Natural Language Processing with Deep Learning

IFT6289, Winter 2022

Lecture 8: Prompting

Suyuchen Wang

2 Lecture Outline

1. Paradigms of NLP Technical Development
2. Architectures for Pretrained Language Models
3. Prompting
4. Development of Prompting
5. The Prompt-based Massive Multi-task Learning

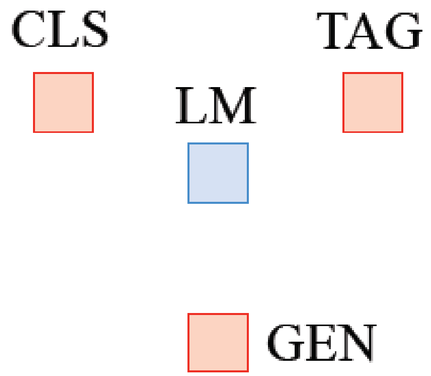
Paradigms of NLP Technical Development

4 Four Paradigms of NLP Technical Development

1. Feature Engineering
2. Architecture Engineering
3. Objective Engineering
4. Prompt Engineering

5 Feature Engineering

- **Paradigm**: Fully supervised learning (Non-neural network)
- **Time Period**: Most popular through 2015
- **Characteristics**:
  - Non-neural machine learning models mainly used
  - Require manually defined feature extraction
- **Representative Work**:
  - Manual features → linear or kernelized SVM
  - Manual features → conditional random fields (CRF)

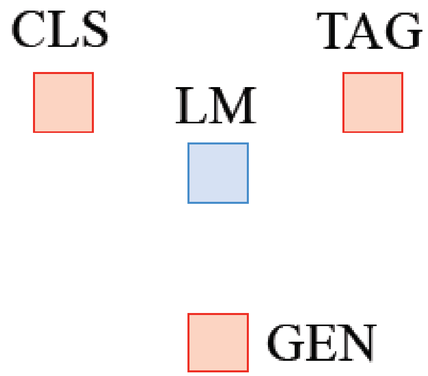


CLS: Classification
TAG: Seq tagging
GEN: Text generation
: Unsupervised training
: Supervised training
: Sup. + Unsup. training
: Textual prompt

Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

6 Architecture Engineering

- **Paradigm**: Fully supervised learning (Neural networks)
- **Time Period**: About 2013-2018
- **Characteristics**:
  - Rely on neural networks
  - Don’t need manually defined features
  - Should modify network structure (e.g., LSTM vs. CNN)
  - Sometimes use pretraining of LMs but often only for shallow features such as embeddings (word2vec / GloVe)
- **Representative Work**:
  - CNN / LSTM for text classification

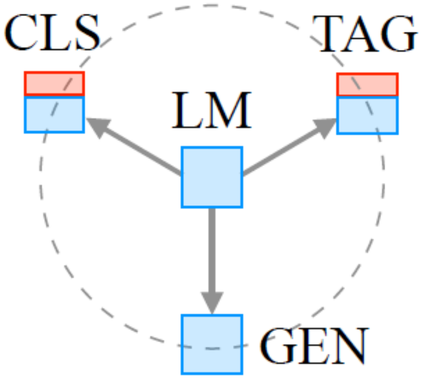


CLS: Classification
TAG: Seq tagging
GEN: Text generation
: Unsupervised training
: Supervised training
: Sup. + Unsup. training
: Textual prompt

Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

7 Objective Engineering

- **Paradigm**: Pre-train → Fine-tune
- **Time Period**: 2017-Now
- **Characteristics**:
  - Pre-trained LMs (PLMs) used as initialization of full model – both shallow and deep features
  - Less work on architecture design, but engineer objective functions
- **Representative Work**:
  - BERT + Fine-tuning

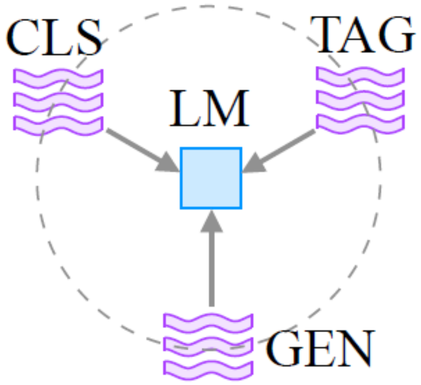


CLS: Classification
TAG: Seq tagging
GEN: Text generation
: Unsupervised training
: Supervised training
: Sup. + Unsup. training
: Textual prompt

Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

8 Prompt Engineering

- **Paradigm**: Pre-train → Prompt → Predict
- **Time Period**: 2019-Now
- **Characteristics**:
  - NLP tasks are modeled entirely by relying on LMs
  - The tasks of shallow & deep feature extraction, and prediction of the data are all given to the LM
  - Engineering of prompts is required
- **Representative Work**:
  - GPT-3



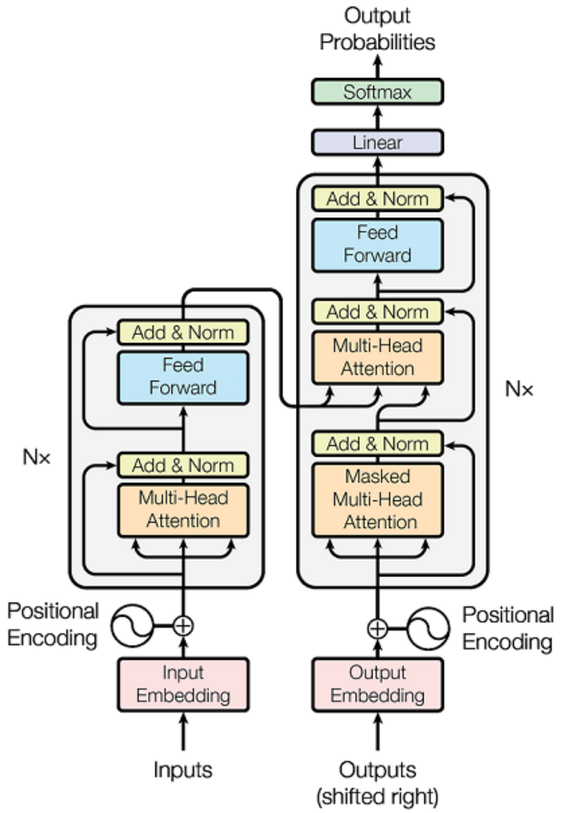
CLS: Classification
TAG: Seq tagging
GEN: Text generation
: Unsupervised training
: Supervised training
: Sup. + Unsup. training
: Textual prompt

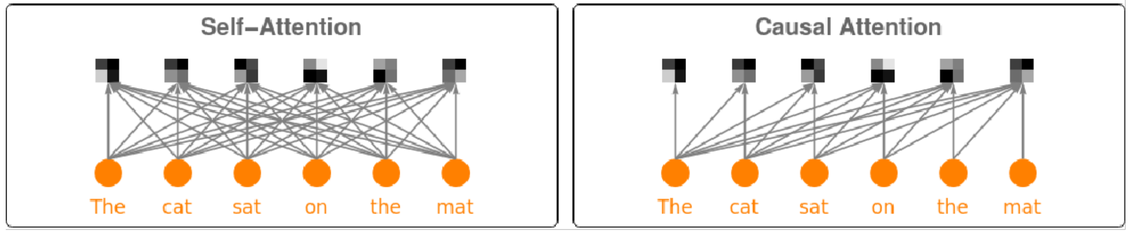
Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Architectures for Pretrained Language Models (PLMs)

PLMs Categorized by Architectures

- **Transformer Encoder**
  - BERT, RoBERTa, SpanBERT, XLNet…
- **Transformer Decoder**
  - GPT, GPT-2, GPT-3…
- **Transformer Encoder-Decoder**
  - T5, BART, mBART, MASS, XNLG…





PLMs Categorized by Frameworks

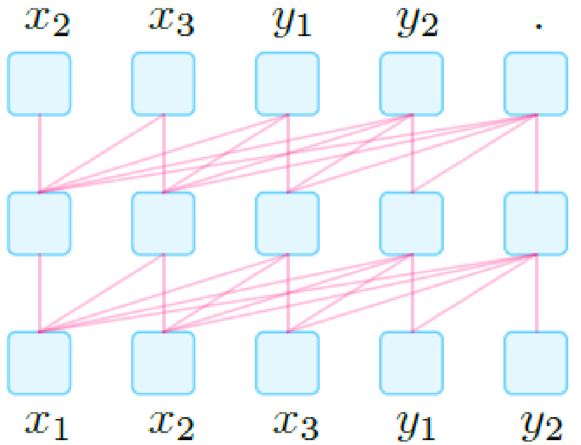
- Some popular frameworks include…
  - Left-to-Right LM
  - Masked LM
  - Prefix LM
  - Encoder-decoder

Left-to-Right Language Model

- **Characteristics**
  - First proposed by Markov (1913)
  - Count-based → Neural network-based
  - Specifically suitable to highly larger-scale LMs

- **Example**: GPT, GPT-2, GPT-3

- **Roles in Prompting Methods**
  - The earliest architecture chosen for prompting

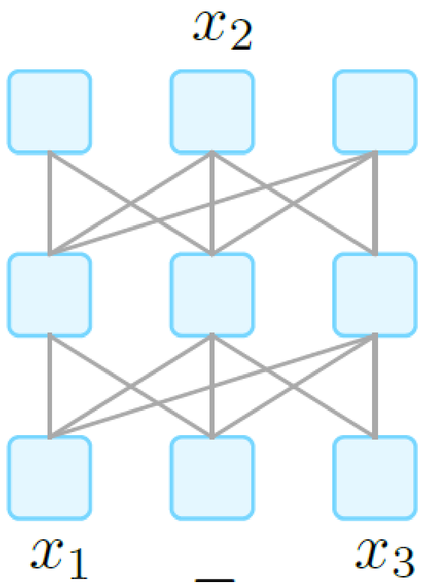


(a) Left-to-right LM.

Masked Language Model

- **Characteristics**
  - Unidirectional → Bi-directional prediction
  - Suitable for NLU tasks
  - Not suitable for NLG tasks

- **Example**: BERT, ERNIE

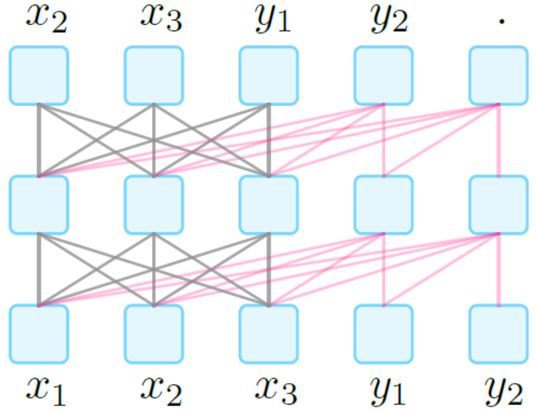


(b) Masked LM.

Prefix Language Model

- **Characteristics**
  - A combination of Left-to-Right LM and Masked LM
  - Use a Transformer but 2 different mask mechanisms to handle text X and y separately
  - Corruption operations can be introduced when encoding X

- **Example**: UniLMv1/v2, ERNIE-M

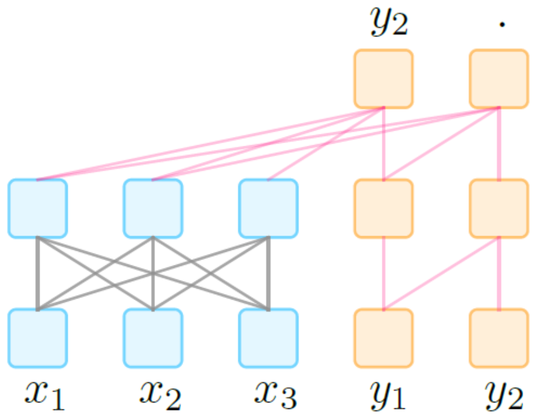


(c) Prefix LM.

Encoder-Decoder

- **Characteristics**
  - A denoised auto-encoder
  - Use 2 Transformers and 2 different mask mechanisms to handle text X and y separately
  - Corruption operations can be introduced when encoding X

- **Example**: T5, BART

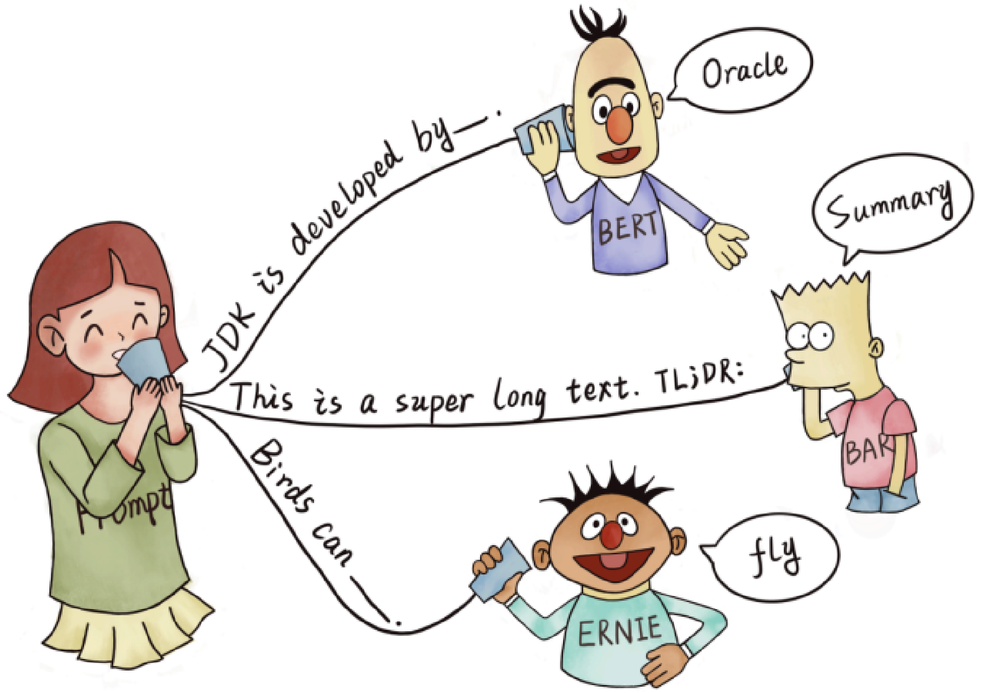


(d) Encoder-Decoder.

