Applied Deep Learning

Prompt-Based Learning

June 13th, 2022 http://adl.miulab.tw

National Taiwan University
國立臺灣大學

Slides credited from Mohit Lyyer (Umass) and Hung-Yi Lee

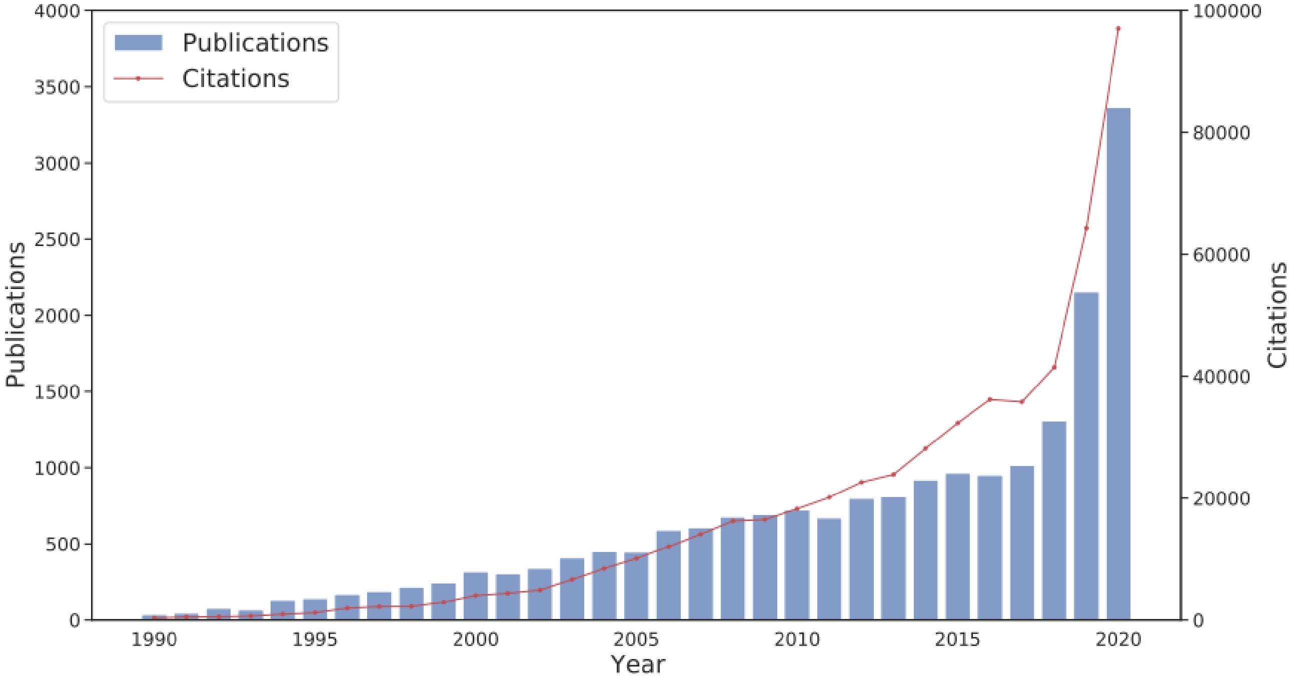
Fine-Tuning on Pretrained LMs

- (Standard) fine-tuning: use the pre-trained LMs for initialization and tuning the parameters for a **downstream** task

System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Wide Usage of PLMs (Han et al., 2021)

- Increasing usage of PLMs



Issue 1: Data Scarcity

- Downstream annotated data may not be large

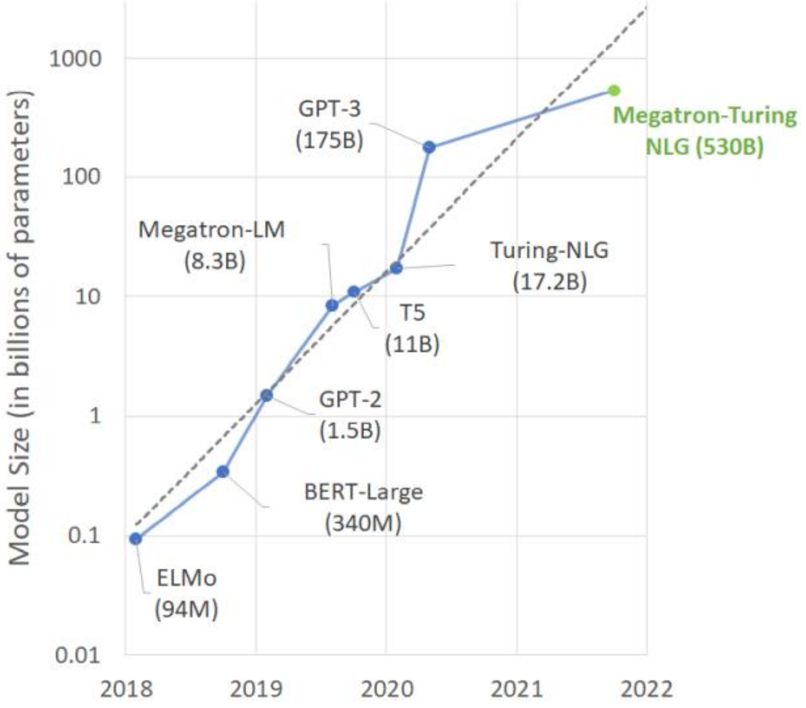
Task	MNLI	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE
Size	391K	363K	108K	67K	8.5K	5.7K	3.5K	2.5K

→ More practical cases are few-shot, one-shot or even zero-shot settings

Issue 2: Large-Scale PLMs

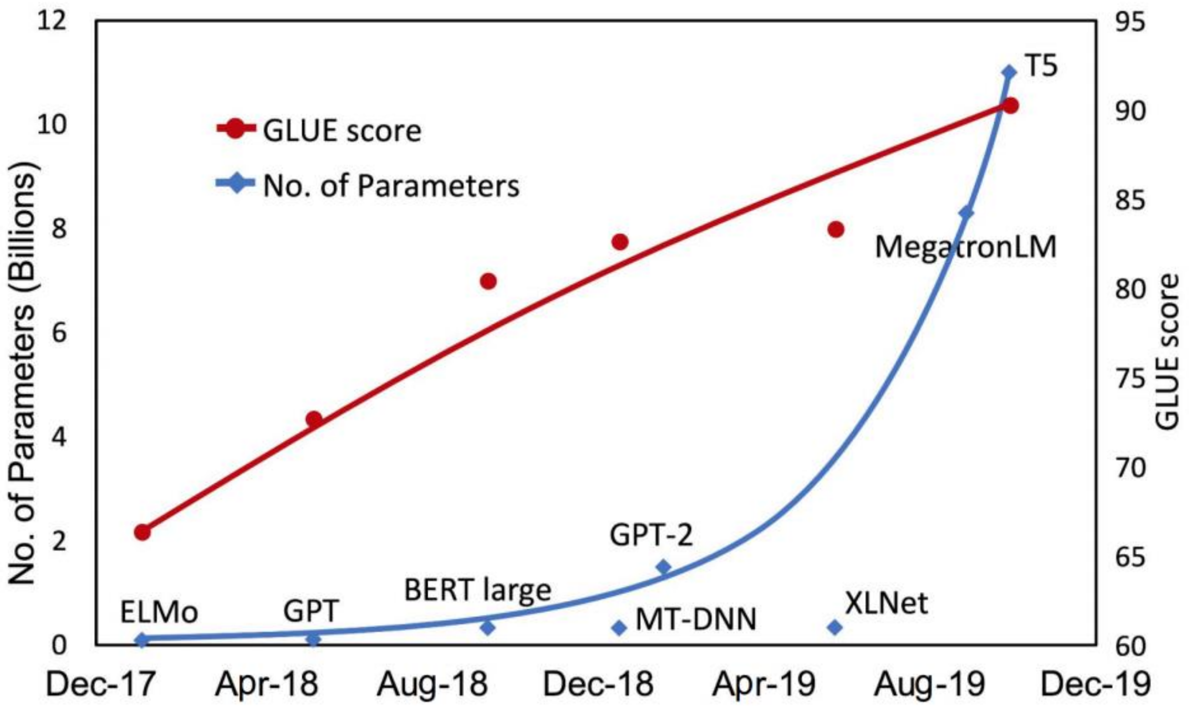
- PLMs are larger and larger

Model	#Params	#Layers
ELMo	93M	2 (BiLSTM)
BERT Base	110M	12
BERT Large	340M	24
GPT-3 Small	125M	12
GPT-3 Medium	350M	24
GPT-3 Large	760M	24
GPT-3 XL	1.3B	24
GPT-3 2.7B	2.7B	32
GPT-3 6.7B	6.7B	32
GPT-3 13B	13B	40
GPT-3 175B ("GPT-3")	175.0B	96



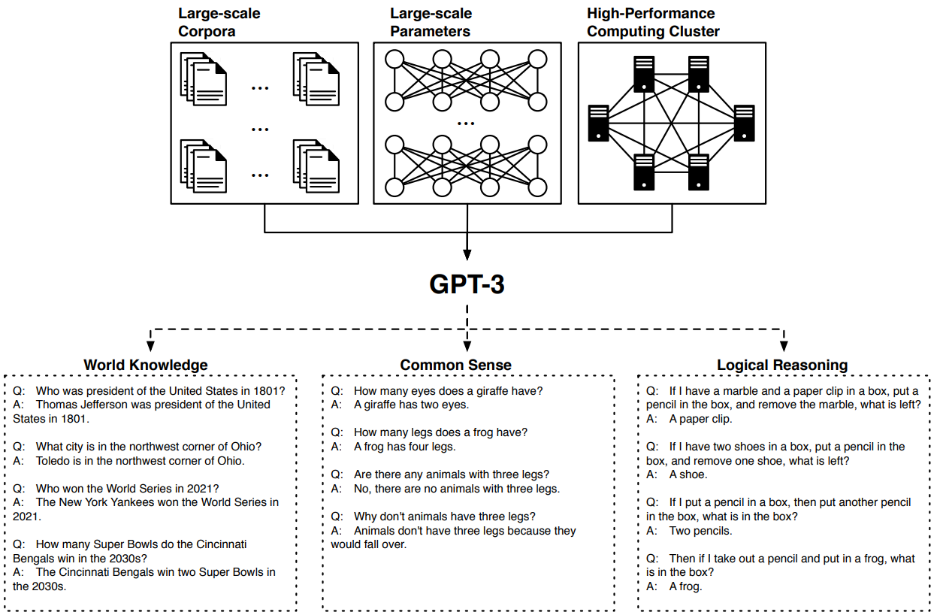
Better Performance from Larger Models

- Language understanding performance (Ahmet & Abdullah, 2021)



