Announcements

- Yejin Choi talk Friday 11am, GDC 6.302 or https://utexas.zoom.us/j/97223751833
- FP check-ins due Tuesday. No slip days!

Recap

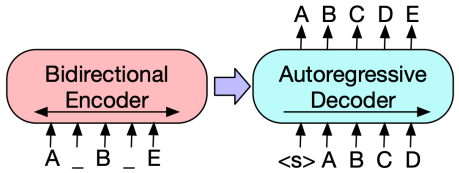
Today

- Seq2seq pre-trained models (BART, T5)
- GPT-2/GPT-3
- Prompting

Seq2seq Pre-trained Models: BART, T5

BART

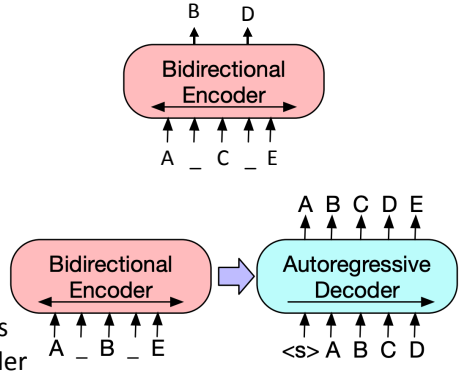
- BERT is good for "analysis" tasks, GPT is a good language model
- What to do for seq2seq tasks?
- Sequence-to-sequence BERT variant: permute/make/delete tokens, then predict full sequence autoregressively
- Uses the transformer encoder-decoder we discussed for MT (decoder attends to encoder)



Lewis et al. (2019)

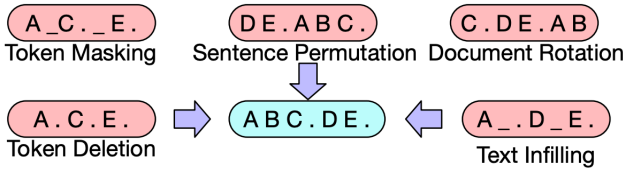
BERT vs. BART

- BERT: only parameters are an encoder, trained with masked language modeling objective
  - No way to do translation or left-to-right language modeling tasks
- BART: both an encoder and a decoder
  - Typically used for enc-dec tasks but also can just use the encoder as a replacement for BERT



Lewis et al. (2019)

BART



Infilling is longer spans than masking

- They try several strategies for generating training data. Infilling is a particularly helpful strategy for better performance

Lewis et al. (2019)

BART for Summarization

- **Pre-train** on the BART task: take random chunks of text, noise them according to the schemes described, and try to "decode" the clean text
- **Fine-tune** on a summarization dataset: a news article is the input and a summary of that article is the output (usually 1-3 sentences depending on the dataset)
- Can achieve good results even with **few summaries to fine-tune on**, compared to basic seq2seq models which require 100k+ examples to do well

Lewis et al. (2019)

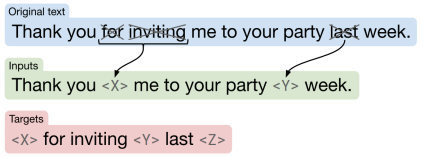
BART for Summarization: Outputs

This is the first time anyone has been recorded to run a full marathon of 42.195 kilometers (approximately 26 miles) under this pursued landmark time. It was not, however, an officially sanctioned world record, as it was not an "open race" of the IAAF. His time was 1 hour 59 minutes 40.2 seconds. Kipchoge ran in Vienna, Austria. It was an event specifically designed to help Kipchoge break the two hour barrier.	Kenyan runner Eliud Kipchoge has run a marathon in less than two hours.
PG&E stated it scheduled the blackouts in response to forecasts for high winds amid dry conditions. The aim is to reduce the risk of wildfires. Nearly 800 thousand customers were scheduled to be affected by the shutoffs which were expected to last through at least midday tomorrow.	Power has been turned off to millions of customers in California as part of a power shutoff plan.

Lewis et al. (2019)

T5

- Pre-training: similar denoising scheme to BART (they were released within a week of each other in fall 2019)
- Input: text with gaps. Output: a series of phrases to fill those gaps.