Prompting

What is Prompting?

- Encouraging a pre-trained model to make particular predictions by providing a “prompt” specifying the task to be done

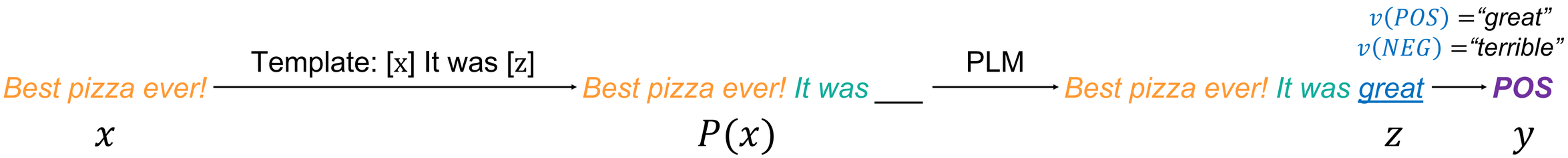


Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Terms About Prompts

- **Input** x: the original input text of a task
- **Label** y: the original output of a task
- **Template/Pattern** $P(x)$: a sentence that contains one masked section
- **Verbalizer** $v(y)$: transforming label to a token or a text span
- **Answer** z: the text filled to the template's masked section by the model

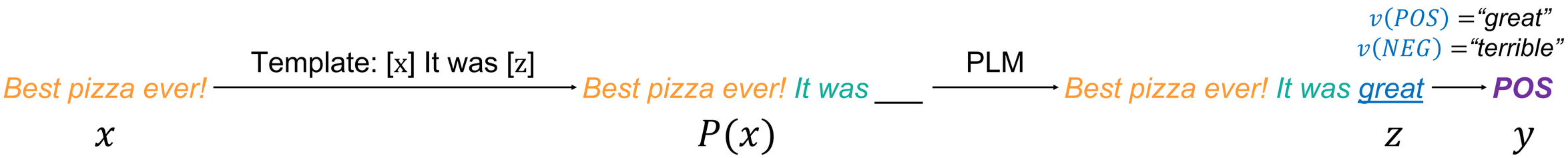
E.g., For a sentiment classification task,



Schick, T, Schütze, H. 2020. Exploiting cloze questions for few shot text classification and natural language inference. In EACL 2021.

Typical Workflow of Prompting

1. **Prompt Addition**: Generate prompt with masked section by pattern P
2. **Answer Prediction**: Fill in answer z to the masked section by PLM
3. **Mapping**: Given predicted answer z, map it back to the label

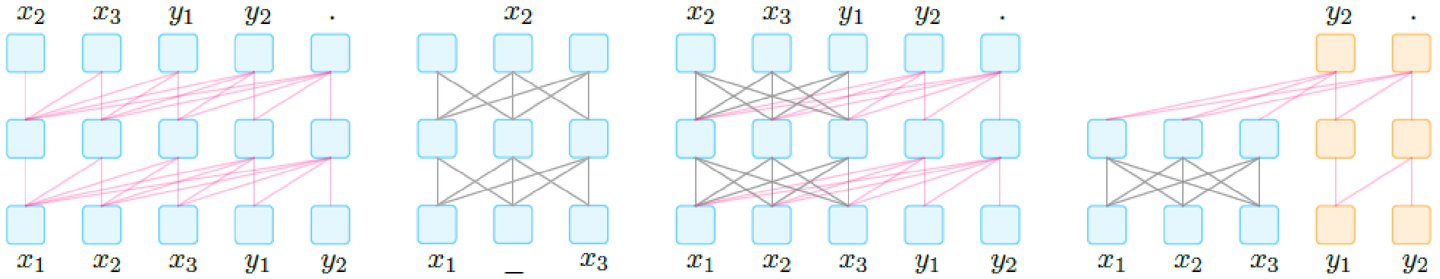


Schick, T, Schütze, H. 2020. Exploiting cloze questions for few shot text classification and natural language inference. In EACL 2021.

Different Types of Prompts

- **Cloze Prompt**: *[x] I think it is a [z] restaurant*
  - Suitable for Masked LMs (BERT, RoBERTa…)

- **Prefix Prompt**: *[x] I think it is [z]*
  - Suitable for Left-to-right LMs (GPT-2, GPT-3…), Prefix LMs (UniLMv1/v2), and Encoder-Decoders (T5, BART…)



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Different Types of Answers

- **Token**: Answer is one token in the vocabulary
  - E.g., Sentiment classification: *I love this movie. This movie is {great, bad…}*

- **Span**: A short multi-token span. Typically used with **cloze prompts**
  - E.g., Topic classification: *He trained a neural network. This sentence is about {machine learning, quantum physics…}*

- **Sentence**: An arbitrary length sentence. Typically used with **prefix prompts**
  - E.g., Machine translation: *English: I love natural language processing. French: {J'adore le traitement automatique du langage naturel}*

Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Different Types of Training Strategies

Strategy	LM Params Tuned	Additional Trainable Params for Prompt	Examples
Promptless Fine-tuning	✓	N/A	BERT Fine-tuning
Tuning-free Prompting	✗	✗	GPT-3
Fixed-LM Prompt Tuning	✗	✓	Prefix-tuning
Fixed-prompt LM Tuning	✓	✗	PET
Prompt+LM Fine-tuning	✓	✓	PADA

Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Different Types of Settings

- **Zero-shot:** without any explicit training samples of the downstream task
- **Few-shot:** only few training samples (e.g., 1-100) of the downstream task
- **Full-data:** Use plenty of training samples (e.g., 10K) from the full dataset

Zero-shot

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

```
1  Translate English to French:        ← task description
2  cheese =>                           ← prompt
```

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, T. et al. 2020. Language Models are Few-shot Learners. In NeurIPS 2020.

Some Examples of Prompting

Type	Task	Input ([X])	Template	Answer ([Z])
Text CLS	Sentiment	I love this movie.	[X] The movie is [Z].	great fantastic ...
	Topics	He prompted the LM.	[X] The text is about [Z].	sports science ...
	Intention	What is taxi fare to Denver?	[X] The question is about [Z].	quantity city ...
Text-span CLS	Aspect Sentiment	Poor service but good food.	[X] What about service? [Z].	Bad Terrible ...
Text-pair CLS	NLI	[X1]: An old man with ... [X2]: A man walks ...	[X1]? [Z], [X2]	Yes No ...
Tagging	NER	[X1]: Mike went to Paris. [X2]: Paris	[X1][X2] is a [Z] entity.	organization location ...
Text Generation	Summarization	Las Vegas police ...	[X] TL;DR: [Z]	The victim ... A woman ... ...
	Translation	Je vous aime.	French: [X] English: [Z]	I love you. I fancy you. ...

Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Using Prompts in More Complicated Tasks

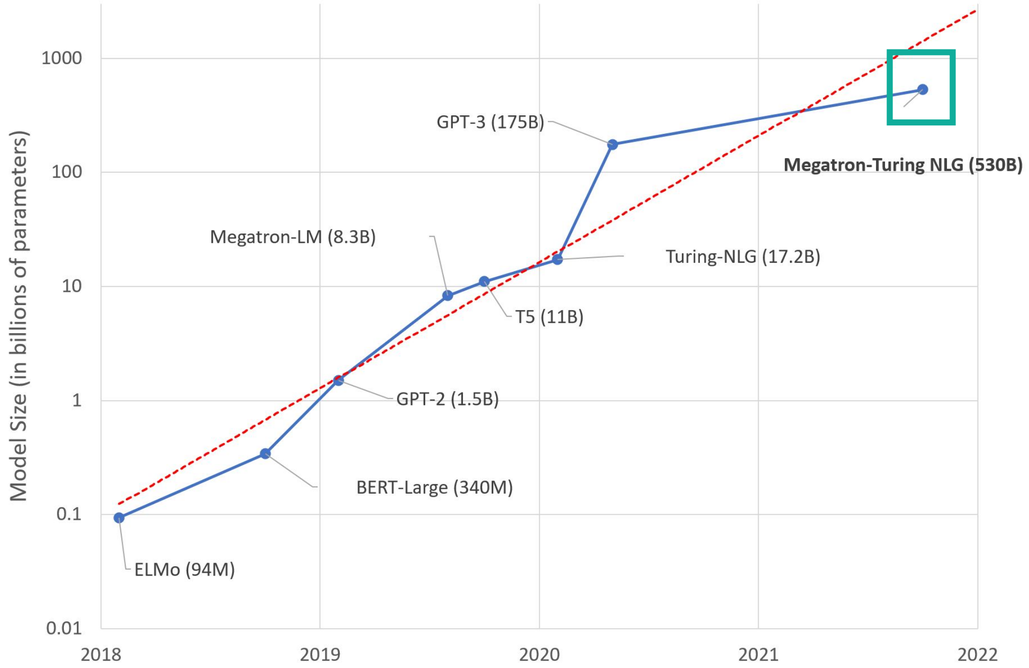
- Natural Language Inference (NLI)
  - *[x1] ? {Yes (Entailment), No (Contradiction), Maybe (Neutral)} [x2]*
  - *A soccer game with multiple males playing ? Yes. Some men are playing a sport.*

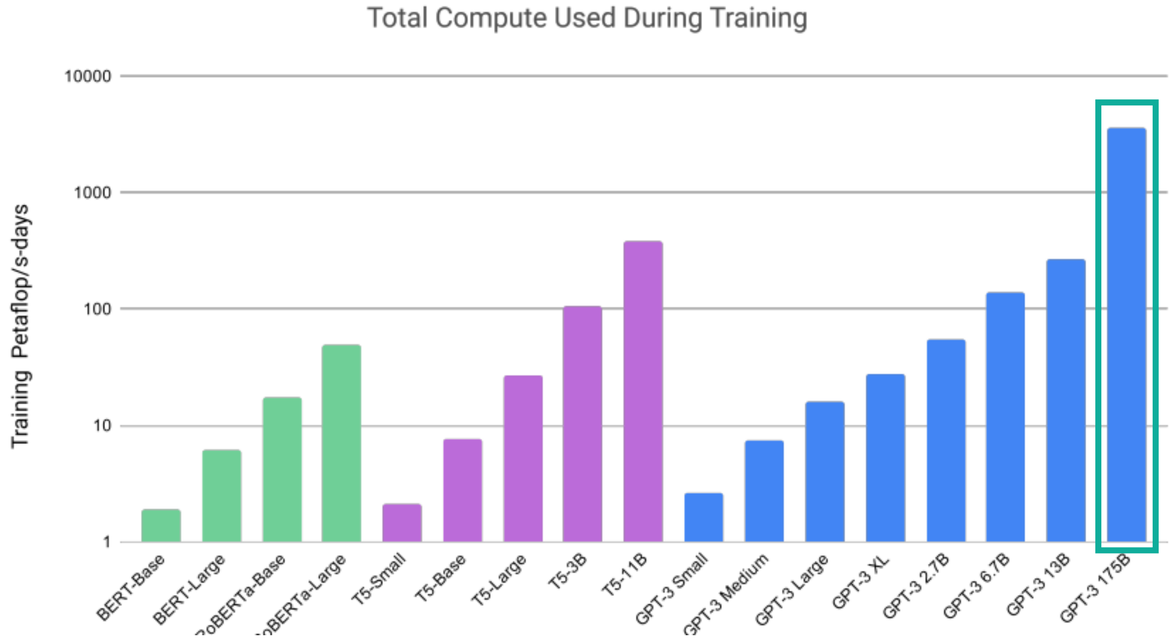
- Aspect-Based Sentiment Analysis (ABSA)
  - *[x] The [Aspect] is [Opinion] ? {Yes, No}. This is {POS, NEG, NEU}.*
  - *The owners are great fun and the beer selection is worth staying for. The owners are great fun ? Yes. This is POS.*

Li, C. et al. 2021. Sentiprompt: Sentiment knowledge enhanced prompt-tuning for aspect-based sentiment analysis. arXiv:2109.08306
Schick T, Schütze H. 2020. Exploiting cloze questions for few shot text classification and natural language inference. arXiv:2001.07676.

Why Prompting?

- **The PLM may be too large to fine-tune.**



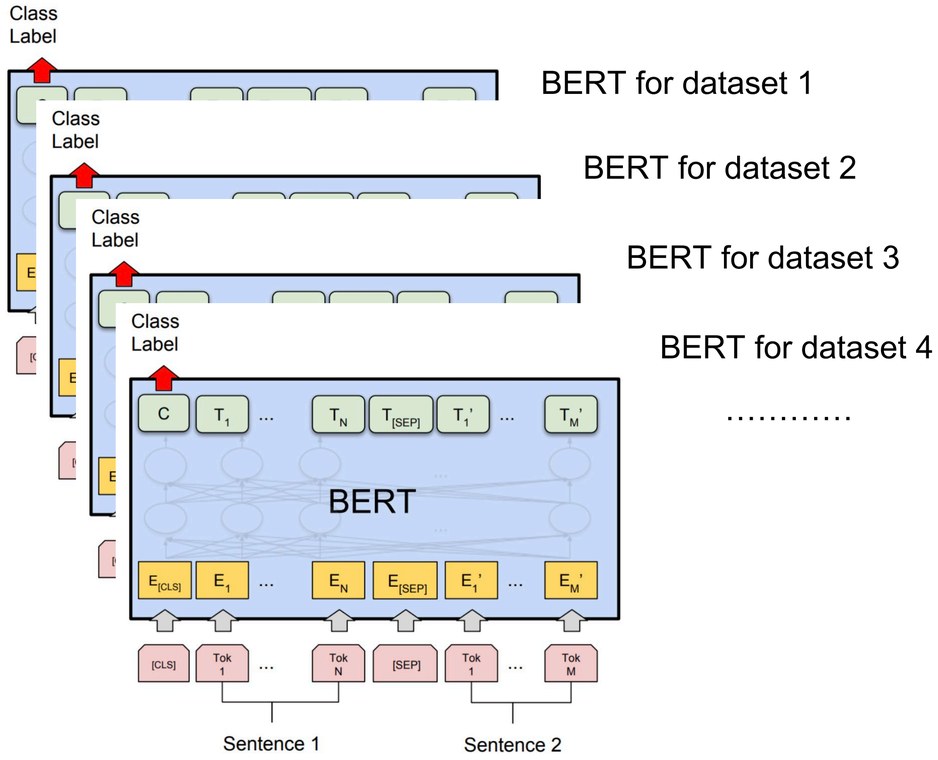


Brown, T. et al. 2020. Language Models are Few-shot Learners. In NeurIPS 2020.
Smith, S, et al. 2022. Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, A Large-Scale Generative Language Model. arXiv:2201.11990.

