2016), Match- LSTM (Wang et al., 2017), BiDAF (Seo et al., 2017), Dynamic coattention network (Xiong et al., 2017), DrQA (Chen et al., 2017), R-Net (Wang et al., 2017), ReasoNet (Shen et al., 2017)..
- Fine-tuning BERT-like models for reading comprehension (2019+)

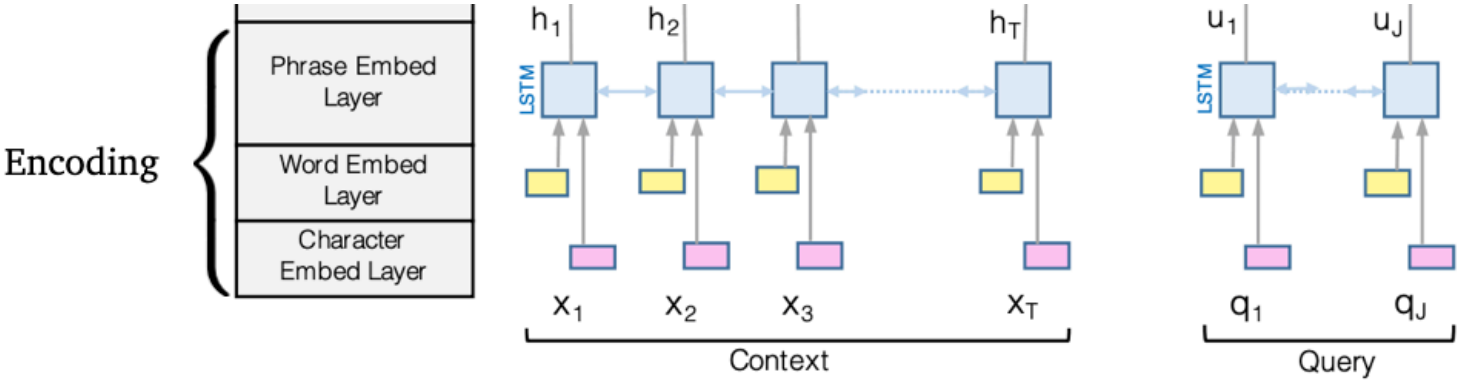
BiDAF: the Bidirectional Attention Flow model

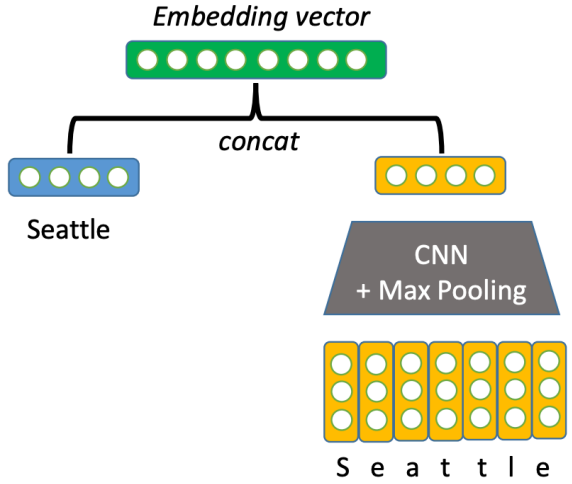


(Seo et al., 2017): Bidirectional Attention Flow for Machine Comprehension

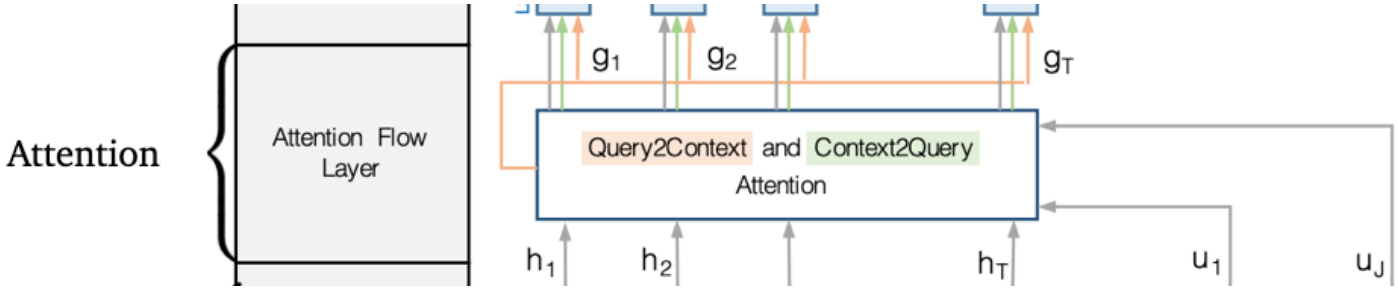
BiDAF: Encoding

- Use a concatenation of word embedding (GloVe) and character embedding (CNNs over character embeddings) for each word in context and query.
- Then, use two bidirectional LSTMs separately to produce contextual embeddings for both context and query.





BiDAF: Attention



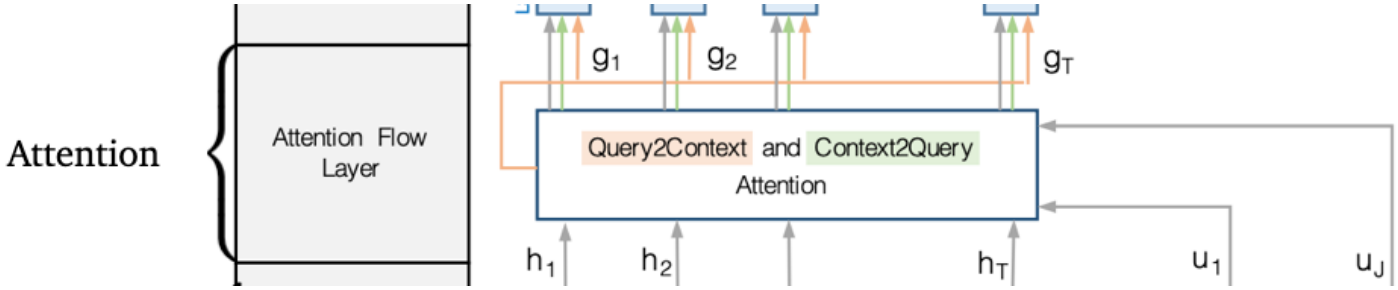
- Context-to-query attention: For each context word, choose the most relevant words from the query words.

Q: *Who leads the United States?*

C: *Barak Obama is the president of the USA.*

For each context word, find the most relevant query word.

BiDAF: Attention

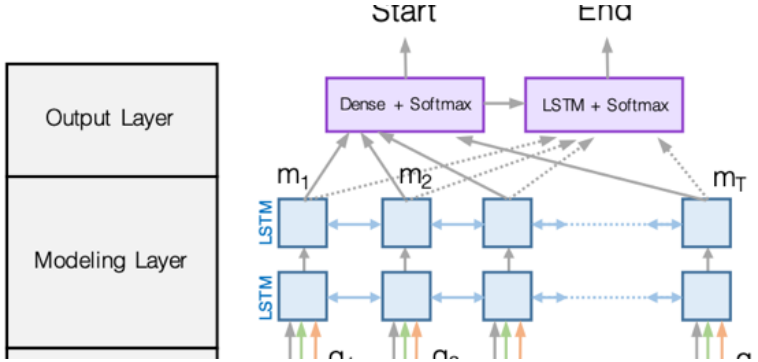


- Query-to-context attention: choose the context words that are most relevant to one of query words.

