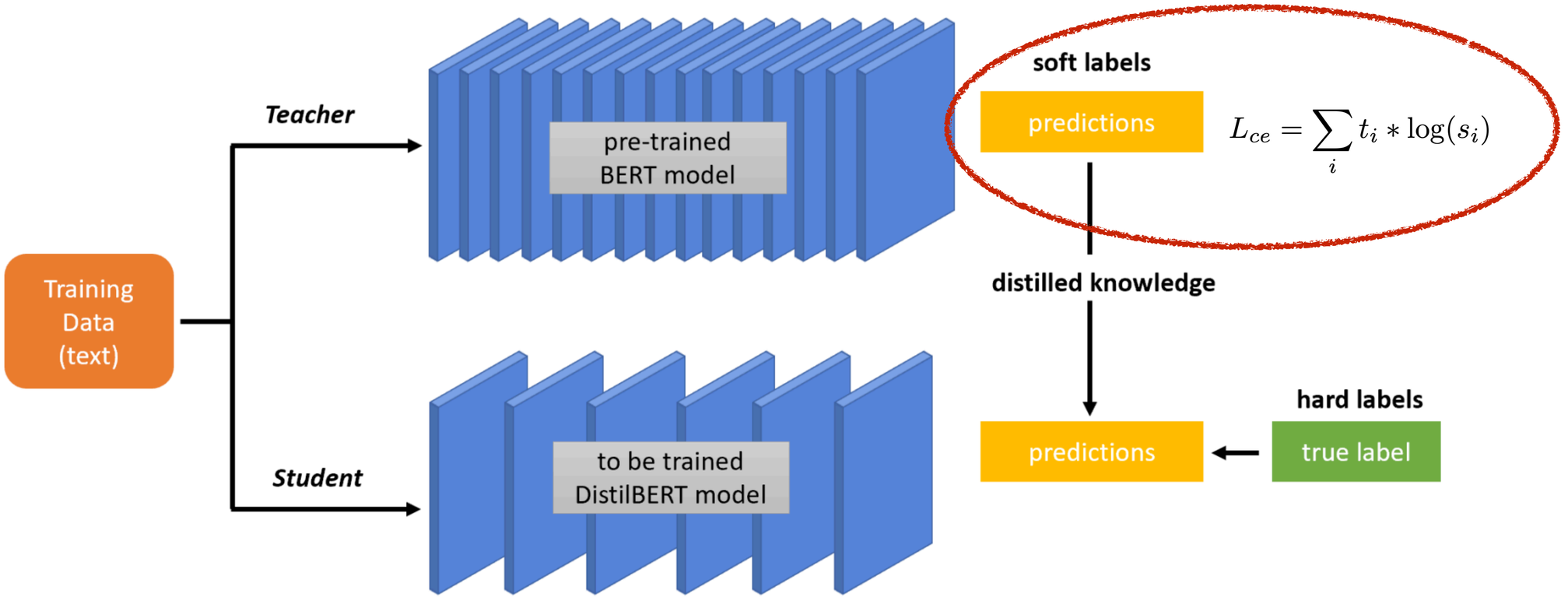
Pretraining Language Models (part 2)

Wei Xu

(some slides from Greg Durrett, Alan Ritter)

DistilBERT

- DistilBERT is pretrained by knowledge distillation to create a smaller model with faster inference and requires less compute to train.

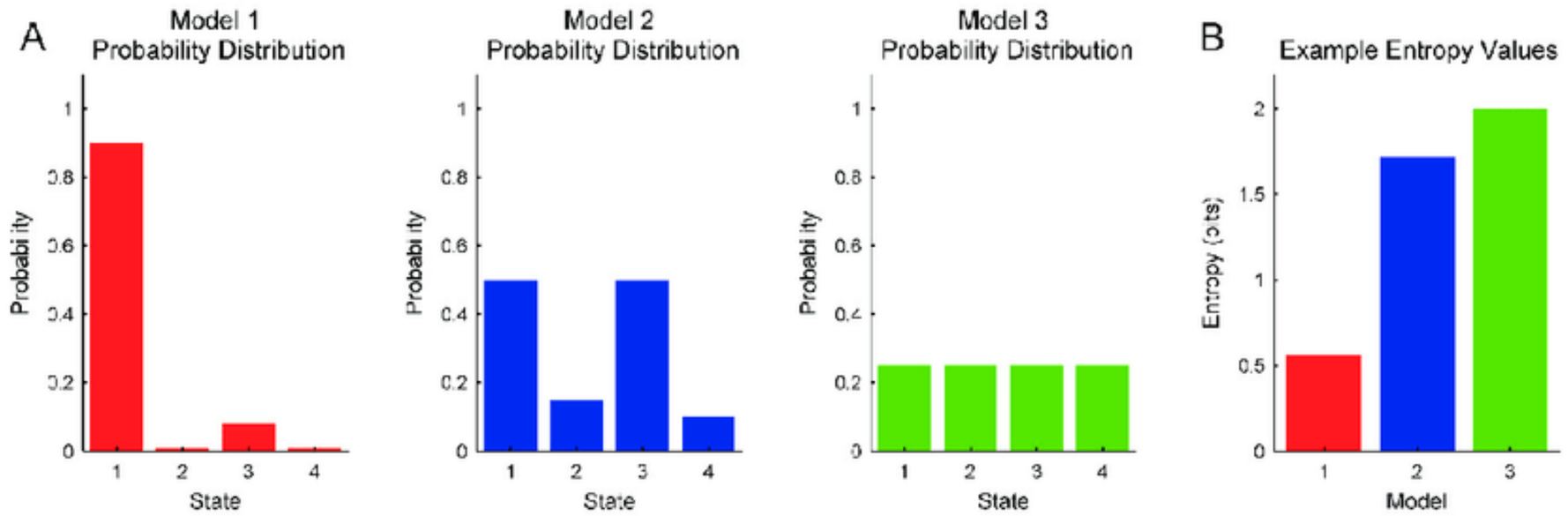


Sanh et al. (2019)

Entropy

- Entropy measures the inherent randomness or uncertainty of a single probability distribution

$$H(p) = -\sum_{i=1}^{n} p_i \log p_i$$



Cross-Entropy

- Cross Entropy is an extension of the concept of entropy, when two different probability distributions are present

$$H(p, q) = -\sum_{i=1}^{n} p_i \log q_i$$

p_1	1.00
p_2	0.00
p_3	0.00

q_1	0.21
q_2	0.77
q_3	0.02

= - 1.00 * log(0.21) = log(0.21)

(Recap) Multiclass Logistic Regression

$$P_w(y|x) = \frac{\exp\left(w^\top f(x,y)\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(w^\top f(x,y')\right)}$$

sum over output space to normalize

too many drug trials, too few patients

Health: +1.8

Sports: +3.1

Science: -0.6

$w^\top f(x,y)$

exp

probabilities must be >= 0

6.05

22.2

0.55

unnormalized probabilities

normalize

probabilities must sum to 1

0.21

0.77

0.02

$P_w(y|x)$

log(0.21) = - 1.56

compare

$\mathcal{L}(x_j, y_j^*) = \log P(y_j^*|x_j)$

1.00

0.00

0.00

correct (gold) probabilities

(Recap) Multiclass Logistic Regression

$$P_w(y|x) = \frac{\exp\left(w^\top f(x,y)\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(w^\top f(x,y')\right)}$$

sum over output space to normalize

i.e. minimize negative log likelihood or **cross-entropy loss**

- Training: maximize $\mathcal{L}(x,y) = \sum_{j=1}^{m} \log P(y_j^*|x_j)$

index of data points (*j*)

Knowledge Distillation

Distilling the Knowledge in a Neural Network

Geoffrey Hinton[*†]
Google Inc.
Mountain View
geoffhinton@google.com

Oriol Vinyals[†]
Google Inc.
Mountain View
vinyals@google.com

Jeff Dean
Google Inc.
Mountain View
jeff@google.com

Abstract

A very simple way to improve the performance of almost any machine learning algorithm is to train many different models on the same data and then to average their predictions [3]. Unfortunately, making predictions using a whole ensemble of models is cumbersome and may be too computationally expensive to allow deployment to a large number of users, especially if the individual models are large neural nets. Caruana and his collaborators [1] have shown that it is possible to compress the knowledge in an ensemble into a single model which is much easier to deploy and we develop this approach further using a different compression technique. We achieve some surprising results on MNIST and we show that we can significantly improve the acoustic model of a heavily used commercial system by distilling the knowledge in an ensemble of models into a single model. We also introduce a new type of ensemble composed of one or more full models and many specialist models which learn to distinguish fine-grained classes that the full models confuse. Unlike a mixture of experts, these specialist models can be trained rapidly and in parallel.

1 Introduction

Many insects have a larval form that is optimized for extracting energy and nutrients from the environment and a completely different adult form that is optimized for the very different requirements of traveling and reproduction. In large-scale machine learning, we typically use very similar models for the training stage and the deployment stage despite their very different requirements: For tasks like speech and object recognition, training must extract structure from very large, highly redundant datasets but it does not need to operate in real time and it can use a huge amount of computation. Deployment to a large number of users, however, has much more stringent requirements on latency and computational resources. The analogy with insects suggests that we should be willing to train very cumbersome models if that makes it easier to extract structure from the data. The cumbersome model could be an ensemble of separately trained models or a single very large model trained with a very strong regularizer such as dropout [9]. Once the cumbersome model has been trained, we can then use a different kind of training, which we call "distillation" to transfer the knowledge from the cumbersome model to a small model that is more suitable for deployment. A version of this strategy has already been pioneered by Rich Caruana and his collaborators [1]. In their important paper they demonstrate convincingly that the knowledge acquired by a large ensemble of models can be transferred to a single small model.

A conceptual block that may have prevented more investigation of this very promising approach is that we tend to identify the knowledge in a trained model with the learned parameter values and this makes it hard to see how we can change the form of the model but keep the same knowledge. A more abstract view of the knowledge, that frees it from any particular instantiation, is that it is a learned

[*]Also affiliated with the University of Toronto and the Canadian Institute for Advanced Research.
[†]Equal contribution.

Hinton et al. (2015)

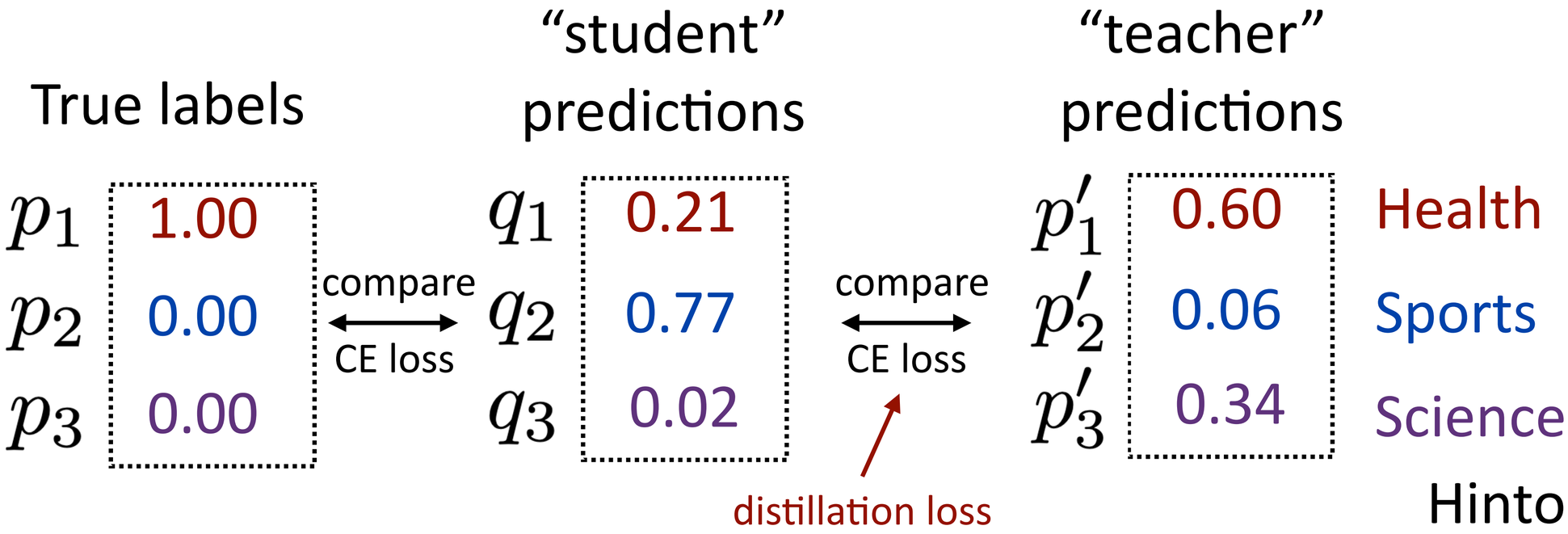
Knowledge Distillation

- Compress a large and complex model (the teacher model) into a smaller and simpler model (the student model)

Hinton et al. (2015)

Knowledge Distillation

- Compress a large and complex model (the teacher model) into a smaller and simpler model (the student model)
- The relative probabilities of incorrect answers tell us a lot about how the large model tends to generalize.



Hinton et al. (2015)

Knowledge Distillation

- Formally, NNs typically use a softmax function to convert logits z_i into class probabilities:

temperature to smooth softmax distribution

$$P(z_i, T) = \frac{\exp(z_i/T)}{\sum_j \exp(z_j/T)}$$

index of output space $\mathcal{Y}$

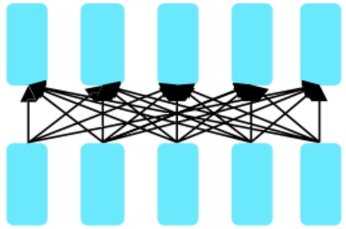
- One classic way of knowledge distillation is to align the class probability distribution from teacher and student networks.
- More advanced KD include aligning intermediate weights, gradients of attention maps, sparsity patterns after ReLU activation, etc.

