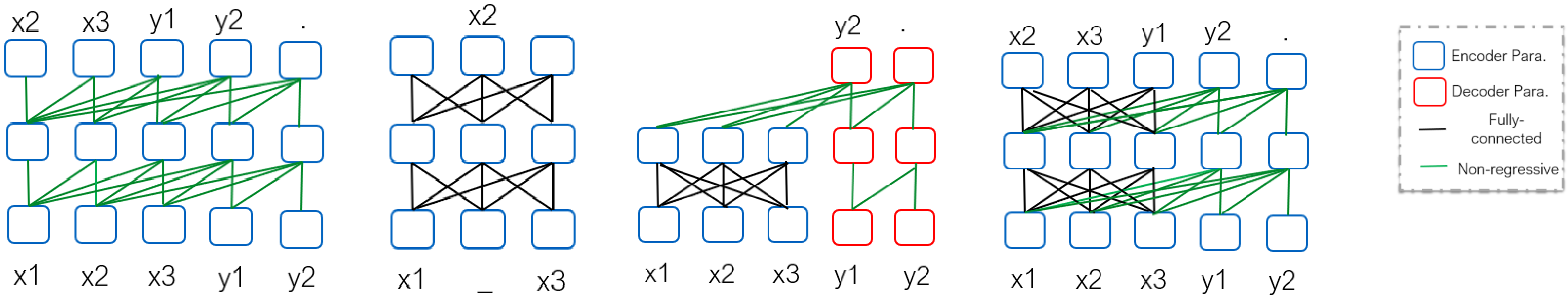
上一节课内容目标

- 预训练模型
  - 掌握常见四大预训练结构
  - 理解LLaMa网络架构
  - 了解ChatGPT形成历史
- 提示学习
  - 掌握提示学习的概念和意义
  - 掌握提示学习的基本方法
  - 理解提示学习中的设计考虑因素
  - 了解最新提示学习的内容

复习： “Big Four” Pretraining Framework



Left-to-right	Masked LM	Encode-decoder	Prefixed LM
unidirectional	**no decoder**	**more params**	**limited capacity**
GPT1/2/3	BERT	MASS/T5/BART	UNiLM/T5

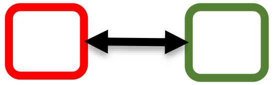
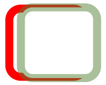
复习：提示学习

- Prompt is the technique of making better use of the knowledge from the pre-trained model by adding additional texts to the input.

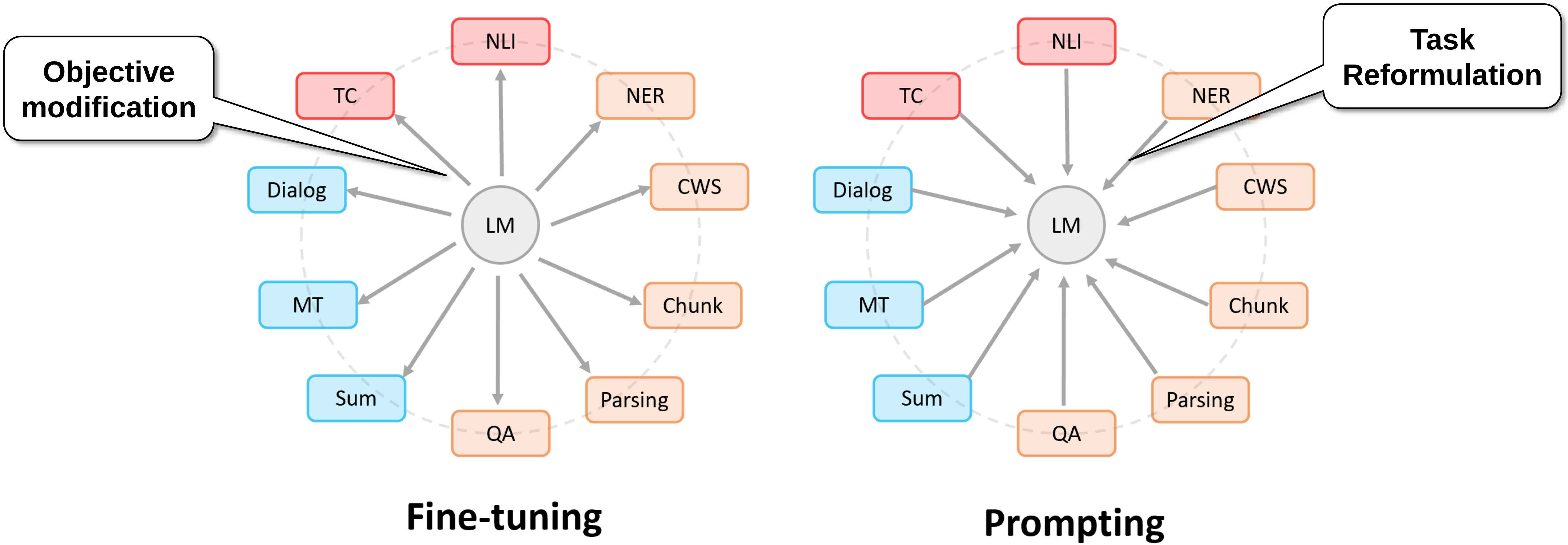
purpose

Method

复习：PLMs and Downstream Task Models

Stages	Downstream Task Models	Pre-trained LMs	Reasons
Traditional machine learning			No pre-training language model
Neural network methods enhanced by word2vec			The pre-trained language model plays the role of initializing the input text signal
The fine-tune method represented by BERT			The pre-trained language model is **responsible for extracting** high-level features from the input text
The prompt approach represented by GPT3			Pre-training language models **take on more responsibilities**: feature extraction, result prediction

复习：任务的“大一统”



Objective modification
NLI
TC
NER
Dialog
CWS
LM
MT
Chunk
Sum
Parsing
QA
Fine-tuning
Task Reformulation
NLI
TC
NER
Dialog
CWS
LM
MT
Chunk
Sum
Parsing
QA
Prompting

复习：提示学习

Input: x = I love this movie.