While Seattle's weather is very nice in summer, its weather is very rainy in winter, making it one of the most gloomy cities in the U.S. LA is …

Q: Which city is gloomy in winter?

BiDAF: Modeling and output layers



The final training loss is

$$\mathcal{L} = -\log p_{\text{start}}(s^*) - \log p_{\text{end}}(e^*)$$

- Modeling layer: pass g_i to another two layers of bi-directional LSTMs.
  - Attention layer is modeling interactions between query and context
  - Modeling layer is modeling interactions within context words
- Output layer: two classifiers predicting the start and end positions

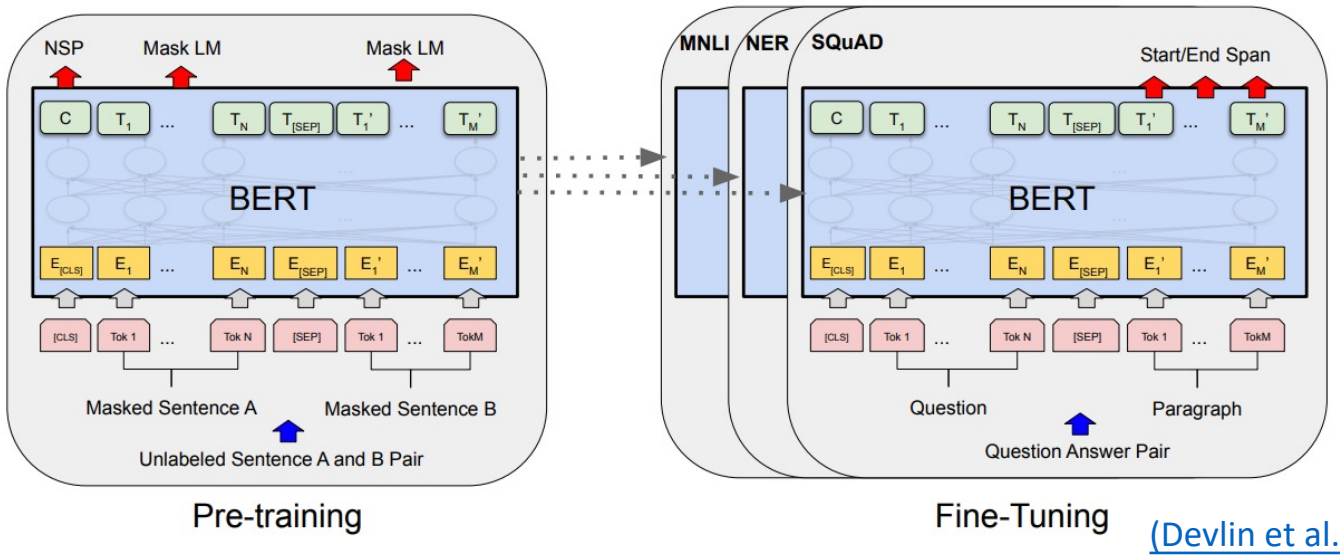
BiDAF: Performance on SQuAD

- This model achieved 77.3 F1 on SQuAD v1.1.
  - Without context-to-query attention ⟹ 67.7 F1
  - Without query-to-context attention ⟹ 73.7 F1
  - Without character embeddings ⟹ 75.4 F1

	F1
Logistic regression	51.0
Fine-Grained Gating (Carnegie Mellon U)	73.3
Match-LSTM (Singapore Management U)	73.7
DCN (Salesforce)	75.9
BiDAF (UW & Allen Institute)	77.3
Multi-Perspective Matching (IBM)	78.7
ReasoNet (MSR Redmond)	79.4
DrQA (Chen et al. 2017)	79.4
r-net (MSR Asia) [Wang et al., ACL 2017]	79.7
Human performance	91.2

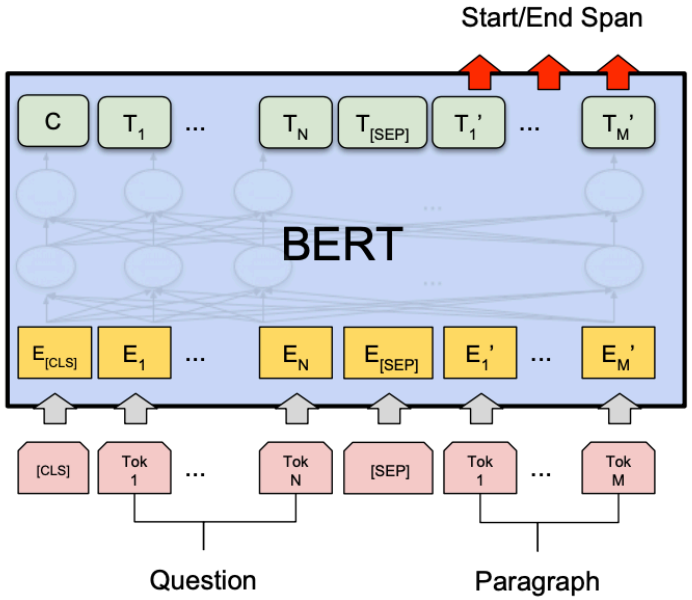
BERT for reading comprehension

- BERT is a deep bidirectional Transformer encoder pre-trained on large amounts of text (Wikipedia + BooksCorpus)
- BERT is pre-trained on two training objectives:
  - Masked language model (MLM)
  - Next sentence prediction (NSP)
- BERTbase has 12 layers and 110M parameters, BERTlarge has 24 layers and 330M parameters



(Devlin et al., 2018)

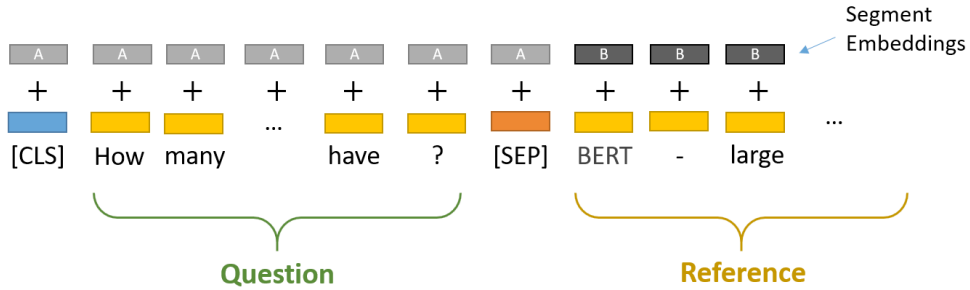
BERT for reading comprehension



Question = Segment A

Passage = Segment B

Answer = predicting two endpoints in segment B



Question: `How many parameters does BERT-large have?`

Reference Text: `BERT-large is really big... it has 24 layers and an embedding size of 1,024, for a total of 340M parameters! Altogether it is 1.34GB, so expect it to take a couple minutes to download to your Colab instance.`

Image credit: https://mccormickml.com/

$$\mathcal{L} = -\log p_{\text{start}}(s^*) - \log p_{\text{end}}(e^*)$$

$$p_{\text{start}}(i) = \text{softmax}_i(\mathbf{w}_{\text{start}}^\intercal \mathbf{h}_i)$$