Hinton et al. (2015)

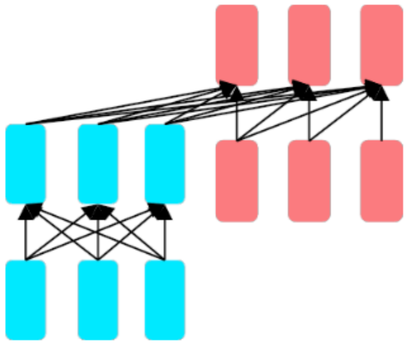
This Lecture

- Decoder-only LMs: GPT / GPT-2 / GPT-3
- Instruction-tuning: T0/Flan/PaLM
- Decoding strategies (if time)

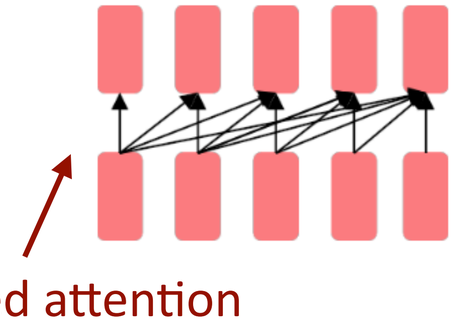
3 main types of Transformer LMs



- Encoders:
  - e.g., BERT, RoBERTa, mmBERT
  - captures bidirectional context
  - trained with masked/denoising LM objectives

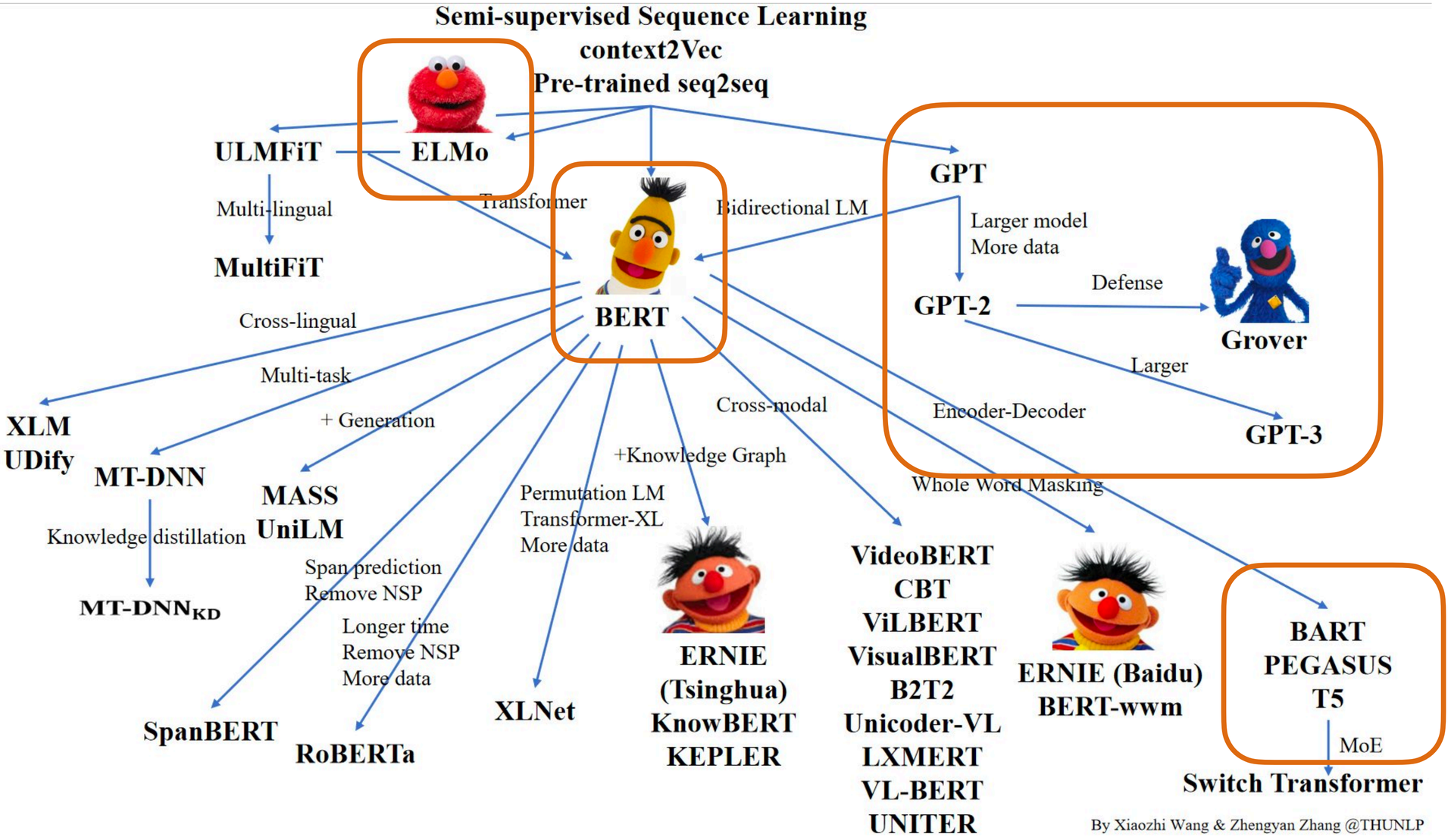


- Encoder-Decoders:
  - e.g., BART, T5



- Decoders:
  - e.g., GPT, LLaMA, most frontier LMs
  - Other names: casual/generative/auto-regressive LMs

Explosion of Pre-trained LMs

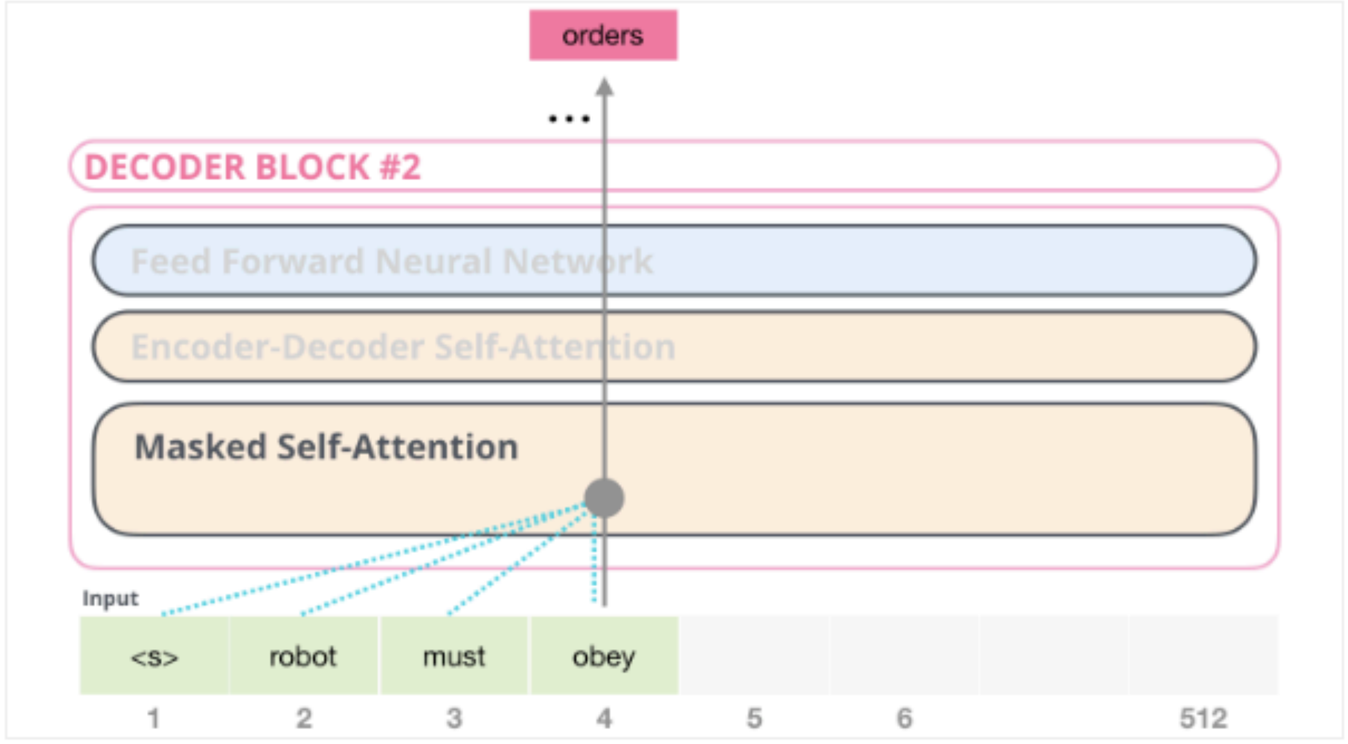


(This is only by 09/2019! ELMo late 2017, BERT 10/2018; ChatGPT 11/2022)

GPT/GPT2

OpenAI GPT/GPT2

- "ELMo with transformers" (works better than ELMo)
- Train a single unidirectional transformer LM on long contexts
- Masked self-attention: each token can only attend to past tokens



Radford et al. (2019)

OpenAI GPT/GPT2

- GPT2: trained on 40GB of text collected from upvoted links from reddit
- 1.5B parameters — the largest of these models trained as of March 2019

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

- Because it's a language model, we can **generate** from it

Radford et al. (2019)

OpenAI GPT2

SYSTEM PROMPT (HUMAN-WRITTEN)	*Miley Cyrus was caught shoplifting from Abercrombie and Fitch on Hollywood Boulevard today.*
MODEL COMPLETION (MACHINE-WRITTEN, SECOND TRY)	The 19-year-old singer was caught on camera being escorted out of the store by security guards. The singer was wearing a black hoodie with the label 'Blurred Lines' on the front and 'Fashion Police' on the back. Scroll down for video Shoplifting: Miley Cyrus was caught shoplifting from Abercrombie and Fitch on Hollywood Boulevard today (pictured) The singer was wearing a black hoodie with the label 'Blurred Lines' on the front and 'Fashion Police' on the back The singer was also wearing a pair of black-rimmed glasses, a black jacket, black jeans and black sandals. She was carrying a pair of black and white striped gloves and a small black bag.

slide credit: OpenAI

Open Questions

1) How novel is the stuff being generated? (Is it just doing nearest neighbors on a large corpus?)

2) How do we understand and distill what is learned in this model?

3) How do we harness these priors for conditional generation tasks (summarization, generate a report of a basketball game, etc.)

4) Is this technology dangerous? (OpenAI pursued a “staged release”)

Ethical Considerations

Grover

- Sample from a large language model conditioned on a domain, date, authors, and headline

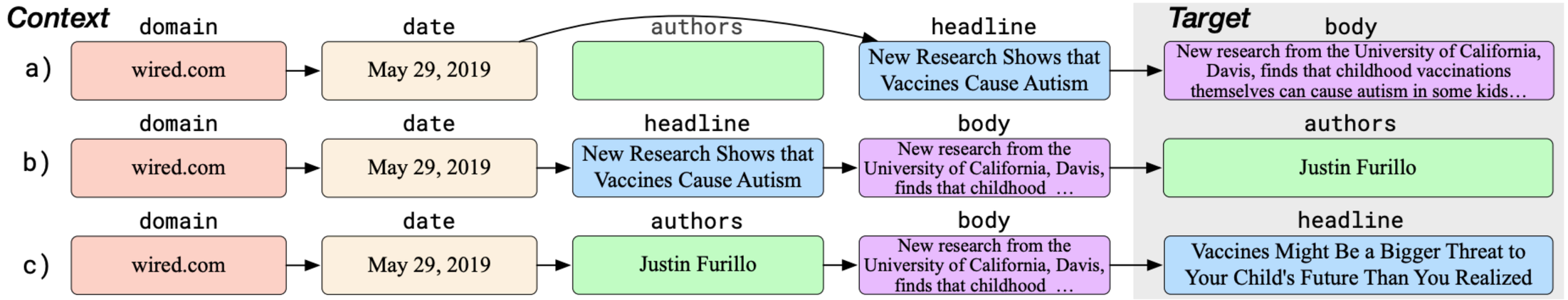


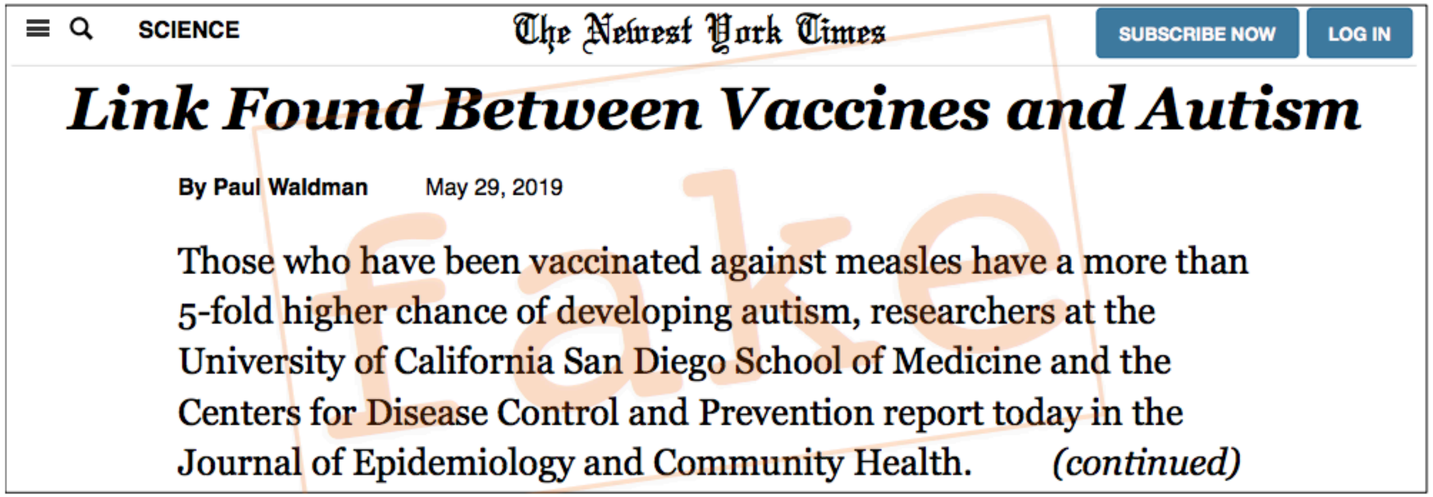
Figure 2: A diagram of three GROVER examples for article generation. In row a), the body is generated from partial context (the authors field is missing). In b), the model generates the authors. In c), the model uses the new generations to regenerate the provided headline to one that is more realistic.