Predicting: ☺

Traditional Method

Input: x = I love this movie.

Template: [x] Overall, it was a [z] movie.

Answer: {fantastic:☺, boring:☹}

Prompting: x’ = I love this movie. Overall, it was a [z] movie.

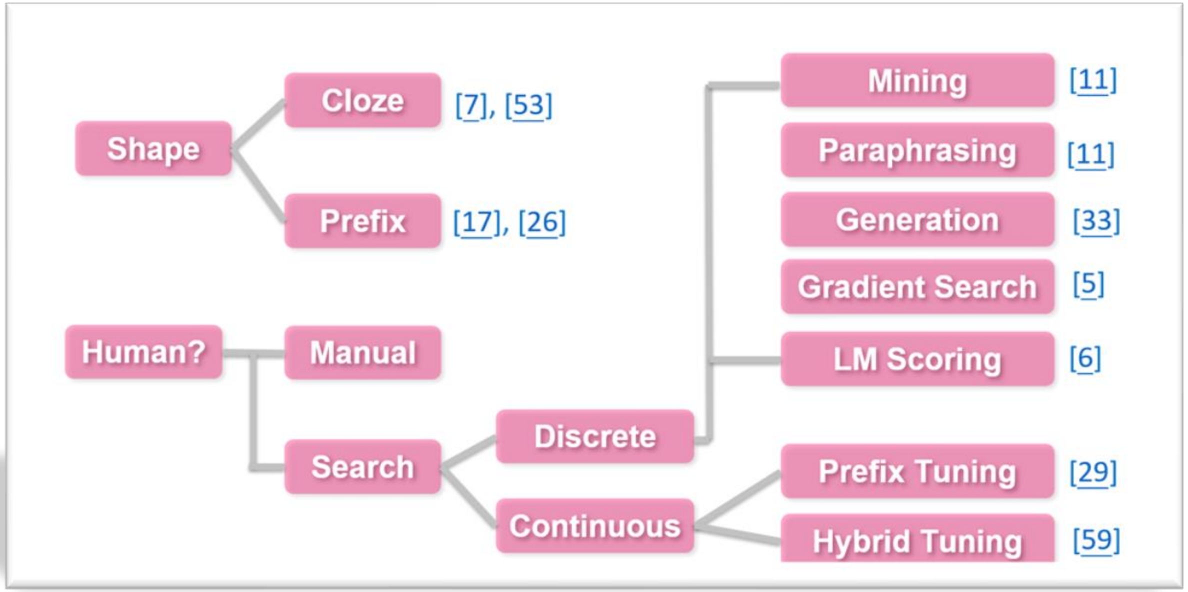
Predicting: x’ = I love this movie. Overall, it was a fantastic movie.

Prompting Method

Mapping: fantastic =>☺

复习：Design Considerations for Prompt-based Methods

- Prompt Template Engineering
- Answer Engineering
- Pre-trained Model Choice
- Expanding the Paradigm
- Prompt-based Training Strategies