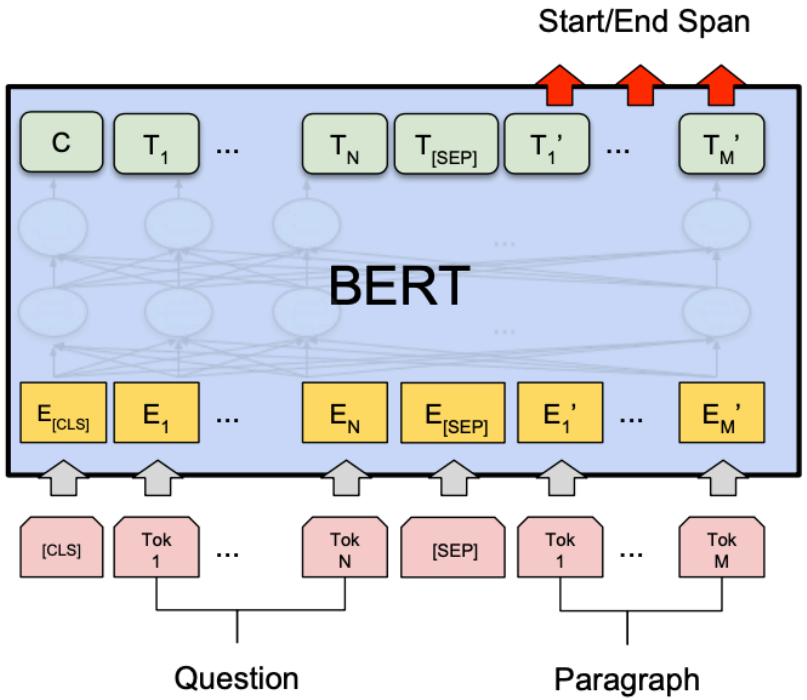
$$p_{\text{end}}(i) = \text{softmax}_i(\mathbf{w}_{\text{end}}^\intercal \mathbf{h}_i)$$

where $\mathbf{h}_i$ is the hidden vector of c_i, returned by BERT

BERT for reading comprehension

$$\mathcal{L} = -\log p_{\text{start}}(s^*) - \log p_{\text{end}}(e^*)$$

- All the BERT parameters (e.g., 110M) as well as the newly introduced parameters h_{start} , h_{end} (e.g., 768 x 2 = 1536) are optimized together for ***L***.
- It works amazing well. Stronger pre-trained language models can lead to even better performance and SQuAD becomes a standard dataset for testing pre-trained models.



	F1	EM
Human performance	91.2*	82.3*
BiDAF	77.3	67.7
BERT-base	88.5	80.8
BERT-large	90.9	84.1
XLNet	94.5	89.0
RoBERTa	94.6	88.9
ALBERT	94.8	89.3

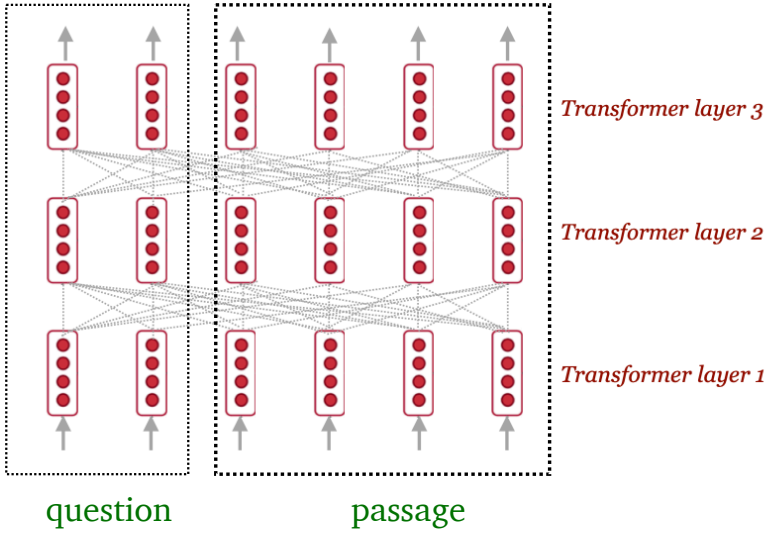
(dev set, except for human performance)

Comparisons between BiDAF and BERT models

- BERT model has many many more parameters (110M or 330M).
  BiDAF has ~2.5M parameters.
- BiDAF is built on top of several bidirectional LSTMs while BERT is built on top of Transformers (no recurrence architecture and easier to parallelize).
- BERT is **pre-trained** while BiDAF is only built on top of GloVe (and all the remaining parameters need to be learned from the supervision datasets).
- Pre-training is clearly a game changer but it is expensive..

Comparisons between BiDAF and BERT models

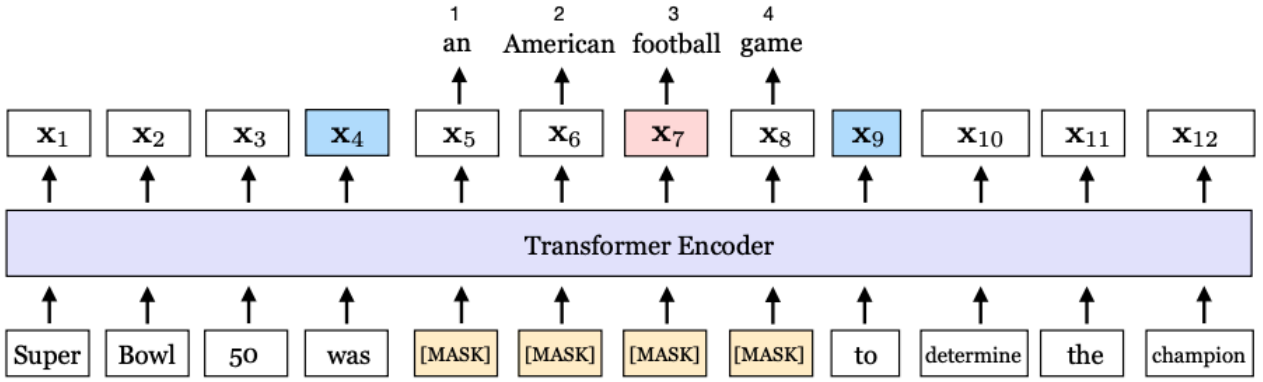
- BiDAF and other models aim to model the interactions between question and passage.
- BERT uses self-attention between the **concatenation** of question and passage = **attention(P, P) + attention(P, Q) + attention(Q, P) + attention(Q, Q)**
- (Clark and Gardner, 2018) shows that adding a self-attention layer for the passage **attention(P, P)** to BiDAF also improves performance.



Can we design better pre-training objectives?