Why Prompting? (contd.)

- **When dealing with multiple tasks, only need to keep one copy of PLM!**

Fine-tuning



BERT for dataset 1

BERT for dataset 2

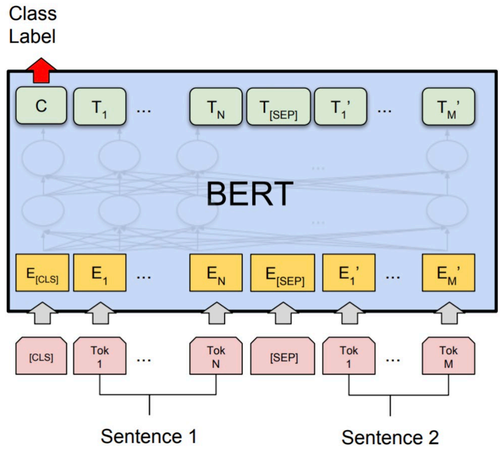
BERT for dataset 3

BERT for dataset 4

.....

Vs.

Prompting



[x]…[z]: Prompt for dataset 1

[x]…[z]: Prompt for dataset 2

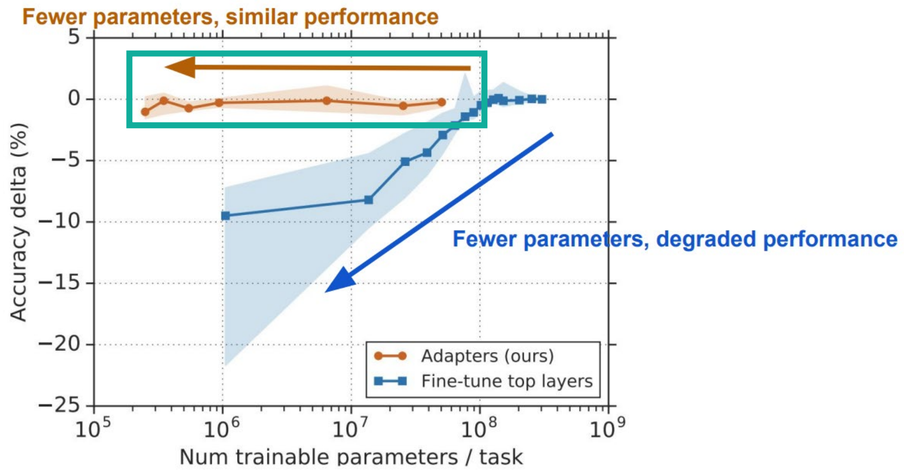
[x]…[z]: Prompt for dataset 3

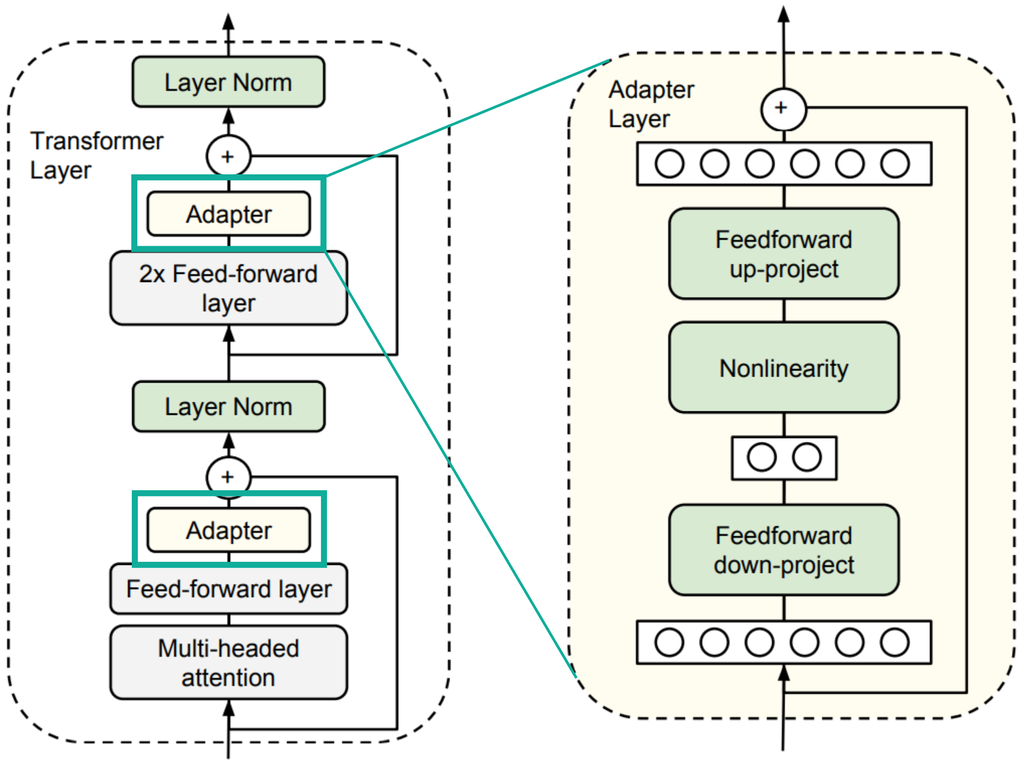
[x]…[z]: Prompt for dataset 4

.....

Why Prompting? (contd.)

- **When dealing with multiple tasks, only need to keep one copy of PLM!**
- Note: the previous most popular solution for this is called **Adapters.**
- Adapters reduce #params/task by **30x** at only 0.4% accuracy drop.





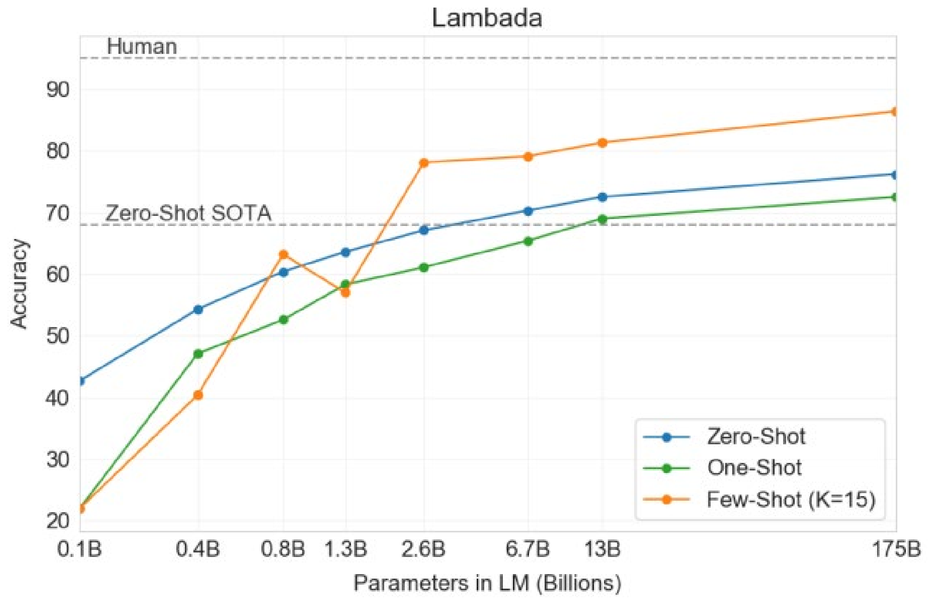
Houlsby, N. et al. 2019. Parameter-Efficient Transfer Learning for NLP. In ICML 2019.

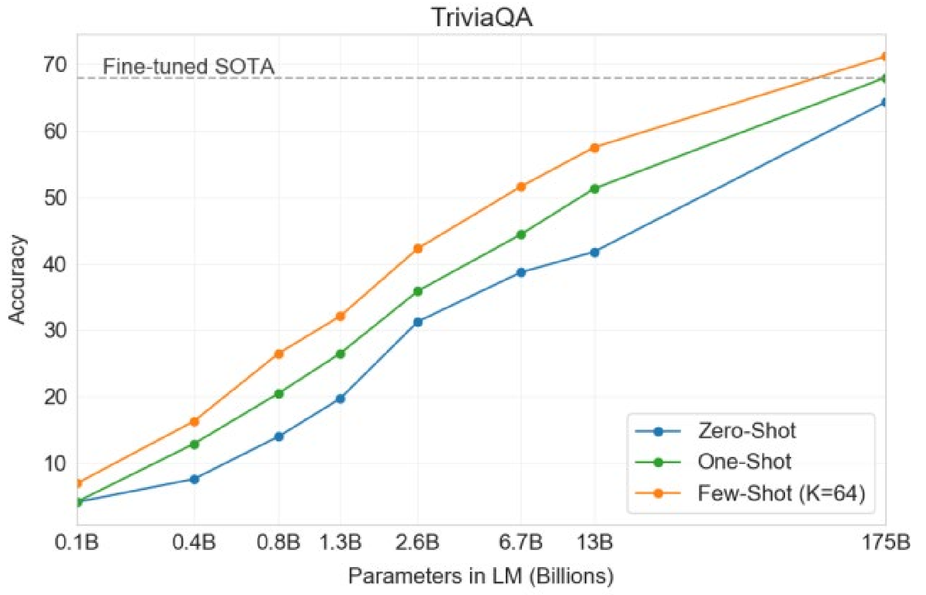
Why Prompting? (contd.)

- **Large PLMs performs well under even zero-shot setting using prompts.**

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

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

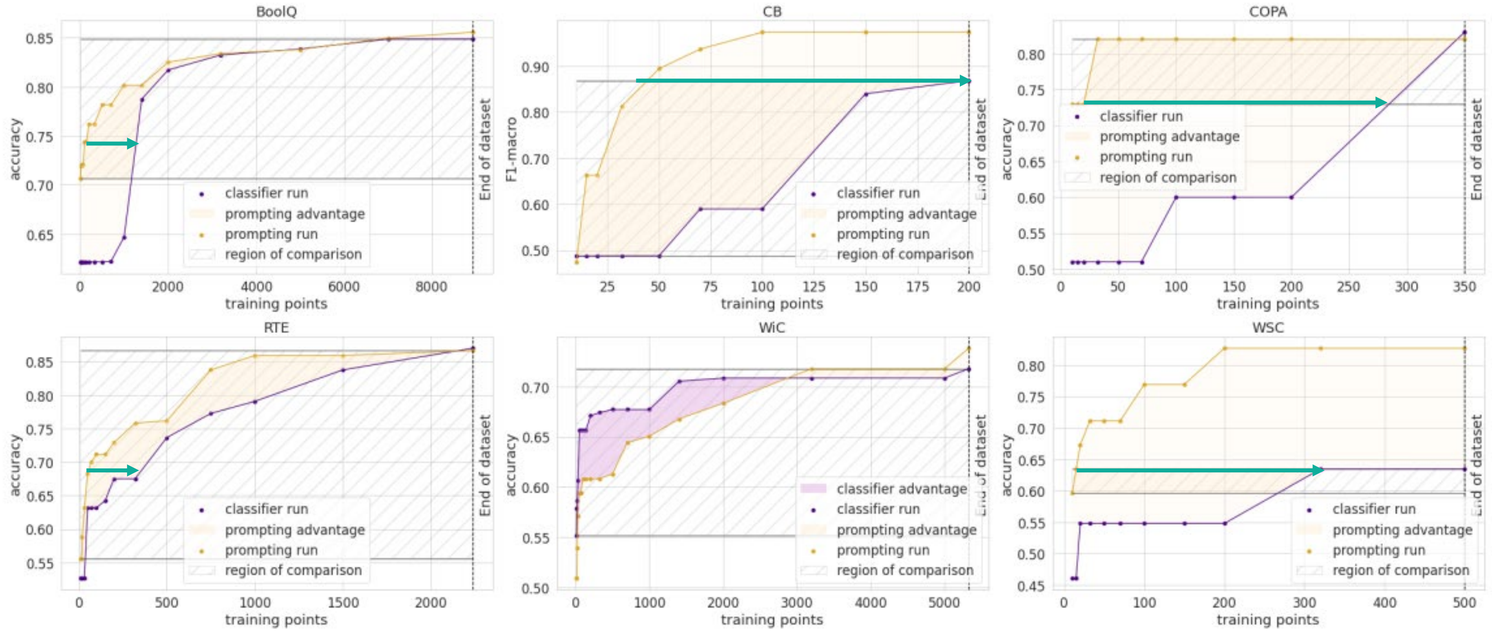




Brown, T. et al. 2020. Language Models are Few-shot Learners. In NeurIPS 2020.

Why Prompting? (contd.)