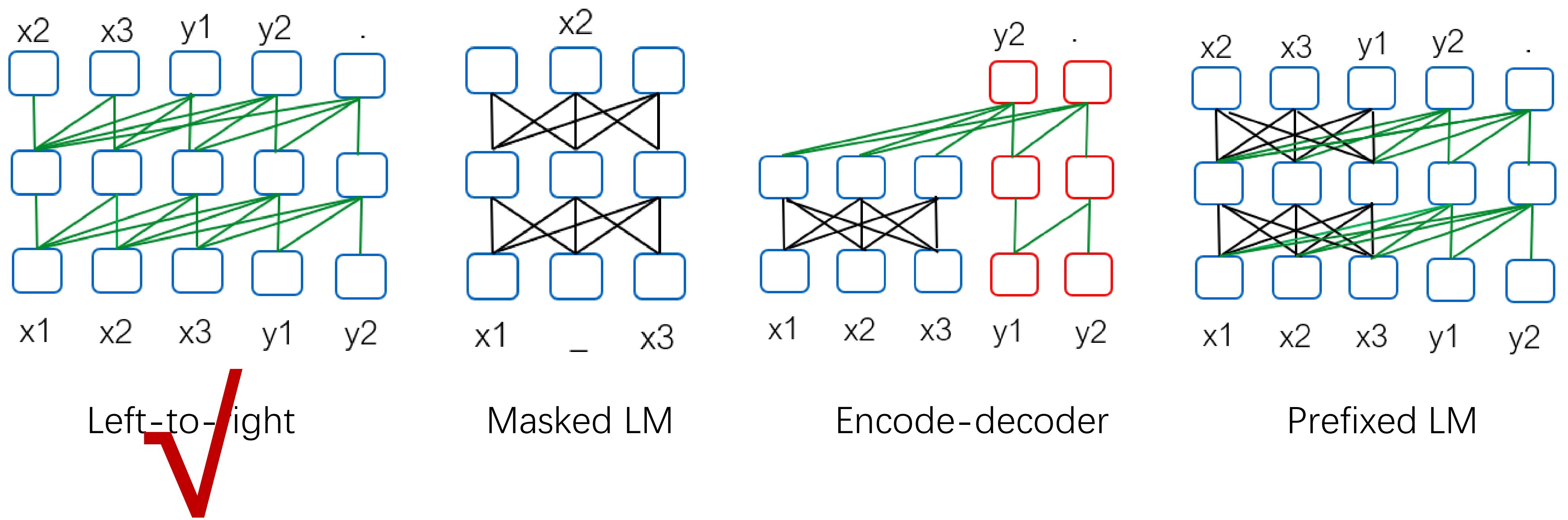


Revisit “Prompt Engineering” in the era of ChatGPT

Changes brought by ChatGPT

- Left-to-right models dominate the world

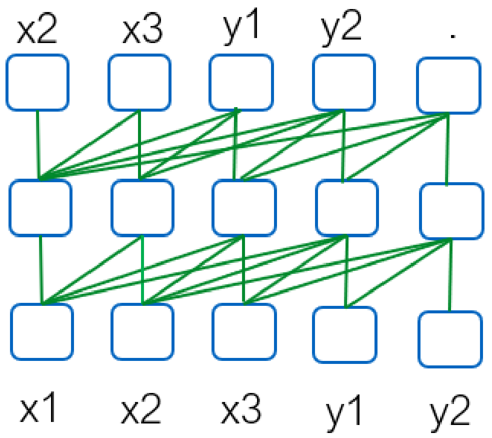
Cloze prompts fade into history



Changes brought by ChatGPT

- Left-to-right models dominate the world

Cloze prompts fade into history



Left-to-right

Changes brought by ChatGPT

- Left-to-right models dominate the world
- Solving traditional NLP tasks are not the most important things

Cloze prompts fade into history

Prompt distribution matters a lot

Grammar correction
Convert ungrammatical statements into standard English.

Parse unstructured data
Create tables from unstructured text.

Calculate time complexity
Find the time complexity of a function.

Keywords
Extract keywords from a block of text.

Python bug fixer
Find and fix bugs in source code.

Tweet classifier
Detect sentiment in a tweet.

Mood to color
Turn a text description into a color.

Marv the sarcastic chat bot
Marv is a factual chatbot that is also sarcastic.

Interview questions
Create interview questions.

Improve code efficiency
Provide ideas for efficiency improvements to Python code.

Rap battle writer
Generate a rap battle between two characters.

Summarize for a 2nd grader
Simplify text to a level appropriate for a second-grade student.

Emoji Translation
Translate regular text into emoji text.

Explain code
Explain a complicated piece of code.

Product name generator
Generate product names from a description and seed words.

Spreadsheet creator
Create spreadsheets of various kinds of data.

Airport code extractor
Extract airport codes from text.

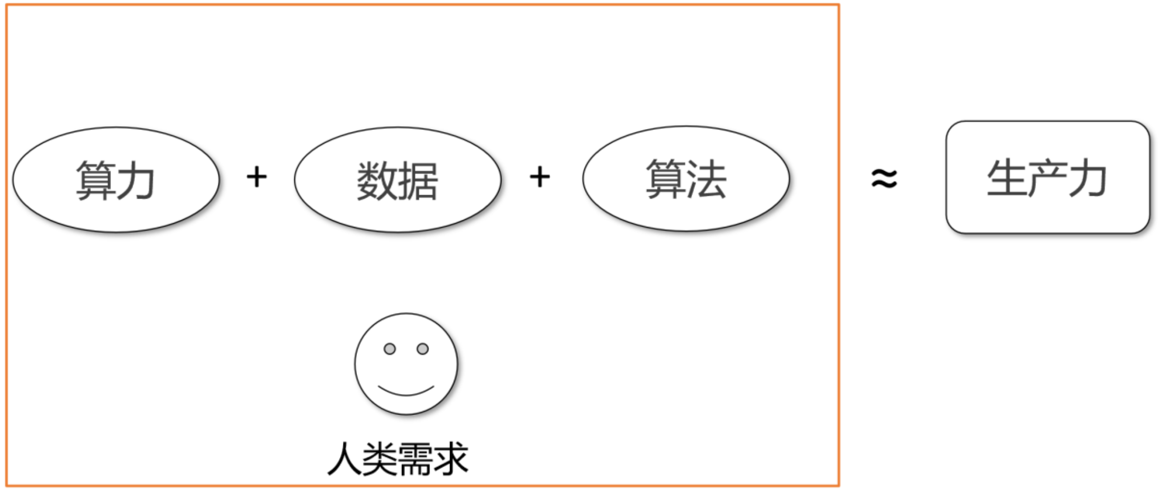
VR fitness idea generator
Generate ideas for fitness promoting virtual reality games.

Turn by turn directions
Convert natural language to turn-by-turn directions.

Function from specification
Create a Python function from a specification.

Single page website creator
Create a single page website.

Memo writer
Generate a company memo based on provided points.



Changes brought by ChatGPT

- Left-to-right models dominate the world
- Solving traditional NLP tasks are not the most important things
- API-based research become more popular

Cloze prompts fade into history

Prompt distribution matters a lot

Zero-shot & few-shot prompting

Changes brought by ChatGPT

- Left-to-right models dominate the world
- Solving traditional NLP tasks are not the most important things
- API-based research become more popular
- Supervised fine-tuning become popular

Cloze prompts fade into history

Prompt distribution matters a lot

Zero-shot & few-shot prompting

Prompt scaling law

Changes brought by ChatGPT

- Left-to-right models dominate the world
- Solving traditional NLP tasks are not the most important things
- API-based research become more popular
- Supervised fine-tuning become popular
- Evaluation is difficult