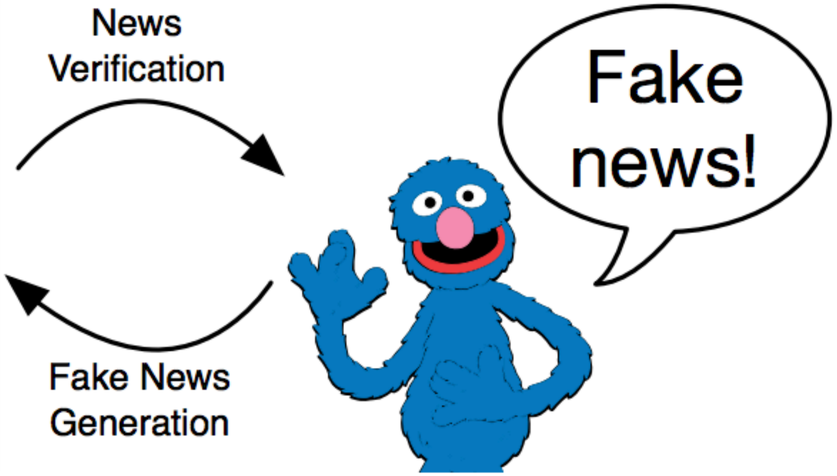
- NOTE: Not a GAN, discriminator trained separately from the generator

Zellers et al. (2019)

Grover

- Humans rank Grover-generated propaganda as more realistic than real “fake news”



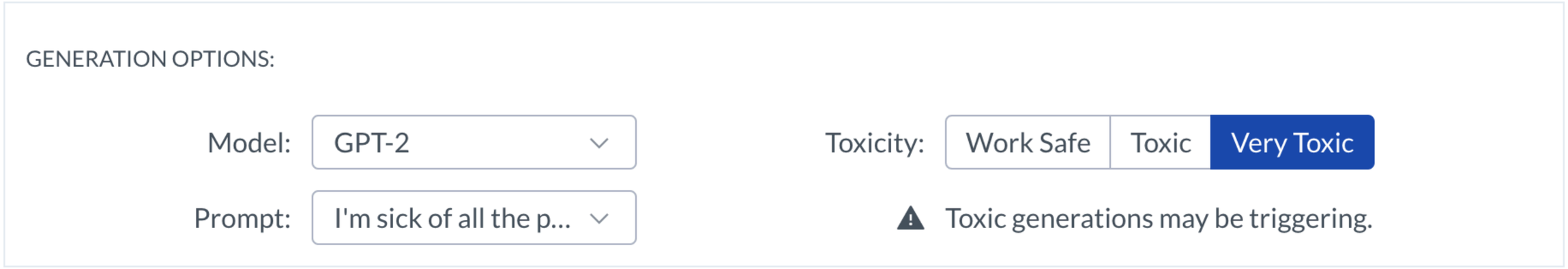


- Fine-tuned Grover can detect Grover propaganda easily — authors argue for releasing it for this reason

Zellers et al. (2019)

Bias and Toxicity

- "Toxic degeneration": systems that generate toxic stuff



I'm sick of all the politically correct talk and crying and looking down at your pathetic self and wishing you could just get outta there and...|

- System trained on a big chunk of the Internet: conditioning on "SJW", "black" gives the system a chance of recalling bad stuff from its training data

https://toxicdegeneration.allenai.org/

Pre-Training Cost (with Google/AWS)

- BERT: Base $500, Large (340M parameters) $7000
- Grover-MEGA (1.5B parameters): $25,000
- XLNet (BERT variant): $30,000 — $60,000 (unclear)
- This is for a single pre-training run…developing new pre-training techniques may require many runs
- *Fine-tuning* these models can typically be done with a single GPU (but may take 1-3 days for medium-sized datasets)

https://syncedreview.com/2019/06/27/the-staggering-cost-of-training-sota-ai-models/

Pre-Training Cost (with Google/AWS)

- GPT-3: estimated to be $4~10M. This cost has a large carbon footprint
  - Carbon footprint: equivalent to driving 700,000 km by car (source: Anthropocene magazine)
  - (Counterpoints: GPT-3 isn't trained frequently, equivalent to 100 people traveling 7000 km for a conference, can use renewables)
- BERT-Base pre-training: carbon emissions roughly on the same order as a single passenger on a flight from NY to San Francisco

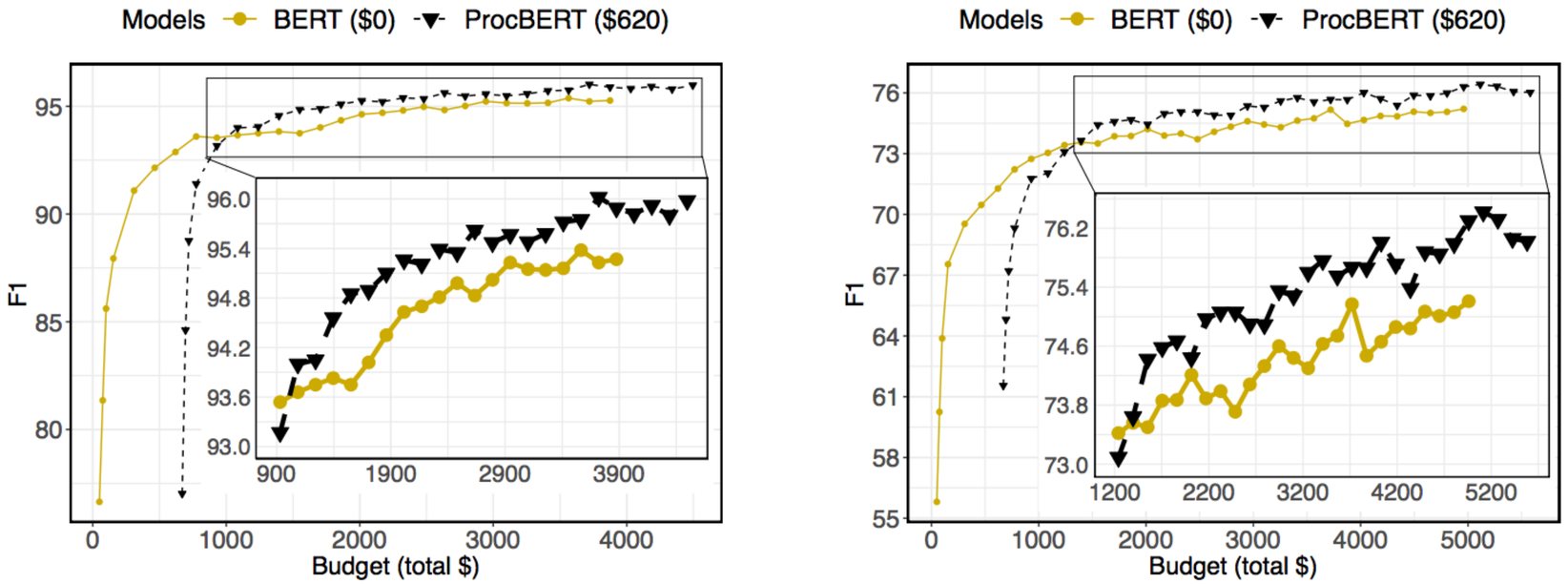
Strubell et al. (2019)

https://lambdalabs.com/blog/demystifying-gpt-3/

https://www.technologyreview.com/2019/06/06/239031/training-a-single-ai-model-can-emit-as-much-carbon-as-five-cars-in-their-lifetimes/

Pre-Training Cost (with Google/AWS)

- Cost-aware Domain Adaptation



(a) CHEMSYN (b) WLP

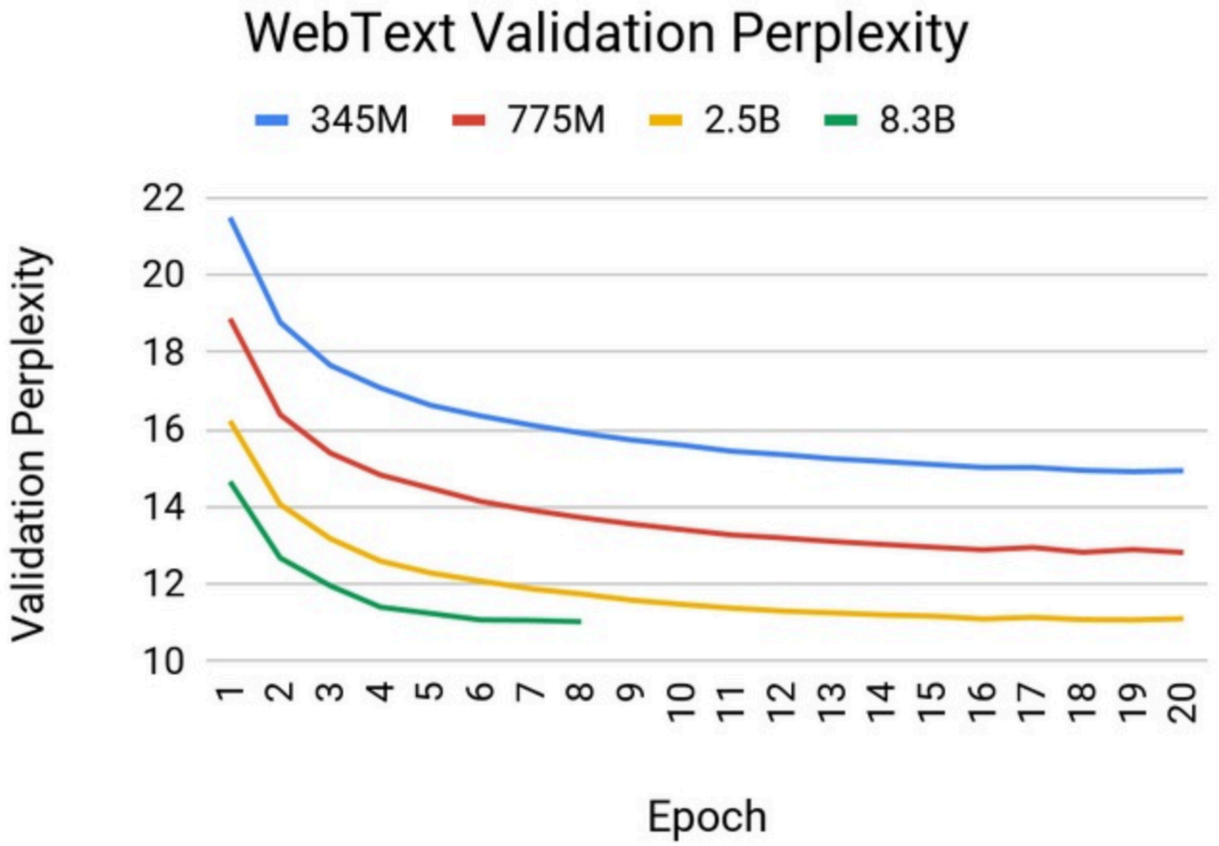
Figure 3: Comparison of spending the entire budget on data annotation (●) and pre-training followed by in-domain annotation (▼), where models are trained on **target domain labeled data only**. The crossover point for WLP moves from 775 USD (adapted from CHEMSYN) to around 1395 USD (WLP only) demonstrating that a large source domain dataset can reduce the need for target domain annotation.

Bai et al. (2021)

GPT-3

Scaling Up

- Question: what are the scaling limits of large language models?
- NVIDIA: trained 8.3B parameter GPT model (5.6x the size of GPT-2), showed lower perplexity from this
- Didn’t catch on and wasn't used for much



LM Evaluation - Perplexity

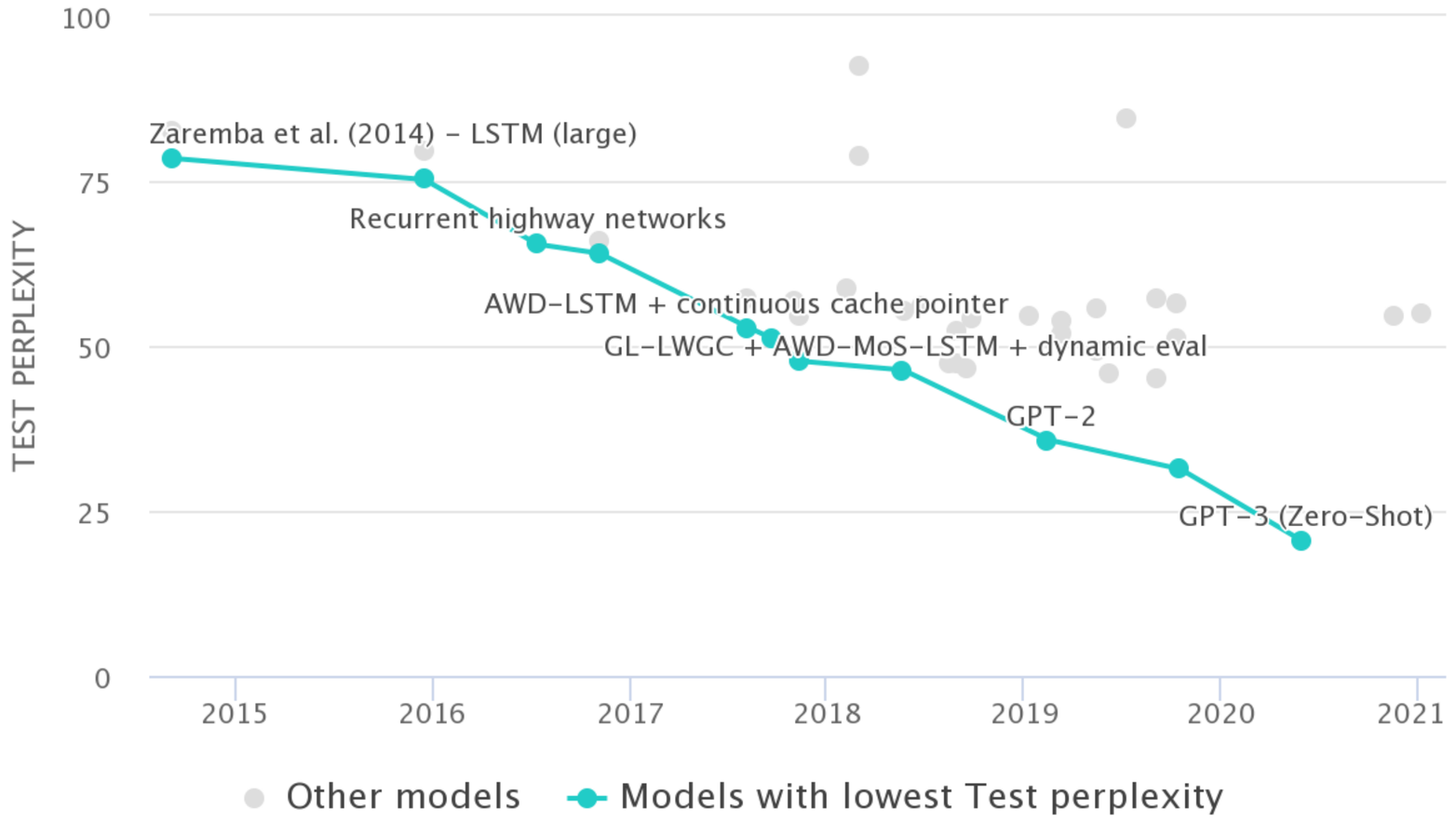
- Accuracy doesn't make sense — predicting the next word is generally impossible so accuracy values would be very low
- Evaluate LMs on the likelihood of held-out data (averaged to normalize for length)

$$\frac{1}{n}\sum_{i=1}^{n}\log P(w_i|w_1,\ldots,w_{i-1})$$

- Perplexity: exp(average negative log likelihood). Lower is better
  - Suppose we have probs 1/4, 1/3, 1/4, 1/3 for 4 predictions
  - Avg NLL (base e) = 1.242 Perplexity = 3.464 ← geometric mean of denominators

LM Evaluation - Perplexity

- The perplexity of modern language models have consistently been going down.