- **A good prompt is worth hundreds to thousands of labeled data.**

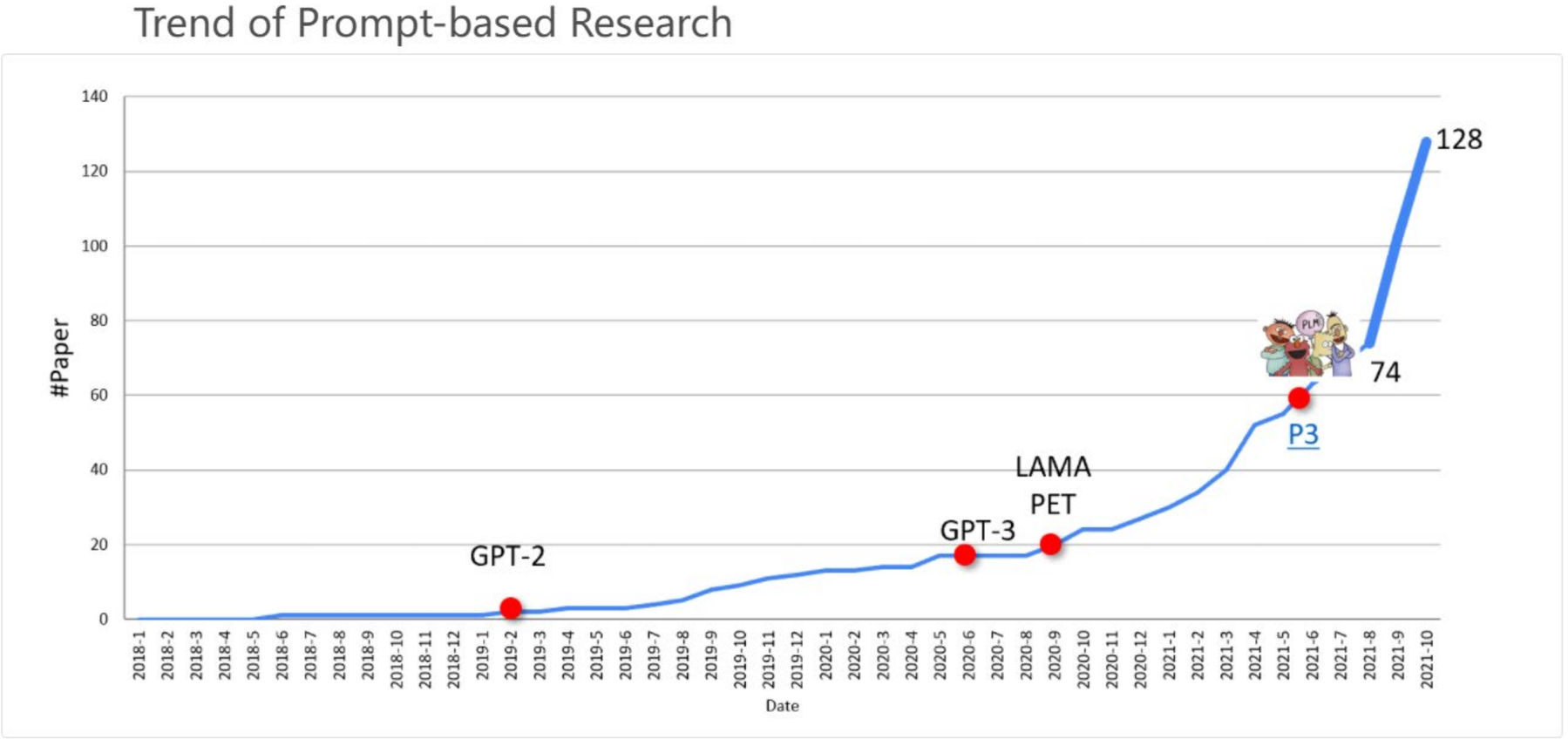


	Average Advantage (# Training Points)							
	MNLI	BoolQ	CB	COPA	MultiRC*	RTE	WiC	WSC
P vs H	3506 ± 536	752 ± 46	90 ± 2	288 ± 242	384 ± 378	282 ± 34	-424 ± 74	281 ± 137
P vs N	150 ± 252	299 ± 81	78 ± 2	-	74 ± 56	404 ± 68	-354 ± 166	-
N vs H	3355 ± 612	453 ± 90	12 ± 1	-	309 ± 320	-122 ± 62	-70 ± 160	-

Scao, T L, Rush, A M. 2021. How many data points is a prompt worth?. In NAACL-HLT 2021.

Why Prompting? (contd.)

- **Researchers' interest in prompting is HIGH!**



What is the Problem with Prompting?

- **It is hard to manually design a "good" prompt.**

Prompt	P@1
[X] is located in [Y]. *(original)*	31.29
[X] is located in which country or state? [Y].	19.78
[X] is located in which country? [Y].	31.40
[X] is located in which country? In [Y].	51.08

Table 1. Case study on LAMA-TREx P17 with bert-base-cased. A single-word change in prompts could yield a drastic difference.

Prompts	
manual	DirectX *is developed by* y_{man}
mined	y_{mine} *released the* DirectX
paraphrased	DirectX *is created by* y_{para}

Top 5 predictions and log probabilities

	y_{man}		y_{mine}		y_{para}	
1	Intel	-1.06	Microsoft	-1.77	Microsoft	-2.23
2	Microsoft	-2.21	They	-2.43	Intel	-2.30
3	IBM	-2.76	It	-2.80	default	-2.96
4	Google	-3.40	Sega	-3.01	Apple	-3.44
5	Nokia	-3.58	Sony	-3.19	Google	-3.45

Figure 1: Top-5 predictions and their log probabilities using different prompts (manual, mined, and paraphrased) to query BERT. Correct answer is underlined.

Liu, X, et al. 2021. GPT Understands, Too. arXiv: 2103.10385.
Jiang, Z, et al. 2020. How Can We Know What Language Models Know? In TACL 2020.

How to Select a Strategy to Use PLMs (Currently)

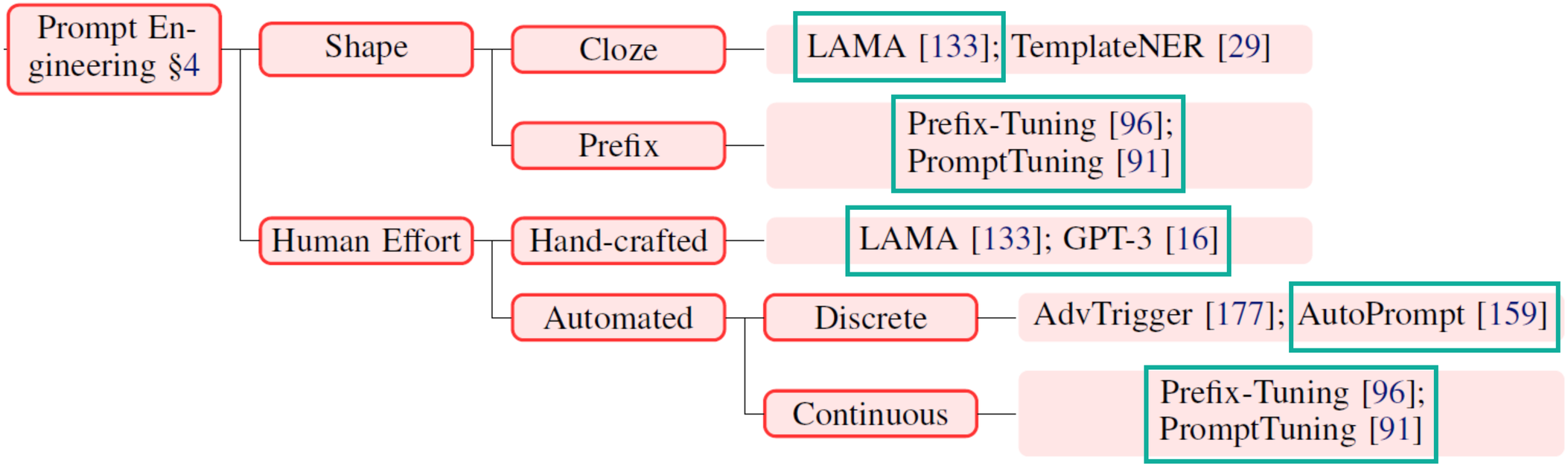
- **Promptless Fine-tuning**
- **Fixed-prompt Tuning**
- **Prompt+LM Fine-tuning**
- **Tuning-free Prompting**
- **Fixed-LM Prompt Tuning**

- If you have a huge PLM to use (e.g., GPT-3)?
- If you have few training examples?
- If you have lots of training examples?

Development of Prompting

From a General View

- In this section, we will *very* briefly introduce the papers in green rectangles in an approximately chronological order (More details will be covered in your mini-lecture!):



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

LAMA: Patterns to Probe Knowledge in PLMs

- Use a prompt to probe knowledge in unfine-tuned PLMs, like querying a KG
- Contains templates for a set of datasets for knowledge probing (a.k.a LAMA Probes), which forms a knowledge probing task for PLMs

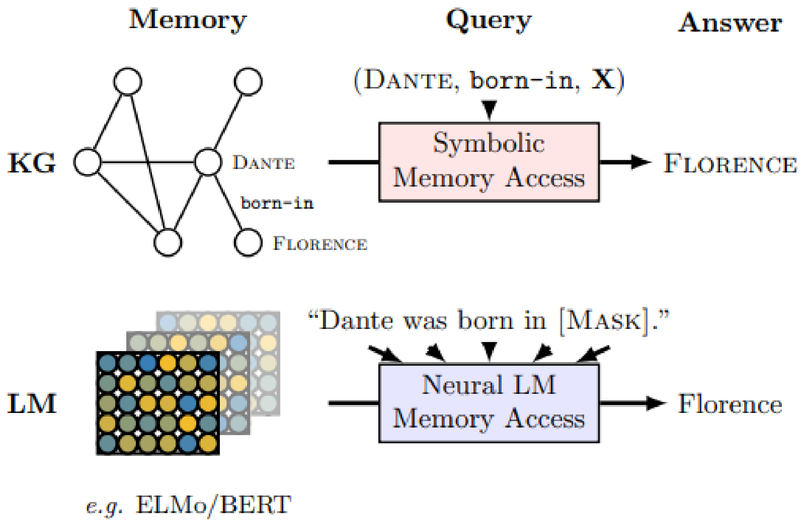


Figure 1: Querying knowledge bases (KB) and language models (LM) for factual knowledge.

Petroni, F, et al. 2019. Language Models as Knowledge Bases?. In EMNLP 2019.

LPAQA: Easily Generate More Patterns for Ensembling

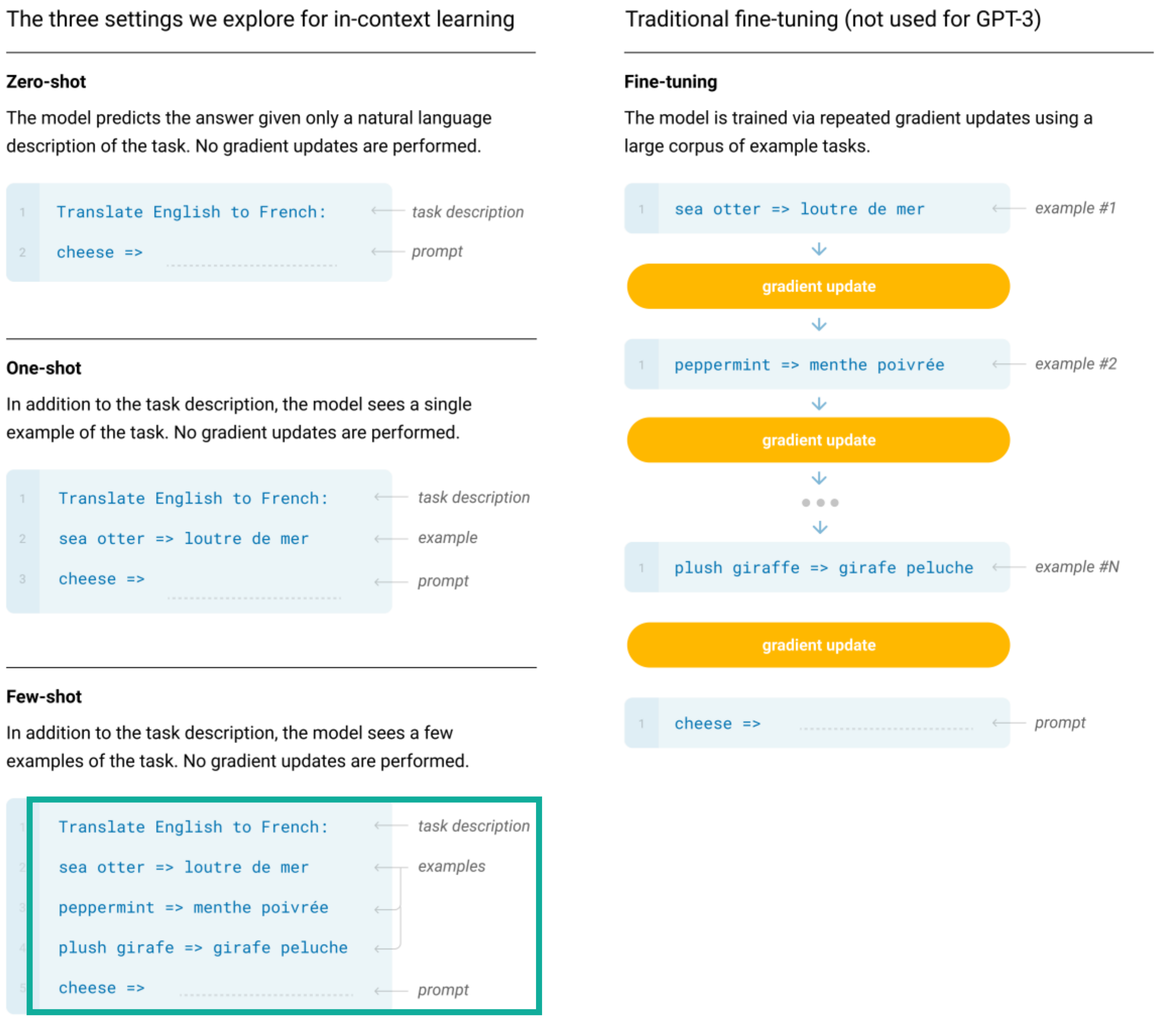
- **Simple ways to generate & select prompts (especially for relation extraction tasks)**

  - **Prompt mining (answers → prompts)**
    - Middle-word prompts: *Barack Obama was born in Hawaii → [x] was born in [y]*
    - Dependency-based prompts: *The capital of France is Paris → capital of [x] is [y]*
  - **Prompt paraphrasing (existing prompts → new prompts)**
    - Back-translation: *[x] shares a border with [y]* → En-De translation → En-De translation → *[x] adjoins [y]*

Jiang, Z, et al. 2020. How Can We Know What Language Models Know?. In TACL 2020.

GPT-3: Fine-tuning is Not Needed!