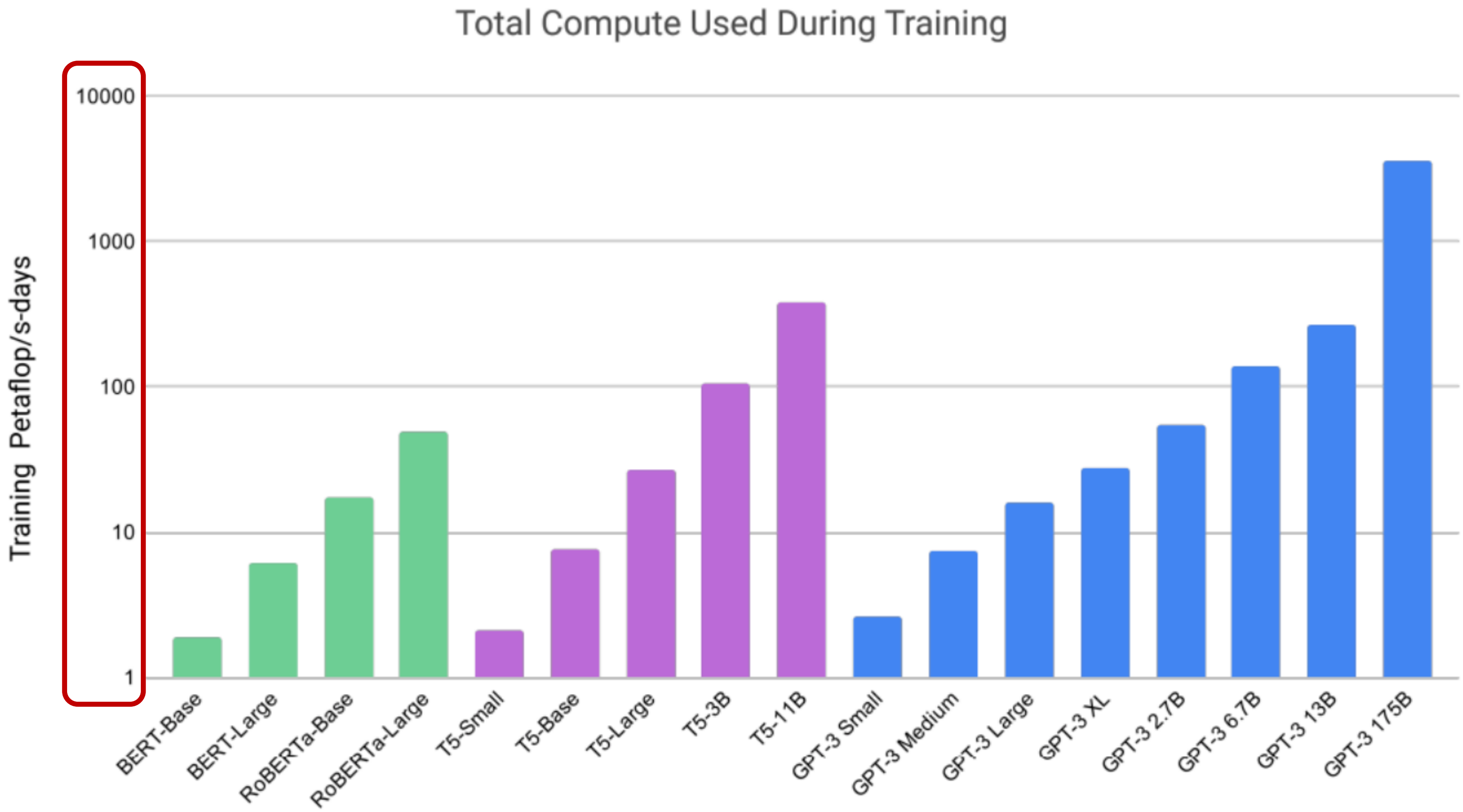
Better Performance from Large Models

- More types of data for pre-training → diverse capability



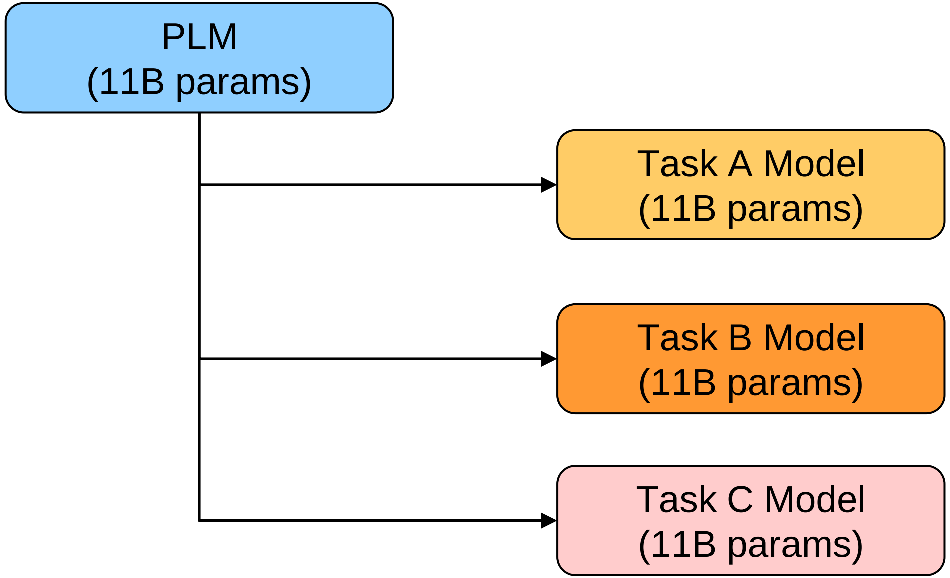
What is the problem of large PLMs?

Computing Cost of Large PLMs



Large Space Requirement

- Each task requires a copy of a large model



Practical Issues of PLMs

1) Data scarcity
2) Large PLMs
   - Higher training cost
   - Larger space requirement

→ **Solution: Prompt-Based Learning**

11

Prompt-Based Learning

Leveraging big pre-trained models

GPT-3 “In-Context” Learning

- **Zero-Shot**

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

prompt (提示)

- **One-Shot**

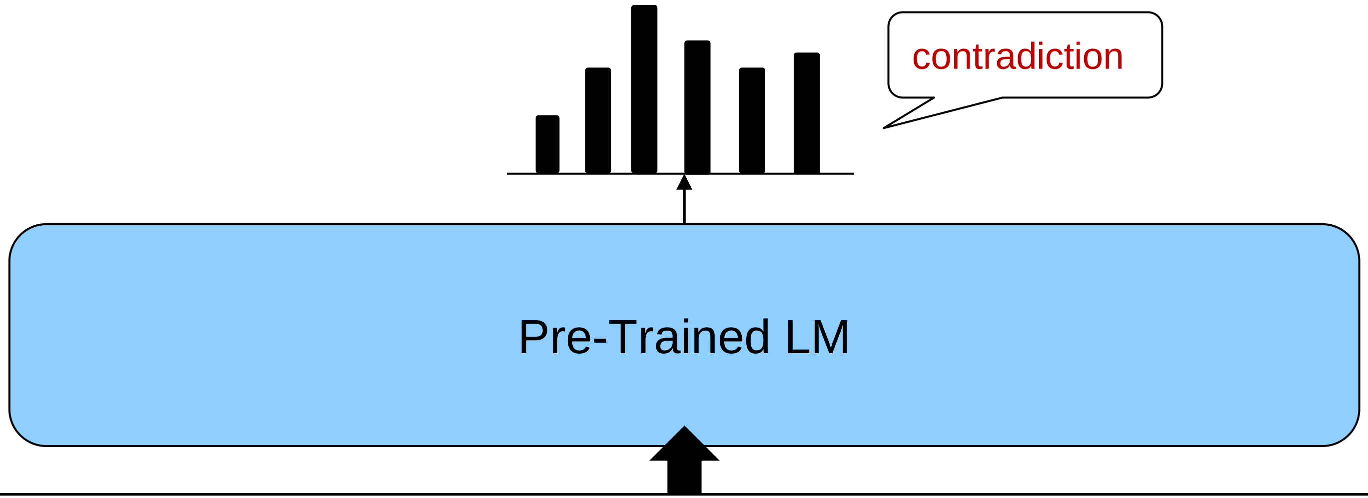
```
Translate English to French:   ← task description
sea otter => loutre de mer     ← example
cheese =>                      ← prompt
```

natural language instruction and/or a few task demonstrations

- **Few-Shot**

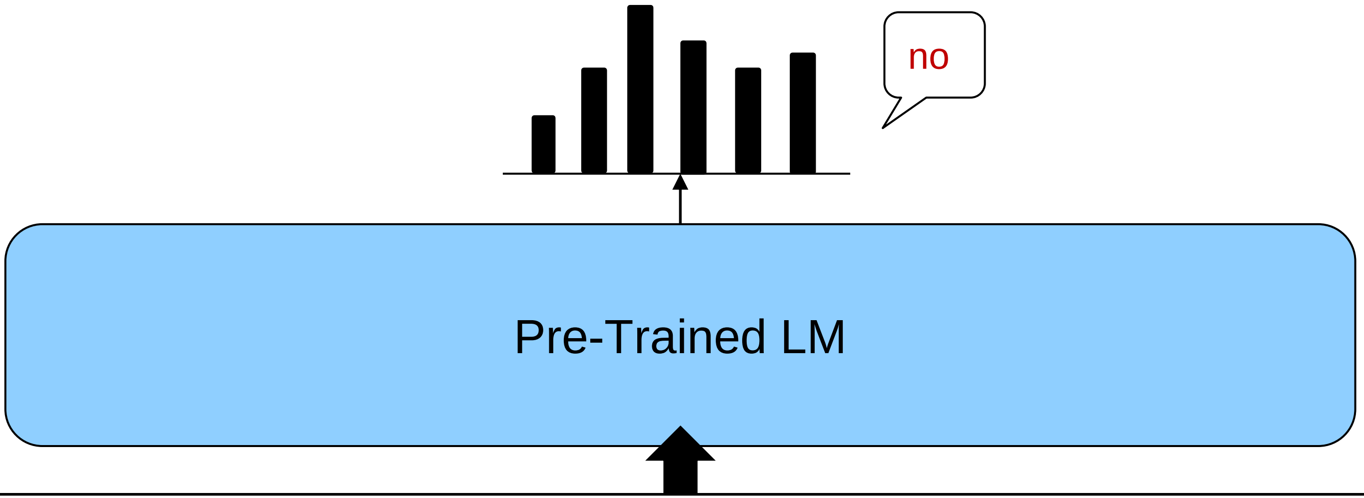
```
Translate English to French:          ← task description
sea otter => loutre de mer            ← examples
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>                             ← prompt
```

Prompt-Tuning



[CLS] Vivian likes dancing. [SEP] Vivian loves singing. [SEP] >>> neutral
[CLS] The vacation is coming soon. [SEP] The vacation was over. [SEP] >>> contradiction
[CLS] I am going to have dinner. [SEP] I am going to eat something. [SEP] >>> entailment
……
[CLS] I like strawberries. [SEP] I hate strawberries. [SEP] >>>