Raffel et al. (2019)

T5

Number of tokens	Repeats	GLUE	CNNDM	SQuAD	SGLUE	EnDe	EnFr	EnRo
★ Full dataset	0	**83.28**	**19.24**	**80.88**	**71.36**	**26.98**	**39.82**	**27.65**
2^{29}	64	**82.87**	**19.19**	**80.97**	**72.03**	**26.83**	**39.74**	**27.63**
2^{27}	256	82.62	**19.20**	79.78	69.97	**27.02**	**39.71**	27.33
2^{25}	1,024	79.55	18.57	76.27	64.76	26.38	39.56	26.80
2^{23}	4,096	76.34	18.33	70.92	59.29	26.37	38.84	25.81

summarization machine translation

- Colossal Cleaned Common Crawl: 750 GB of text
- We still haven't hit the limit of bigger data being useful for pre-training: here we see stronger MT results from the biggest data

Raffel et al. (2019)

Successes of T5

- How can we handle a task like QA by framing it as a seq2seq problem?
- Need to have text input and text output

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

- Format: *Question \n Passage* —> *Answer*
  encoder decoder

Raffel et al. (2019)

UnifiedQA

Extractive

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

Abstractive

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

- Past work: different architectures for every QA formulation. (Span selection, answer generation, multiple choice, …)
- Now: one 11B parameter T5 model

Khashabi et al. (2020)

UnifiedQA

Multiple choice

MC	**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

Yes/no

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

- Past work: different architectures for every QA formulation. (Span selection, answer generation, multiple choice, …)
- Now: one 11B parameter T5 model

Khashabi et al. (2020)

UnifiedQA

Seen dataset?	Model ↓ - Evaluated on →	NewsQA	Quoref	Quoref-CS	ROPES	ROPES-CS	DROP	DROP-CS	QASC	CommonsenseQA	NP-BoolQ	BoolQ-CS	MultiRC	Avg
No	UnifiedQA [EX]	58.7	64.7	53.3	43.4	29.4	24.6	24.2	55.3	62.8	20.6	12.8	7.2	38.1
	UnifiedQA [AB]	58.0	**68.2**	57.6	48.1	41.7	30.7	36.8	54.1	59.0	27.2	39.9	28.4	45.8
	UnifiedQA [MC]	48.5	67.9	**58.0**	61.0	44.4	28.9	37.2	67.9	75.9	2.6	5.7	9.7	42.3
	UnifiedQA [YN]	0.6	1.7	1.4	0.0	0.7	0.4	0.1	14.8	20.8	79.1	78.6	**91.7**	24.2
	UnifiedQA	**58.9**	63.5	55.3	**67.0**	**45.5**	**32.5**	**40.1**	**68.5**	**76.2**	**81.3**	**80.4**	59.9	**60.7**
Yes	Previous best	66.8	86.1	55.4	61.1	32.5	89.1	54.2	85.2	79.1	78.4	71.1	--	
		Retro Reader	TASE	XLNet	ROBERTa	RoBERTa	ALBERT	MTMSN	KF+SIR+2Step	freeLB-RoBERT	RoBERTa	RoBERTa	--	

NewsQA	Quoref	Quoref-CS	ROPES	ROPES-CS
58.7	64.7	53.3	43.4	29.4
58.0	**68.2**	57.6	48.1	41.7
48.5	67.9	**58.0**	61.0	44.4
0.6	1.7	1.4	0.0	0.7
58.9	63.5	55.3	**67.0**	**45.5**
66.8	86.1	55.4	61.1	32.5
Retro Reader	TASE	XLNet	ROBERTa	RoBERTa

DROP	DROP-CS
24.6	24.2
30.7	36.8
28.9	37.2
0.4	0.1
32.5	**40.1**
89.1	54.2
ALBERT	MTMSN

Khashabi et al. (2020)

UnifiedQA

- How well does this really work?
  Open-book: retrieval-based models
  Closed-book: just generate the answer with T5 or BART
  Nearest neighbor: NN over training data
- Models do well because the training sets contain answers shared in the test data
- T5 gets 9.4% accuracy if you filter to include no overlap