GPT-3 vs. GPT-2

- GPT-3 but even larger —> 175B parameter models (3640 PF-days)
- sparse factorizations of the attention matrix to reduce computing time and memory use. context window is set to 2048 tokens.
- Data: filtered Common Crawl (410B tokens downsampled x0.44) + WebText dataset (19B x2.9) + two Internet-based book corpora (12Bx1.9, 55Bx0.43) + English Wiki (3B upsampled x3.4)

https://twitter.com/cocoweixu/status/1285727605568811011

Brown et al. (2020)

GPT-3

- GPT-2 but even larger: 1.3B -> 175B parameter models

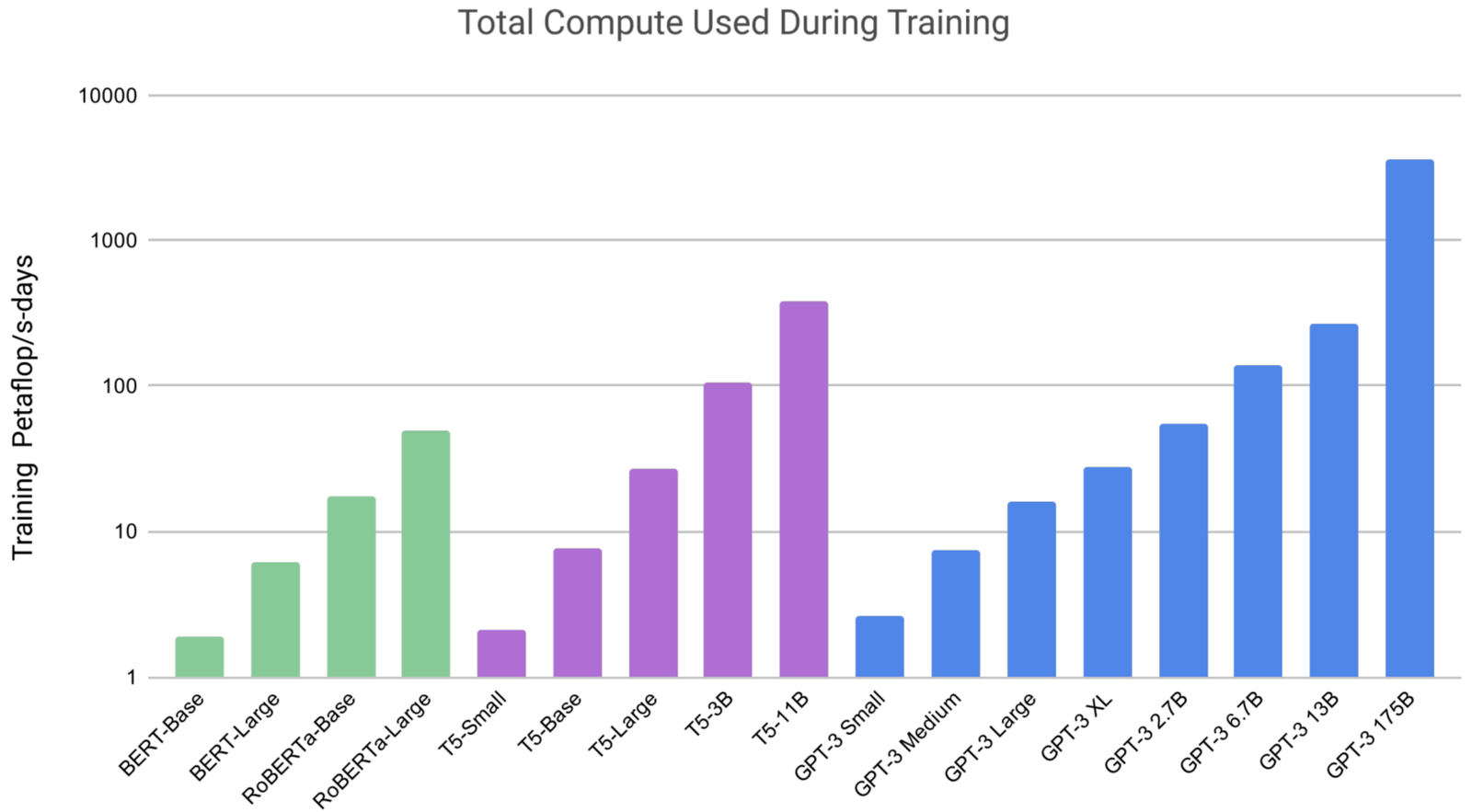
Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or “GPT-3”	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

- Trained on 570GB of Common Crawl
- 175B parameter model’s parameters alone take >400GB to store (4 bytes per param). Trained in parallel on a “high bandwidth cluster provided by Microsoft”

Brown et al. (2020)

Pre-training Cost

- Trained on Microsoft Azure, estimated to cost $4~10M (1000x BERT-large)

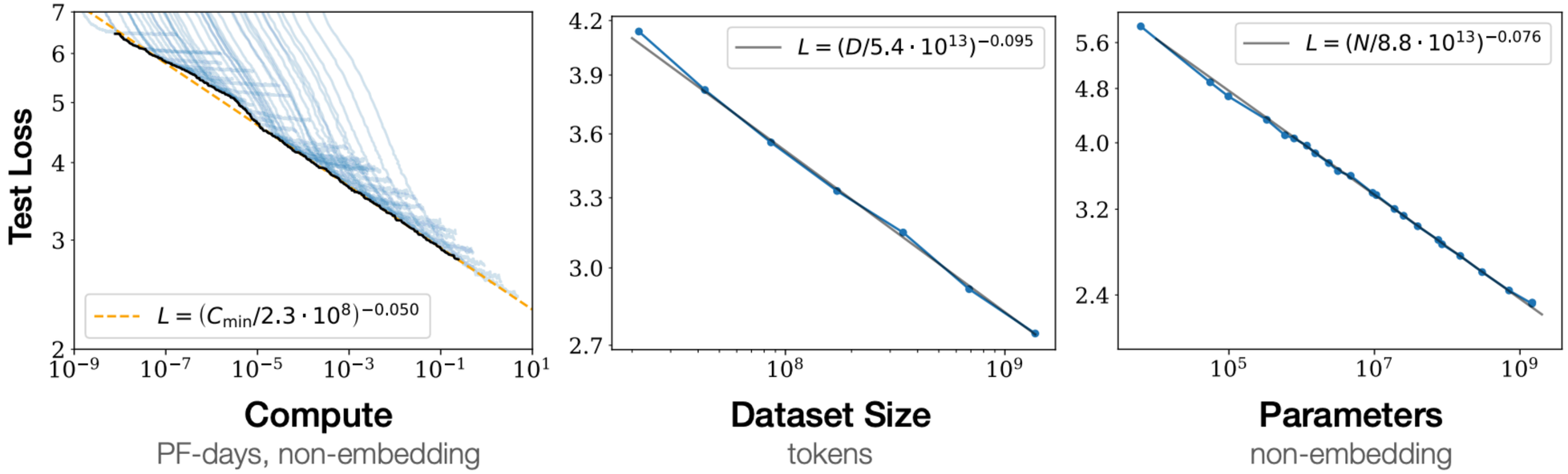


1 petaflop/s-day is equivalent to 8 V100 GPUs at full efficiency of a day

Brown et al. (2020)

Scaling Laws

- Each model is a different-sized LM (GPT-style)
- With more compute, larger models get further down the loss “frontier”
- Building a bigger model (increasing compute) will decrease test loss!



petaflop (10^{20})/s-days

Kaplan et al. (2020)

GPT-3

- This is the “normal way” of doing learning in models like GPT-2, BERT …

Brown et al. (2020)

GPT-3: Few-shot Prompting

- Model is frozen and is given a few demonstrations.

Few-shot

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

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

Brown et al. (2020)

GPT-3: Few-shot Prompting

- Model is frozen and is given a few demonstrations.

Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying off the national debt will be extremely painful. // Negative

The company anticipated its operating profit to improve. // _____

Circulation revenue has increased by 5% in Finland. // Finance

They defeated … in the NFC Championship Game. // Sports

Apple … development of in-house chips. // Tech

The company anticipated its operating profit to improve. // _____

- "in-context learning" - unlike conventional machine learning in that there's no optimization of any parameters.
- Model "learns" by conditioning on a few examples of the task.

Brown et al. (2020), Schick and Schütze (2021)

Prompt Engineering

Yelp For the Yelp Reviews Full Star dataset (Zhang et al., 2015), the task is to estimate the rating that a customer gave to a restaurant on a 1- to 5-star scale based on their review's text. We define the following patterns for an input text a:

$P_1(a) =$ It was ____. a $\quad P_2(a) =$ Just ____! $\|$ a

$P_3(a) =$ a. All in all, it was ____.

$P_4(a) =$ $a \,\|$ In summary, the restaurant is ____.

← patterns

We define a single verbalizer v for all patterns as

$v(1) =$ terrible $\quad v(2) =$ bad $\quad v(3) =$ okay

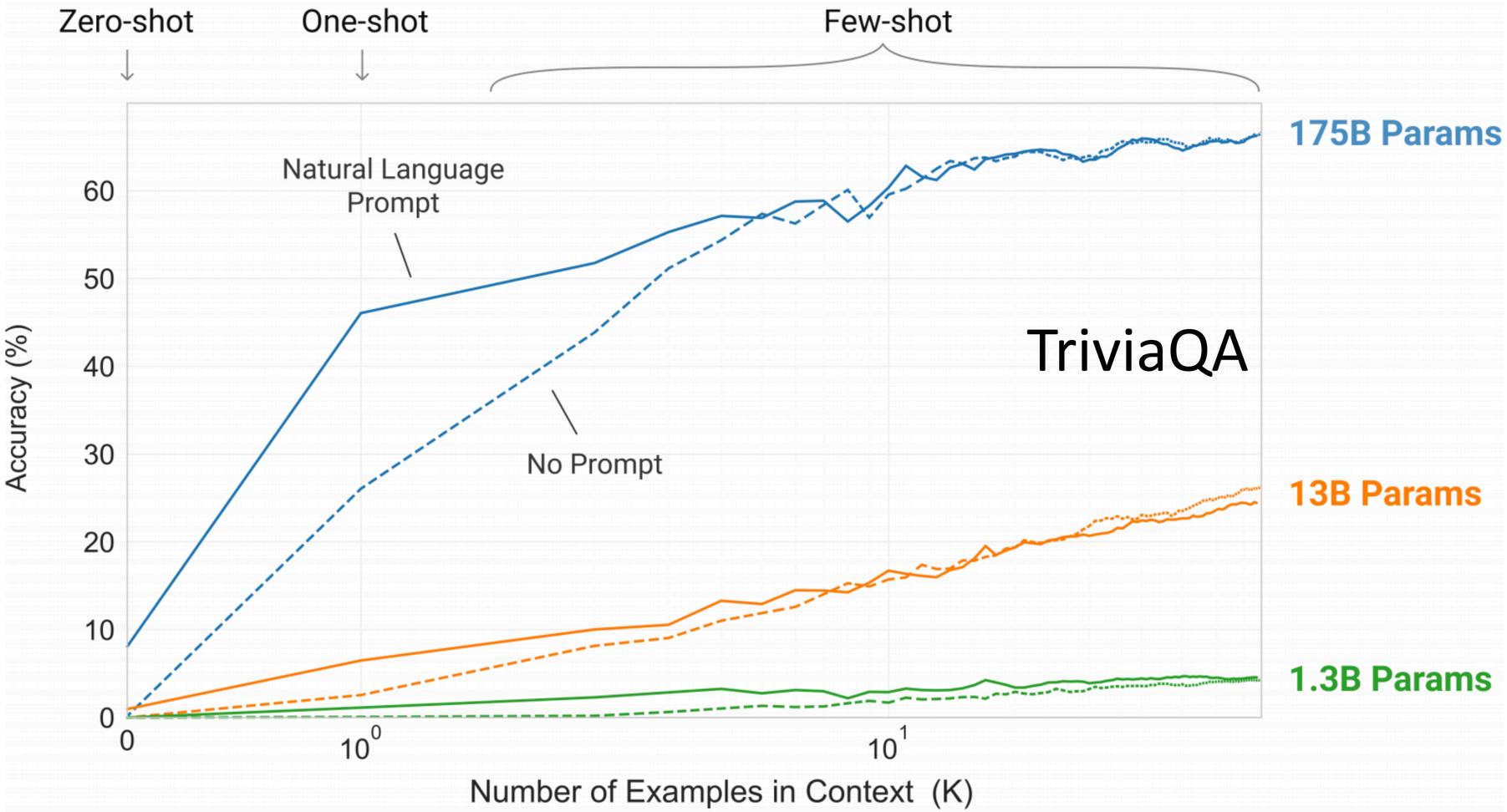
$v(4) =$ good $\quad v(5) =$ great

↖ "verbalizer" of labels

Schick and Schutze et al. (2020)

GPT-3: Few-shot Learning

- **Key observation:** few-shot learning only works with the very largest models!



Brown et al. (2020)

TriviaQA

Context →	Q: ‘Nude Descending A Staircase’ is perhaps the most famous painting by which 20th century artist? A:
Target Completion →	MARCEL DUCHAMP
Target Completion →	r mutt
Target Completion →	duchamp
Target Completion →	marcel duchamp
Target Completion →	R.Mutt
Target Completion →	Marcel duChamp
Target Completion →	Henri-Robert-Marcel Duchamp
Target Completion →	Marcel du Champ
Target Completion →	henri robert marcel duchamp
Target Completion →	Duchampian
Target Completion →	Duchamp
Target Completion →	duchampian
Target Completion →	marcel du champ
Target Completion →	Marcel Duchamp
Target Completion →	MARCEL DUCHAMP

Figure G.34: Formatted dataset example for TriviaQA. TriviaQA allows for multiple valid completions.

GPT-3

	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

- Sometimes very impressive, (MultiRC, ReCoRD), sometimes very bad
- Results on other datasets are equally mixed — but still strong for a few-shot model!

Brown et al. (2020)

MultiRC (multi-sentence)

Sent 1: The hijackers attacked at 9:28.
Sent 2: While traveling 35,000 feet above eastern Ohio, United 93 suddenly dropped 700 feet.
Sent 3: Eleven seconds into the descent, the FAA's air traffic control center in Cleveland received the first of two radio transmissions from the aircraft.
Sent 4: During the first broadcast, the captain or first officer could be heard declaring "Mayday" amid the sounds of a physical struggle in the cockpit.
Sent 5: The second radio transmission, 35 seconds later, indicated that the fight was continuing.
Sent 6: The captain or first officer could be heard shouting: "Hey get out of here-get out of here-get out of here."
Sent 7: On the morning of 9/11, there were only 37 passengers on United 93-33 in addition to the 4 hijackers.
Sent 8: This was below the norm for Tuesday mornings during the summer of 2001.
Sent 9: But there is no evidence that the hijackers manipulated passenger levels or purchased additional seats to facilitate their operation.
Sent 10: The terrorists who hijacked three other commercial flights on 9/11 operated in five-man teams.
Sent 11: They initiated their cockpit takeover within 30 minutes of takeoff.
Sent 12: On Flight 93, however, the takeover took place 46 minutes after takeoff and there were only four hijackers.

Question: Which two factors were different between the three other hijacked planes and United 93?
the day of the takeover
A) The amount of time that passed before the takeover started
B)* United 93 took longer and had less hijackers
C) The airline operating the planes
D) The weather and fuel used by the airplane
E) The navigation system used by the planes
Reasoning needed: Discourse relation (contrast)
One needs to identify that the discourse marker *however* in Sent 12 indicates a contrast relation between Flight 93 and the flights mentioned in Sent 10. Also, *only* in Sent 12 indicates that the number of hijackers were fewer than in the contrasted other flights.

Question: What was below average for this particular day?
A) the number of passengers in the first class.
B)* the number of passengers on board.
C) the number of hijackers
D) the amount of air traffic in the skies
E) the temperature
Reasoning needed: Event coreference
One needs to identify that *This* in Sent 8 co-refers to (event of) number of passengers in Sent 7. Note that Sent 12 contains *only four hijackers* and understanding that *only* indicates a smaller number of entities than expected (as in previous question), might mislead a system into believing that (C) is the correct answer.