Cloze prompts fade into history

Prompt distribution matters a lot

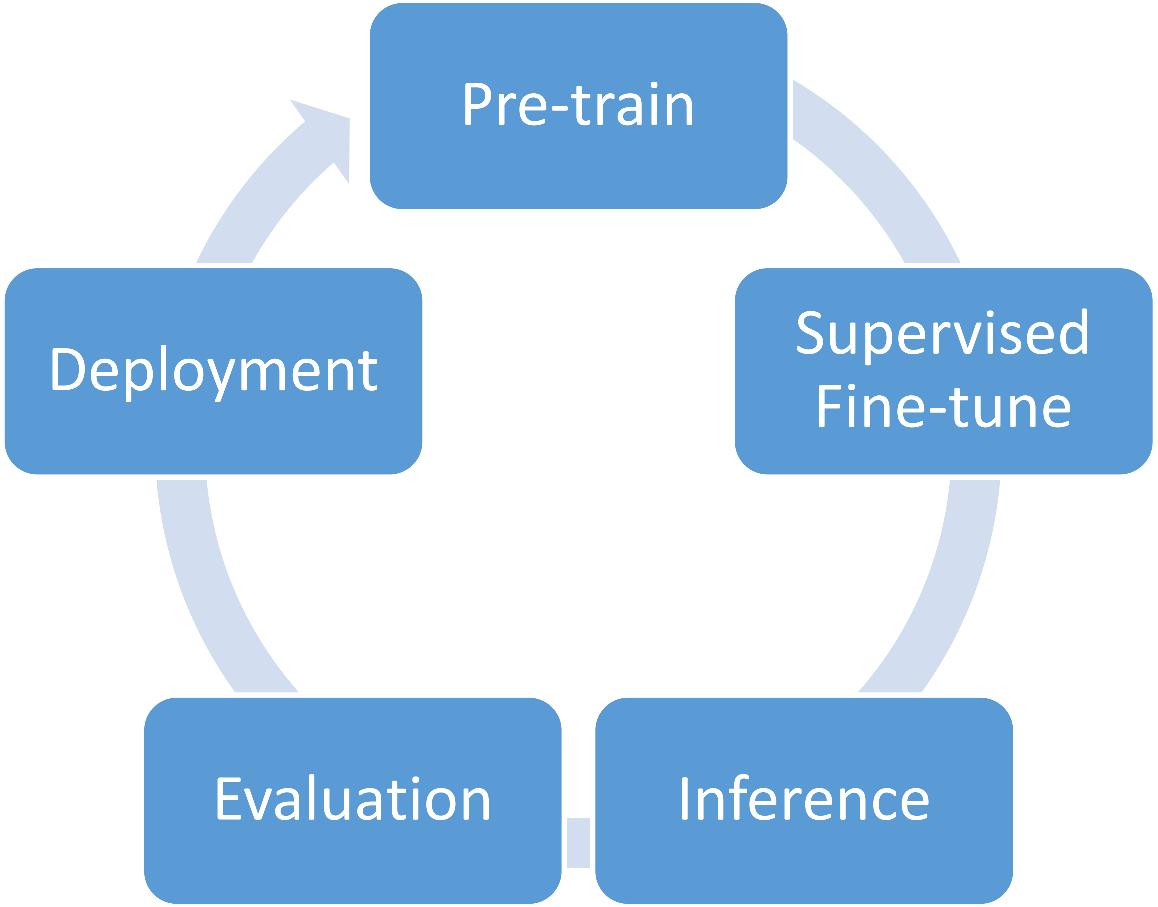
Zero-shot & few-shot prompting

Prompt scaling law

Prompt-based evaluation

Prompt Engineering 2.0: Design Considerations

Prompt Engineering in LLMOps



Prompt Engineering: Supervised Fine-tuning

- Prompt Diversity
  - How does prompt diversity affect model's performance?
- Prompt number
  - How does the number of prompts affect model's performance?
- Response Quality
  - How does the quality of response affect model's performance?

Prompt Engineering: Supervised Fine-tuning

Table 3: English Instruction Data (Continued from Table 2)

Dataset	**# Tasks**	**# Instructions**	**Lan**	**Collection Method**	**Usage**	**Access**	**Human Verified?**
OIG (AI, 2021)	30	43M	English	Mixed	Instruct. Tuning	Open	No
Baize (Xu et al., 2023)	3	100K+	English	Model Generated	Chat	Open	No
Camel (Guohao et al., 2023)	-	115K	English	Model Generated	Instruct. Tuning, Chat	Open	No
UltraChat (Ding et al., 2023)	-	675K	English	Model Generated	Chat	Open	No
Dolly (Databricks, 2022)	7	15,000	English	Human Annotated	Instruct. Tuning	Open	Yes
Guanaco-Dataset (JosephusCheung, 2021)	175	534,530	Multilingual	Mixed	Instruct. Tuning	Open	No
ChatLLaMA Chinese-ChatLLaMA (YDli-ai, 2021)	-	-	Multilingual	Mixed	Instruct. Tuning	Open	No
GPT-4-LLM (Peng et al., 2023)	175	165K	Multilingual	Model Generated	RLHF, Instruct. Tuning	Open	No
ShareGPT (ShareGPT, 2021)	-	-	Multilingual	Model Generated	Instruct. Tuning, Chat	Closed	Yes
SHP (Ethayarajh et al., 2023)	18	385K	English	Existing, Human Annotated	RLHF, Instruct. Tuning	Open	Yes
HH-RLHF (Bai et al., 2022; Anthropic, 2022; Ganguli et al., 2022)	-	169,550	English	Mixed	RLHF, Instruct. Tuning	Open	Yes
HC3 (Guo et al., 2023)	12	37,175	Multilingual	Mixed	Instruct. Tuning	Open	Yes

A Survey of Recently Released "Instructions" (Zhang et al)