Model		Natural Questions			
		Total	Question Overlap	Answer Overlap Only	No Overlap
Open book	RAG	44.5	70.7	34.9	24.8
	DPR	41.3	69.4	34.6	19.3
	FID	51.4	71.3	48.3	34.5
Closed book	T5-11B+SSM	36.6	77.2	22.2	9.4
	BART	26.5	67.6	10.2	0.8
Nearest Neighbor	Dense	26.7	69.4	7.0	0.0
	TF-IDF	22.2	56.8	4.1	0.0

Khashabi et al. (2020)

Takeaways

- BART and T5 are useful for all sorts of seq2seq tasks involving language — so if you were going to use a seq2seq model, use one of these.
  (Caveat: need specialized models for things like language-to-code, but there's PLBART and CodeT5)
- UnifiedQA suggests that big generative models are good at generalizing across tasks and even to new tasks (although QA results have a long way to go)
- If we have a strong enough pre-trained model and train on enough tasks, can we generalize to new tasks?
- How do we specify those new tasks if they're not close to tasks we've already run on?
- Answer: **prompting**.

GPT

OpenAI GPT/GPT2

- Very large language models using the Transformer architecture
- Straightforward left-to-right language model, trained on raw text
- GPT2: trained on 40GB of text collected from upvoted links from reddit
- 1.5B parameters — by far the largest of these models trained when it came out in 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" strategy and didn't release biggest model)

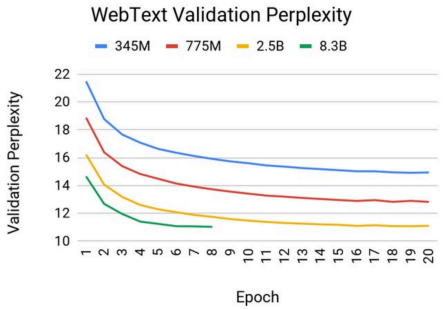
Pre-Training Cost (with Google/AWS)

- BERT: Base $500, Large $7000
- GPT-2 (as reported in other work): $25,000
- 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/

Pushing the Limits

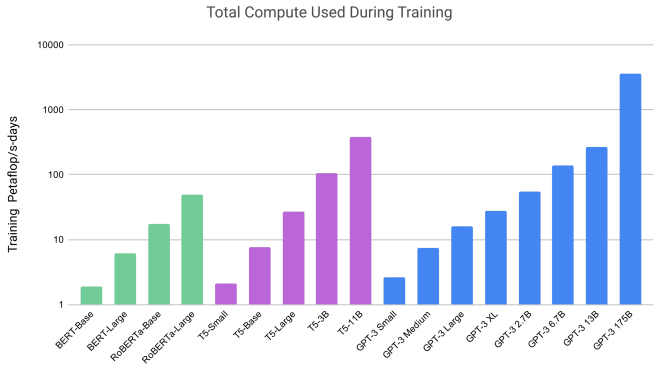
- NVIDIA: trained 8.3B parameter GPT model (5.6x the size of GPT-2)
- Arguable these models are still underfit: larger models still get better held-out perplexities



NVIDIA blog (Narasimhan, August 2019)

Pushing the Limits: GPT-3

- 175B parameter model: 96 layers, 96 heads, 12k-dim vectors
- Trained on Microsoft Azure, estimated to cost roughly $10M



Brown et al. (2020)

Pre-GPT-3: Fine-tuning

- Fine-tuning: this is the "normal way" of doing learning in models like GPT-2
- Requires computing the gradient and applying a parameter update on every example
- **This is super expensive with 175B parameters**

Brown et al. (2020)

GPT-3: Few-shot Learning

- GPT-3 proposes an alternative: **in-context learning.** Just uses the off-the-shelf model, no gradient updates
- This procedure depends heavily on the examples you pick as well as the prompt ("*Translate English to French*")

Few-shot

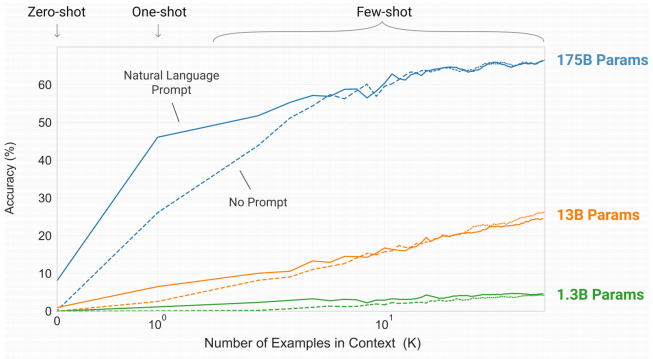
In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