Open Questions

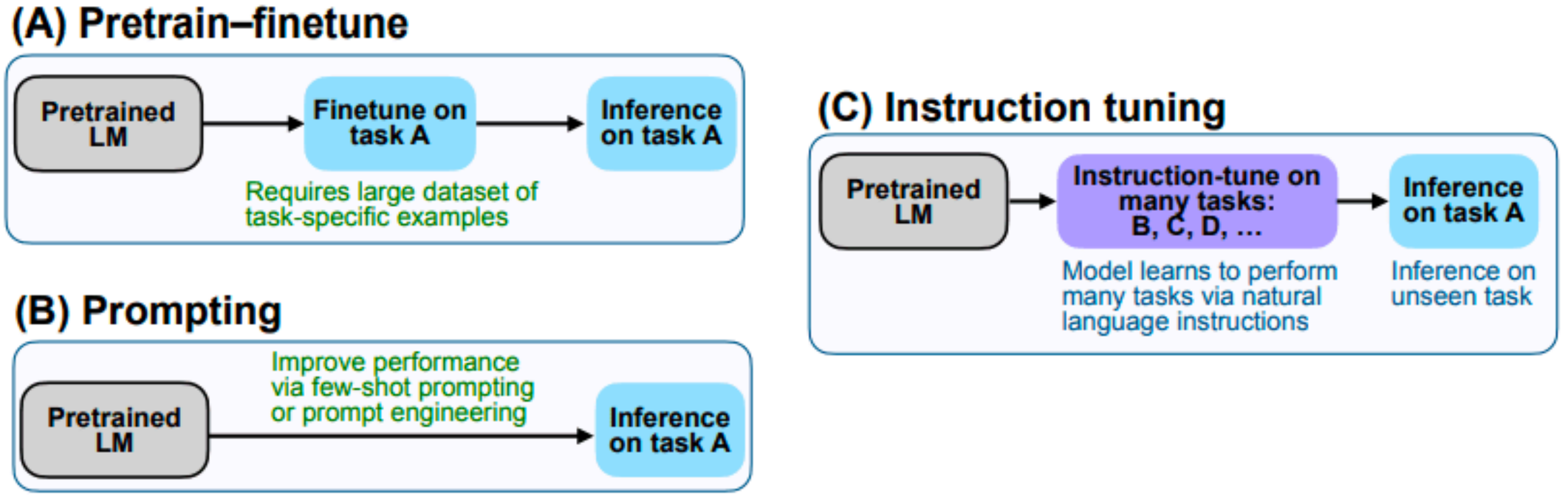
1) How much farther can we scale these models?

2) How do we get them to work for languages other than English?

3) Which will win out: prompting or fine-tuning?

New Models from 2022

Instruction Tuning



- We want to optimize models for P(answer | prompt, input), but they're learned on a basic language model objective.
- Instruction tuning: supervised fine-tuning on data derived from many NLP tasks (with natural language instructions in prompts)

Chung et al. (2022)

Instruction Tuning

- Early ideas from UnifiedQA (Khashabi et al. 2020) and Meta-tuning (Zhong et al. 2021)

Unified QA

Extractive [SQuAD]

Question: At what speed did the turbine operate?
Context: (Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...
Gold answer: 16,000 rpm

Abstractive [NarrativeQA]

Question: What does a drink from narcissus's spring cause the drinker to do?
Context: Mercury has awakened Echo, who weeps for Narcissus, and states that a drink from Narcissus's spring causes the drinkers to "Grow dotingly enamored of themselves." ...
Gold answer: fall in love with themselves

Multiple-Choice [ARC-challenge]

Question: What does photosynthesis produce that helps plants grow?
Candidate Answers: (A) water (B) oxygen (C) protein (D) sugar
Gold answer: sugar

Yes/No [BoolQ]

Question: Was America the first country to have a president?
Context: (President) The first usage of the word president to denote the highest official in a government was during the Commonwealth of England ...
Gold answer: no

EX	**Dataset**	SQuAD 1.1
	Input	`At what speed did the turbine operate? \n (Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...`
	Output	`16,000 rpm`
AB	**Dataset**	NarrativeQA
	Input	`What does a drink from narcissus's spring cause the drinker to do? \n Mercury has awakened Echo, who weeps for Narcissus, and states that a drink from Narcissus's spring causes the drinkers to ``Grow dotingly enamored of themselves.'' ...`
	Output	`fall in love with themselves`
MC	**Dataset**	ARC-challenge
	Input	`What does photosynthesis produce that helps plants grow? \n (A) water (B) oxygen (C) protein (D) sugar`
	Output	`sugar`
	Dataset	MCTest
	Input	`Who was Billy? \n (A) The skinny kid (B) A teacher (C) A little kid (D) The big kid \n Billy was like a king on the school yard. A king without a queen. He was the biggest kid in our grade, so he made all the rules during recess. ...`
	Output	`The big kid`
YN	**Dataset**	BoolQ
	Input	`Was America the first country to have a president? \n (President) The first usage of the word president to denote the highest official in a government was during the Commonwealth of England ...`
	Output	`no`

Khashabi et al. (2020)

Meta-Tuning

- Turn binary classification tasks into a “Yes"/“No" QA format

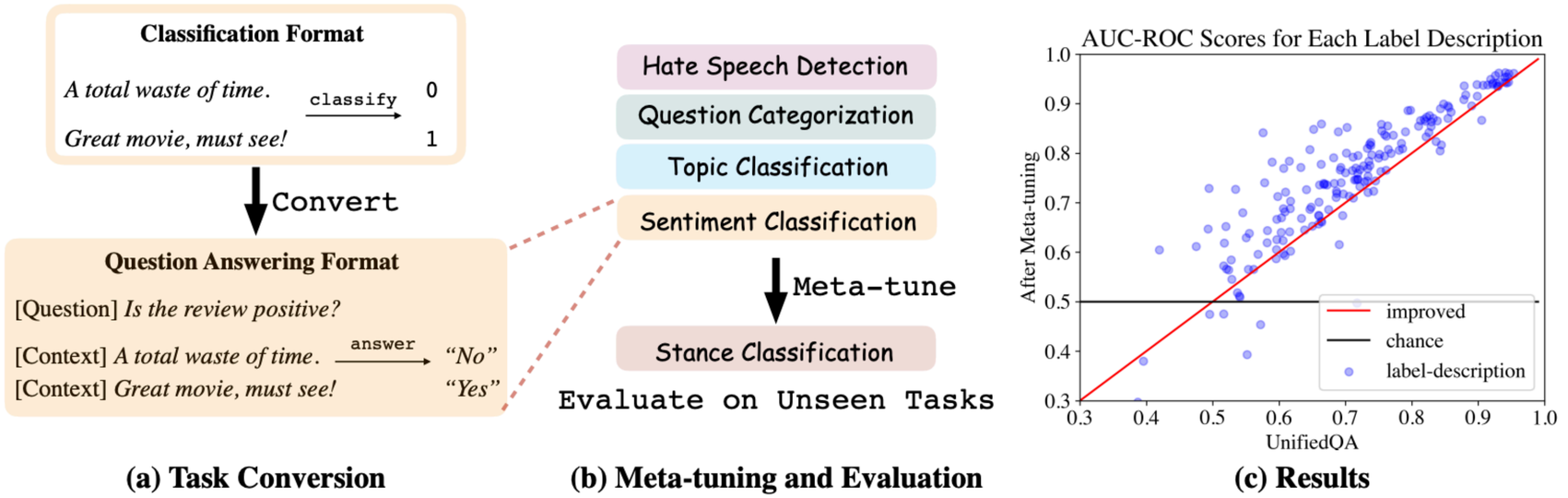


Figure 1: **(a)** We convert the format to question answering. We manually annotate label descriptions (questions) ourselves (Section 2). **(b)** We finetune the UnifiedQA (Khashabi et al., 2020) model (with 770 M parameters) on a diverse set of tasks (Section 4), and evaluate its 0-shot classification (ZSC) performance on an unseen task. **(c)** For each label description (question) we evaluate the AUC-ROC score for the “Yes" answer, and each dot represents a label description (Section 3). The x-value is the ZSC performance of UnifiedQA; the y-value is the performance after meta-tuning. In most cases, the y-value improves over the x-value (above the red line) and is better than random guesses (above the black line) by a robust margin (Section 5).

Zhong et al. (2021)

T0

- Extended from LM-adapted T5 model (Lester et al. 2021)
- “Instruction Tuning” — using existing labeled training datasets from many tasks + crowdsourced prompts

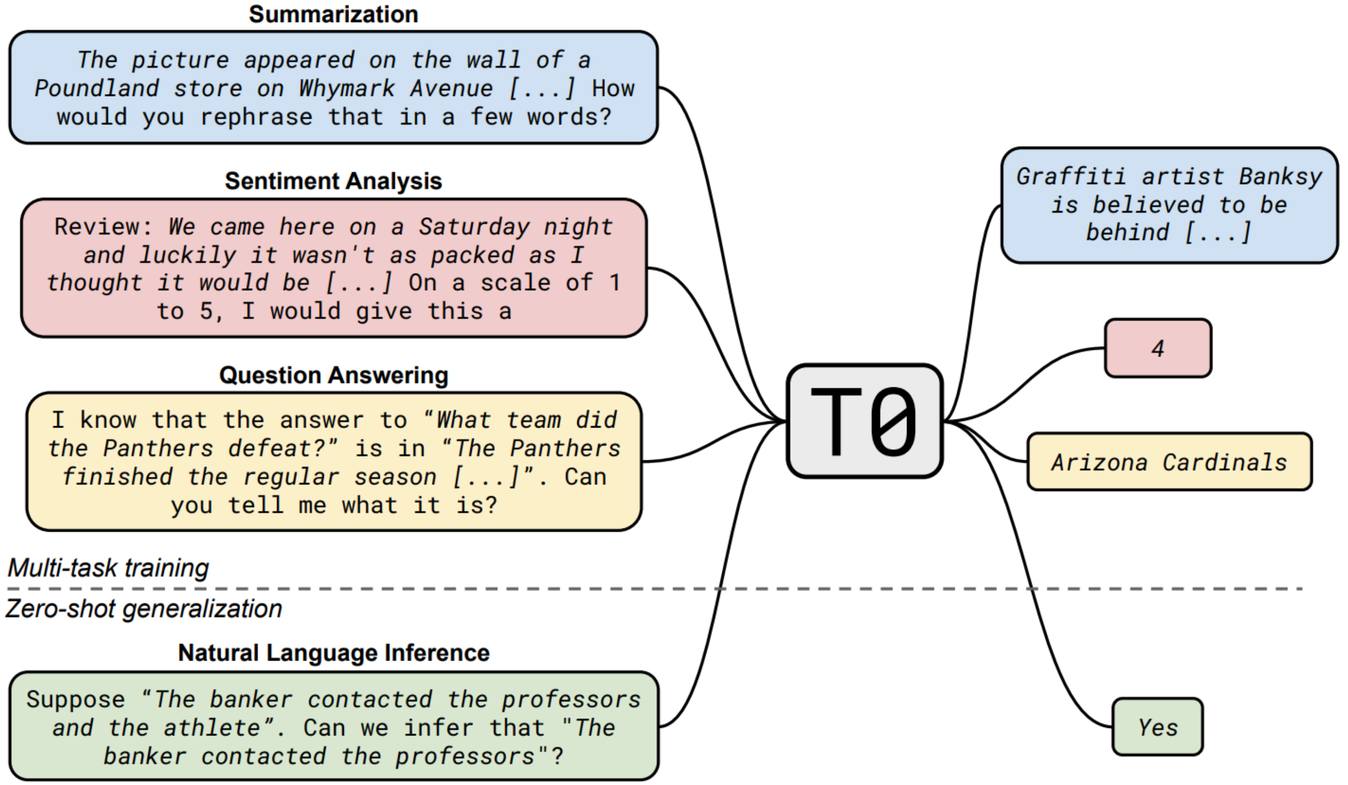
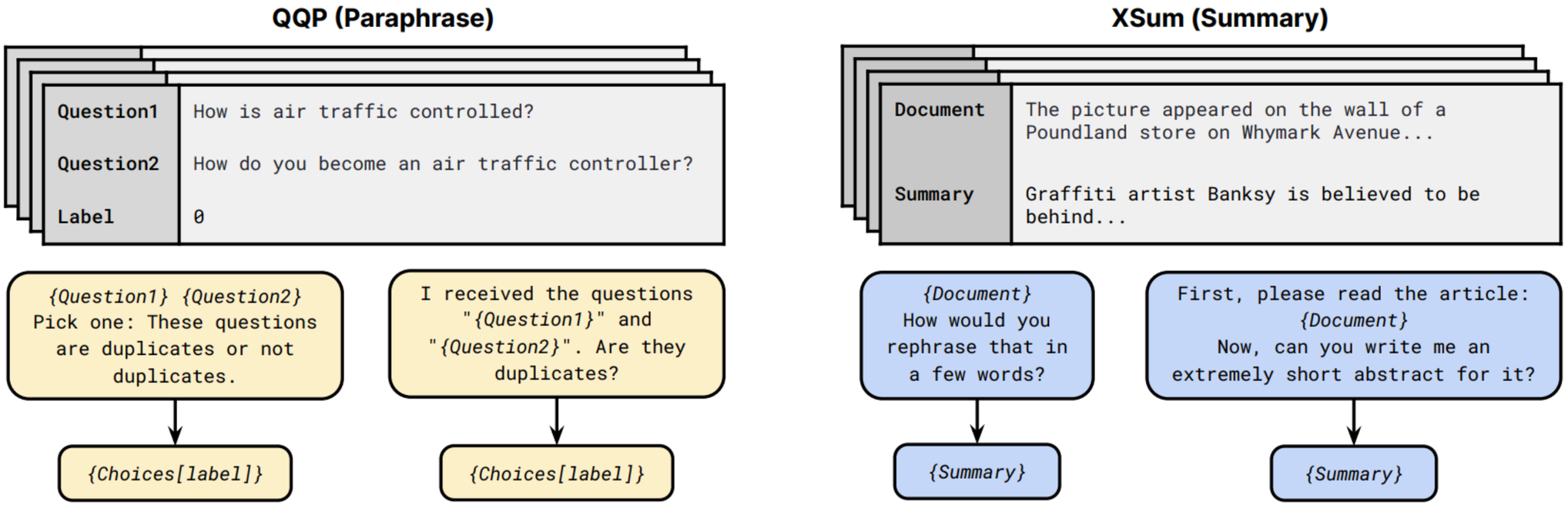


Figure 1: Our model and prompt format. T0 is an encoder-decoder model that consumes textual inputs and produces target responses. It is trained on a multitask mixture of NLP datasets partitioned into different tasks. Each dataset is associated with multiple prompt templates that are used to format example instances to input and target pairs. Italics indicate the inserted fields from the raw example data. After training on a diverse mixture of tasks (top), our model is evaluated on zero-shot generalization to tasks that are not seen during training (bottom).