The answer is yes!

$$\mathcal{L}(\text{football}) = \mathcal{L}_{\text{MLM}}(\text{football}) + \mathcal{L}_{\text{SBO}}(\text{football})$$
$$= -\log P(\text{football} \mid \mathbf{x}_7) - \log P(\text{football} \mid \mathbf{x}_4, \mathbf{x}_9, \mathbf{p}_3)$$

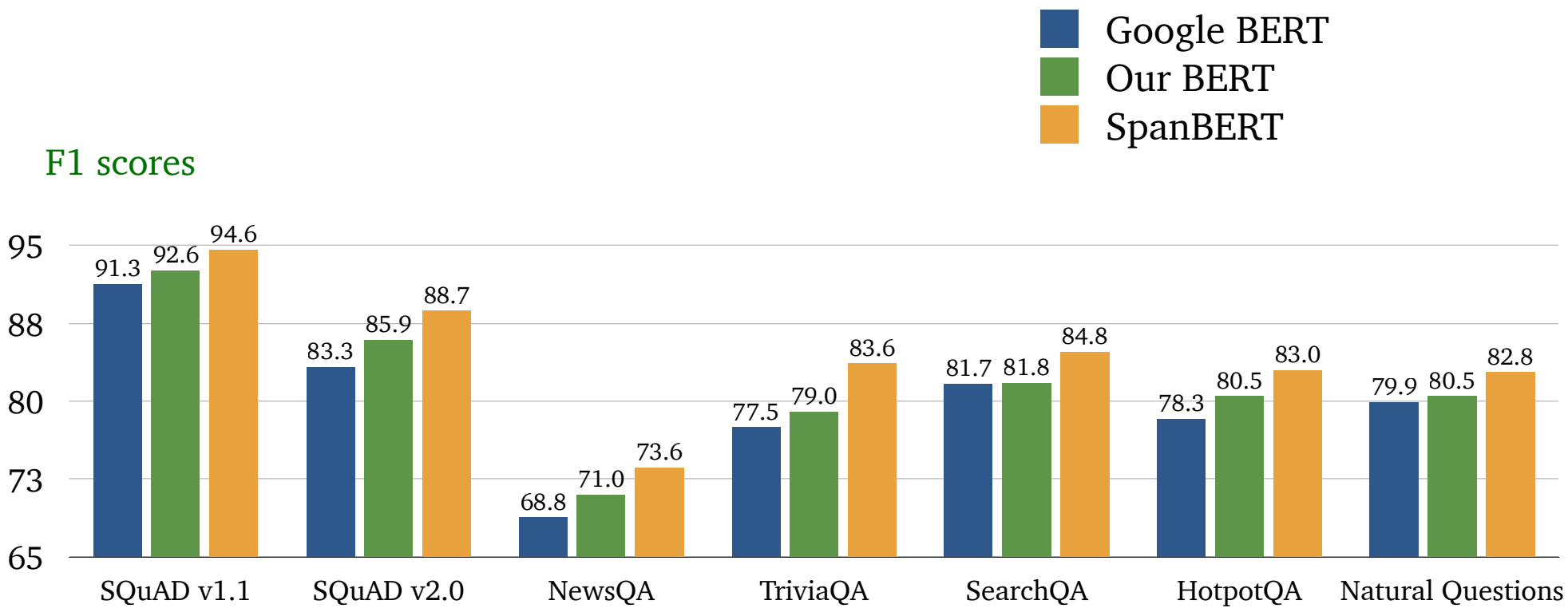


- Two ideas:
  - 1) masking contiguous spans of words instead of 15% random words
  - 2) using the two end points of span to predict all the masked words in between = compressing the information of a span into its two endpoints

$$\mathbf{y}_i = f(\mathbf{x}_{s-1}, \mathbf{x}_{e+1}, \mathbf{p}_{i-s+1})$$

(Joshi & Chen et al., 2020): SpanBERT: Improving Pre-training by Representing and Predicting Spans

SpanBERT performance



Is reading comprehension solved?

- ❑ We have already surpassed human performance on SQuAD. Does it mean that reading comprehension is already solved? Of course not!
- ❑ The current systems still perform poorly on adversarial examples or examples from out-of- domain distributions

Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

	Match Single	Match Ens.	BiDAF Single	BiDAF Ens.
Original	71.4	75.4	75.5	80.0
ADDSENT	27.3	29.4	34.3	34.2
ADDONESENT	39.0	41.8	45.7	46.9
ADDANY	7.6	11.7	4.8	2.7
ADDCOMMON	38.9	51.0	41.7	52.6

(Jia and Liang, 2017): Adversarial Examples for Evaluating Reading Comprehension Systems

Is reading comprehension solved?

- Systems trained on one dataset can't generalize to other datasets:

Fine-tuned on	Evaluated on: SQuAD	TriviaQA	NQ	QuAC	NewsQA
SQuAD	**75.6**	46.7	48.7	20.2	41.1
TriviaQA	49.8	**58.7**	42.1	20.4	10.5
NQ	53.5	46.3	**73.5**	21.6	24.7
QuAC	39.4	33.1	33.8	**33.3**	13.8
NewsQA	52.1	38.4	41.7	20.4	**60.1**

(Sen and Saffari, 2020): What do Models Learn from Question Answering Datasets?

Is reading comprehension solved?

BERT-large model trained on SQuAD

	Test *TYPE* and Description	Failure Rate (🤖)	Example Test cases (with expected behavior and 🤖 prediction)
Vocab	***MFT:*** comparisons	20.0	**C:** Victoria is younger than Dylan. **Q:** Who is less young? **A:** Dylan 🤖: Victoria
	MFT: intensifiers to superlative: most/least	91.3	**C:** Anna is worried about the project. Matthew is extremely worried about the project. **Q:** Who is least worried about the project? **A:** Anna 🤖: Matthew
Taxonomy	***MFT:*** match properties to categories	82.4	**C:** There is a tiny purple box in the room. **Q:** What size is the box? **A:** tiny 🤖: purple
	MFT: nationality vs job	49.4	**C:** Stephanie is an Indian accountant. **Q:** What is Stephanie's job? **A:** accountant 🤖: Indian accountant
	MFT: animal vs vehicles	26.2	**C:** Jonathan bought a truck. Isabella bought a hamster. **Q:** Who bought an animal? **A:** Isabella 🤖: Jonathan
	MFT: comparison to antonym	67.3	**C:** Jacob is shorter than Kimberly. **Q:** Who is taller? **A:** Kimberly 🤖: Jacob
	MFT: more/less in context, more/less antonym in question	100.0	**C:** Jeremy is more optimistic than Taylor. **Q:** Who is more pessimistic? **A:** Taylor 🤖: Jeremy
Robust.	***INV:*** Swap adjacent characters in **Q** (typo)	11.6	**C:** ...Newcomen designs had a duty of about 7 million, but most were closer to 5 million.... **Q:** What was the ideal duty → udty of a Newcomen engine? **A:** INV 🤖: 7 million → 5 million
	INV: add irrelevant sentence to **C**	9.8	(no example)

(Ribeiro et al., 2020): Beyond Accuracy: Behavioral Testing of NLP Models with CheckList

Is reading comprehension solved?