Prompt Engineering: Supervised Fine-tuning

	MMLU (factuality)	GSM (reasoning)	BBH (reasoning)	TydiQA (multilinguality)	Codex-Eval (coding)	AlpacaFarm (open-ended)	Average
	EM (0-shot)	EM (8-shot, CoT)	EM (3-shot, CoT)	F1 (1-shot, GP)	P@10 (0-shot)	Win % vs Davinci-003	
Vanilla LLaMa 13B	42.5	14.0	36.9	47.4	26.6	-	-
+SuperNI	49.8	4.0	2.8	51.4	13.1	5.0	21.0
+CoT	44.5	**39.5**	39.0	**52.2**	23.3	4.7	33.9
+Flan V2	**50.7**	21.0	39.2	47.5	16.2	5.3	30.0
+Dolly	45.3	17.0	26.0	46.8	31.4	18.3	30.8
+Open Assistant 1	43.1	16.0	38.5	38.3	31.8	55.2	37.1
+Self-instruct	30.3	9.0	29.6	40.4	13.4	7.3	21.7
+Unnatural Instructions	46.2	7.5	32.8	39.3	24.8	10.8	26.9
+Alpaca	45.1	8.0	34.5	32.8	27.6	33.2	30.2
+Code-Alpaca	42.6	12.0	36.6	41.3	34.5	21.3	31.4
+GPT4-Alpaca	47.0	14.0	38.3	24.4	32.5	63.6	36.6
+Baize	43.5	8.5	36.7	33.9	27.3	33.9	30.6
+ShareGPT	49.2	16.0	40.1	30.1	31.6	**69.1**	39.3
+ Human data mix	50.4	36.5	39.4	49.8	23.7	38.5	39.7
+Human+GPT data mix.	49.2	36.5	**42.8**	46.1	**35.0**	57.2	**44.5**

Which “instruction” data is the best? (Wang et al)

Prompt Engineering: Supervised Fine-tuning

Source	#Examples	Avg Input Len.	Avg Output Len.
Training			
Stack Exchange (STEM)	200	117	523
Stack Exchange (Other)	200	119	530
wikiHow	200	12	1,811
Pushshift r/WritingPrompts	150	34	274
Natural Instructions	50	236	92
Paper Authors (Group A)	200	40	334
Dev			
Paper Authors (Group A)	50	36	N/A
Test			
Pushshift r/AskReddit	70	30	N/A
Paper Authors (Group B)	230	31	N/A

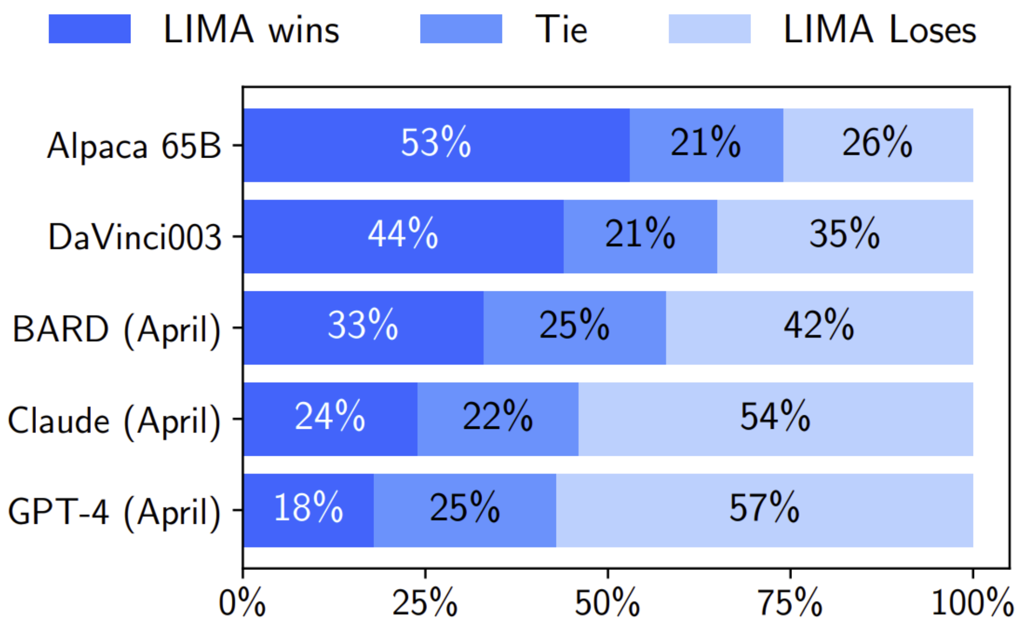


Figure 1: Human preference evaluation, comparing LIMA to 5 different baselines across 300 test prompts.

LIMA: Less Is More for Alignment (Zhou et al)

Prompt Engineering: Inference

- Zero-shot Prompting:
  - How to ask a good question that ChatGPT can better understand you?

Prompt Engineering: Inference

你是一个中文人工智能助手，你需要仿照示例，根据给定的除示例外的所有法律生成一个包含题目、选项分析和答案的单项选择题。在生成单项选择题时，你必须遵守以下几个原则：

题目构成 1. 题目由题目描述和4个选项构成
题目描述 2. 单项选择题的题目描述需要合理
题目生成的整体限制 3. 尽可能根据除示例外的所有法律生成题目，避免使用单条法律生成题目

题目选项
4. 在生成4个选项时，结合题目描述与除示例外的所有法律，首先设计1个正确答案的选项，然后再设计3个错误的选项，接着这4个选项以随机的顺序排列
5. 选项互有差异，避免选项之间的明显重复或相似性
6. 在设计选项时，不要使得某些选项明显不可能是正确答案
7. 每个选项需要和题目描述相关
8. 每个选项需要前后内容一致
9. 不能直接从给定的法律中复制文本作为选项内容，需要结合给定的法律生成合理的选项

生成顺序 10. 依次生成题目、选项分析和答案
选项分析 11. 选项分析是结合题目与除示例外的所有法律，对每个选项进行分析
答案 12. 选项分析中的正确答案是最终答案