- **Instead of fine-tuning, GPT-3 uses “in-context learning”**
- The task description and examples forms the “context”, while the prompt completes the task
- Large PLM + in-context learning works surprisingly well
- Later works refine the way to choose and orders in-context examples

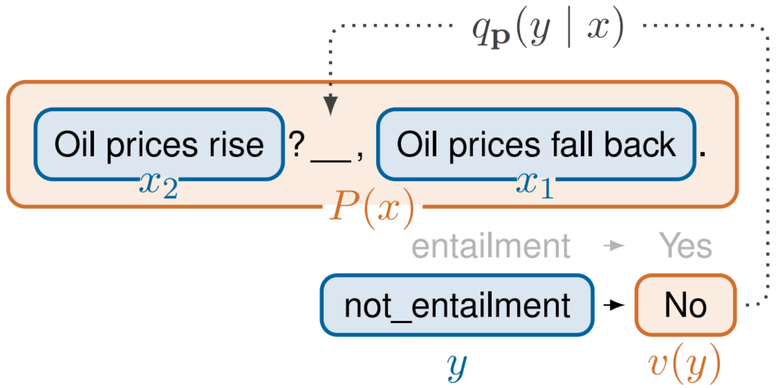


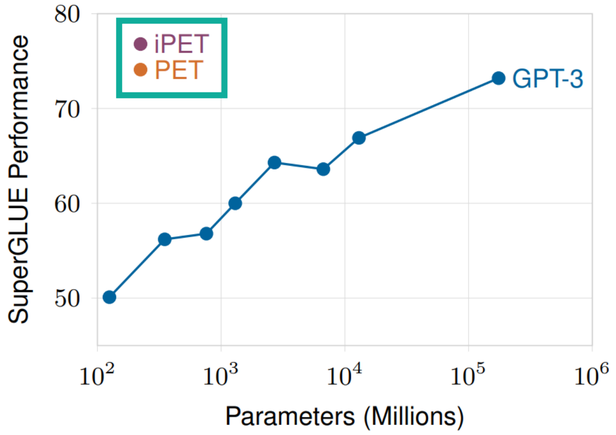
Brown, T. et al. 2020. Language Models are Few-shot Learners. In NeurIPS 2020.

PET: You Don't Need a Huge PLM to Beat GPT-3

- **Fixed-prompt LM Tuning + ALBERT-xxlarge-v2 >= GPT-3 in few-shot setting**

	Model	Params (M)	BoolQ Acc.	CB Acc. / F1	COPA Acc.	RTE Acc.	WiC Acc.	WSC Acc.	MultiRC EM / F1a	ReCoRD Acc. / F1	Avg –
dev	GPT-3 Small	125	43.1	42.9 / 26.1	67.0	52.3	49.8	58.7	6.1 / 45.0	69.8 / 70.7	50.1
	GPT-3 Med	350	60.6	58.9 / 40.4	64.0	48.4	55.0	60.6	11.8 / 55.9	77.2 / 77.9	56.2
	GPT-3 Large	760	62.0	53.6 / 32.6	72.0	46.9	53.0	54.8	16.8 / 64.2	81.3 / 82.1	56.8
	GPT-3 XL	1,300	64.1	69.6 / 48.3	77.0	50.9	53.0	49.0	20.8 / 65.4	83.1 / 84.0	60.0
	GPT-3 2.7B	2,700	70.3	67.9 / 45.7	83.0	56.3	51.6	62.5	24.7 / 69.5	86.6 / 87.5	64.3
	GPT-3 6.7B	6,700	70.0	60.7 / 44.6	83.0	49.5	53.1	67.3	23.8 / 66.4	87.9 / 88.8	63.6
	GPT-3 13B	13,000	70.2	66.1 / 46.0	86.0	60.6	51.1	75.0	25.0 / 69.3	88.9 / 89.8	66.9
	GPT-3	175,000	77.5	82.1 / 57.2	92.0	72.9	**55.3**	75.0	32.5 / 74.8	**89.0 / 90.1**	73.2
	PET	223	79.4	85.1 / 59.4	**95.0**	69.8	52.4	**80.1**	**37.9 / 77.3**	86.0 / 86.5	74.1
	iPET	223	**80.6**	**92.9 / 92.4**	**95.0**	**74.0**	52.2	**80.1**	33.0 / 74.0	86.0 / 86.5	**76.8**
test	GPT-3	175,000	76.4	75.6 / 52.0	**92.0**	69.0	49.4	80.1	30.5 / 75.4	**90.2 / 91.1**	71.8
	PET	223	79.1	87.2 / 60.2	90.8	67.2	**50.7**	**88.4**	**36.4 / 76.6**	85.4 / 85.9	74.0
	iPET	223	**81.2**	**88.8 / 79.9**	90.8	**70.8**	49.3	**88.4**	31.7 / 74.1	85.4 / 85.9	**75.4**
	SotA	11,000	91.2	93.9 / 96.8	94.8	92.5	76.9	93.8	88.1 / 63.3	94.1 / 93.4	89.3

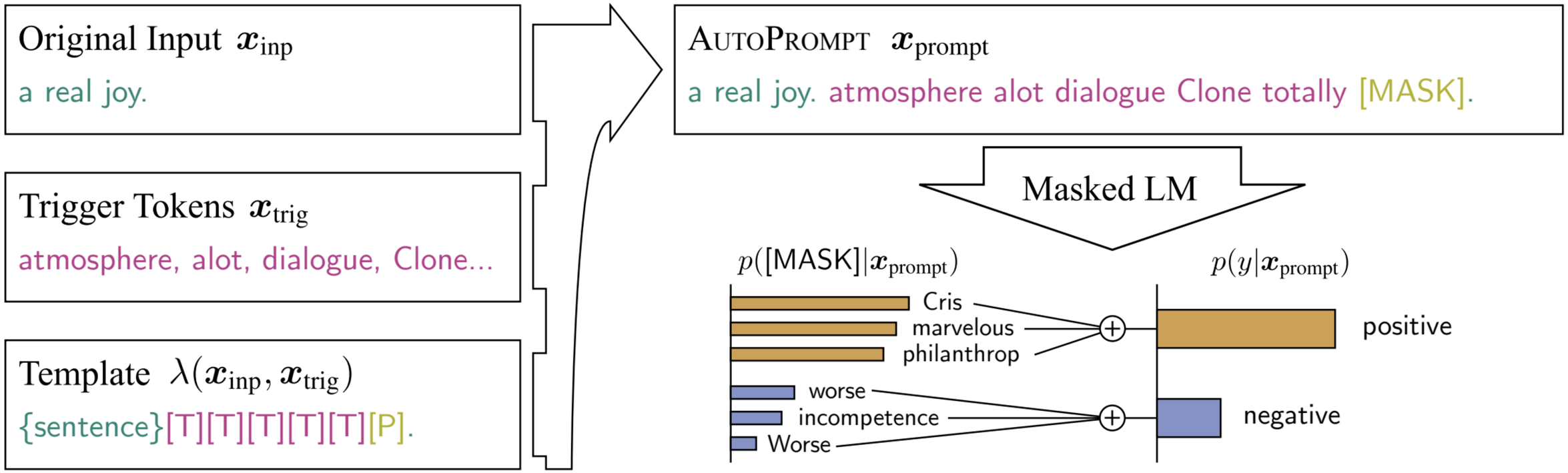




Schick, T, Schütze, H. 2021. It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners. In NAACL-HLT 2021.

40 AutoPrompt: Prompts Can Be Automatically Optimized

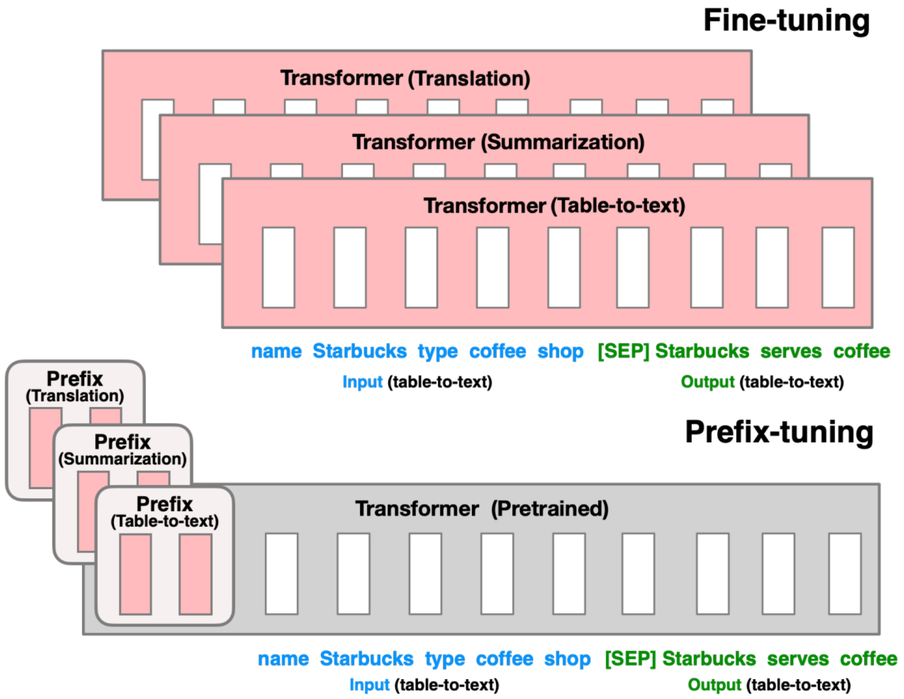
- **Automatically optimize arbitrary prompts based on existing words**
- Train “trigger tokens” as prompt using SGD. Doesn’t have to be meaningful.

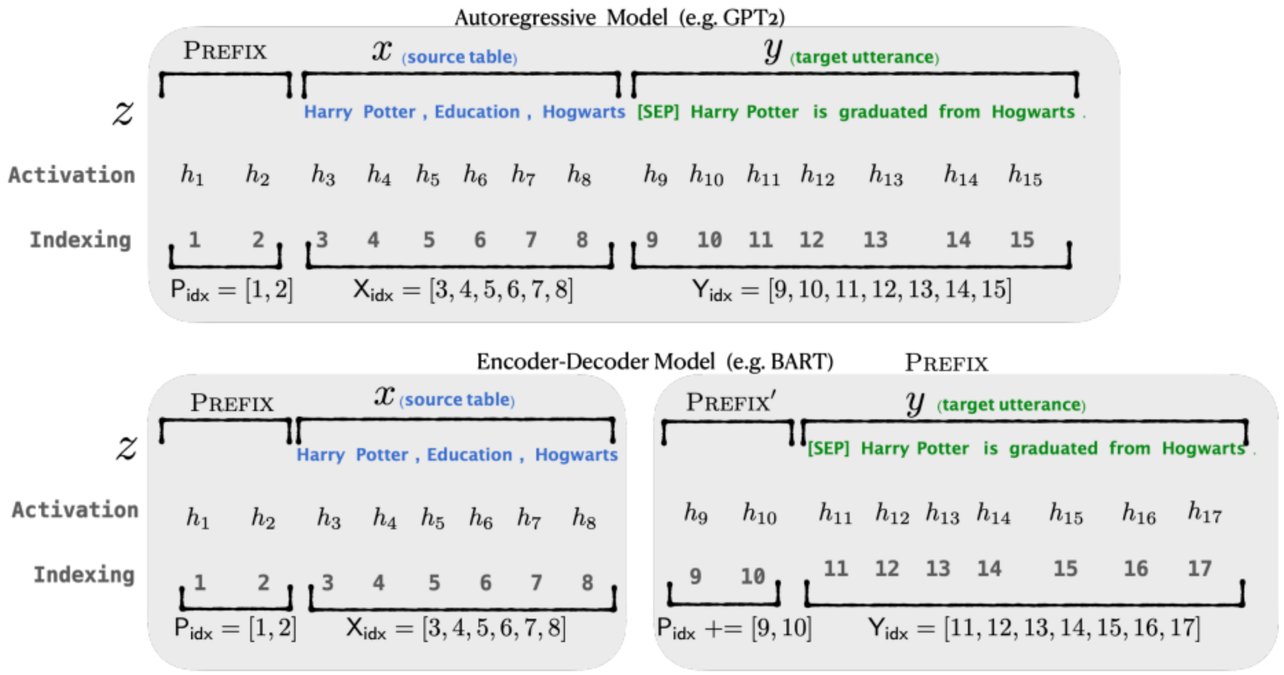


Shin, T, et al. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In EMNLP 2020.

Prefix-Tuning: Do Prompts Have to Be Discrete?