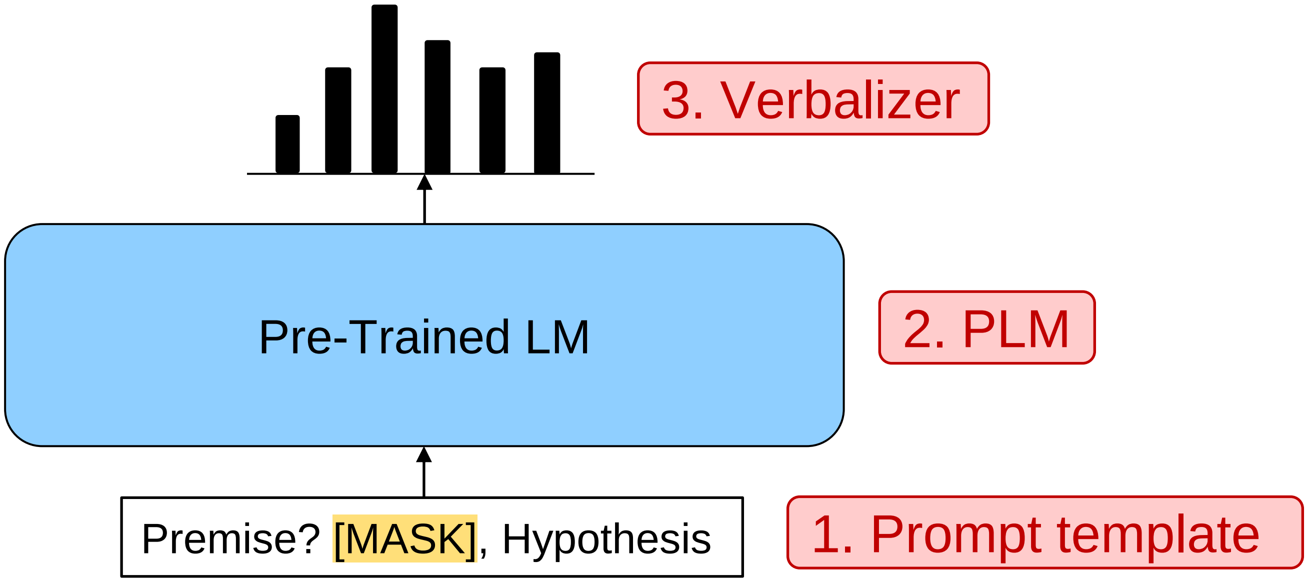
Prompt-Tuning



[CLS] Vivian likes dancing. Is it true that Vivian loves singing? [SEP] >>> maybe
[CLS] The vacation is coming soon. Is it true that the vacation was over? [SEP] >>> no
[CLS] I am going to have dinner. Is it true that I am going to eat something? [SEP] >>> yes
……
[CLS] I like strawberries. Is it true that I hate strawberries? [SEP] >>>

Prompt-Tuning

- Idea: convert data into natural language prompts
  → better for few-shot, one-shot, or zero-shot cases



Prompt-Tuning

1. Prompt template: manually designed natural language input for a task

NLI sample datapoint

Premise	Vivian is Jolin’s fans
Hypothesis	Vivian loves Jolin.
Label	0

0: “entailment”
1: “neutral”
2: “contradiction”

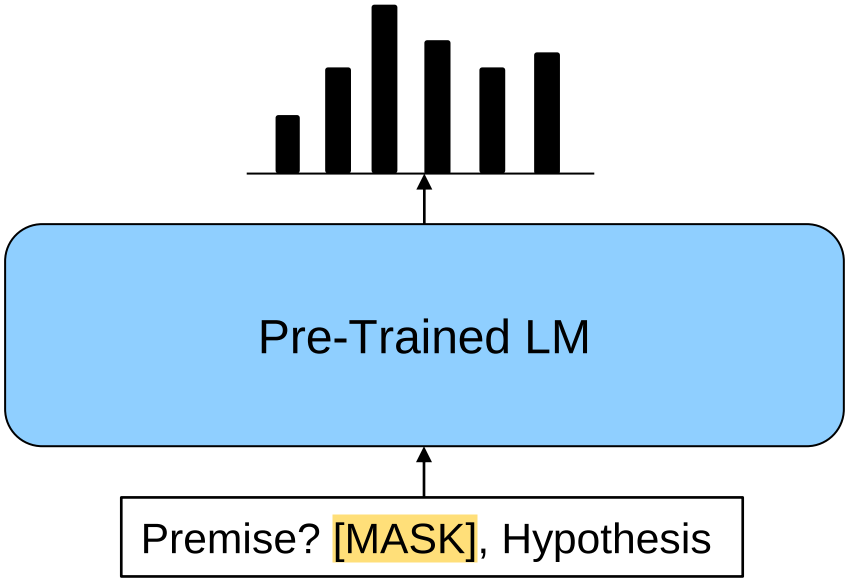
[CLS] Vivian is Jolin’s fans? [MASK], Vivian loves Jolin.

Premise? [MASK], Hypothesis

prompt template

Prompt-Tuning

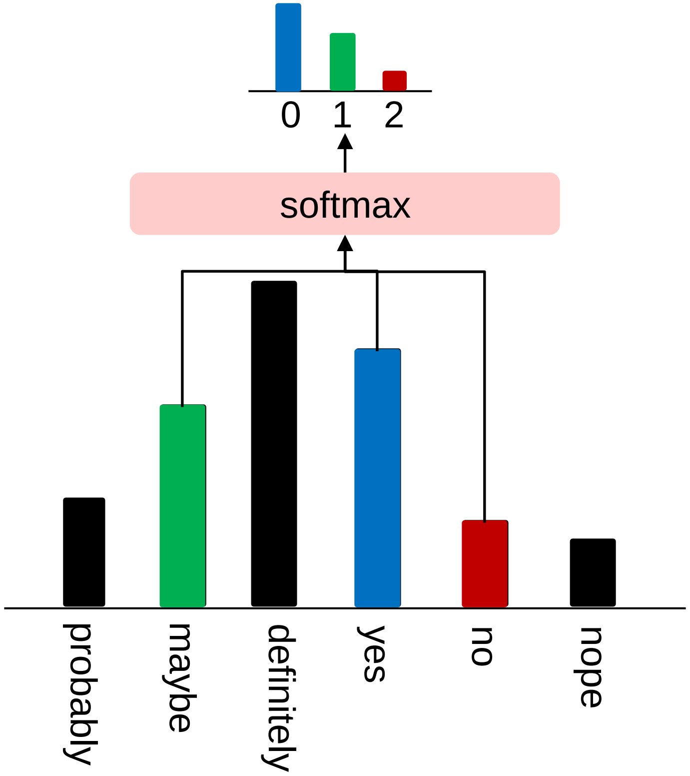
2. PLM: perform language modeling (masked LM or auto-regressive LM)



Prompt-Tuning

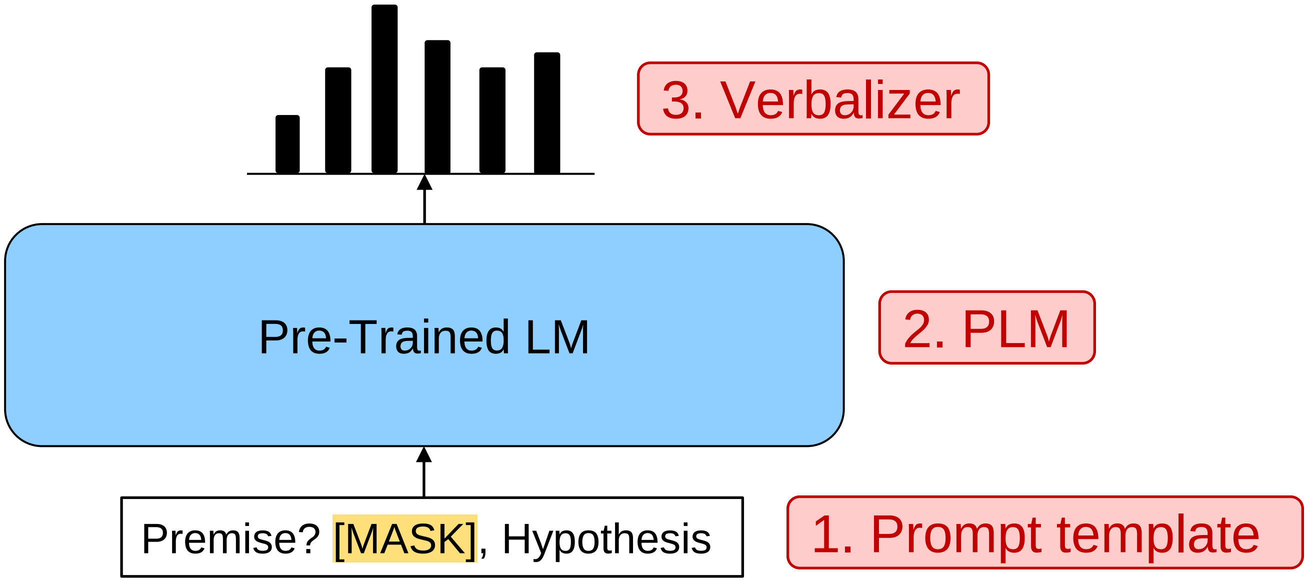
3. Verbalizer: mapping from the vocabulary to labels

0: “entailment” → yes
1: “neutral” → maybe
2: “contradiction” → no



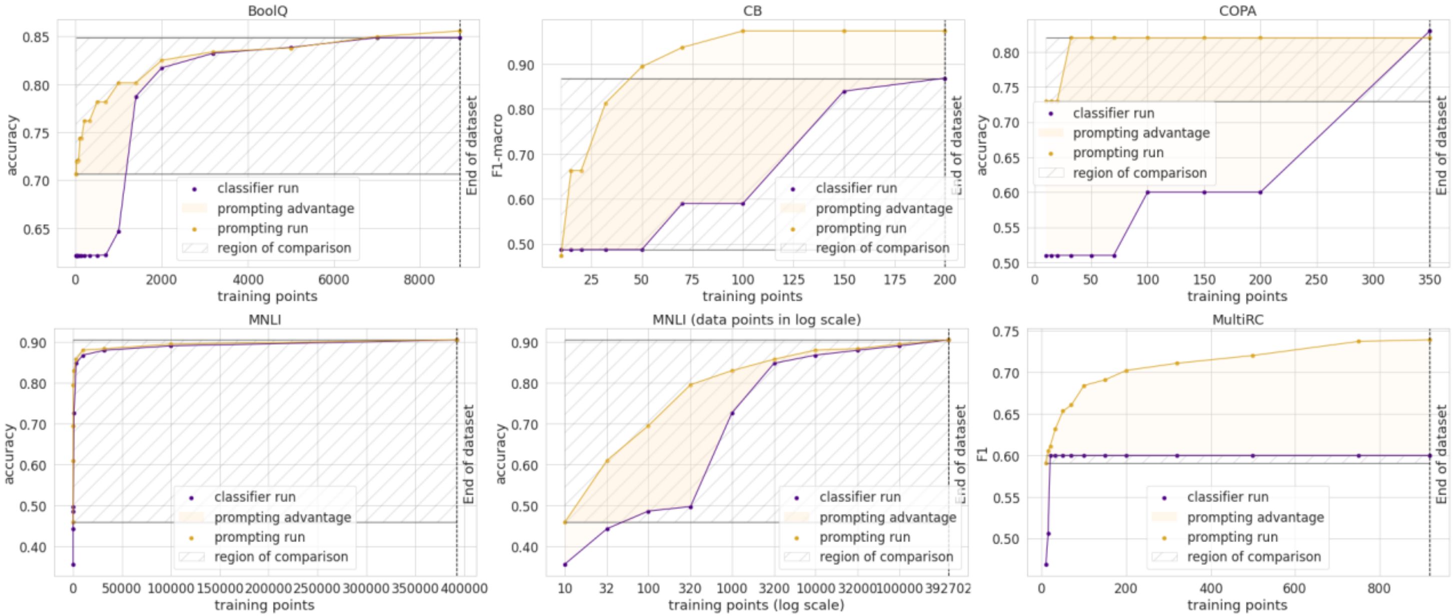
Prompt-Tuning

- Fine-tuning PLMs based on few annotated data samples
  - No parameter tuning when zero-shot settings