Sanh et al. (2022)

Natural Language Prompts

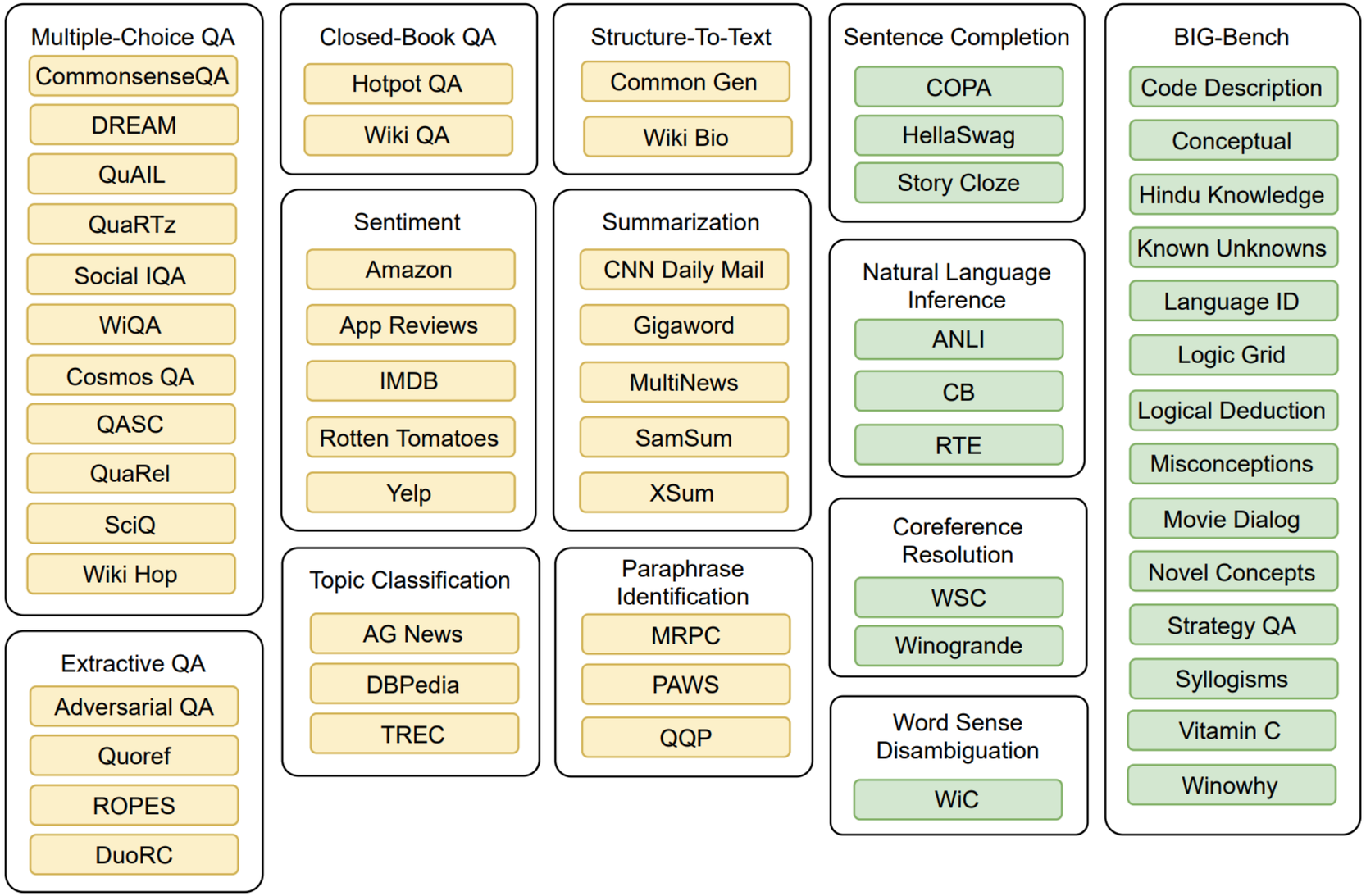
- Some examples from T0 paper:



Sanh et al. (2022)

Task Generalization: T0

- Pre-train: T5
- Train: a collection of tasks with prompts. **This uses existing labelled training data.**
- Test: a new task specified only by a new prompt. **No training data in this task.**



Flan

- Pre-train, then fine-tune on a bunch of tasks, generalize to unseen tasks
- Scaling the number of tasks, models size (Flan-T5, Flan-Palm), and fine-tuning on chain-of-thought data

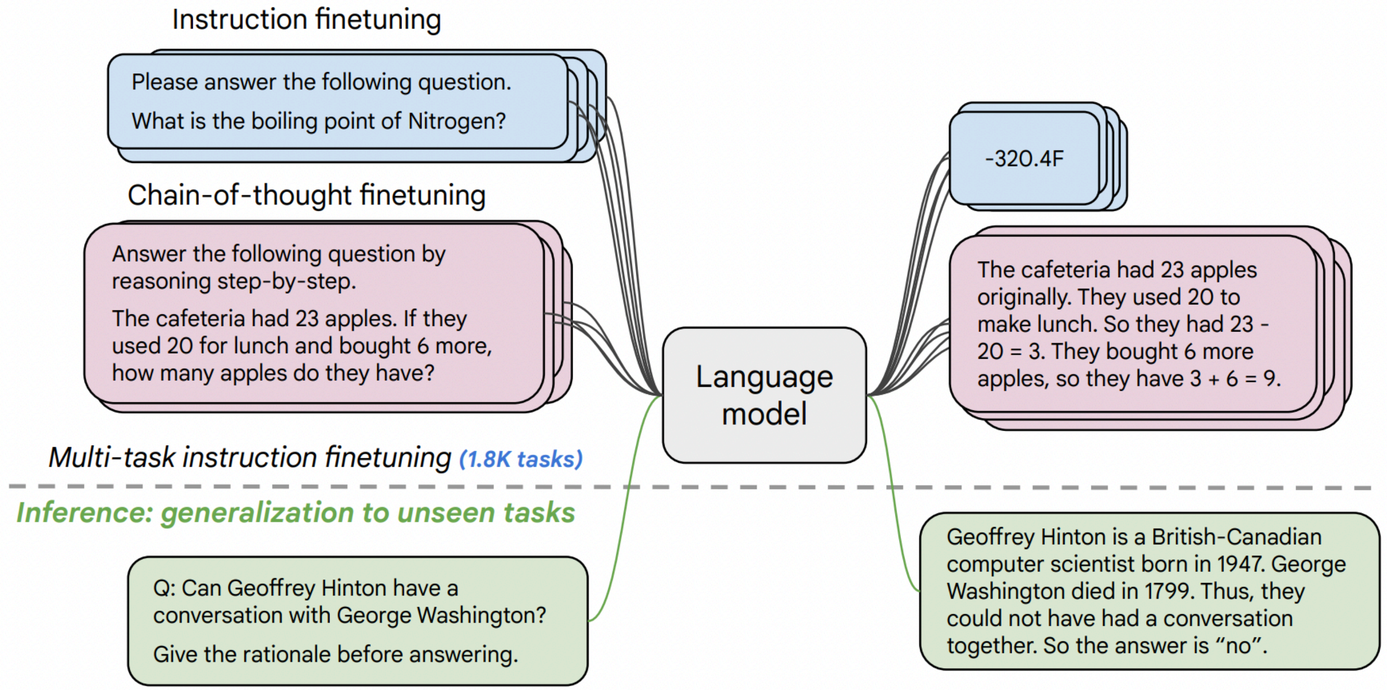
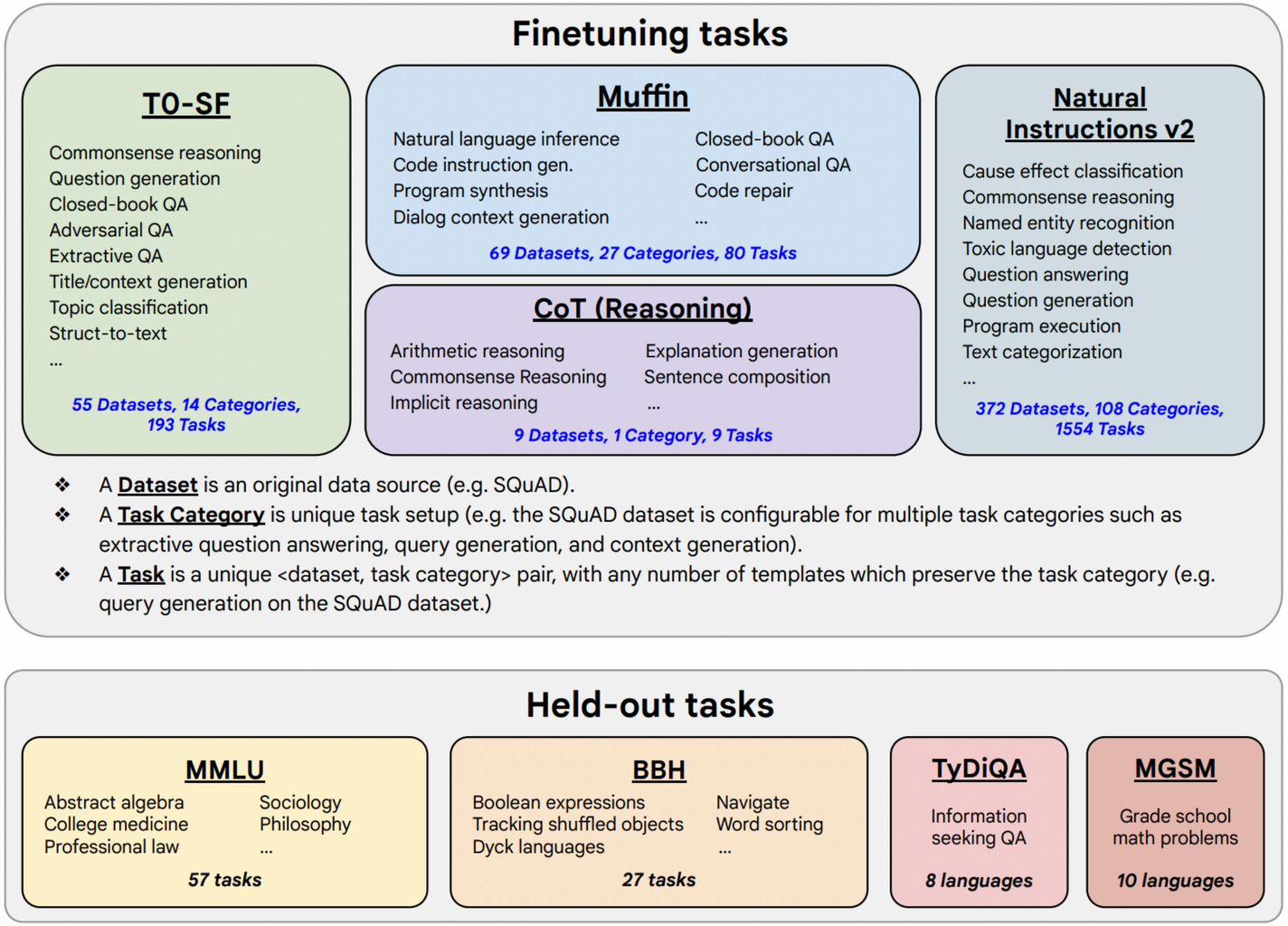


Figure 1: We finetune various language models on 1.8K tasks phrased as instructions, and evaluate them on unseen tasks. We finetune both with and without exemplars (i.e., zero-shot and few-shot) and with and without chain-of-thought, enabling generalization across a range of evaluation scenarios.

Chung et al. (2022)

Flan

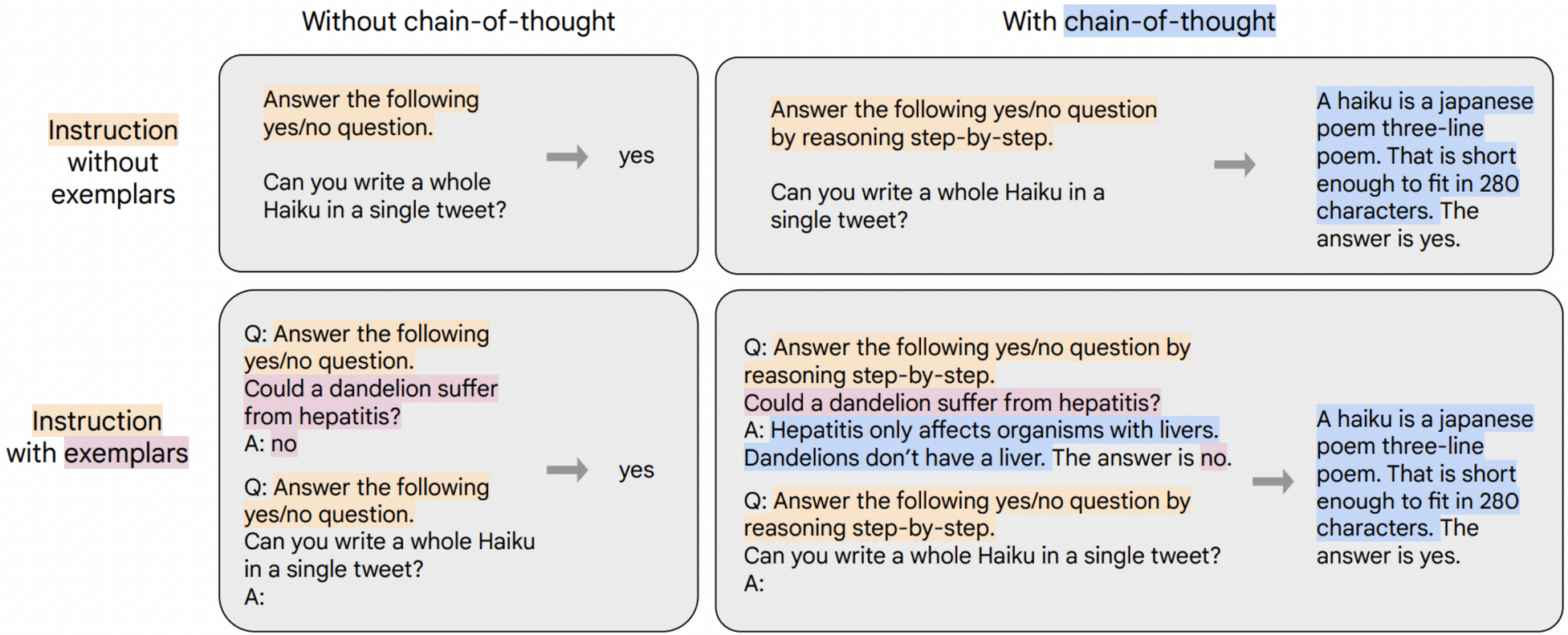


- Fine-tuned on 473 datasets, 1836 tasks.
- Some datasets support multiple tasks
- E.g. SQuAD can be used for QA or question generation.