- **Directly optimize the embedding vectors for the prompt, instead of words**
- Adds fixed-length trainable prefix vectors to **each Transformer layer**

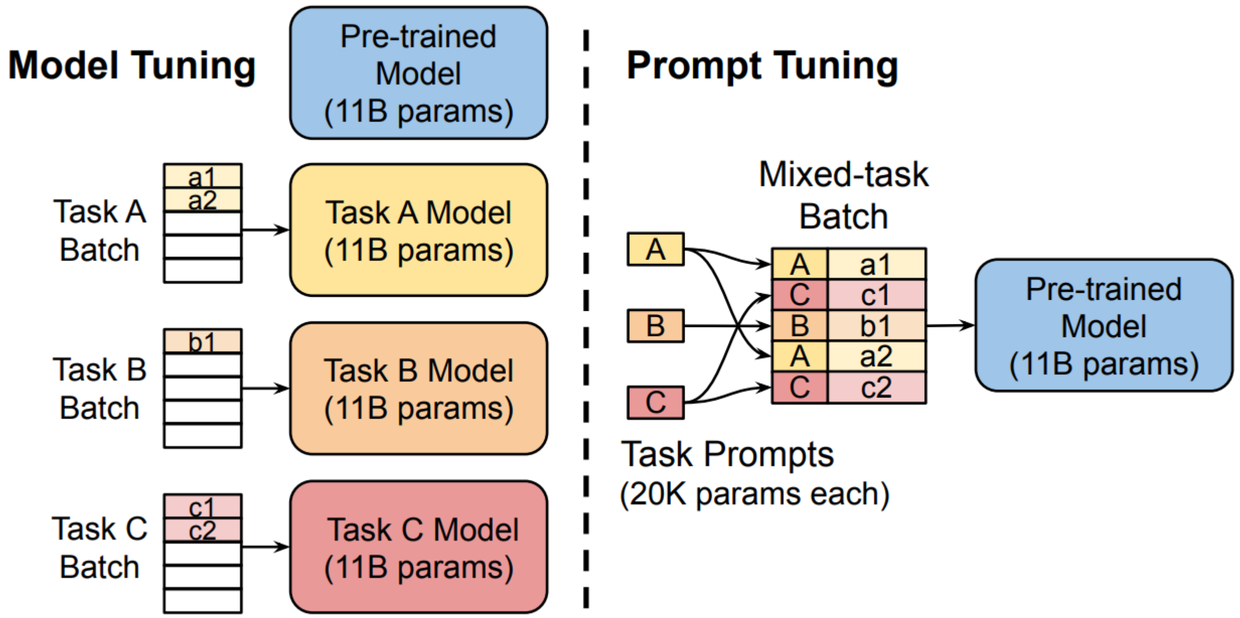


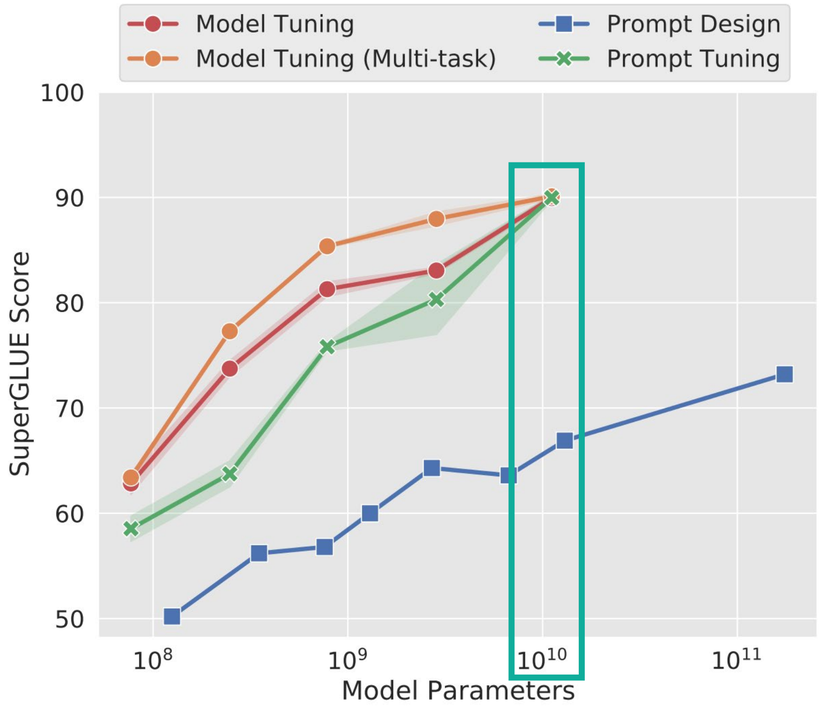


Li, X L, Liang, P. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In ACL 2021.

Prompt-Tuning: Continuous Prompts at its Best

- **Optimizing only the prefix for embedding layer instead of all layers in prefix-tuning**
- Lots of useful ablation studies about different designs!

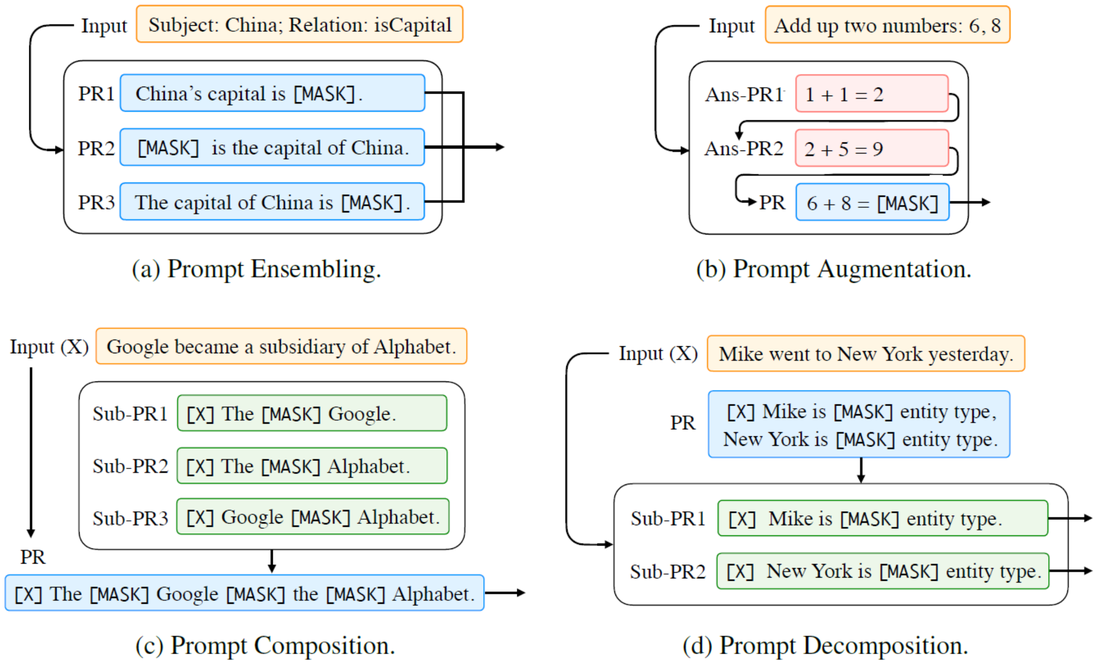




Lester, B, et al. 2021. The Power of Scale for Parameter-Efficient Prompt Tuning. In EMNLP 2021.

Multi-prompt Learning

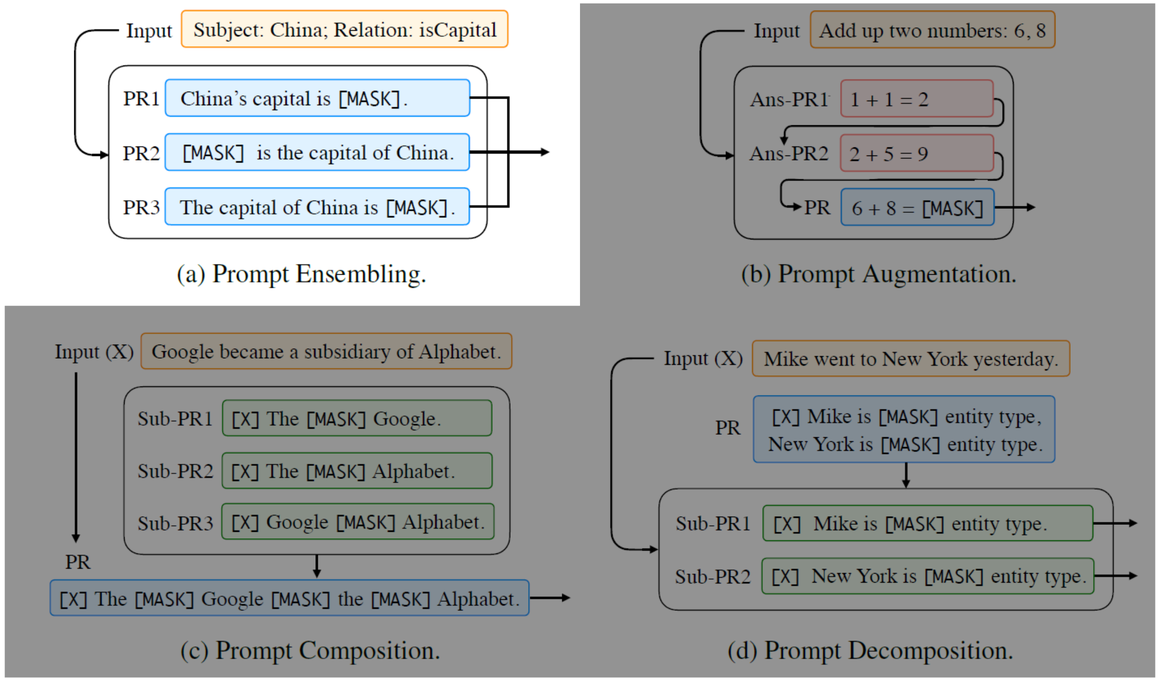
- **4 representative processes:** Prompt ensembling; Prompt augmentation; Prompt composition; Prompt decomposition.



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Multi-prompt Learning

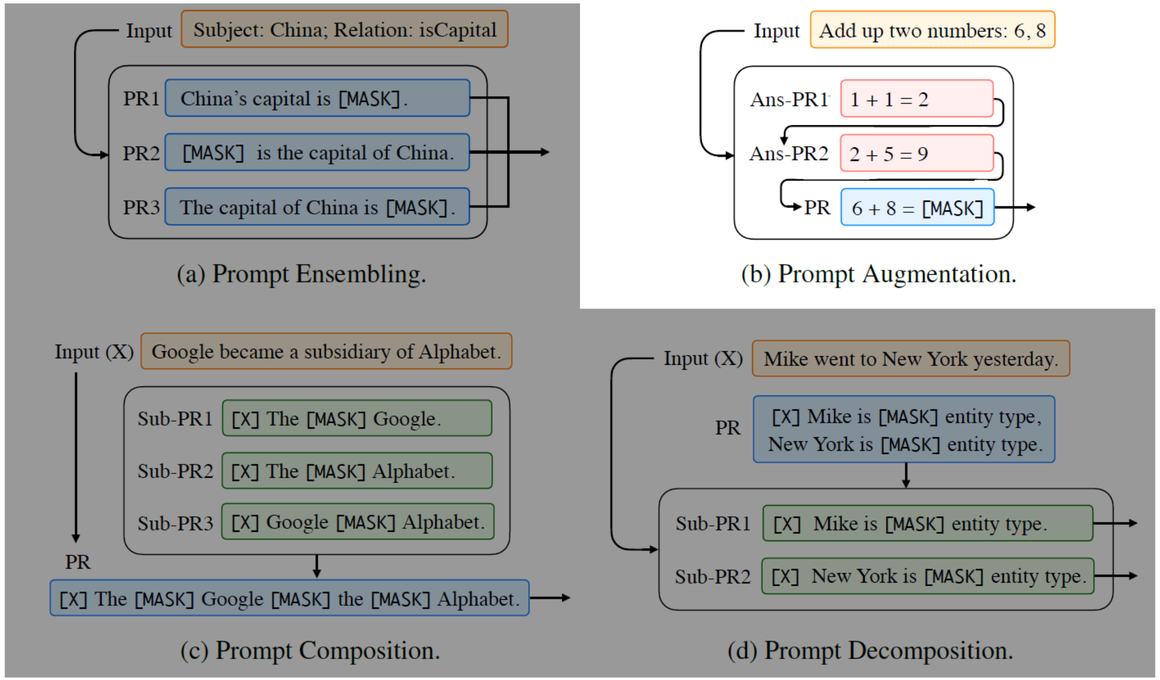
- **Prompt Ensembling:** Use multiple prompts and perform model ensembling techniques like weighted averaging or majority voting



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Multi-prompt Learning

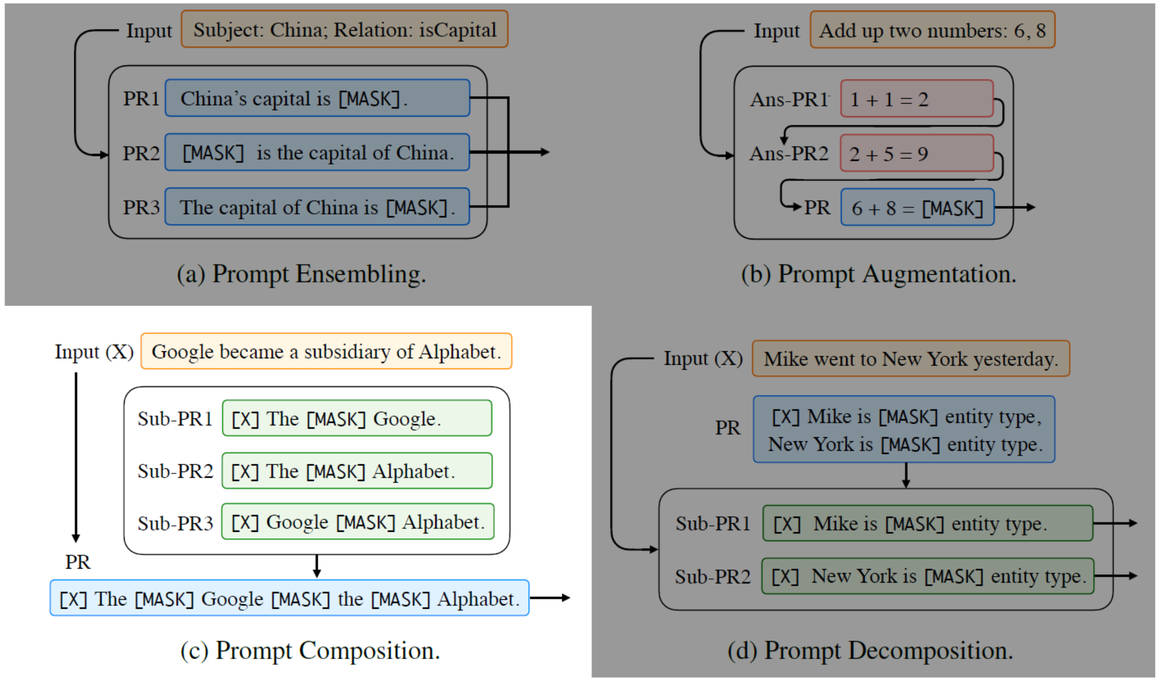
- **Prompt Augmentation:** Provide some examples of correct answers to the prompt. The selection and ordering of the examples are crucial.