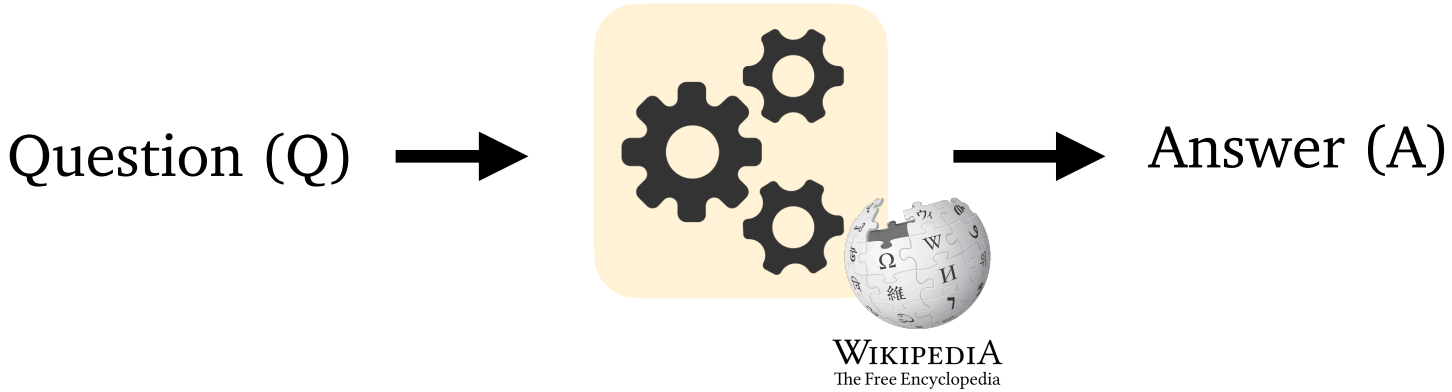
BERT-large model trained on SQuAD

Temporal	***MFT:*** change in one person only	41.5	**C:** Both Luke and Abigail were writers, but there was a change in Abigail, who is now a model. **Q:** Who is a model? **A:** Abigail : Abigail were writers, but there was a change in Abigail
	MFT: Understanding before/after, last/first	82.9	**C:** Logan became a farmer before Danielle did. **Q:** Who became a farmer last? **A:** Danielle : Logan
Neg.	***MFT:*** Context has negation	67.5	**C:** Aaron is not a writer. Rebecca is. **Q:** Who is a writer? **A:** Rebecca : Aaron
	MFT: **Q** has negation, **C** does not	100.0	**C:** Aaron is an editor. Mark is an actor. **Q:** Who is not an actor? **A:** Aaron : Mark
Coref.	***MFT:*** Simple coreference, he/she.	100.0	**C:** Melissa and Antonio are friends. He is a journalist, and she is an adviser. **Q:** Who is a journalist? **A:** Antonio : Melissa
	MFT: Simple coreference, his/her.	100.0	**C:** Victoria and Alex are friends. Her mom is an agent **Q:** Whose mom is an agent? **A:** Victoria : Alex
	MFT: former/latter	100.0	**C:** Kimberly and Jennifer are friends. The former is a teacher **Q:** Who is a teacher? **A:** Kimberly : Jennifer
SRL	***MFT:*** subject/object distinction	60.8	**C:** Richard bothers Elizabeth. **Q:** Who is bothered? **A:** Elizabeth : Richard
	MFT: subj/obj distinction with 3 agents	95.7	**C:** Jose hates Lisa. Kevin is hated by Lisa. **Q:** Who hates Kevin? **A:** Lisa : Jose

(Ribeiro et al., 2020): Beyond Accuracy: Behavioral Testing of NLP Models with CheckList

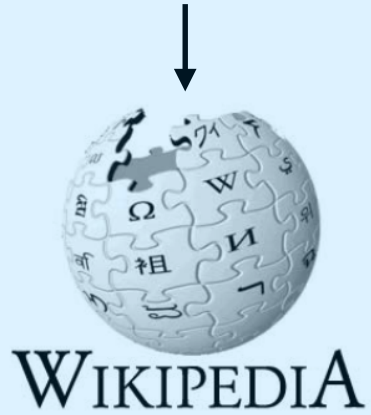
Open-domain question answering

- Different from reading comprehension, we don't assume a given passage.
- Instead, we only have access to a large collection of documents (e.g., Wikipedia). We don't know where the answer is located, and the goal is to return the answer for any open-domain questions.
  - In contrast to closed-domain systems that deal with questions under a specific domain (medicine, technical support).
- Much more challenging and a more practical problem!

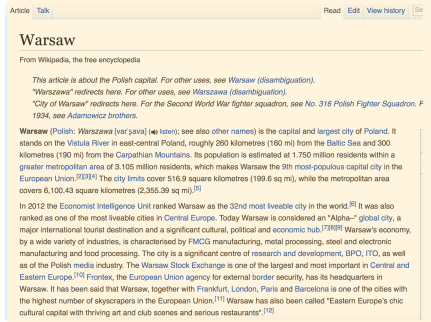


Retriever-reader framework

How many of Warsaw's inhabitants spoke Polish in 1933?



Document Retriever


Article Talk Read Edit View history

Warsaw

From Wikipedia, the free encyclopedia

This article is about the Polish capital. For other uses, see Warsaw (disambiguation).
"Warszawa" redirects here. For other uses, see Warszawa (disambiguation).
"City of Warsaw" redirects here. For the Second World War fighter squadron, see No. 316 Polish Fighter Squadron. F 1934, see Adamowicz brothers.

Warsaw (Polish: *Warszawa* [varˈʂava] ◄) listen); see also other names) is the capital and largest city of Poland. It stands on the Vistula River in east-central Poland, roughly 260 kilometres (160 mi) from the Baltic Sea and 300 kilometres (190 mi) from the Carpathian Mountains. Its population is estimated at 1.750 million residents within a greater metropolitan area of 3.105 million residents, which makes Warsaw the 9th most-populous capital city in the European Union.[2][3][4] The city limits cover 516.9 square kilometres (199.6 sq mi), while the metropolitan area covers 6,100.43 square kilometres (2,355.39 sq mi).[5]

In 2012 the Economist Intelligence Unit ranked Warsaw as the 32nd most liveable city in the world.[6] It was also ranked as one of the most liveable cities in Central Europe. Today Warsaw is considered an "Alpha–" global city, a major international tourist destination and a significant cultural, political and economic hub.[7][8][9] Warsaw's economy, by a wide variety of industries, is characterised by FMCG manufacturing, metal processing, steel and electronic manufacturing and food processing. The city is a significant centre of research and development, BPO, ITO, as well as of the Polish media industry. The Warsaw Stock Exchange is one of the largest and most important in Central and Eastern Europe.[10] Frontex, the European Union agency for external border security, has its headquarters in Warsaw. It has been said that Warsaw, together with Frankfurt, London, Paris and Barcelona is one of the cities with the highest number of skyscrapers in the European Union.[11] Warsaw has also been called "Eastern Europe's chic cultural capital with thriving art and club scenes and serious restaurants".[12]

Document Reader

833,500

https://github.com/facebookresearch/DrQA

Chen et al., 2017. Reading Wikipedia to Answer Open-domain Questions

Retriever-reader framework

- Input: a large collection of documents $\mathcal{D} = D_1, D_2, \ldots, D_N$ and Q
- Output: an answer string A