Chung et al. (2022)

Chain-of-Thought Prompts

- Using explanations (some rationals) to improve model performance, usually in few-shot prompting



Wei et al. (2022)

Figure from Chung et al. (2022)

Flan

- Instruction fine-tuning can be done on various models (PaLM, T5, etc.)
- Flan-T5 models publicly available

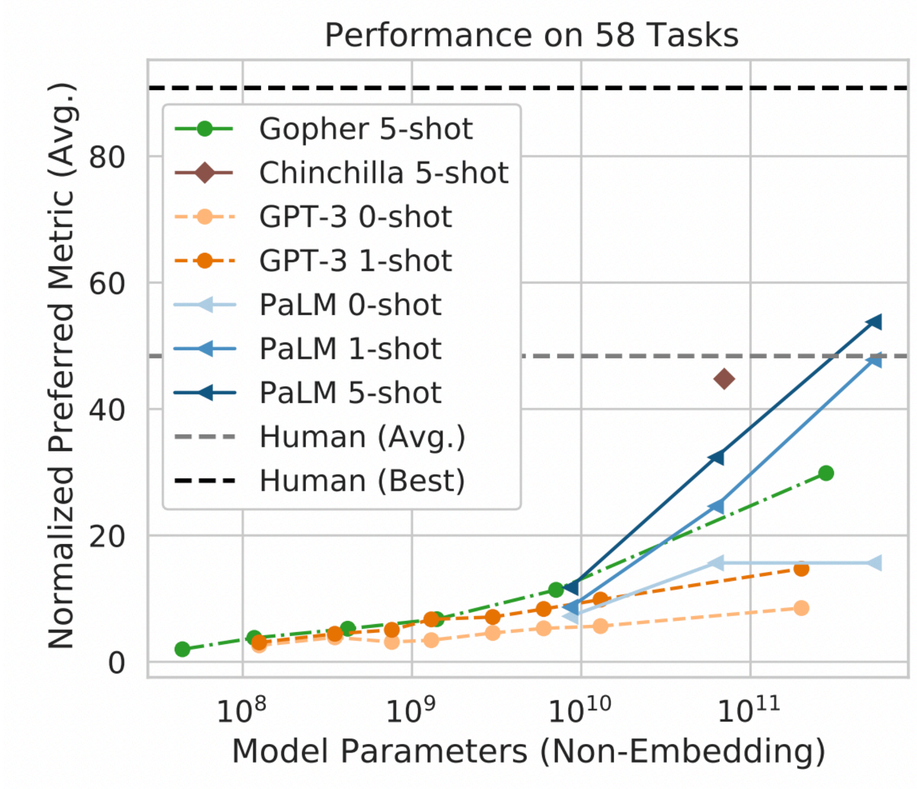
Params	Model	Arhitecture	pre-training Objective	Pretrain FLOPs	Finetune FLOPs	% Finetune Compute
80M	Flan-T5-Small	encoder-decoder	span corruption	1.8E+20	2.9E+18	1.6%
250M	Flan-T5-Base	encoder-decoder	span corruption	6.6E+20	9.1E+18	1.4%
780M	Flan-T5-Large	encoder-decoder	span corruption	2.3E+21	2.4E+19	1.1%
3B	Flan-T5-XL	encoder-decoder	span corruption	9.0E+21	5.6E+19	0.6%
11B	Flan-T5-XXL	encoder-decoder	span corruption	3.3E+22	7.6E+19	0.2%
8B	Flan-PaLM	decoder-only	causal LM	3.7E+22	1.6E+20	0.4%
62B	Flan-PaLM	decoder-only	causal LM	2.9E+23	1.2E+21	0.4%
540B	Flan-PaLM	decoder-only	causal LM	2.5E+24	5.6E+21	0.2%
62B	Flan-cont-PaLM	decoder-only	causal LM	4.8E+23	1.8E+21	0.4%
540B	Flan-U-PaLM	decoder-only	prefix LM + span corruption	2.5E+23	5.6E+21	0.2%

Table 2: Across several models, instruction finetuning only costs a small amount of compute relative to pre-training. T5: Raffel et al. (2020). PaLM and cont-PaLM (also known as PaLM 62B at 1.3T tokens): Chowdhery et al. (2022). U-PaLM: Tay et al. (2022b).

Chung et al. (2022)

PaLM

- 540 billion parameter model created by Google (not publicly available)
- Trained on 780 billion tokens, 6144 TPU v4 chips using Pathways to work across multiple TPU Pods).



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

Chowdhery et al. (2022)

PaLM

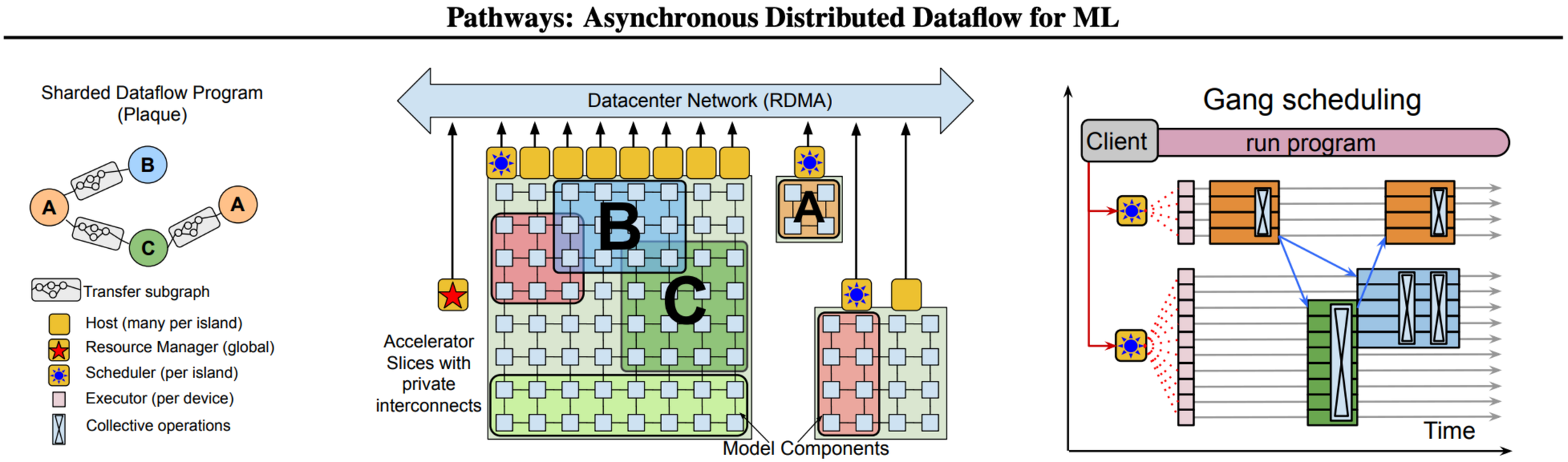


Figure 3. PATHWAYS system overview. (Left) Distributed computation expressed as a DAG where each node represents an individual compiled function, and edges between nodes represent data flows between functions. (Middle) Resource Manager allocates subsets of an island's accelerators ("virtual slices") for each compiled function. (Right) Centralized schedulers for each island gang-schedule computations that are then dispatched by per-shard executors. Red arrows indicate control messages, blue arrows show data-path transfers.

Barham et al. (2022)

PaLM

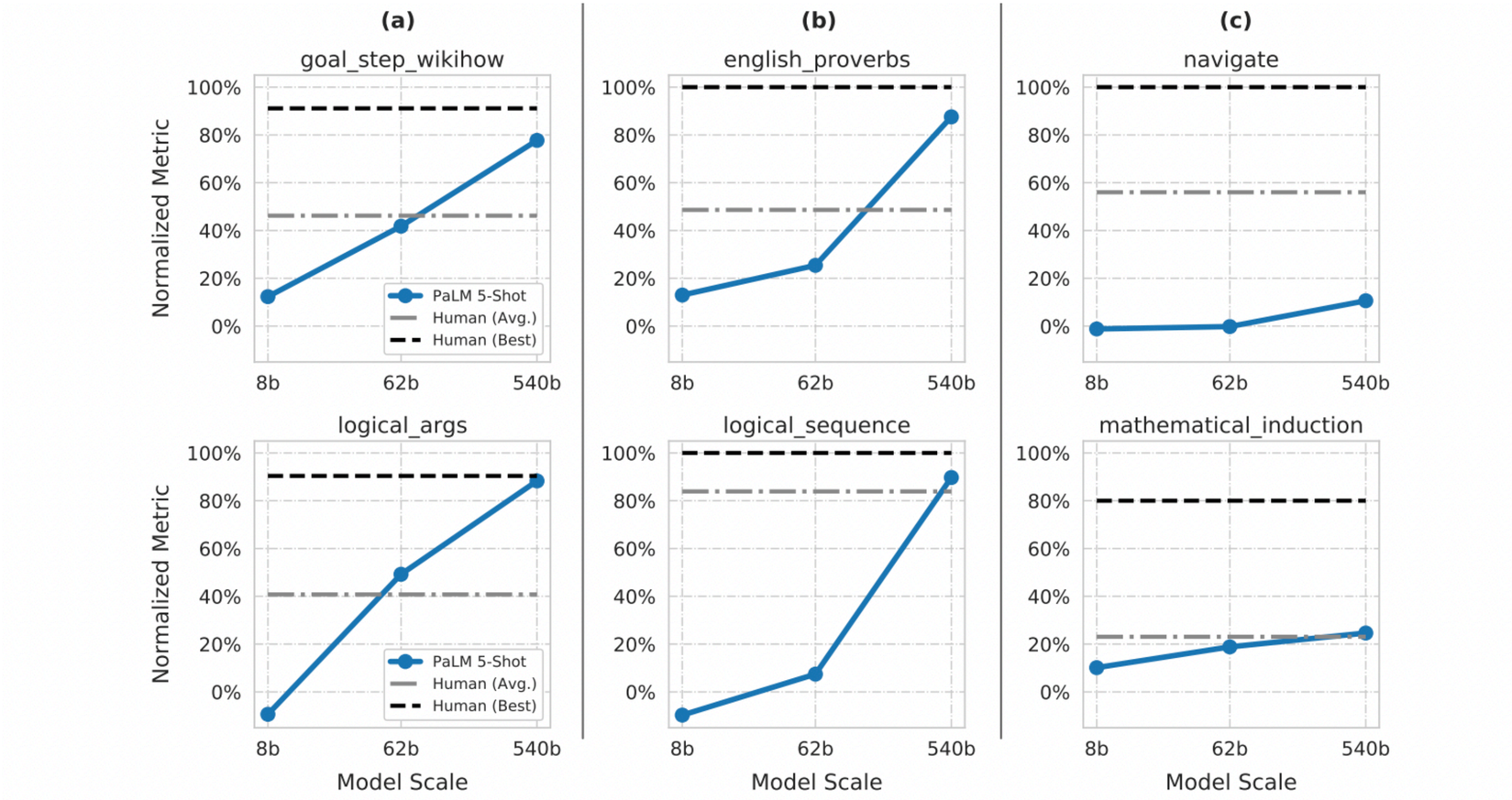


Figure 5: 5-shot evaluations on six individual tasks with interesting scaling characteristics. For each task, the "normalized preferred metric" is used, where 0% corresponds to random guessing for multiple choice tasks. (a) shows log-linear improvements with scale, (b) shows "discontinuous" improvements with scale, and (c) shows relatively flat improvements with scale.

Chowdhery et al. (2022)

PALM

Explaining a Joke

Input: Did you see that Google just hired an eloquent whale for their TPU team? It showed them how to communicate between two different pods!
Model Output: TPUs are a type of computer chip that Google uses for deep learning. A "pod" is a group of TPUs. A "pod" is also a group of whales. The joke is that the whale is able to communicate between two groups of whales, but the speaker is pretending that the whale is able to communicate between two groups of TPUs.

Logical Inference

Input: Shelley is from Virginia, but is visiting that city with that famous market where they throw the fish. Going home next Tuesday!
Question: Is it likely that Shelley will be near the Pacific Ocean this weekend?
Model Output: The city with the famous market where they throw the fish is Seattle, Washington. Seattle is on the Pacific Ocean. Shelley is visiting Seattle, so she will be near the Pacific Ocean this weekend. The answer is "yes", it is likely that Shelley will be near the Pacific Ocean this weekend.

Figure 1: These examples demonstrate exploratory capabilities in prompting PaLM to generate explanations using *chain-of-thought prompting* (Wei et al., 2022b). All examples were generated with greedy (1-best) decoding of PaLM 540B. The prompting was done with 2-shot exemplars, which are shown in Section 9, along with output on more examples using these same exemplars.

Chowdhery et al. (2022)

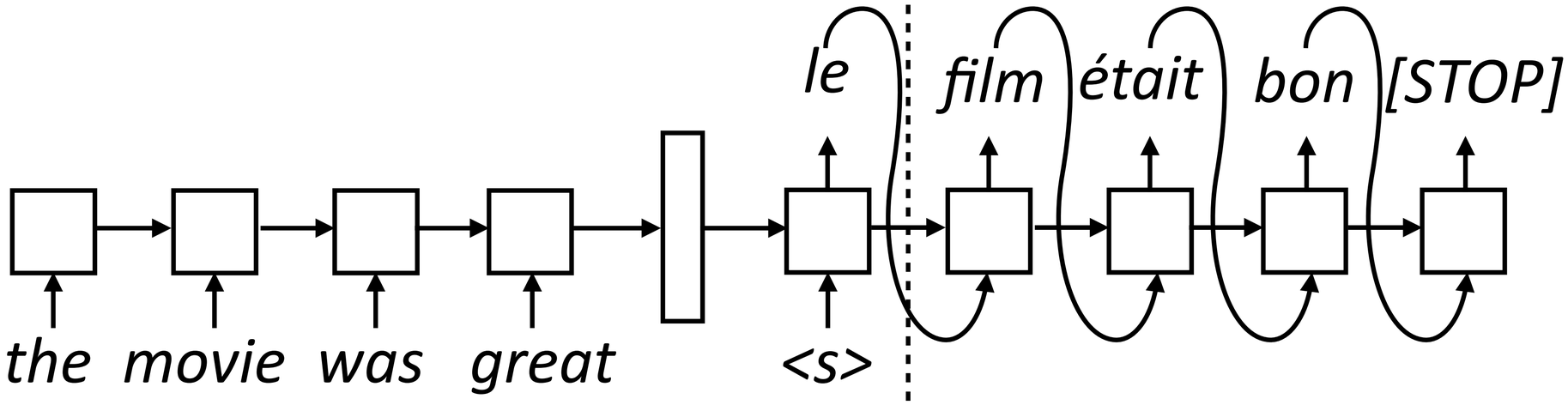
Decoding Strategies

Decoding Strategies

- LMs place a distribution $P(y_i | \ y_1, \ldots, y_{i-1})$
- seq2seq models place a distribution $P(y_i | \mathbf{x}, y_1, \ldots, y_{i-1})$
- Generation from both models looks similar; how do we do it?