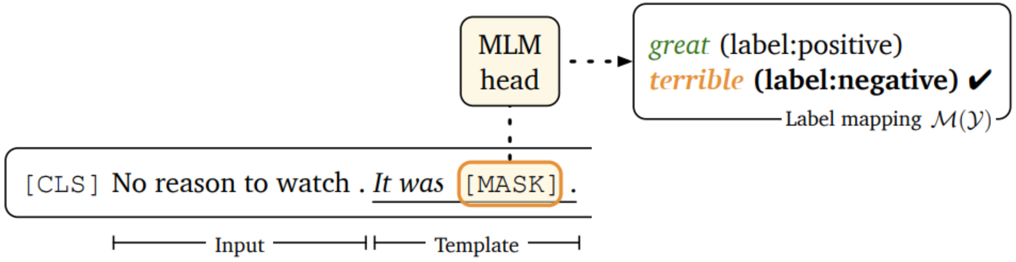
Prompt-Tuning

- Prompt-tuning is better under data scarcity (Le and Rush, 2021) due to
  - It better leverages pre-trained knowledge
  - Pre-trained knowledge can be kept

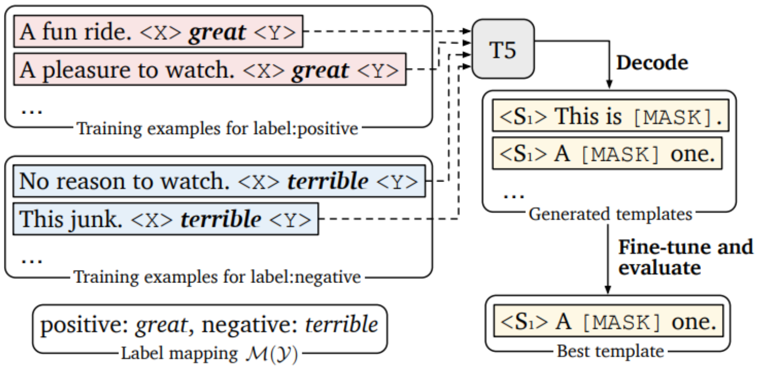


LM-BFF: Better Few-shot Fine-tuning of Language Models (Gao et al., 2021)

- Idea: prompt + demonstration for few-shot learning



- template generation



LM-BFF: Better Few-shot Fine-tuning of Language Models (Gao et al., 2021)

- Performance with RoBERTa-Large

	SST-2 (acc)	**SST-5** (acc)	**MR** (acc)	**CR** (acc)	**MPQA** (acc)	**Subj** (acc)	**TREC** (acc)	**CoLA** (Matt.)
Majority†	*50.9*	*23.1*	*50.0*	*50.0*	*50.0*	*50.0*	*18.8*	*0.0*
Prompt-based zero-shot‡	83.6	35.0	80.8	79.5	67.6	51.4	32.0	2.0
"GPT-3" in-context learning	84.8 (1.3)	30.6 (0.9)	80.5 (1.7)	87.4 (0.8)	63.8 (2.1)	53.6 (1.0)	26.2 (2.4)	-1.5 (2.4)
Fine-tuning	81.4 (3.8)	43.9 (2.0)	76.9 (5.9)	75.8 (3.2)	72.0 (3.8)	90.8 (1.8)	88.8 (2.1)	**33.9** (14.3)
Prompt-based FT (man)	92.7 (0.9)	47.4 (2.5)	87.0 (1.2)	90.3 (1.0)	84.7 (2.2)	91.2 (1.1)	84.8 (5.1)	9.3 (7.3)
+ demonstrations	92.6 (0.5)	**50.6** (1.4)	86.6 (2.2)	90.2 (1.2)	**87.0** (1.1)	**92.3** (0.8)	87.5 (3.2)	18.7 (8.8)
Prompt-based FT (auto)	92.3 (1.0)	49.2 (1.6)	85.5 (2.8)	89.0 (1.4)	85.8 (1.9)	91.2 (1.1)	88.2 (2.0)	14.0 (14.1)
+ demonstrations	**93.0** (0.6)	49.5 (1.7)	**87.7** (1.4)	**91.0** (0.9)	86.5 (2.6)	91.4 (1.8)	**89.4** (1.7)	21.8 (15.9)
Fine-tuning (full)†	*95.0*	*58.7*	*90.8*	*89.4*	*87.8*	*97.0*	*97.4*	*62.6*

	MNLI (acc)	**MNLI-mm** (acc)	**SNLI** (acc)	**QNLI** (acc)	**RTE** (acc)	**MRPC** (F1)	**QQP** (F1)	**STS-B** (Pear.)
Majority†	*32.7*	*33.0*	*33.8*	*49.5*	*52.7*	*81.2*	*0.0*	-
Prompt-based zero-shot‡	50.8	51.7	49.5	50.8	51.3	61.9	49.7	-3.2
"GPT-3" in-context learning	52.0 (0.7)	53.4 (0.6)	47.1 (0.6)	53.8 (0.4)	60.4 (1.4)	45.7 (6.0)	36.1 (5.2)	14.3 (2.8)
Fine-tuning	45.8 (6.4)	47.8 (6.8)	48.4 (4.8)	60.2 (6.5)	54.4 (3.9)	76.6 (2.5)	60.7 (4.3)	53.5 (8.5)
Prompt-based FT (man)	68.3 (2.3)	70.5 (1.9)	77.2 (3.7)	64.5 (4.2)	69.1 (3.6)	74.5 (5.3)	65.5 (5.3)	71.0 (7.0)
+ demonstrations	**70.7** (1.3)	**72.0** (1.2)	**79.7** (1.5)	**69.2** (1.9)	68.7 (2.3)	77.8 (2.0)	**69.8** (1.8)	73.5 (5.1)
Prompt-based FT (auto)	68.3 (2.5)	70.1 (2.6)	77.1 (2.1)	68.3 (7.4)	**73.9** (2.2)	76.2 (2.3)	67.0 (3.0)	75.0 (3.3)
+ demonstrations	70.0 (3.6)	**72.0** (3.1)	77.5 (3.5)	68.5 (5.4)	71.1 (5.3)	**78.1** (3.4)	67.7 (5.8)	**76.4** (6.2)
Fine-tuning (full)†	*89.8*	*89.5*	*92.6*	*93.3*	*80.9*	*91.4*	*81.7*	*91.9*

Issues of Discrete/Hard Prompts

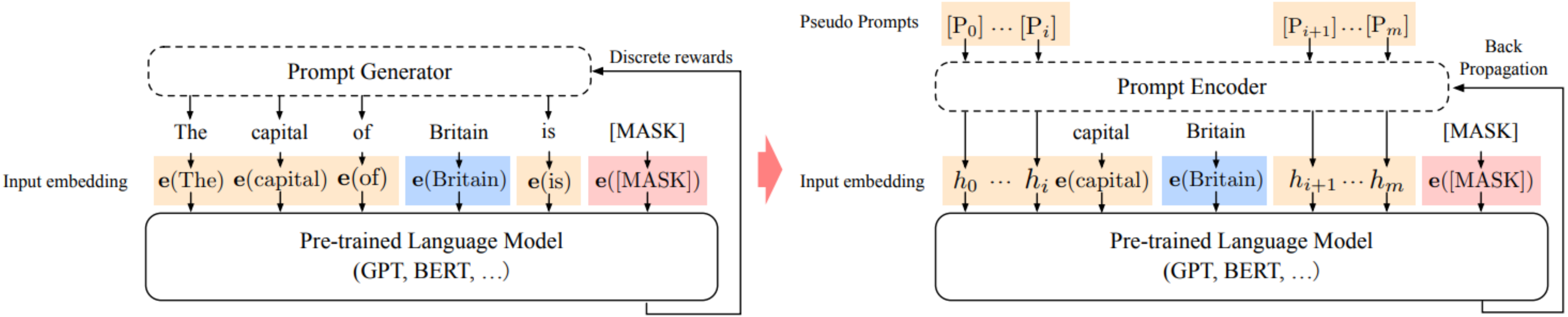
- Difficulty of manually designing prompts
  - Prompts that humans consider reasonable is not necessarily effective for LMs (Liu et al., 2021)
  - Pre-trained LMs are sensitive to the choice of prompts (Zhao et al., 2021)

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

P-Tuning (Liu et al., 2021)

- Idea: direct optimize the embeddings instead of prompt tokens

prompt search for "The capital of Britain is [MASK]".