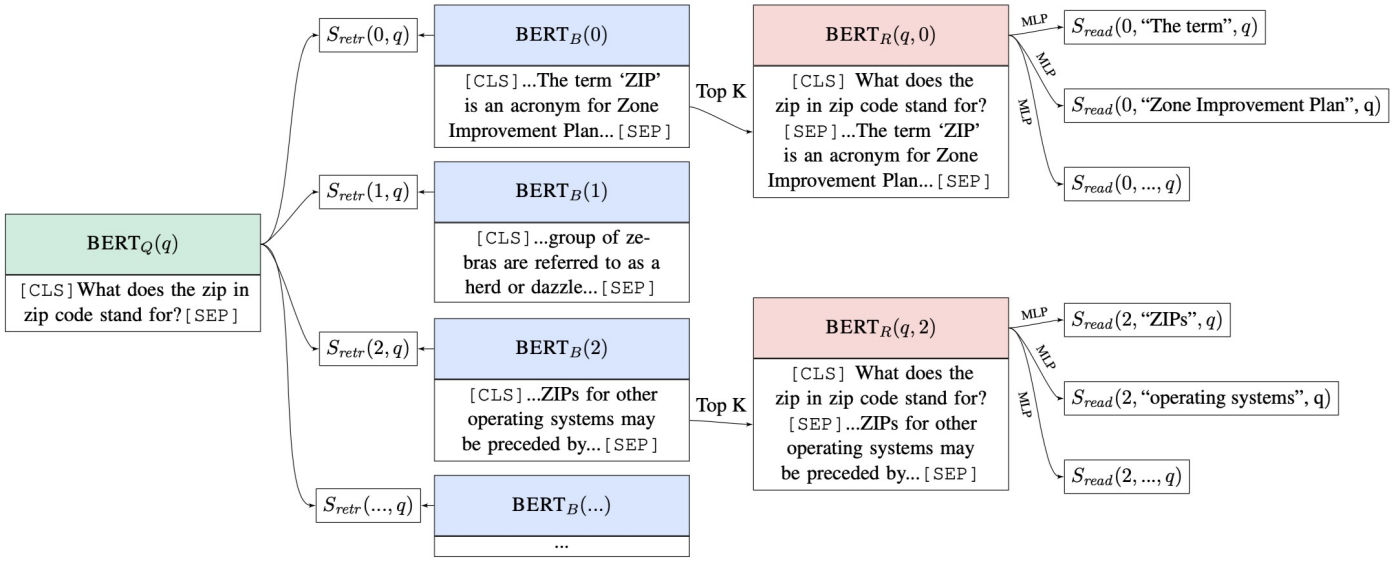
- Retriever: $f(\mathcal{D}, Q) \longrightarrow P_1, \ldots, P_K$ K is pre-defined (e.g.,100)
- Reader: $g(Q, \{P_1, \ldots, P_K\}) \longrightarrow A$ A reading comprehension problem!

- In DrQA,
  - Retriever = A standard TF-IDF information-retrieval sparse model (a fixed module)
  - Reader = a neural reading comprehension model that we just learned

Chen et al., 2017. Reading Wikipedia to Answer Open-domain Questions

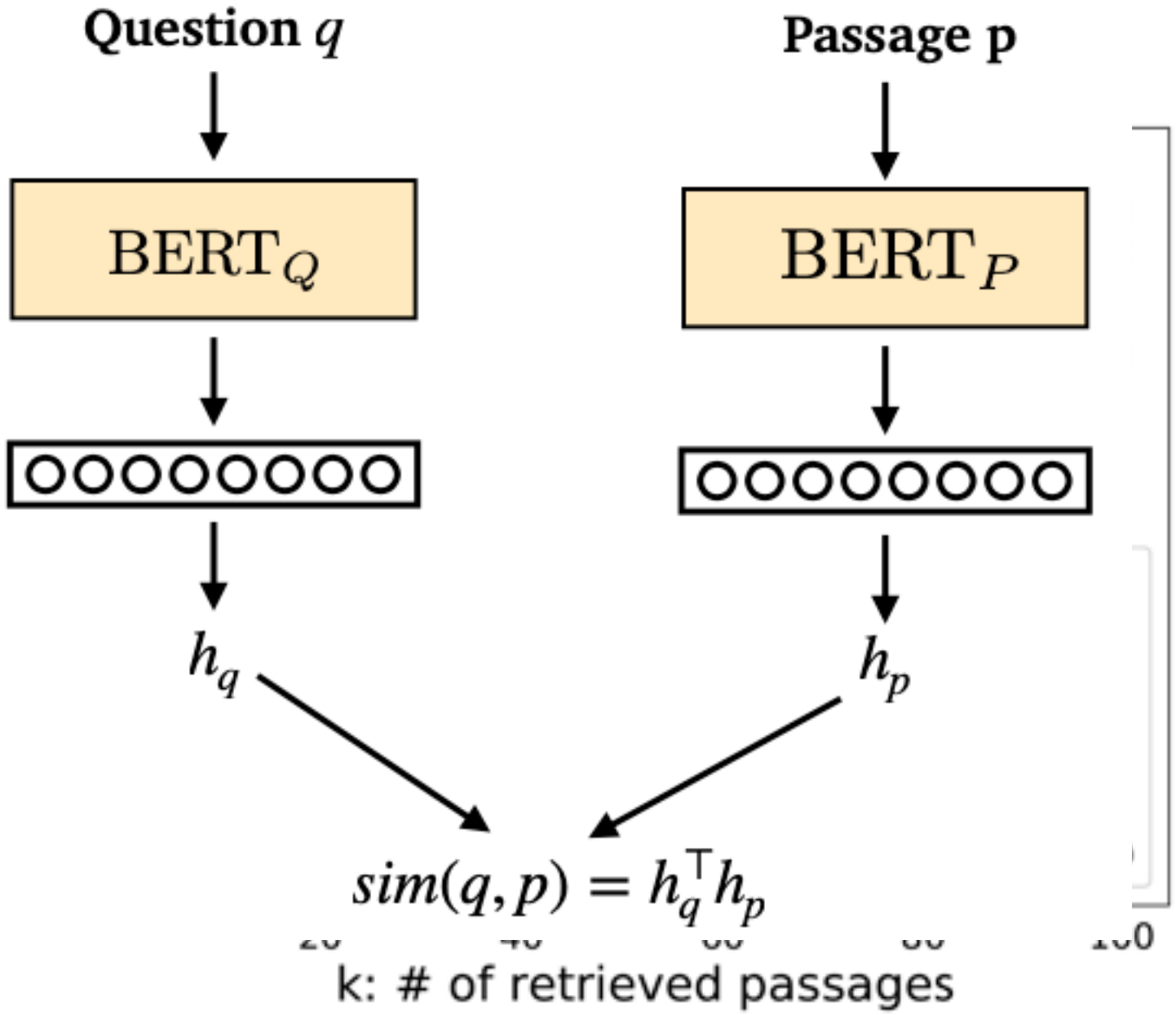
We can train the retriever too

- **Joint training** of retriever and reader



- Each text passage can be encoded as a vector using BERT and the retriever score can be measured as the dot product between the question representation and passage representation.
- However, it is not easy to model as there are a huge number of passages (e.g., 21M in English Wikipedia)