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Multi-prompt Learning

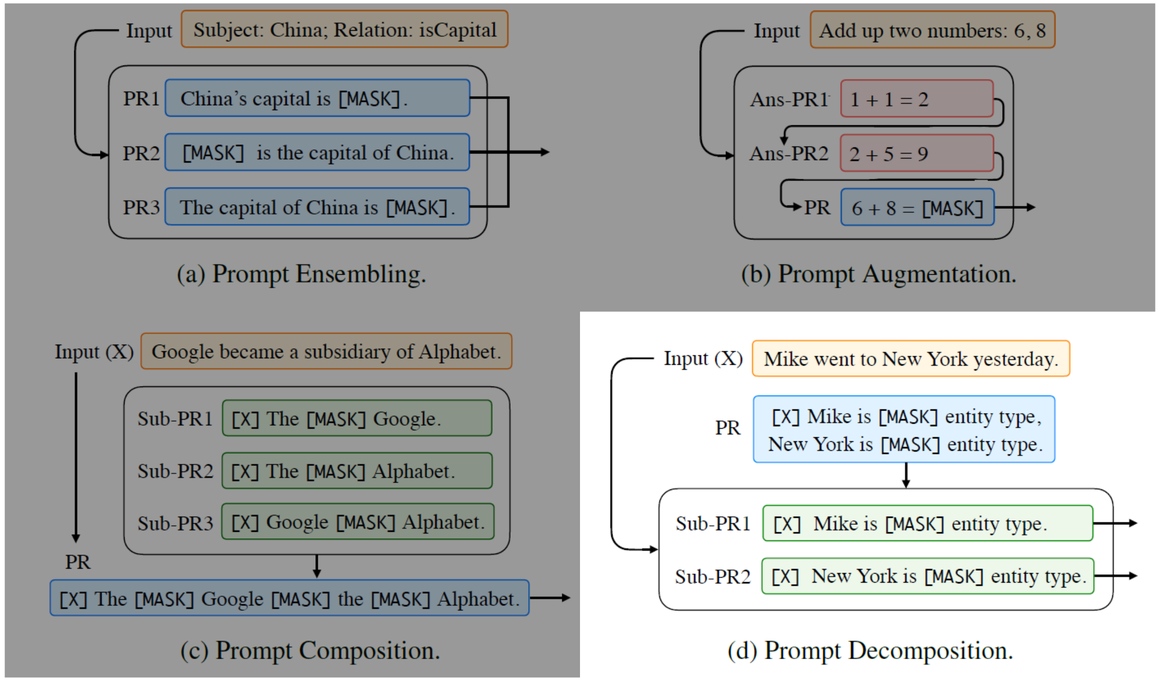
- **Prompt Composition:** For composable tasks (like relation extraction), compose several small sub-prompts into a single complete prompt for the task.



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

Multi-prompt Learning

- **Prompt Decomposition:** For tasks that needs multiple predictions (like sequence labeling), break down into sub-prompts and answer each separately.



Liu, P. et al. 2021. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. arXiv:2107.13586.

The Prompt-based Massive Multi-task Learning

How Does Prompting Affects Pretraining?

- Why different prompts for a single input have huge performance gap?
- A possible reason: **the prompt's expression is not like the ones PLM sees during pretraining; a gap has to be bridged**

- **What if PLM sees such "prompt-like" expressions during pretraining?**

Prompt	P@1
[X] is located in [Y]. *(original)*	31.29
[X] is located in which country or state? [Y].	19.78
[X] is located in which country? [Y].	31.40
[X] is located in which country? In [Y].	51.08

←The expressions in the pretraining corpus are like this.

←**What if we add lots of such kind of sentences for pretraining?**

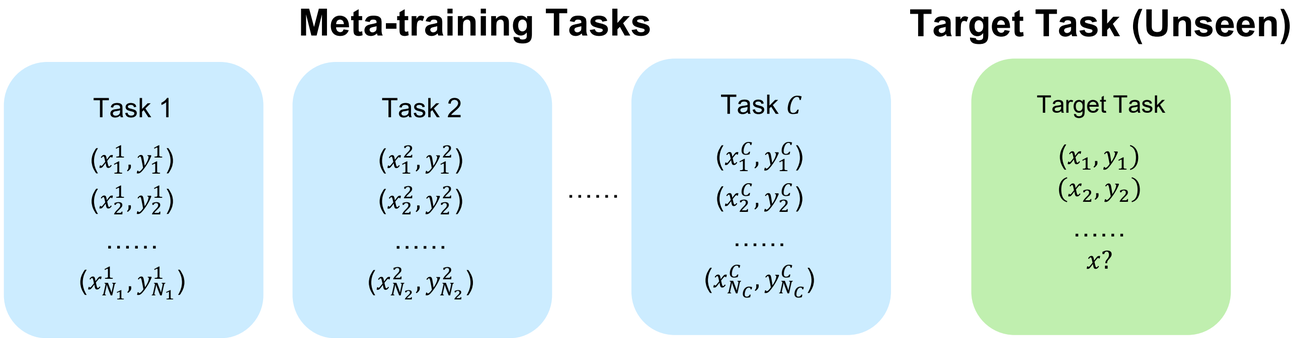
Table 1. Case study on LAMA-TREx P17 with bert-base-cased. A single-word change in prompts could yield a drastic difference.

Liu, X, et al. 2021. GPT Understands, Too. arXiv: 2103.10385.

MetaICL: No Need for Patterns After Meta Learning

- **Meta-learning for in-context learning: train the model to recognize task based on context instances (with meta-learning on 142 tasks)**
- No need for patterns: concatenate k labeled instances with the input

	Meta-training	Inference
Task	C *meta-training* tasks	An unseen *target* task
Data given	Training examples $\mathcal{T}_i = \{(x_j^i, y_j^i)\}_{j=1}^{N_i}$, $\forall i \in [1, C]$ $(N_i \gg k)$	Training examples $(x_1, y_1), \cdots, (x_k, y_k)$, Test input x
Objective	For each iteration, 1. Sample task $i \in [1, C]$ 2. Sample $k+1$ examples from $\mathcal{T}_i$: $(x_1, y_1), \cdots, (x_{k+1}, y_{k+1})$ 3. Maximize $P(y_{k+1} \vert x_{k+1}, x_1, y_1, \cdots, x_k, y_k, x_{k+1})$	$\text{argmax}_{c \in \mathcal{C}} P(c \vert x_1, y_1, \cdots, x_k, y_k, x)$

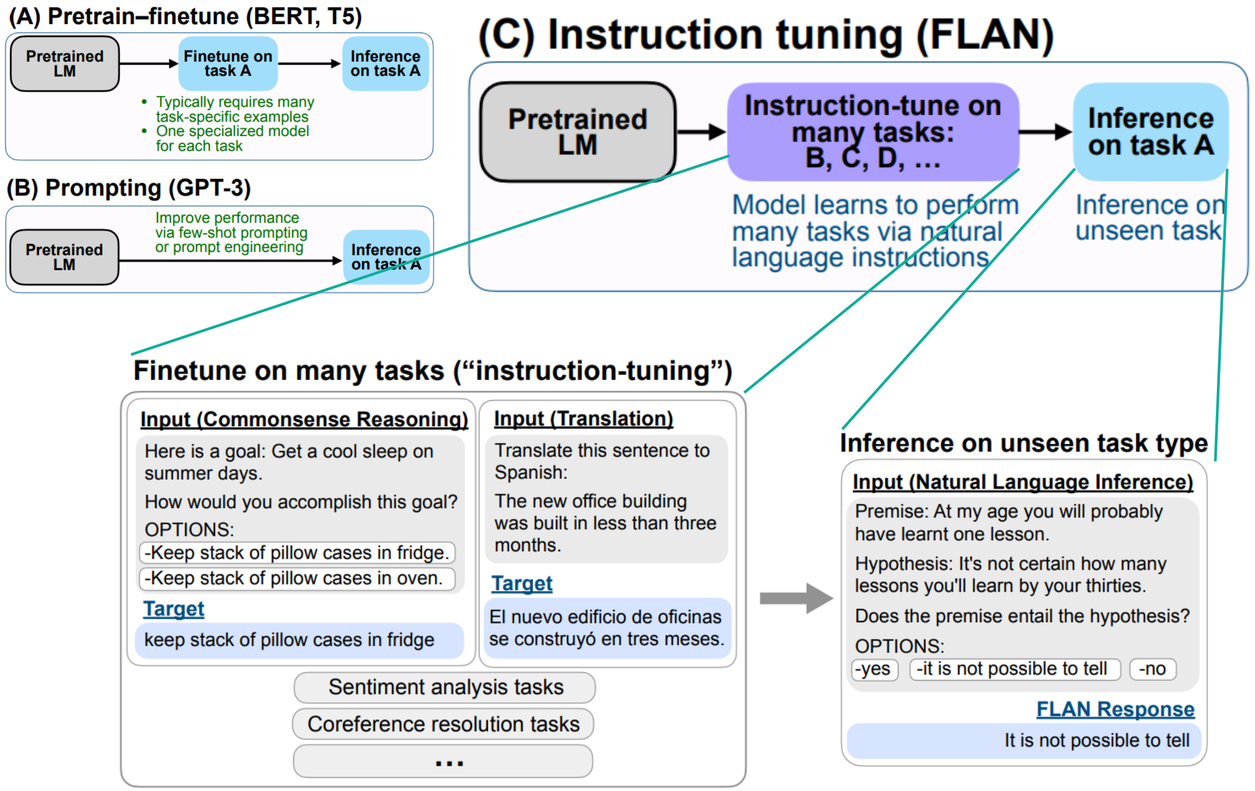


Min, S, et al. 2021. MetaICL: Learning to Learn In Context. arXiv:2110.15943.

FLAN: Bridge the Gap by Instruction Tuning

- **Fine-tune PLM on "instructions" from diverse labeled datasets**

1. Construct 10 templates for each dataset
2. Randomly select instance + template from all datasets to construct "instruction"
3. Instruction-tune PLM on all tasks
4. Zero-shot inference on unseen task with instruction (prompt)

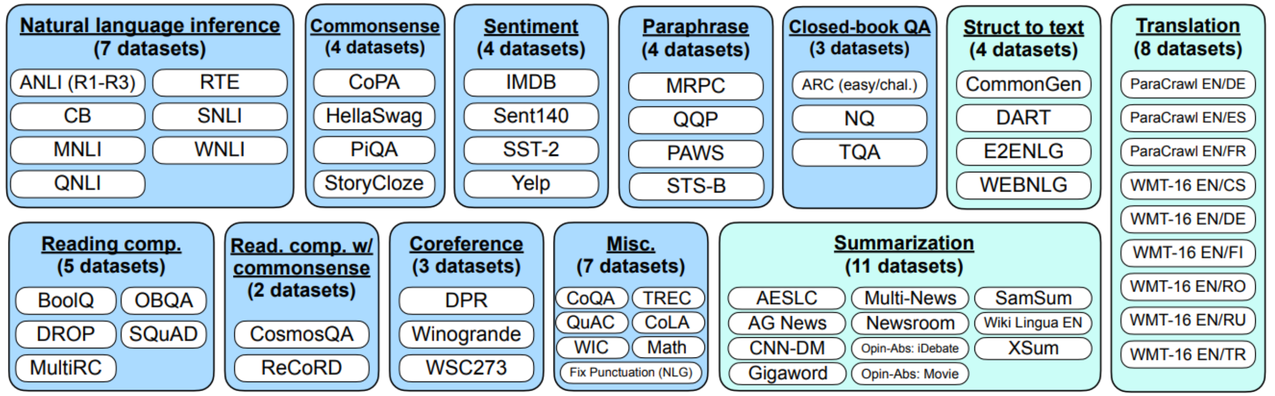


Wei, J, et al. 2021. Finetuned Language Models are Zero-Shot Learners. In ICLR 2022 (Oral).

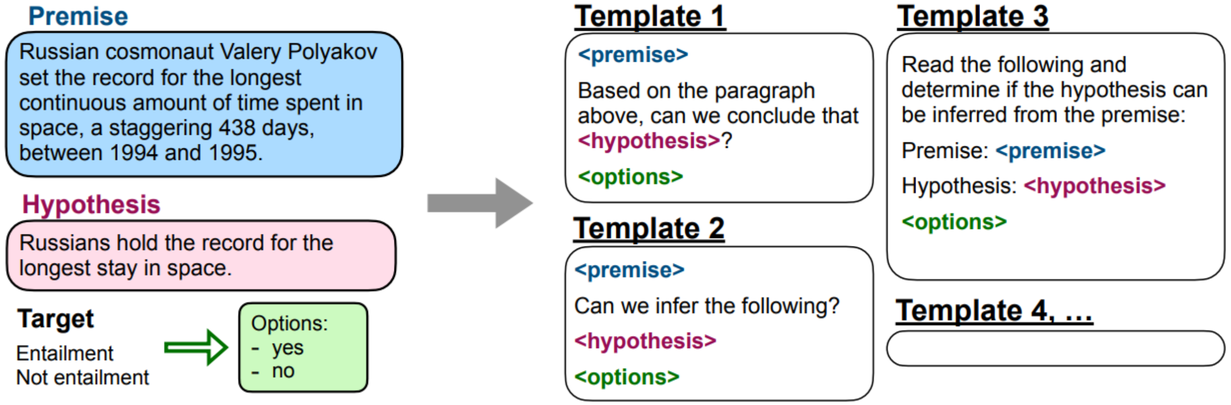
FLAN: Bridge the Gap by Instruction Tuning (contd.)