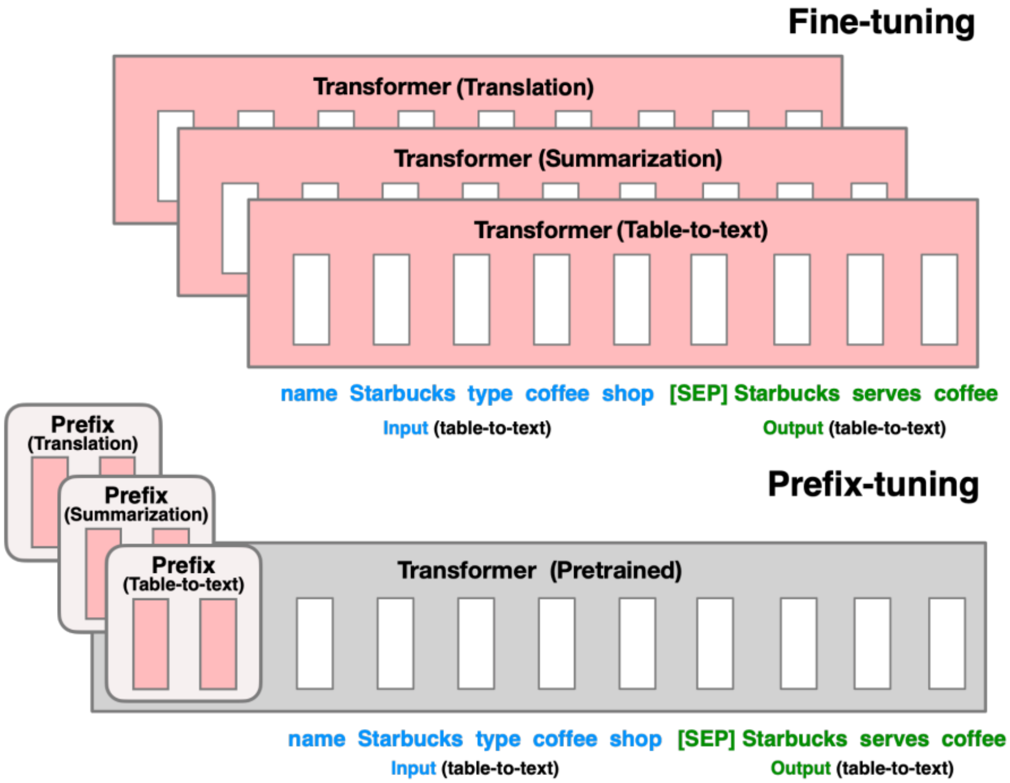


(a) Discrete Prompt Search (b) P-tuning

Prompt	$\mathcal{D}_{dev}$ Acc.	$\mathcal{D}_{dev32}$ Acc.
Does [PRE] agree with [HYP]? [MASK].	57.16	53.12
Does [HYP] agree with [PRE]? [MASK].	51.38	50.00
Premise: [PRE] Hypothesis: [HYP] Answer: [MASK].	68.59	55.20
[PRE] question: [HYP]. true or false? answer: [MASK].	70.15	53.12
P-tuning	76.45	56.25

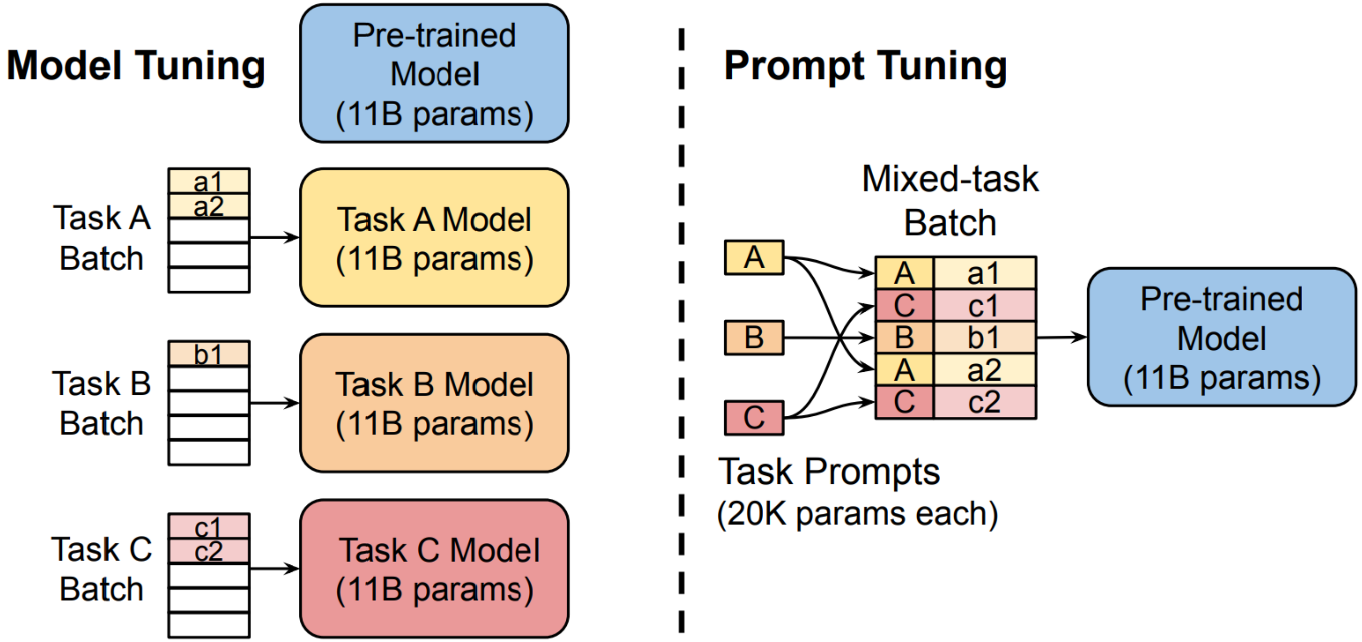
Prefix-Tuning (Li and Liang, 2021)

- Idea: only optimize the <u>prefix embeddings (all layers)</u> for efficiency



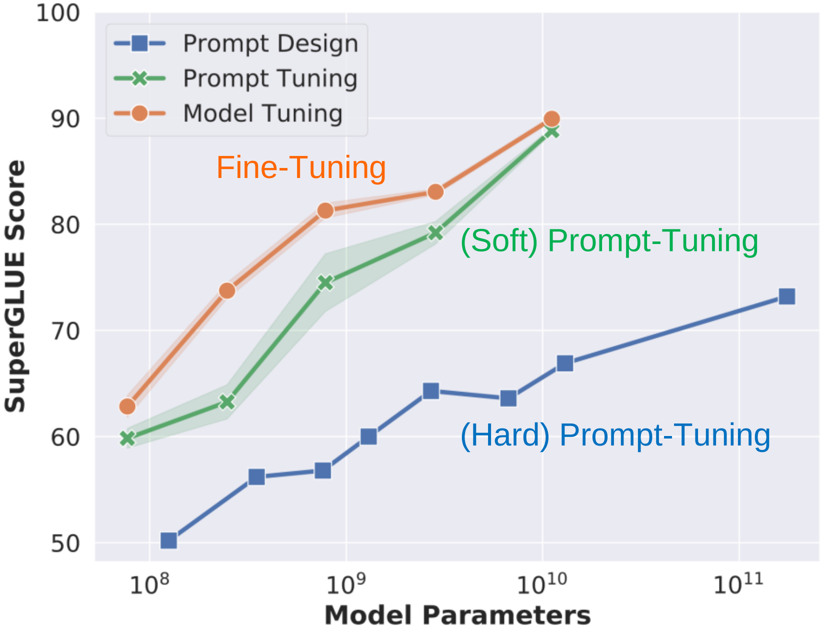
(Soft) Prompt-Tuning (Lester et al., 2021)

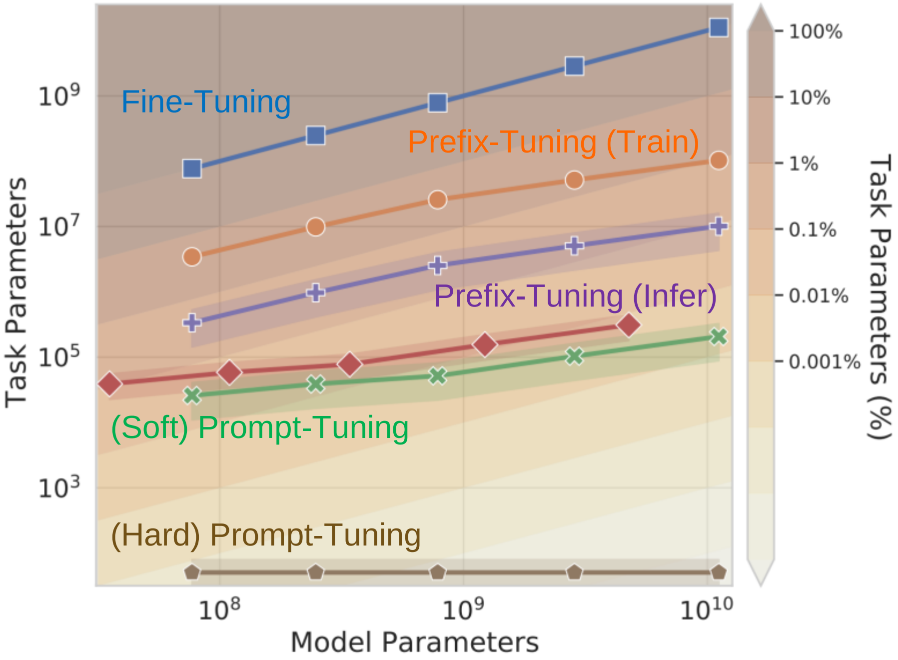
- Idea: only require storing a small task-specific prompt (one layer) for each task and enables mixed-task inference using the original PLMs



(Soft) Prompt-Tuning (Lester et al., 2021)

- Competitive performance and better space efficiency





Instruction Tuning (Wei et al., 2022)

- Idea: improve model's capability of understanding the task description

LM for sentence completion

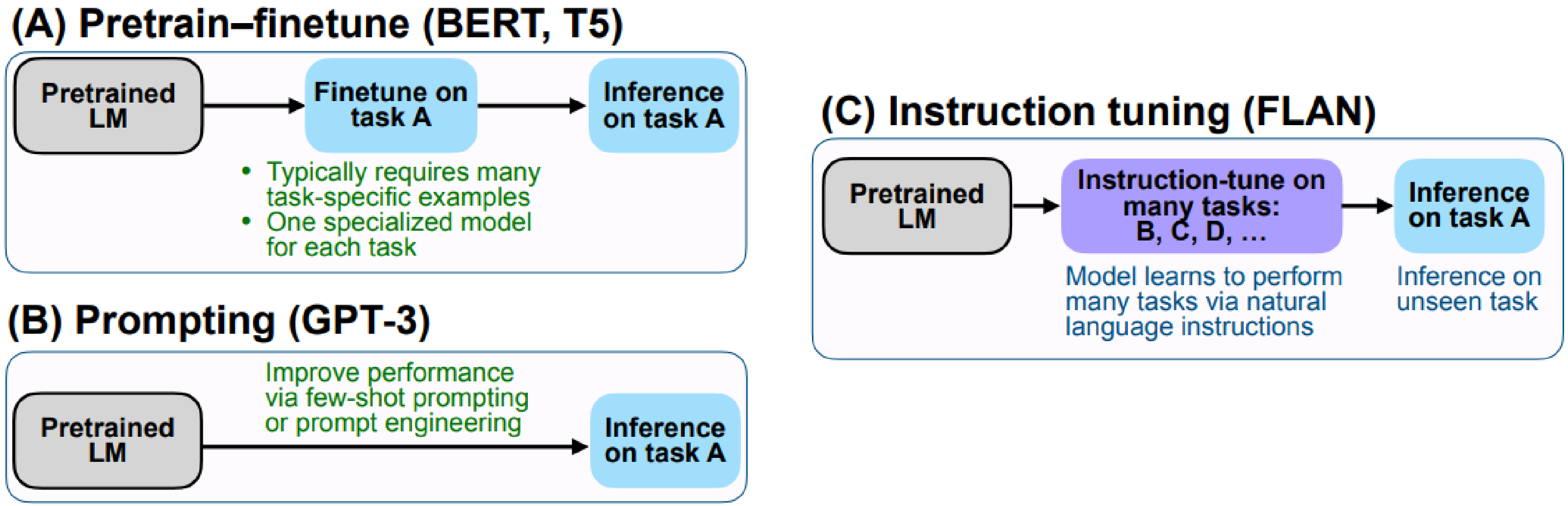
I went to Jolin's concert last night. I really loved her songs and dancing. It was ________

Detailed task instruction for LM generation

Decide the sentiment of the following sentences:
I went to Jolin's concert last night. I really loved her songs and dancing.
OPTIONS: - positive – negative - neutral

FLAN: Finetuned LANguage Models (Wei et al., 2022)

- Idea: fine-tune LM to better understand task descriptions via <u>other</u> tasks



Prompt v.s. Instruction Tuning (Wei et al., 2022)

- Prompt
- Instruction tuning

Training

Input (Commonsense Reasoning)

Here is a goal: Get a cool sleep on summer days.
How would you accomplish this goal?
OPTIONS:
-Keep stack of pillow cases in fridge.
-Keep stack of pillow cases in oven.