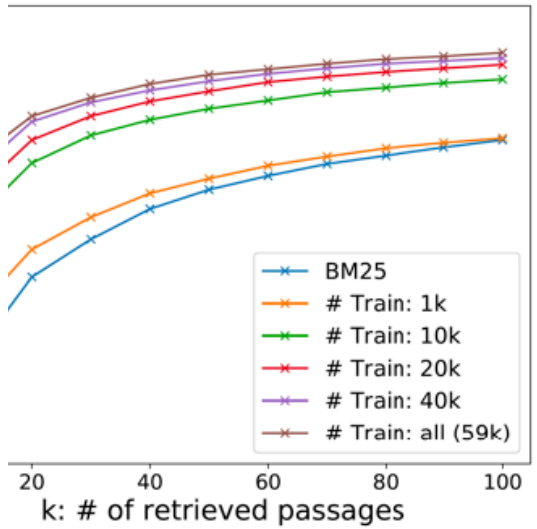
Lee et al., 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering



ver too

ain the retriever using question-

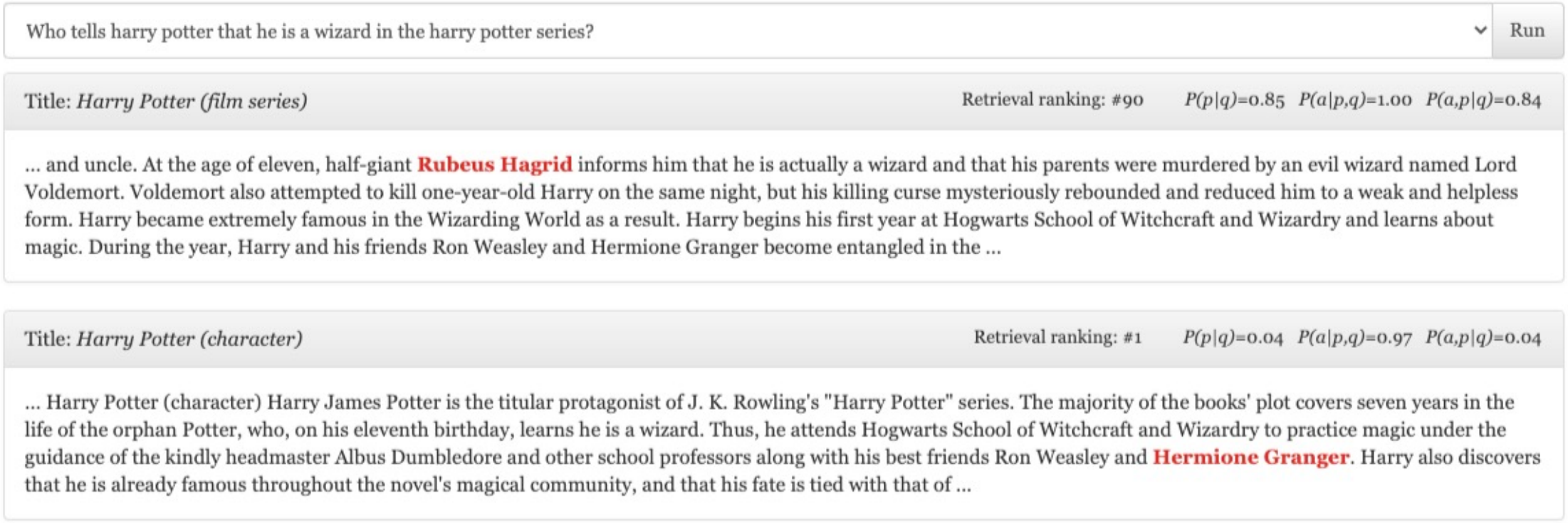
< Q/A pairs beat BM25!



ıs traditional IR retrieval models

Karpukhin et al., 2020. Dense Passage Retrieval for Open-Domain Question Answering

We can train the retriever too

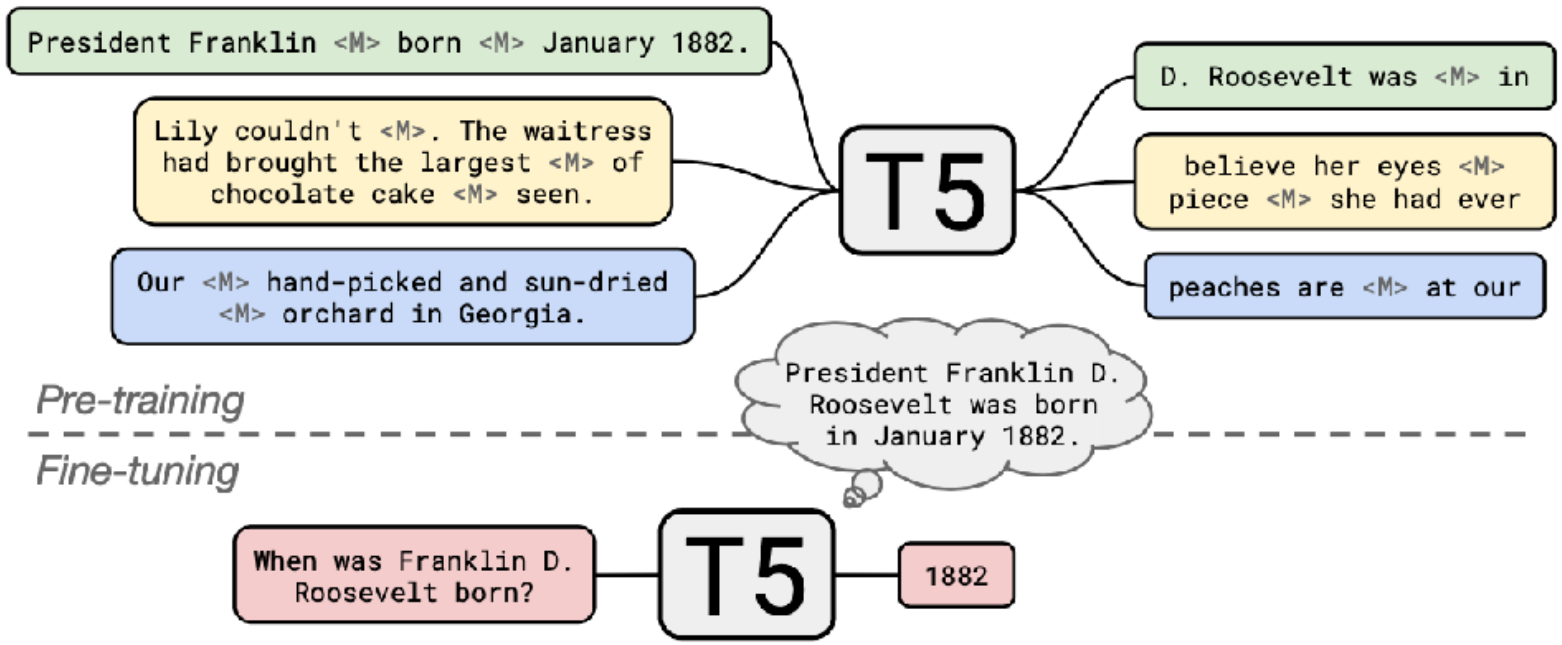


http://qa.cs.washington.edu:2020/

Karpukhin et al., 2020. Dense Passage Retrieval for Open-Domain Question Answering