- **Fine-tune PLM on "instructions" from diverse labeled datasets**

Total 62 datasets for instruction-tuning
NLU tasks in blue, NLG tasks in teal



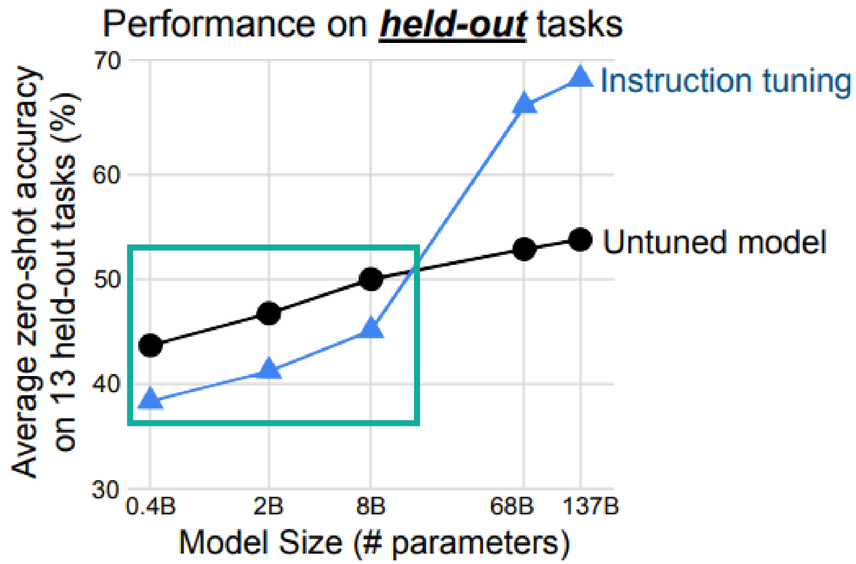
Example instruction templates for NLI
Use textual instructions to describe the task



Wei, J, et al. 2021. Finetuned Language Models are Zero-Shot Learners. In ICLR 2022 (Oral).

FLAN: Bridge the Gap by Instruction Tuning (contd.)

- **Fine-tune PLM on "instructions" from diverse labeled datasets**
- Makes texts seen in pretraining & inference more similar
- Only helps large PLMs to generalize; small models are limited by capacity

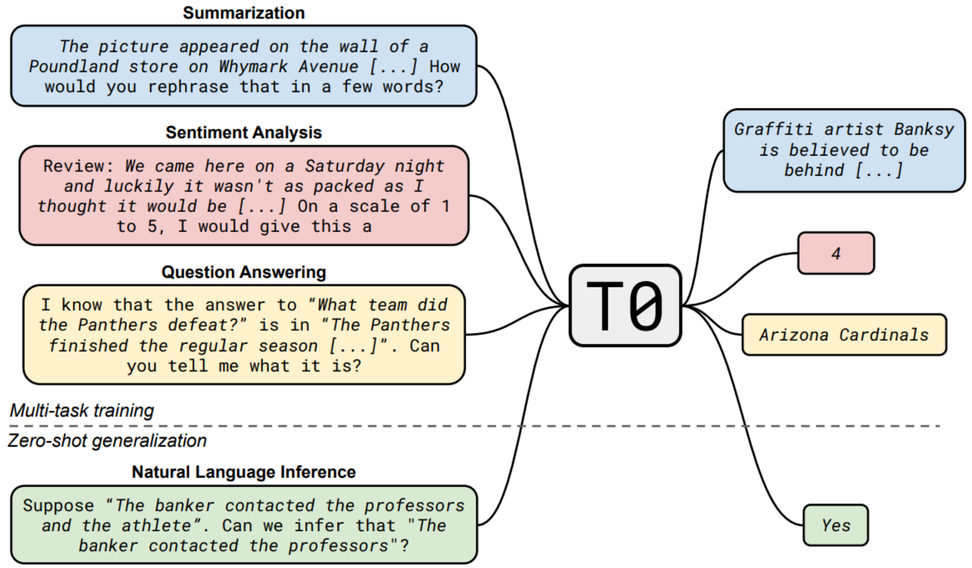


Wei, J, et al. 2021. Finetuned Language Models are Zero-Shot Learners. In ICLR 2022 (Oral).

T0: More Diverse Instructions, Less Parameters

- **Also fine-tune PLM on prompts from diverse labeled datasets**
- **Some differences vs. FLAN:**

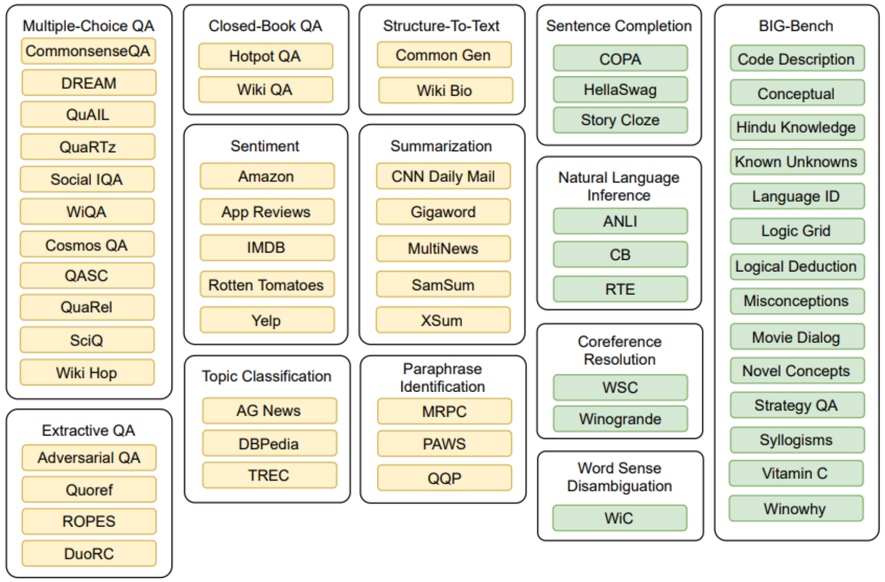
Strategy	T0	FLAN
PLM Selection	**T5+LM (Enc-Dec) trained with MLM**	LaMDA-PT (Dec) trained with LM
Dataset Count	**171**	62
Total Prompts	**1939**	620
Prompt Source	**Crowdsourcing (more diversity)**	Manually design (less diversity)



Sanh, V, et al. 2021. Multitask Prompted Training Enables Zero-Shot Task Generalization. In ICLR 2022 (Spotlight).

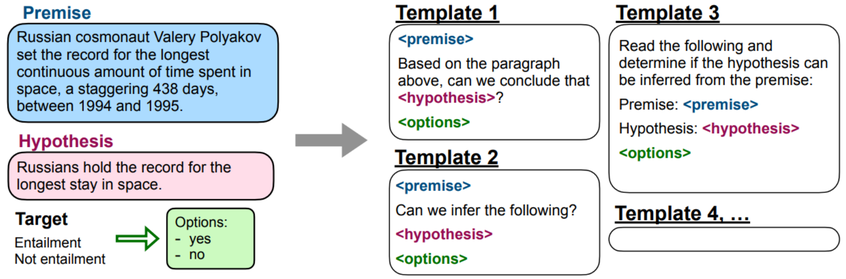
T0: More Diverse Instructions, Less Parameters (Contd.)

- **Also fine-tune PLM on prompts from diverse labeled datasets**



Total 171 datasets

Training tasks in yellow, validation tasks in green



(Compare with FLAN's)

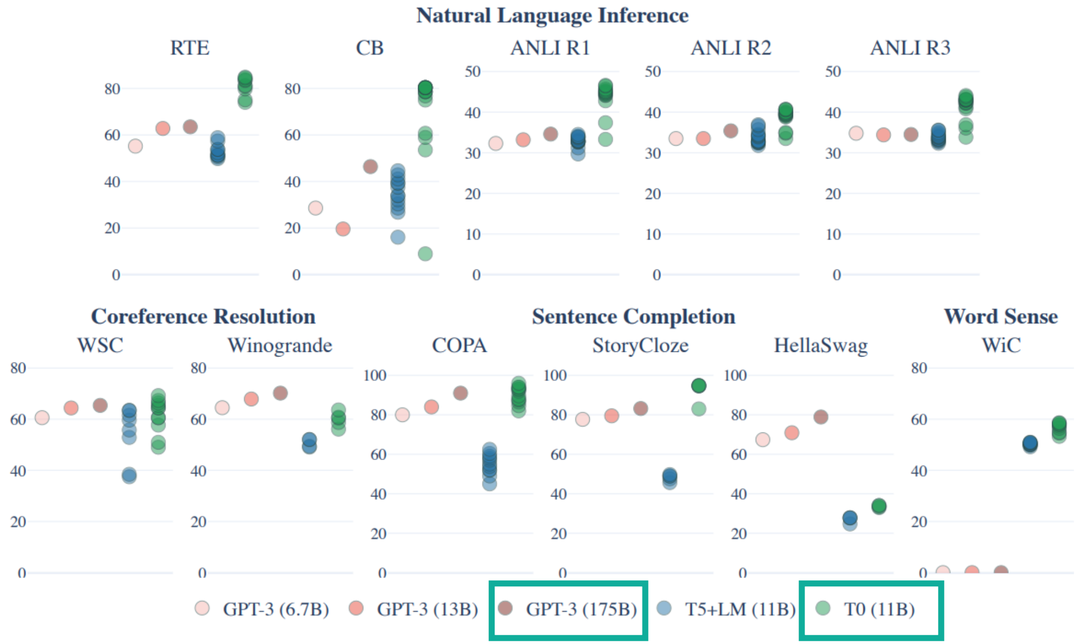
For example, consider one of our prompts for Quora Question Pairs (paraphrasing identification): `I'm an administrator on the website Quora. There are two posts, one that asks "question1" and another that asks "question2". I can merge questions if they are asking the same thing. Can I merge these two questions?` We hypothesize that this diversity could have concrete effects. For

The more diverse prompts from crowdsourcing

Sanh, V, et al. 2021. Multitask Prompted Training Enables Zero-Shot Task Generalization. In ICLR 2022 (Spotlight).

T0: More Diverse Instructions, Less Parameters (Contd.)

- **Also fine-tune PLM on prompts from diverse labeled datasets**
- The difference in PLM and prompt diversity brings different results

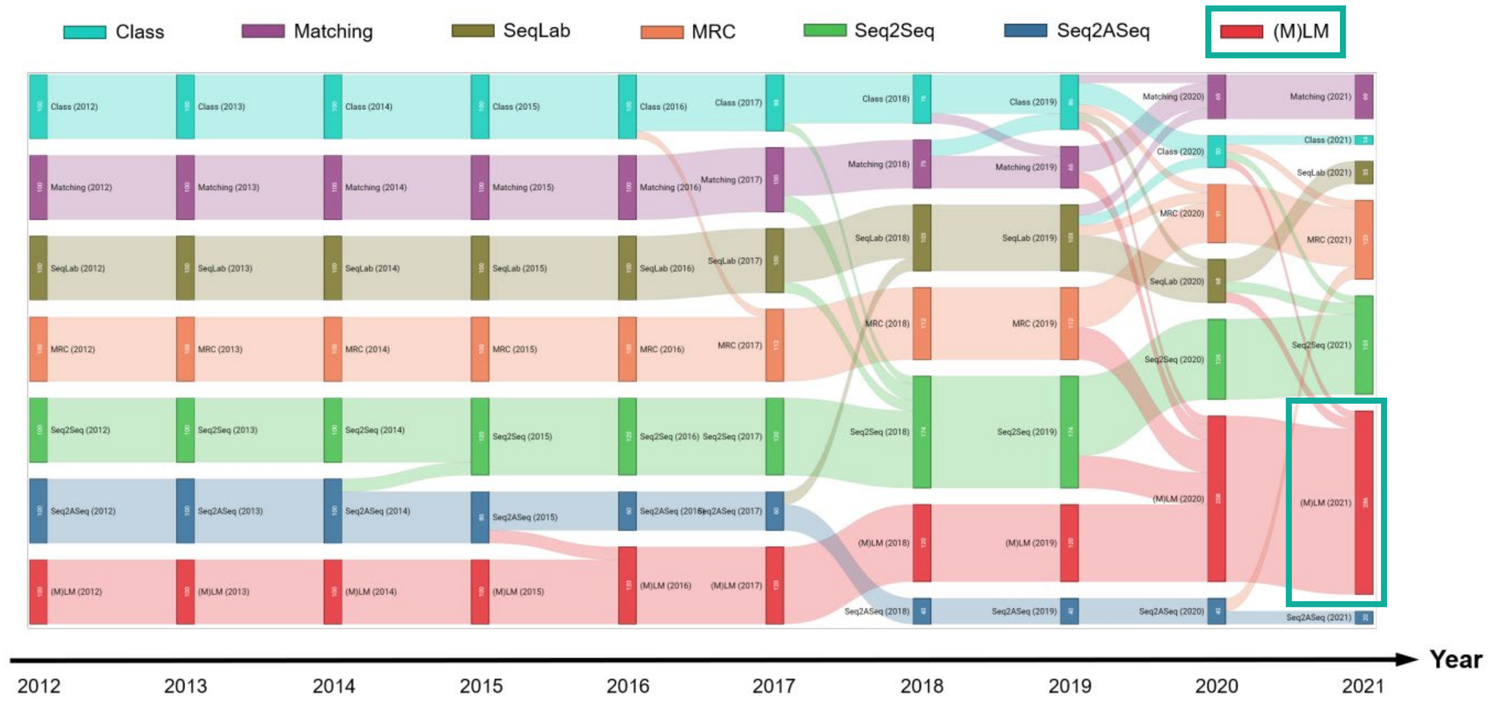


Small model also performs zero-shot well!

Sanh, V, et al. 2021. Multitask Prompted Training Enables Zero-Shot Task Generalization. In ICLR 2022 (Spotlight).

The General Paradigm Shift to LM-based Solutions

- Lots of NLP tasks can be solved by applying prompting to LM…
  Seems language modeling is unifying the task paradigms…?
- **Is this the real unified NLP solution we are seeking? Think about it ☺**



Sun, T, et al. 2021. Paradigm Shift in Natural Language Processing. arXiv:2109.12575.

References

1. CMU LTI CS11-711 Advanced NLP, Fall 2021: http://phontron.com/class/anlp2021/schedule/prompting.html#, Representation 3

- Recommended reading: Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing

About the Assignments and Mini-lectures

- Assignment 2 will be released today. It will be about neural machine translation (NMT) using seq2seq w/ attention. Due: 23:59 EST, March 1st.
- The grades for Assignment 1 will be released this week. We are still looking at your project proposals.
- Don’t forget to submit your mini-lecture slides on **both StudiUM and the slack channel #mini-lectures** before 11:59 a.m. EST, March 18th!
- Check your presentation order on the link posted in the #general channel in advance. Looking forward to your presentation!

Q&A