Target

keep stack of pillow cases in fridge

LM Fine-tuning

Inference

Prompt:

Input (Translation)

Translate this sentence to Spanish: The new office building was built in less than three months.

Target

El nuevo edificio de oficinas se construyó en tres meses.

Instruction tuning:

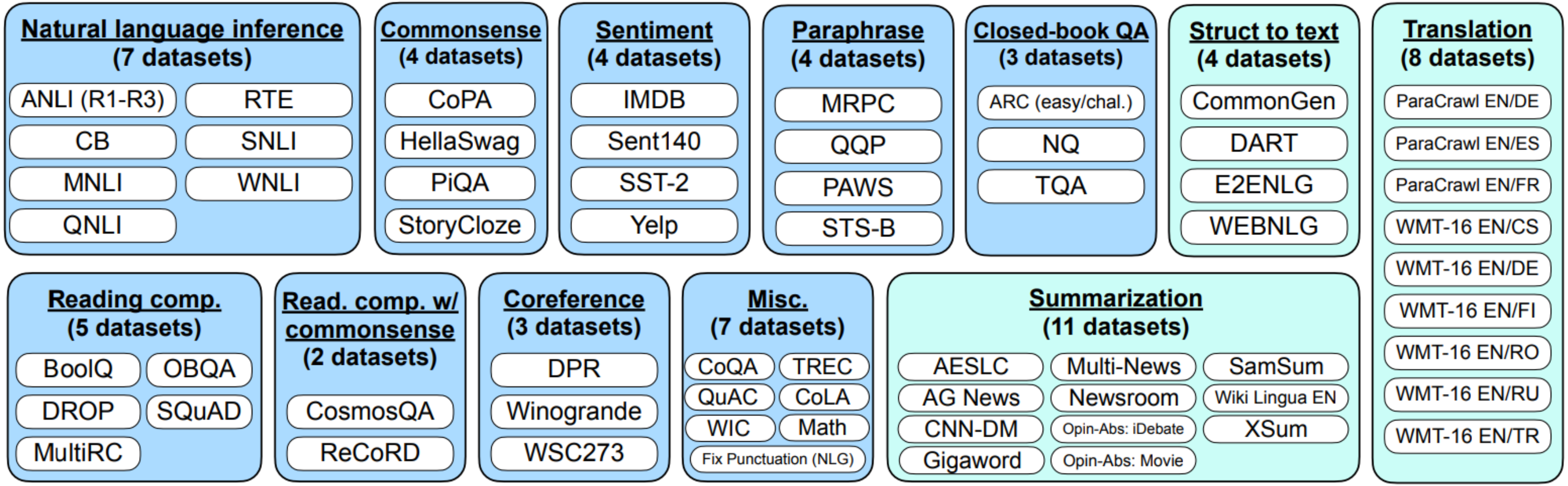
Input (Translation)

Translate this sentence to Spanish: The new office building was built in less than three months.

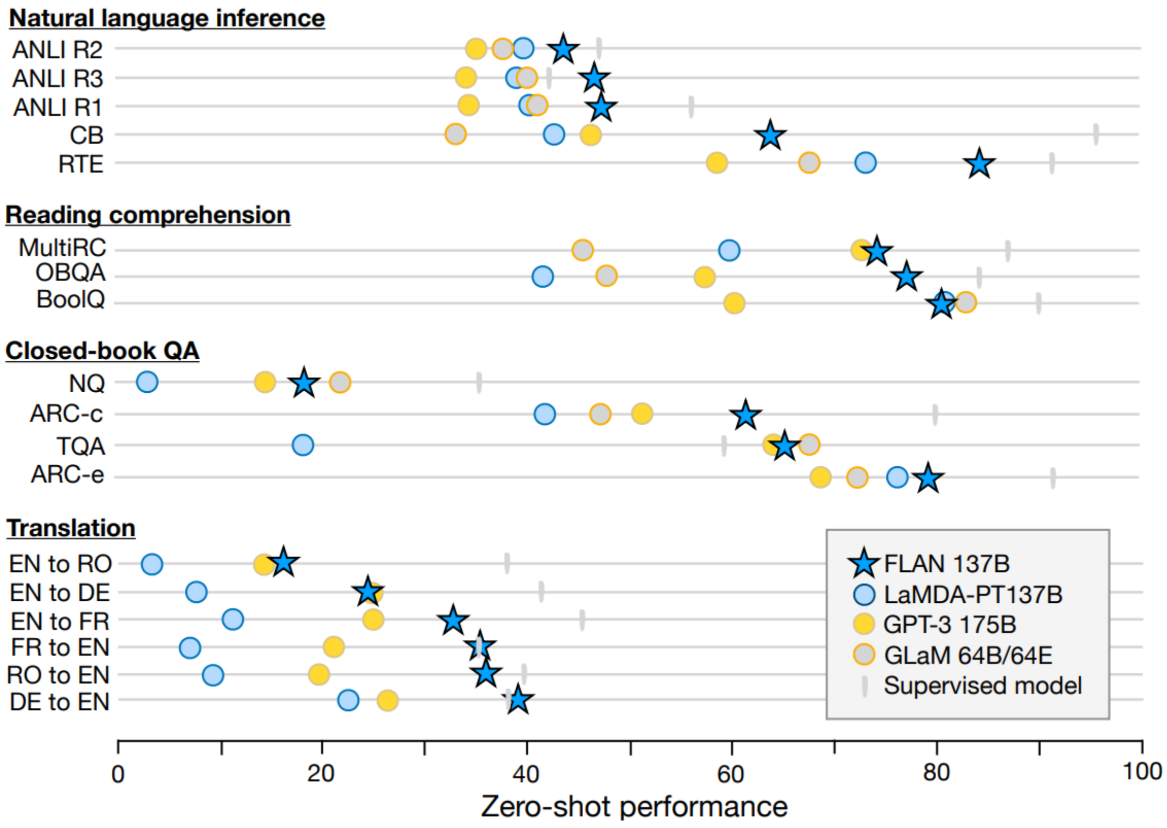
Target

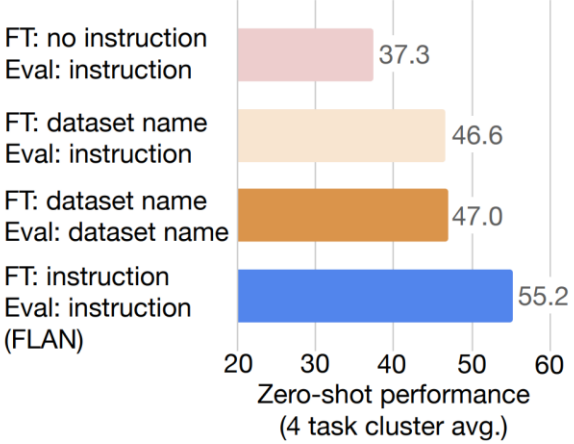
El nuevo edificio de oficinas se construyó en tres meses.

Task Clusters (Wei et al., 2022)