(Recap) Greedy Decoding

- Generate next word conditioned on previous word as well as hidden state



- During inference: need to compute the argmax over the word predictions and then feed that to the next RNN state. This is **greedy decoding**

$$P(y_i|\mathbf{x}, y_1, \ldots, y_{i-1}) = \text{softmax}(W\bar{h})$$

$$y_{\text{pred}} = \text{argmax}_y P(y|\mathbf{x}, y_1, \ldots, y_{i-1})$$

Problems with Greedy Decoding

- Only returns one solution, and it may not be optimal
- Can address this with **beam search**, which usually works better…but even beam search may not find the correct answer! (max probability sequence)

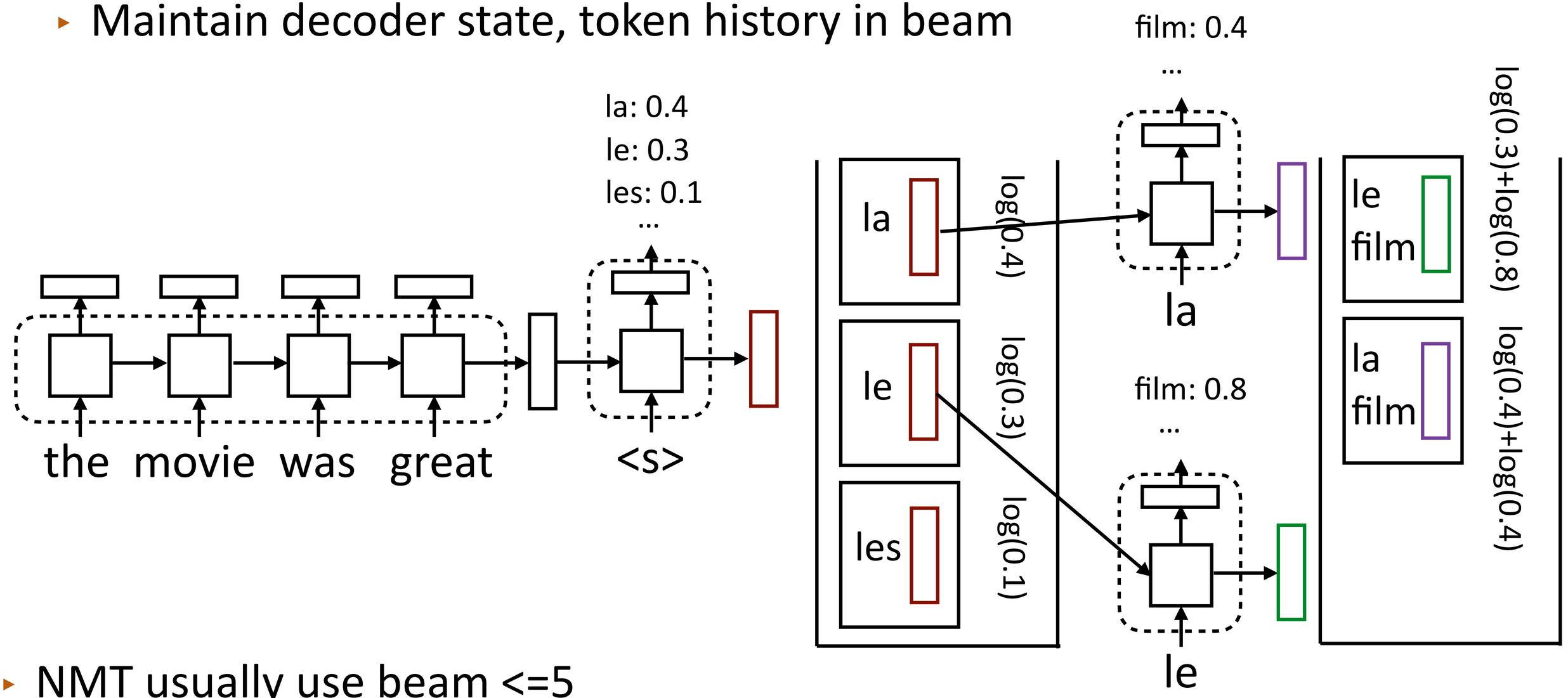
Model	Beam-10	
	BLEU	#Search err.
LSTM*	28.6	58.4%
SliceNet*	28.8	46.0%
Transformer-Base	30.3	57.7%
Transformer-Big*	31.7	32.1%

A sentence is classified as search error if the decoder does not find the global best model score.

Stahlberg and Byrne (2019)

Beam Search

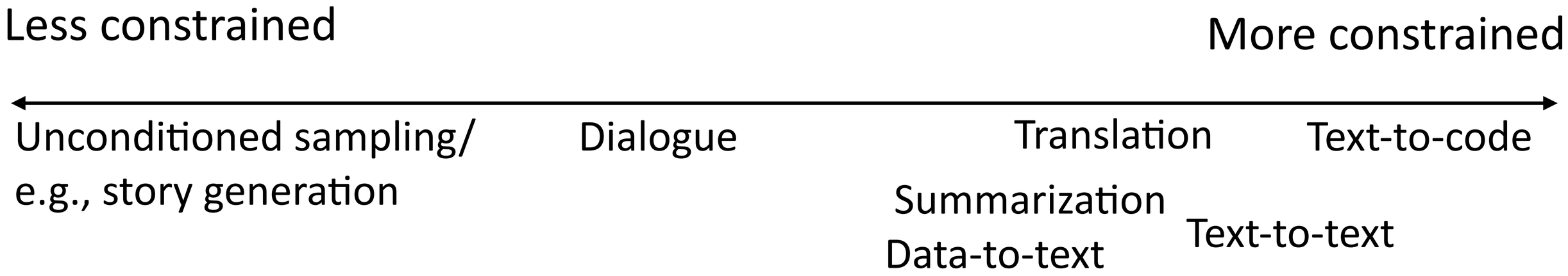
- Maintain decoder state, token history in beam



- NMT usually use beam <=5
- Keep **both** *film* states! Hidden state vectors are different

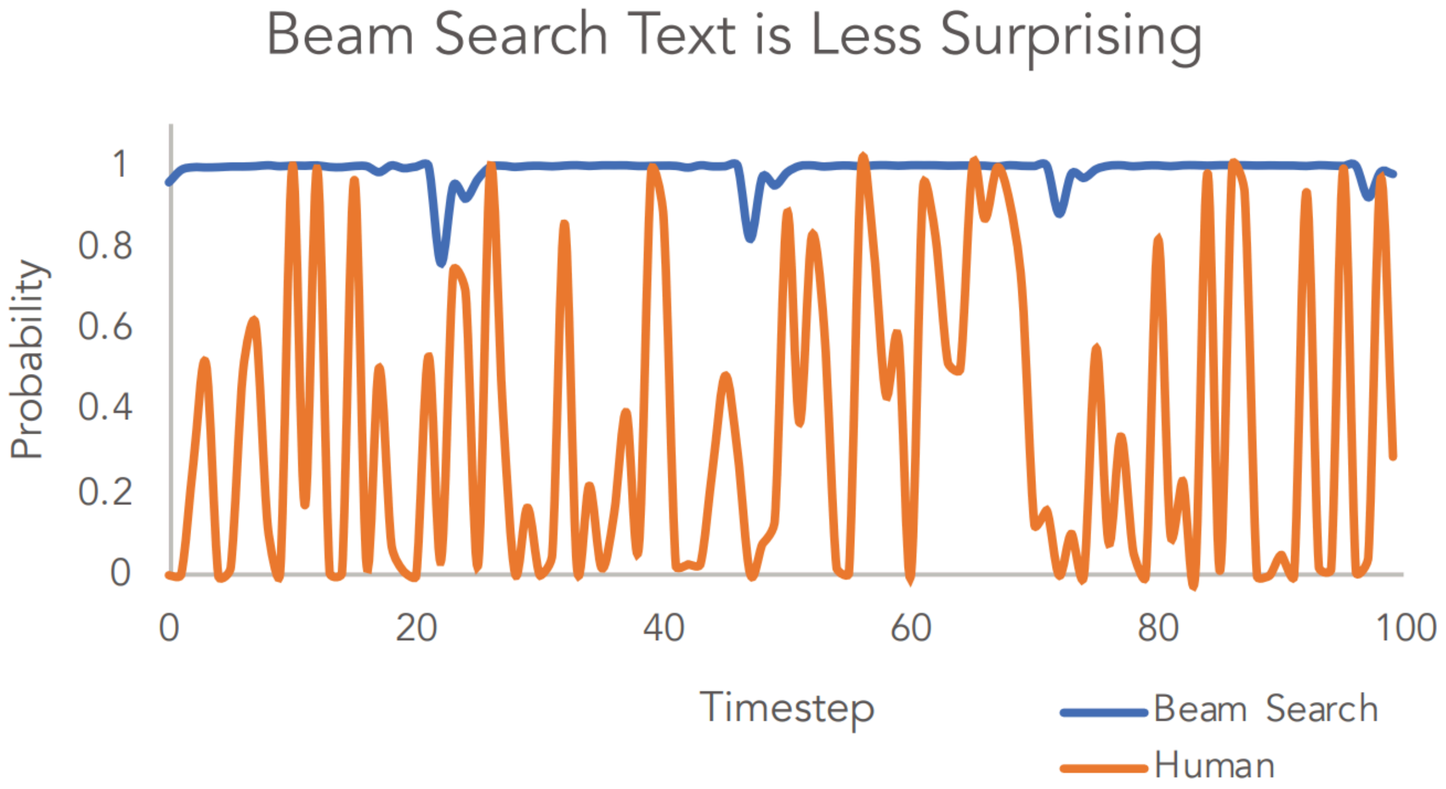
Generation Tasks

- For more constrained problems: greedy/beam decoding are usually best
- For less constrained problems: sampling introduces favorable variation in the output



Beam Search vs. Human

- For less constrained generation tasks (e.g., story generation)



Holtzman et al. (2019)

Sampling

- Greedy solution can be uninteresting / vacuous for various reasons (so called text *degeneration)*.
- Beam search may give many similar sequences, and these actually may be *too close* to the optimal.
- Sampling can help - especially for some text generation tasks.

$$P(y_i|\mathbf{x}, y_1, \ldots, y_{i-1}) = \text{softmax}(W\bar{h})$$

$$y_{\text{sampled}} \sim P(y|\mathbf{x}, y_1, \ldots, y_{i-1})$$

Beam Search vs. Sampling

- These are samples from an unconditioned language model GPT-2 (not seq2seq model)

Context: In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

Beam Search, *b*=32:
"The study, published in the Proceedings of the National Academy of Sciences of the United States of America (PNAS), was conducted by researchers from the Universidad Nacional Autónoma de México (UNAM) and the Universidad Nacional Autónoma de México (UNAM/Universidad Nacional Autónoma de México/Universidad Nacional Autónoma de México/Universidad Nacional Autónoma de México/Universidad Nacional Autónoma de ..."

Pure Sampling:
They were cattle called Bolivian Cavalleros; they live in a remote desert uninterrupted by town, and they speak huge, beautiful, paradisiacal Bolivian linguistic thing. They say, 'Lunch, marge.' They don't tell what the lunch is," director Professor Chuperas Omwell told Sky News. "They've only been talking to scientists, like we're being interviewed by TV reporters. We don't even stick around to be interviewed by TV reporters. Maybe that's how they figured out that they're cosplaying as the Bolivian Cavalleros."

- Sampling is better but sometimes draws too far from the tail of the distribution (relatively low prob. over thousands of candidate tokens).

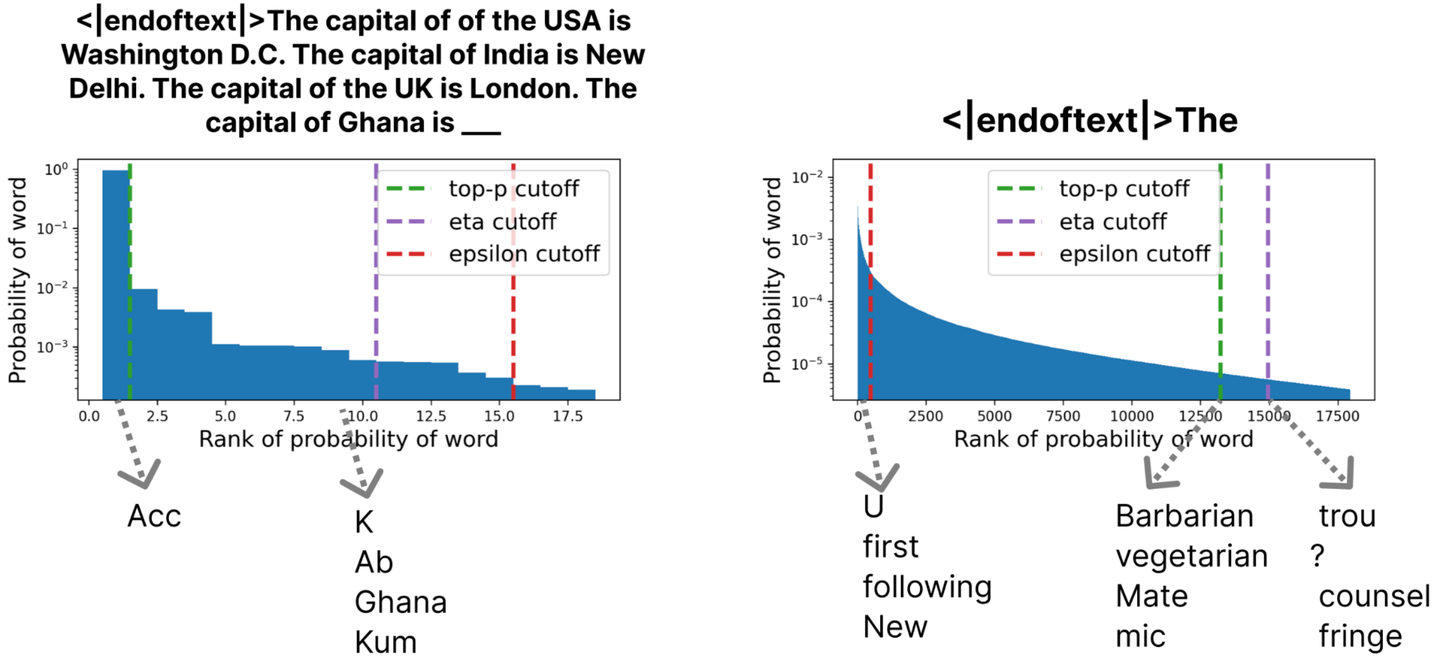
Holtzman et al. (2019)

Basic Decoding Strategies

- Greedy
- Beam search
- Sampling, e.g.:
  - Top-k: take the top k most likely words (e.g. k=5), sample from those
  - Nucleus (top-p): take the top p% (e.g., 95%) of the distribution, sample from within that
  - Epsilon: simple truncation, allow any word with greater than ε probability

Decoding

- Decoding is an inference-time solution to optimize LLM outputs
- Besides data and model size, inference-time algorithms can make a big impact.



Hewitt et al. (2022)

Generation Tasks

WebText

An unprecedented number of mostly young whales have become stranded on the West Australian coast since 2008.

Beam Search, *b*=16

The number of stranded whales has increased by more than 50 per cent in the past year, with the number of stranded whales on the West Australian coast increasing by more than 50 per cent in the past year. The number of whales stranded on the West Australian coast has increased by more than 50 per cent in the past year, with the number of stranded whales on the West Australian coast increasing by more than 50 per cent in the past year.

Pure Sampling

The Australian Food Safety Authority has warned Australia's beaches may be revitalised this year because healthy seabirds and seals have been on the move. More than 50,000 seabirds, sea mammals and seahorses have been swept into the sea by the Holden CS118 and Adelaide Airport CS300 from 2013. A major white-bat and umidauda migration across Australia is under way in Australia for the first time, with numbers reaching an estimated 50,000.

Nucleus, *p*=0.95

There has been an unprecedented number of calves caught in the nets of whaling stations that operate in WA. Pilot whales continue to migrate to feeding grounds to feed their calves. They are now vulnerable due to the decline of wild populations; they are restricted to one breeding site each year. Image copyright Yoon Bo Kim But, with sharp decline in wild populations the size of the Petrels are shrinking and dwindling population means there will only be room for a few new fowl.

Holtzman et al. (2019)

Decoding Strategies

- LMs place a distribution $P(y_i | \ y_1, \ldots, y_{i-1})$
- seq2seq models place a distribution $P(y_i | \mathbf{x}, y_1, \ldots, y_{i-1})$
- Generation from both models looks similar: (1) Greedy, (2) Beam Search, (3) Sampling

- Q: Why the Viterbi algorithm is typically not a good fit for this?