Large language models can do open-domain QA well

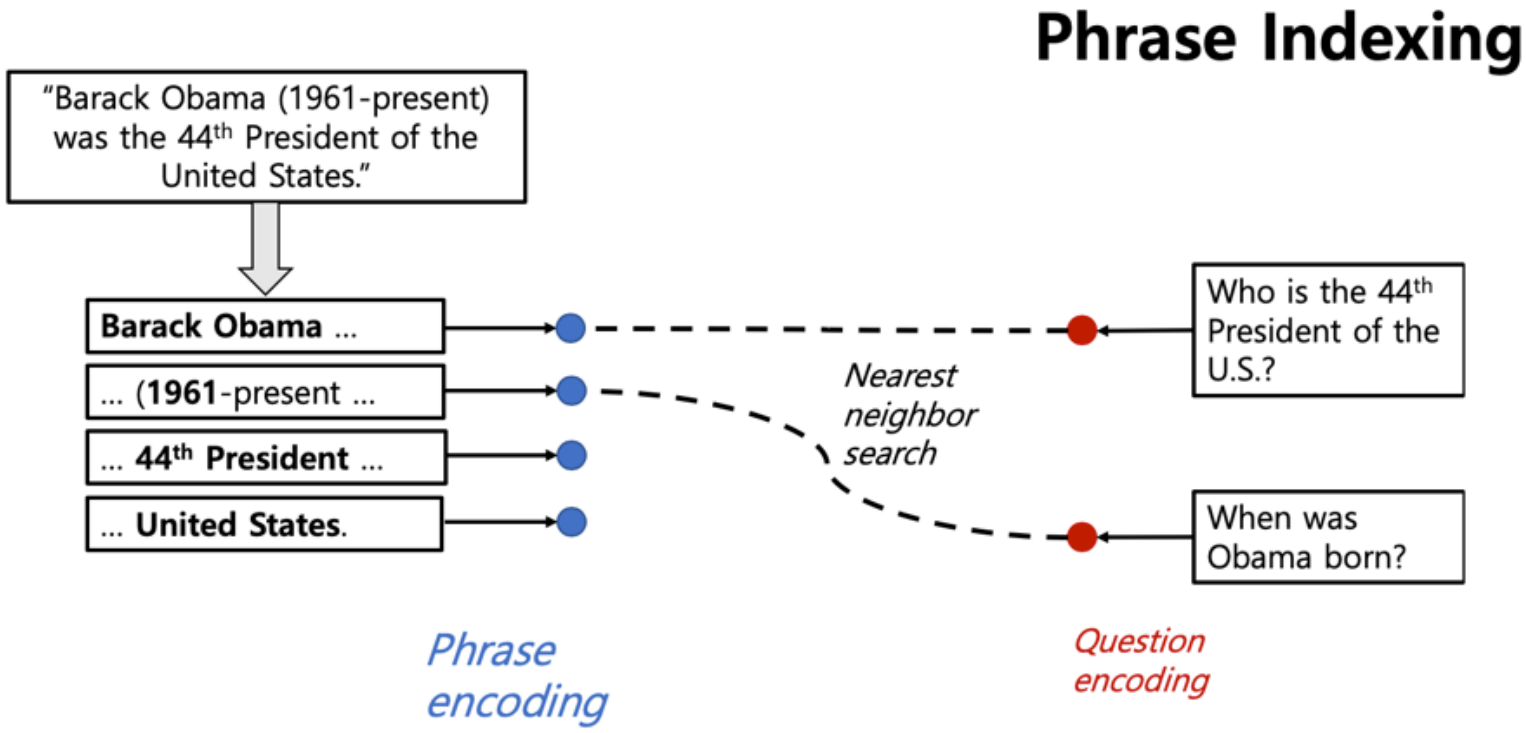
- ... without an explicit retriever stage



Roberts et al., 2020. How Much Knowledge Can You Pack Into the Parameters of a Language Model?

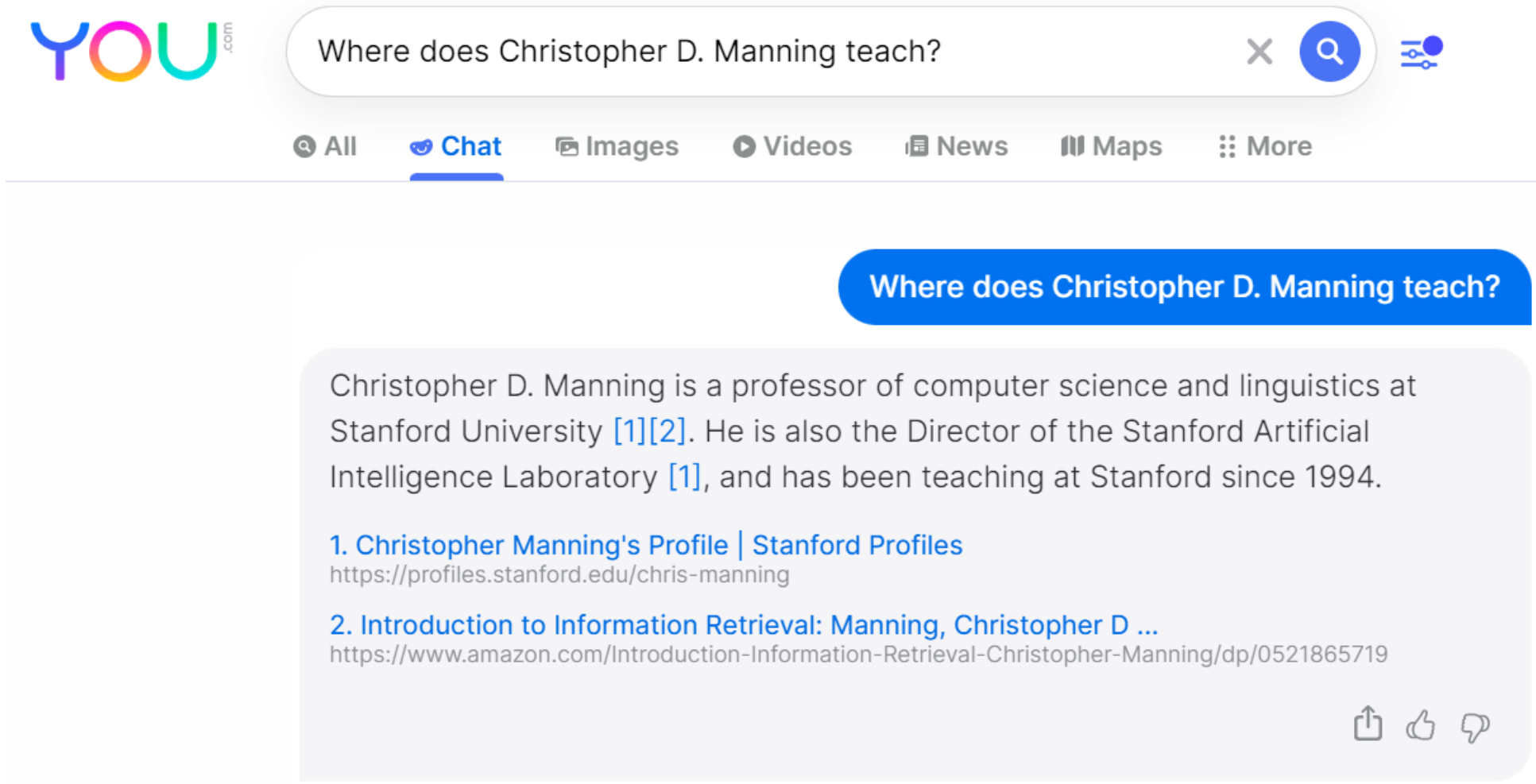
Maybe the reader model is not necessary too!

- It is possible to encode all the phrases (60 billion phrases in Wikipedia) using dense vectors and only do nearest neighbor search without a BERT model at inference time!



Seo et al., 2019. Real-Time Open-Domain Question Answering with Dense-Sparse Phrase Index
Lee et al., 2020. Learning Dense Representations of Phrases at Scale

Large language model-based QA (with web search!)



Problems with large language model-based QA

Seems totally reasonable!

But (1) it's not his most cited paper, and (2) it doesn't have that many citations. Yikes! Also the reference to a web page doesn't help.