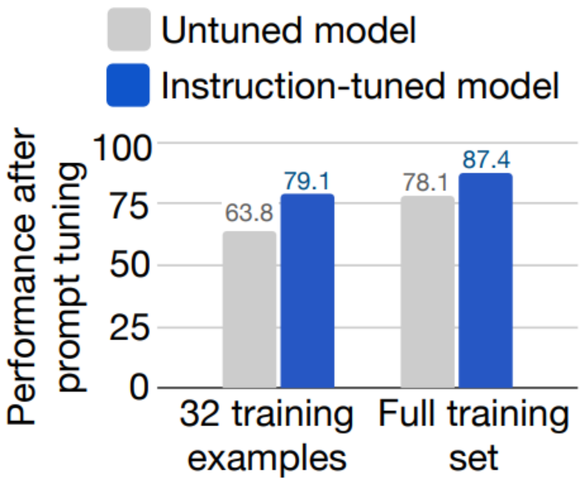
Zero-Shot Performance of FLAN

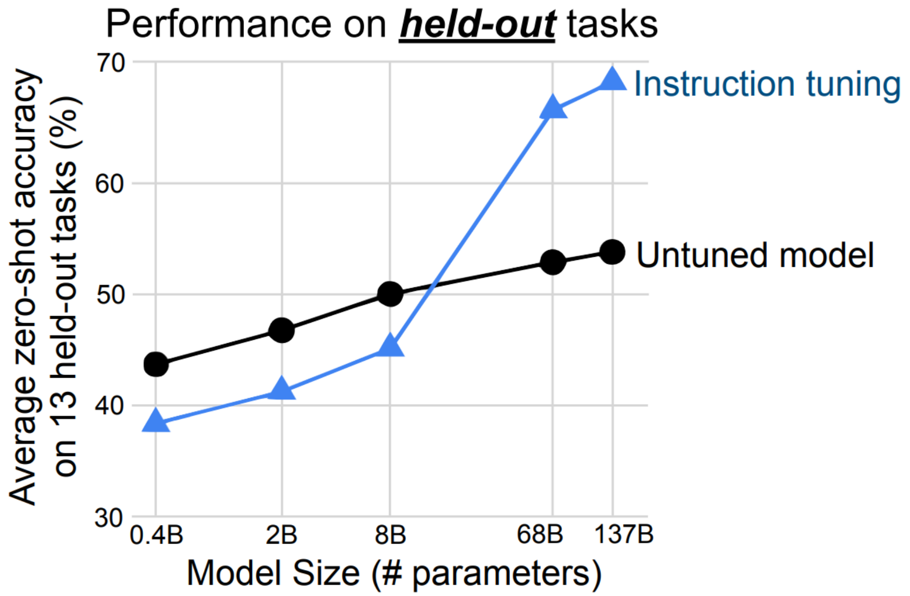




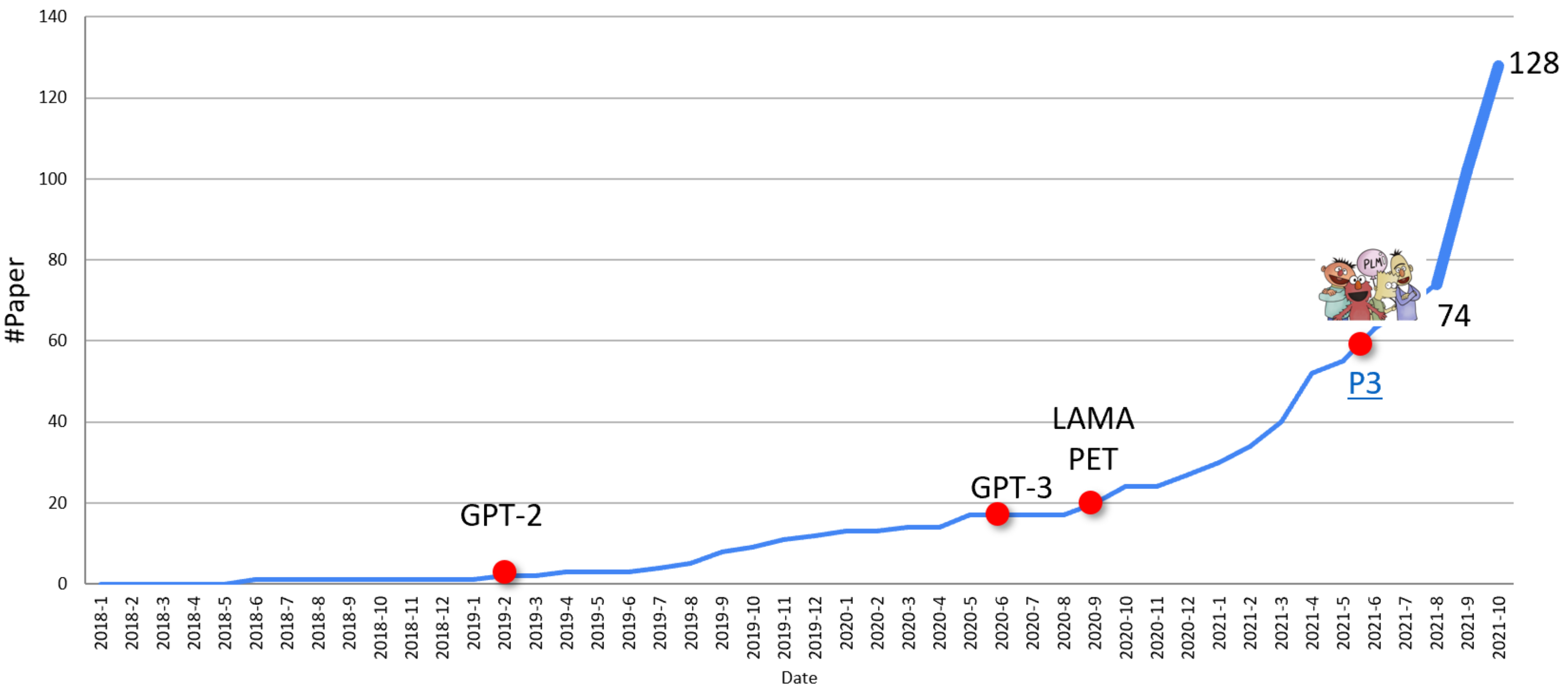
Zero-Shot Performance of FLAN

- Combine with prompt-tuning
- Model size requirement



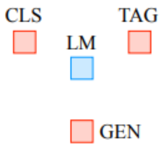
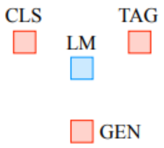
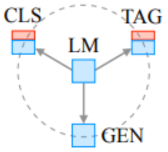
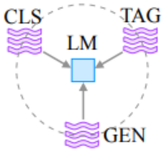


Trend of Prompt-Based Research



http://pretrain.nlpedia.ai/

Prompting Paradigm (Liu et al., 2021)

Paradigm	Engineering	Task Relation
a. Fully Supervised Learning (Non-Neural Network)	Features (e.g. word identity, part-of-speech, sentence length)	
b. Fully Supervised Learning (Neural Network)	Architecture (e.g. convolutional, recurrent, self-attentional)	
c. Pre-train, Fine-tune	Objective (e.g. masked language modeling, next sentence prediction)	
d. Pre-train, Prompt, Predict	Prompt (e.g. cloze, prefix)	

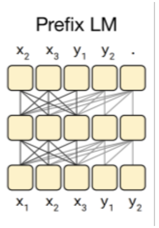
: unsupervised

: supervised

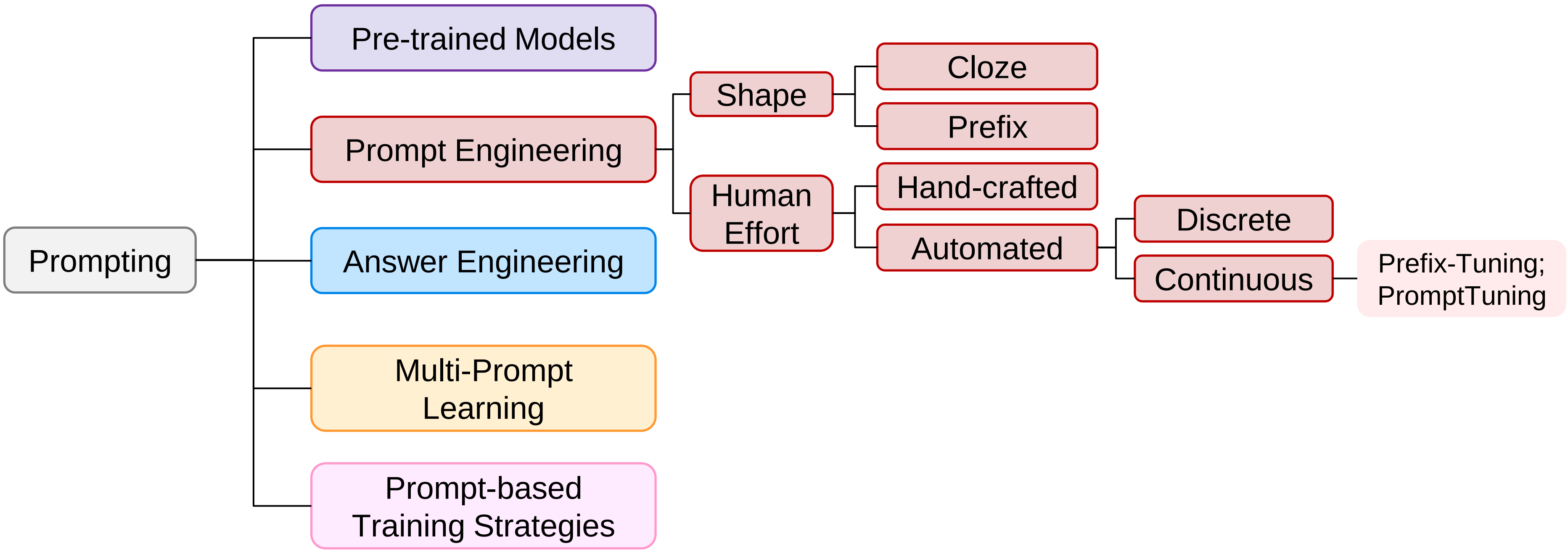
: textual prompt

Prompting Typology (Liu et al., 2021)

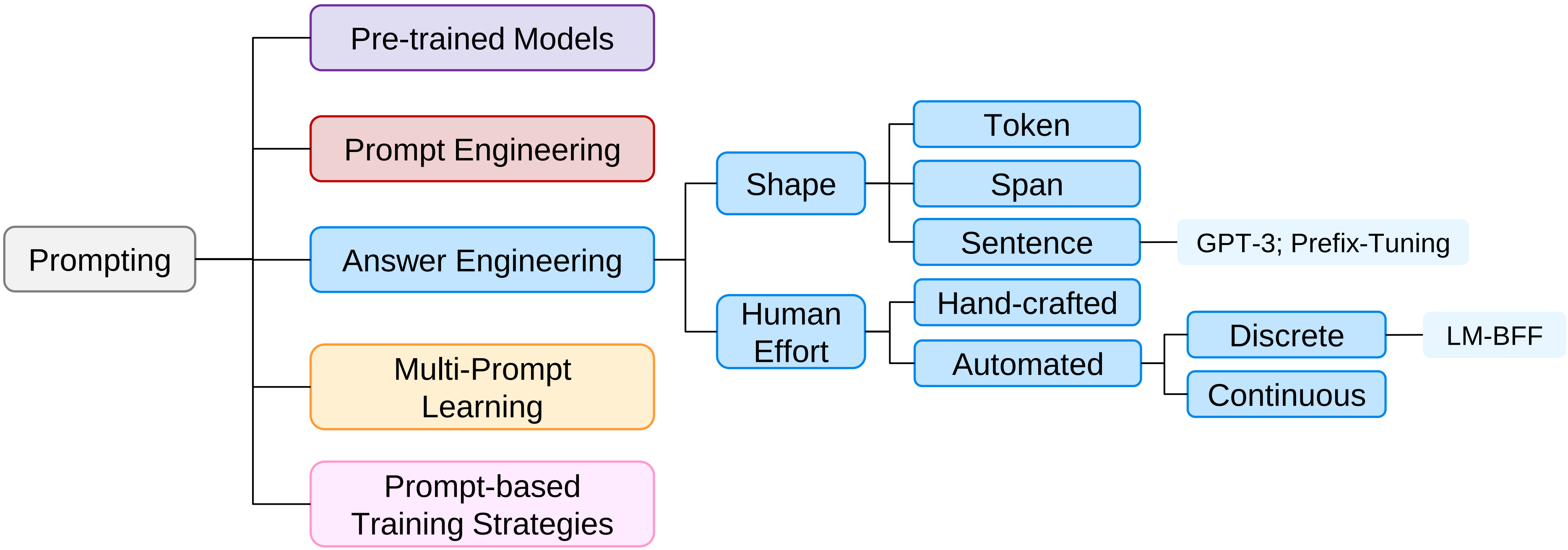
- Prompting
  - Pre-trained Models
    - Left-to-Right LM — GPT; GPT-2; GPT-3
    - Masked LM — BERT; RoBERTa
    - Prefix LM — UniLM
    - Encoder-Decoder — T5; BART
  - Prompt Engineering
  - Answer Engineering
  - Multi-Prompt Learning
  - Prompt-based Training Strategies