以下是1个示例：
示例：
{example}

让我们一步一步思考，参考示例并结合给定法律"{input_law}"{action}，依次生成下面内容：

题目：

选项分析：

答案：

法律：企业破产法：第四十六条　未到期的债权，在破产申请受理时视为到期。附利息的债权自破产申请受理时起停止计息。第四十七条　附条件、附期限的债权
题目：A公司因经营不善，资产已不足以清偿全部债务，经申请进入破产还债程序。关于破产债权的申报，下列哪个表述是正确的?
A.甲对A公司的债权虽未到期，不可以申报
B.乙对A公司的债权因附有条件，故不能申报
C.丙对A公司的债权虽然诉讼未决，但丙仍可以申报
D.职工丁对A公司的伤残补助请求权，应予以申报
选项分析：《企业破产法》第46条第一款规定，未到期的债权，在破产申请受理时视为到期。据此可知，未到期的债权，仍可申报。选项A错误。《企业破产法》
答案：C

中华人民共和国河道管理条例规定：第十条　河道的整治与建设，应当服从流域综合规划，符合国家规定的防洪标准、通航标准和其他有关技术要求，维护堤防安全，保持河势稳定和行洪、航运通畅。第十一条　修建开发水利……

设计一个法律情景/针对给定法律中的某个概念

Prompt Engineering: Changes brought by ChatGPT

- Zero-shot Prompting
- Few-shot Prompting
  - How do I get the model to mimic a given example?
    - Format following
    - Reasoning step decomposition

"X"- of thought

Chain-of-thought

Input

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

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

Q: The bakers at the Beverly Hills Bakery baked 200 loaves of bread on Monday morning. They sold 93 loaves in the morning and 39 loaves in the afternoon. A grocery store returned 6 unsold loaves. How many loaves of bread did they have left?

Model Output

A: The bakers started with 200 loaves. They sold 93 in the morning and 39 in the afternoon. So they sold 93 + 39 = 132 loaves. The grocery store returned 6 loaves. So they had 200 - 132 - 6 = 62 loaves left.
The answer is 62.

Program-of-thought

Input

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

A: Roger started with 5 tennis balls.
```
tennis_balls = 5
```
2 cans of 3 tennis balls each is
```
bought_balls = 2 * 3
```
tennis balls. The answer is
```
answer = tennis_balls + bought_balls
```

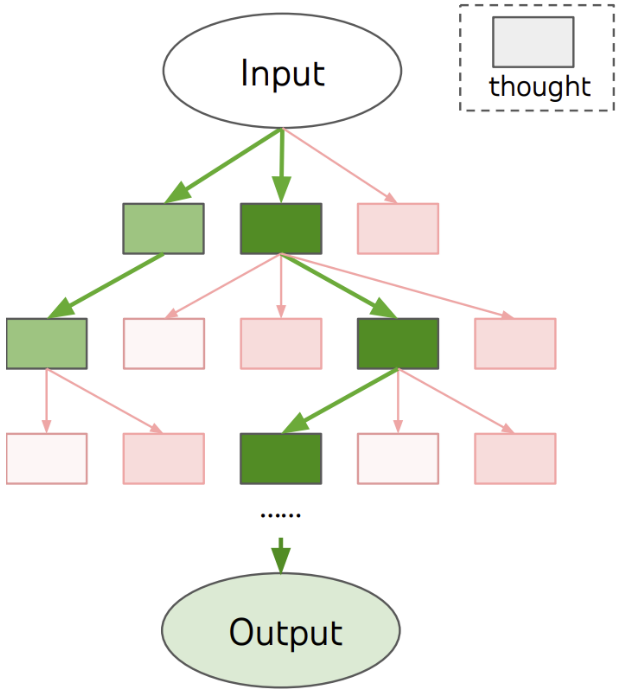
Q: The bakers at the Beverly Hills Bakery baked 200 loaves of bread on Monday morning. They sold 93 loaves in the morning and 39 loaves in the afternoon. A grocery store returned 6 unsold loaves. How many loaves of bread did they have left?

Model Output

A: The bakers started with 200 loaves
```
loaves_baked = 200
```
They sold 93 in the morning and 39 in the afternoon
```
loaves_sold_morning = 93
loaves_sold_afternoon = 39
```
The grocery store returned 6 loaves.
```
loaves_returned = 6
```
The answer is
```
answer = loaves_baked - loaves_sold_morning
  - loaves_sold_afternoon + loaves_returned
```

```
>>> print(answer)
74
```

Tree-of-thought



Prompt Engineering: Evaluation

- How to evaluate a model as you desire?