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

Brown et al. (2020)

GPT-3

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



Brown et al. (2020)

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)

Prompting

Prompts

- Prompts can help induce the model to engage in certain behavior
- In the GPT-2 paper, “tl;dr:” (too long; didn't read) is mentioned as a prompt that frequently shows up in the wild **indicating a summary**
- tl;dr is an indicator that the model should “switch into summary mode” now — and if there are enough clean instances of tl;dr in the wild, maybe the model has been trained on a ton of diverse data?
- **How well does this work? Let’s see!**

Brown et al. (2020)

Prompts

- What about question answering?
- **Let’s see if we can get some questions working both with and without examples in the context**
- How can we take this further?

Brown et al. (2020)

Prompting for Classification

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 $P_2(a) =$ Just ____! $\|$ a

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

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

We define a single verbalizer v for all patterns as

$v(1) = \text{terrible}$ $v(2) = \text{bad}$ $v(3) = \text{okay}$
$v(4) = \text{good}$ $v(5) = \text{great}$

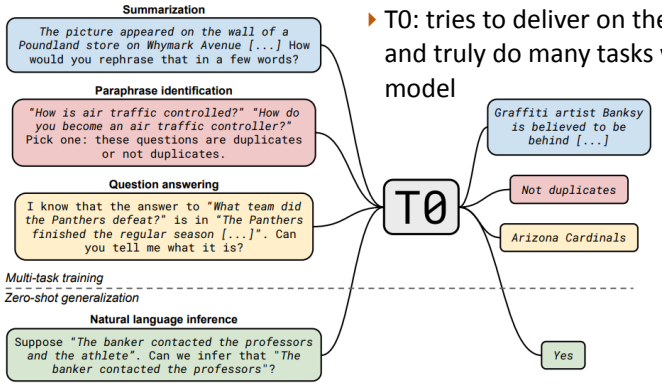
“verbalizer” of labels

patterns

- Compute probability of each pattern + verbalizer
- If P(It was bad …) > P (It was good …) => suggests negative sentiment
- **Can use prompting to pick between discrete options**

Schick and Schutze et al. (2020)

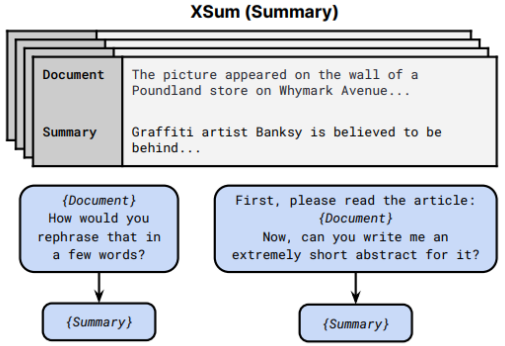
Task Generalization: T0



- T0: tries to deliver on the goal of T5 and truly do many tasks with one model

Sanh et al. (2021)

Task Generalization

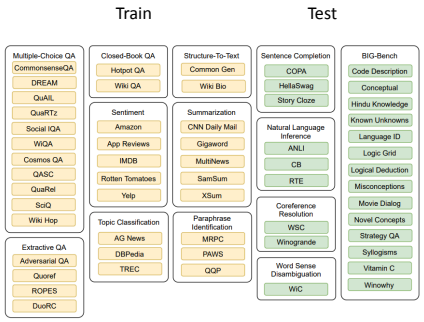


- Multiple prompts per task — they crowdsourced (from researchers) 1700 prompts for various tasks

Sanh et al. (2021)

Task Generalization

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



Sanh et al. (2021)

Prompting

- Lots of work on prompting: soft prompts (vectors that don't necessarily correspond to words) as well as search strategies and best practices
- So far prompted models < fine-tuned models if you have more than a few examples (~100) for your task
- 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?

- More info: see Pengfei Liu et al. survey (link on website)

Takeaways

- Pre-trained seq2seq models and generative language models can do well at lots of generation tasks
- Prompting is a way to harness their power and learn to do many tasks with a single model. Can be done without fine-tuning
- Prompting is a way to harness their power and learn to do many tasks with a single model