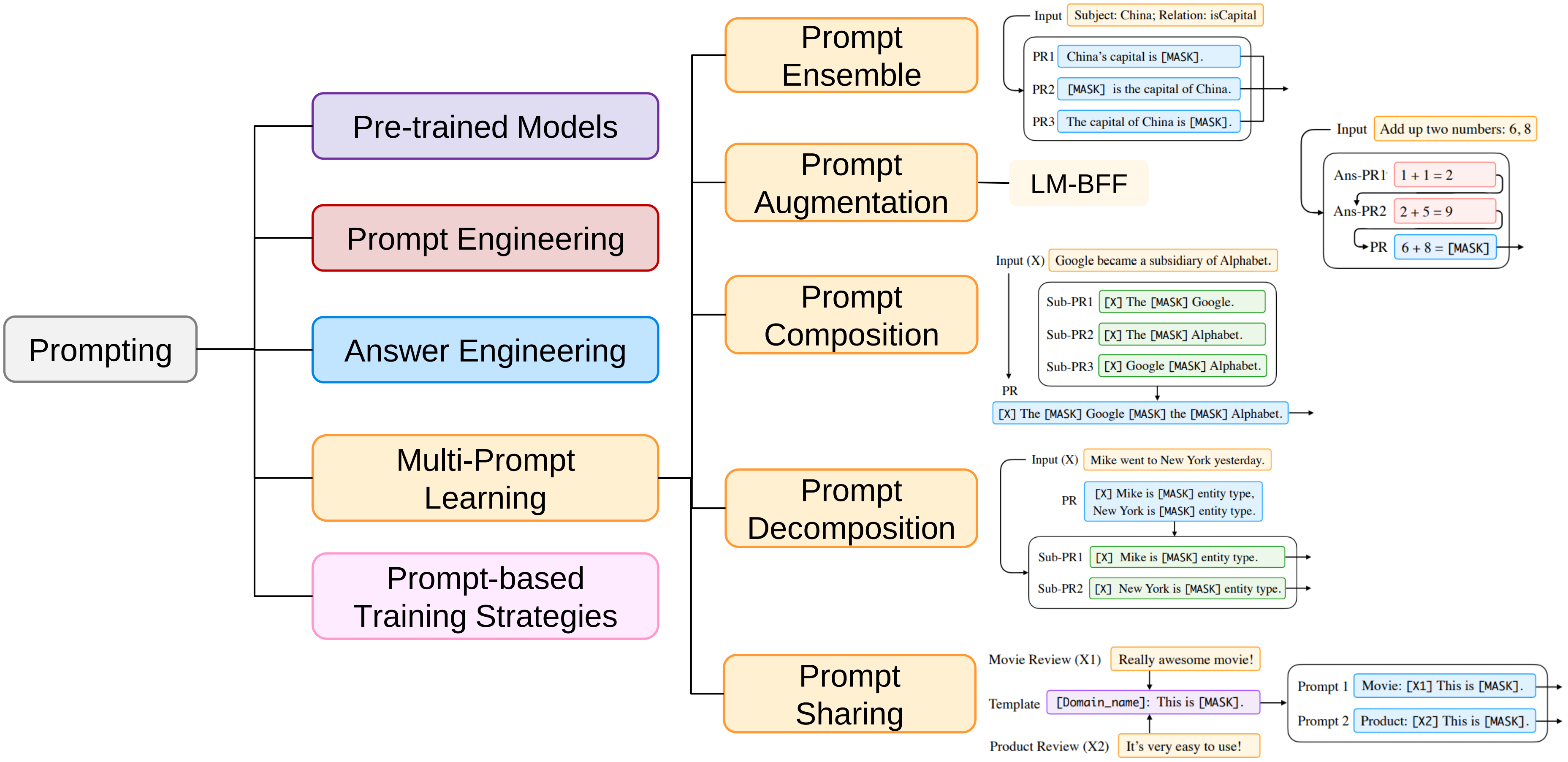
Prompting Typology (Liu et al., 2021)



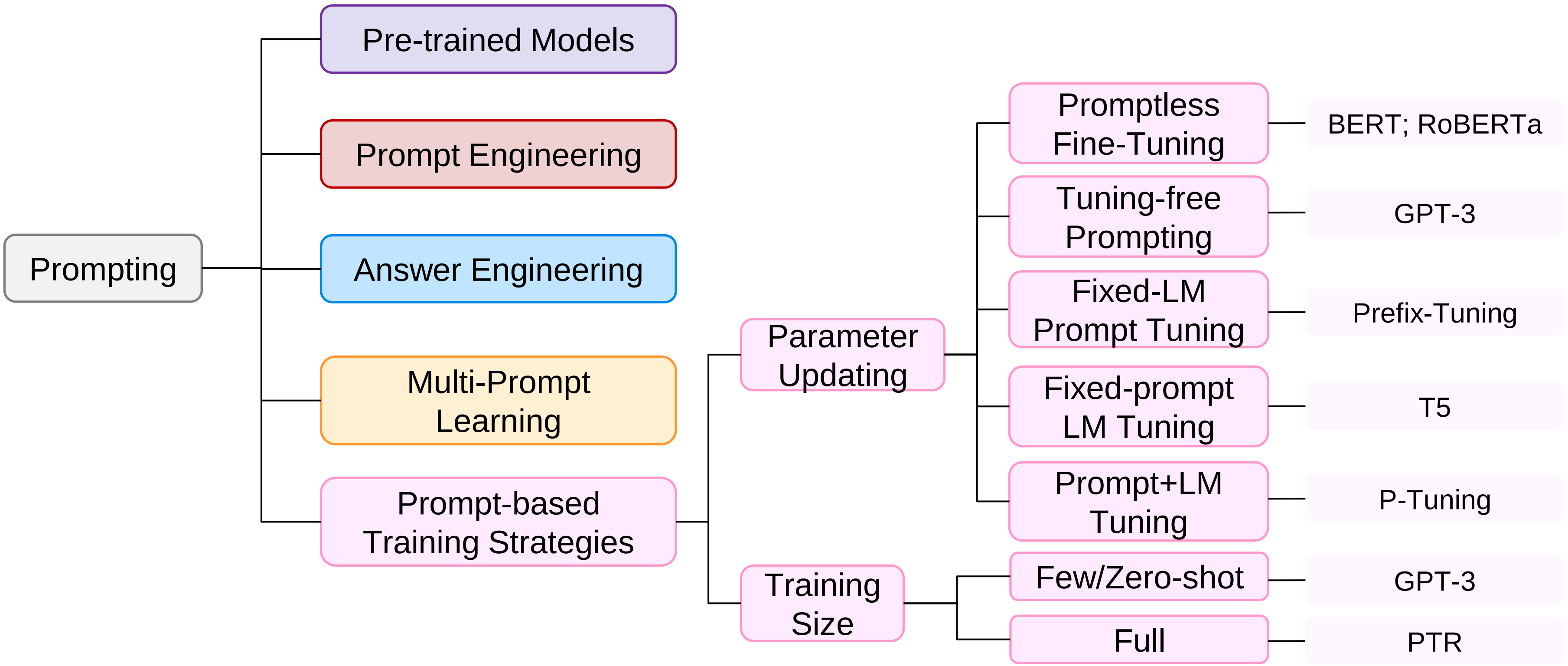
Prompting Typology (Liu et al., 2021)



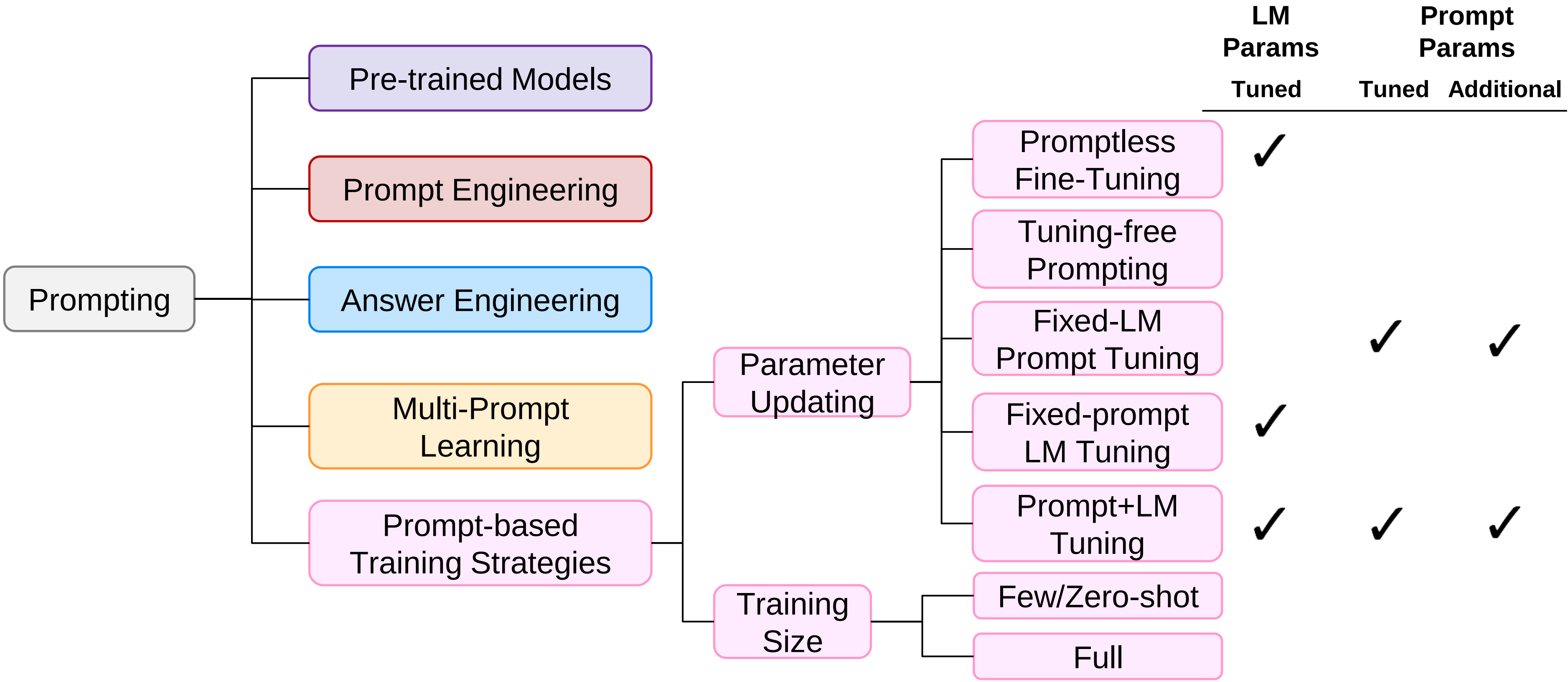
Prompting Typology (Liu et al., 2021)



Prompting Typology (Liu et al., 2021)



Prompting Typology (Liu et al., 2021)



Concluding Remarks

- **Prompt-Tuning**: manually designed natural language prompts
  - Human-understandable prompts
  - Sensitive to choices of prompts
  - Also work for one-shot/zero-shot settings
- **LM-BFF**: prompt-tuning + demonstration + template generation
  - Better performance
- **P-Tuning**: tuning the input (prompt) embeddings
  - Better performance via soft prompts
- **Prefix-Tuning**: only optimize the prefix embeddings (all layers)
  - Better training time/space efficiency
- **(Soft) Prompt-Tuning**: store task prompt and mixed-task learning
  - Updating less parameters
  - Better robustness
- **Instruction Tuning**: tuning LMs for understanding task instructions
  - Better zero-shot performance