Prompt Engineering: Evaluation

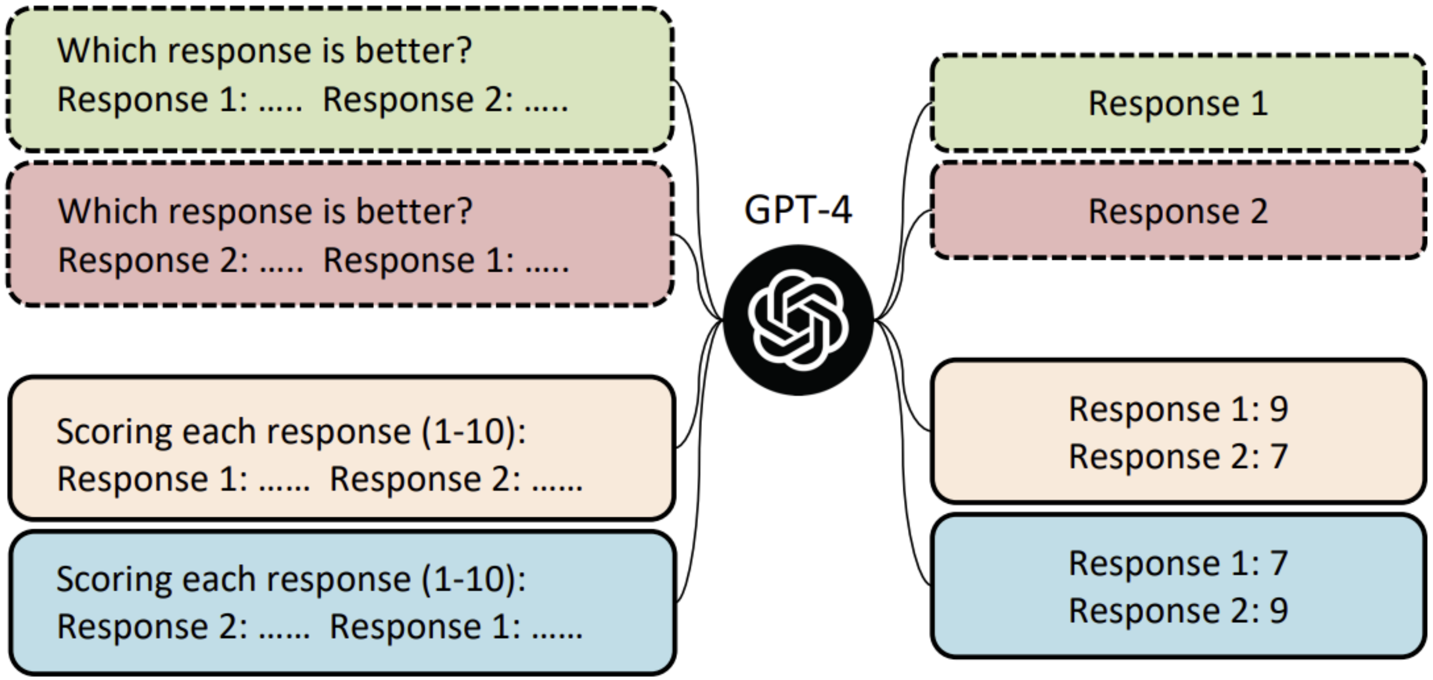
- Evaluation
  - How to evaluate a model as you desire?

ChatGPT Score

```
prompt: |-
  You are evaluating a response that has been submitted for a particular task, using a specific set of standards. Below is the data:
  [BEGIN DATA]
  ***
  [Task]: {input}
  ***
  [Submission]: {completion}
  ***
  [Criterion]: {criteria}
  ***
  [END DATA]
  Does the submission meet the criterion? First, write out in a step by step manner your reasoning about the criterion to be sure that your conclusion is correct. Avoid simply stating the correct answers at
  Reasoning:
eval_type: cot_likert
choice_scores:
  "1": 1.0
  "2": 2.0
  "3": 3.0
  "4": 4.0
  "5": 5.0
  "6": 6.0
criteria:
  helpfulness:
    "1": "Not helpful - The generated text is completely irrelevant, unclear, or incomplete. It does not provide any useful information to the user."
    "2": "Somewhat helpful - The generated text has some relevance to the user's question, but it may be unclear or incomplete. It provides only partial information, or the information provided may not be us
    "3": "Moderately helpful - The generated text is relevant to the user's question, and it provides a clear and complete answer. However, it may lack detail or explanation that would be helpful for the use
    "4": "Helpful - The generated text is quite relevant to the user's question, and it provides a clear, complete, and detailed answer. It offers additional information or explanations that are useful for t
    "5": "Very helpful - The generated text is highly relevant to the user's question, and it provides a clear, complete, and detailed answer. It offers additional information, explanations, or analogies tha
    "6": "Highly helpful -  The generated text provides a clear, complete, and detailed answer. It offers additional information or explanations that are not only useful but also insightful and valuable to t
```

Prompt Engineering: Evaluation

- How to evaluate a model as you desire?

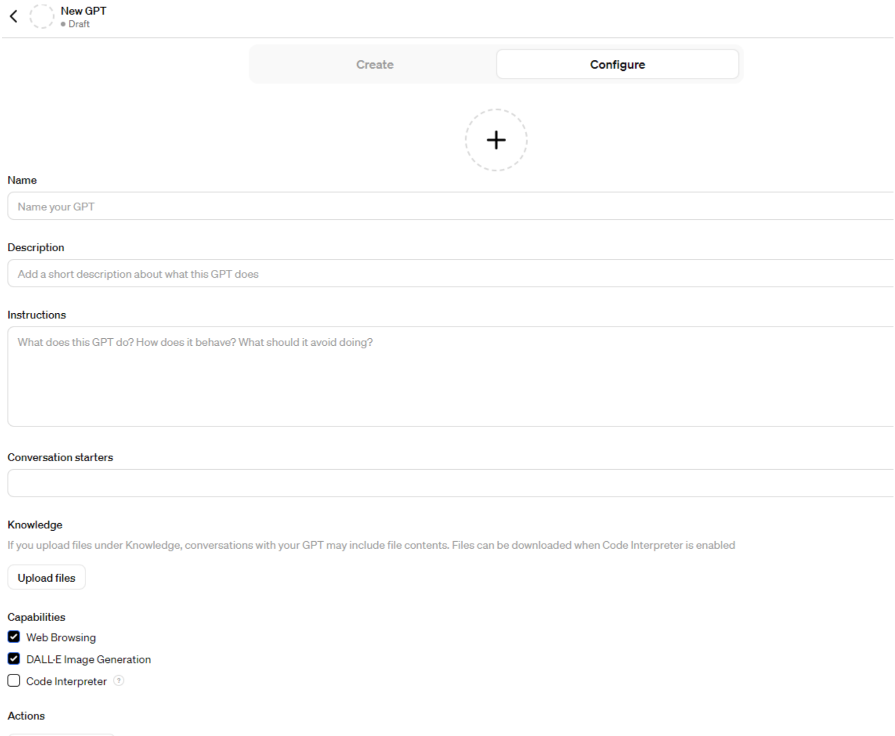


Prompt Engineering: Deployment

- How to design a good preface?
  - GPT Agent
  - System Message
- How to prevent jailbreak prompt?

```
import openai

openai.ChatCompletion.create(
  model="gpt-3.5-turbo",
  messages=[
        {"role": "system", "content": "You are a helpful assistant."},
        {"role": "user", "content": "Who won the world series in 2020?"},
        {"role": "assistant", "content": "The Los Angeles Dodgers won the Worl
        {"role": "user", "content": "Where was it played?"}
    ]
)
```

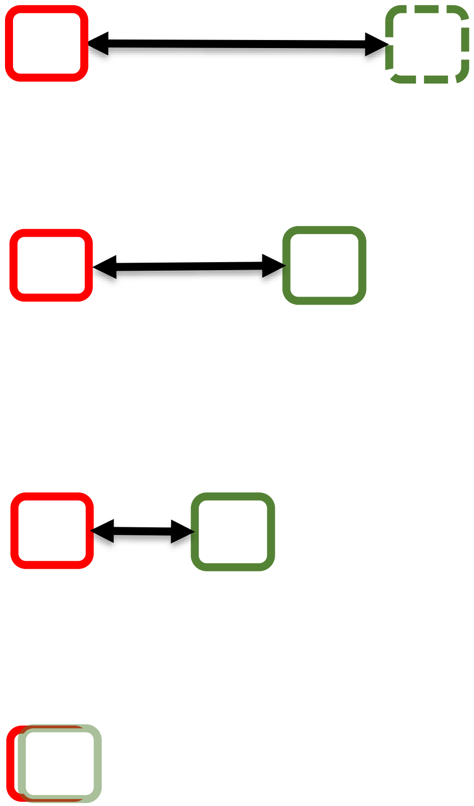


Prompt Engineering: Pre-train

- How to prompt pre-training data so that
  - the next word could be better predicted
  - the stored information can be better elicited

变化中寻找“不变”： **预训练模型与任务模型**越来越近

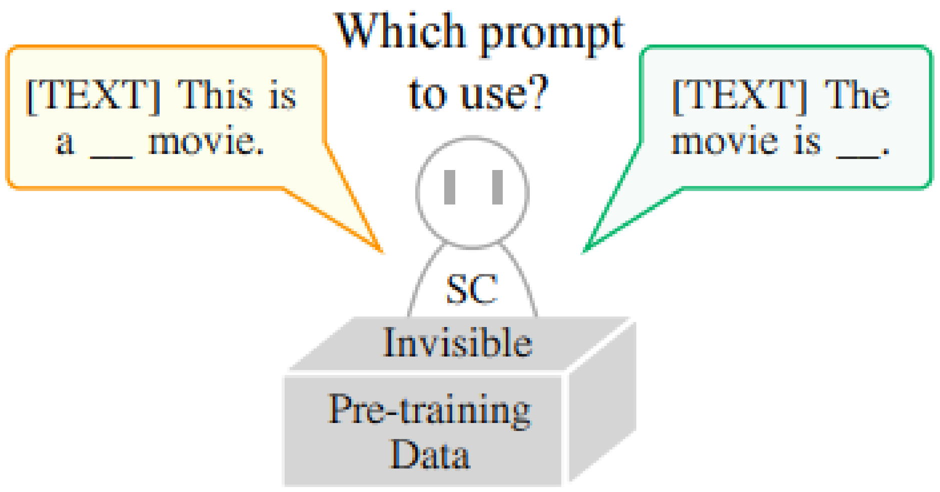
Downstream Task Models — Pre-trained LMs



Stages	Reasons
Traditional machine learning	No pre-training language model
Neural network methods enhanced by word2vec	The pre-trained language model plays the role of initializing the input text signal
The fine-tune method represented by BERT	The pre-trained language model is **responsible for extracting** high-level features from the input text
The prompt approach represented by GPT3	Pre-training language models **take on more responsibilities**: feature extraction, result prediction

变化中寻找“不变”：信息存储和访问的方式越来越近

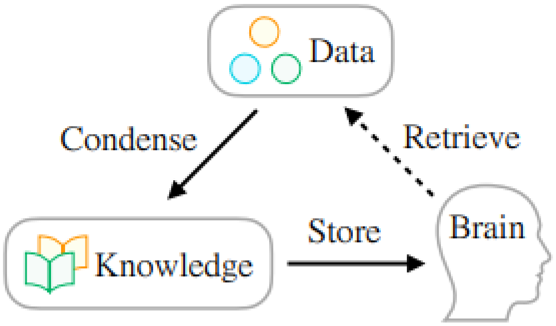
- The way how information is stored is opaque
- There is a gap between data storing and accessing



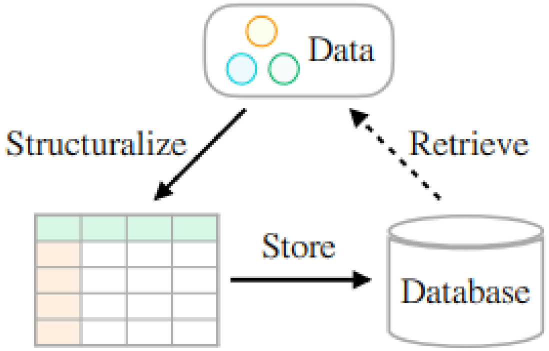
The sentiment classification (SC) task is guessing which prompt should be used

reStructured Pre-training, arXiv 2022, Weizhe Yuan, Pengfei Liu

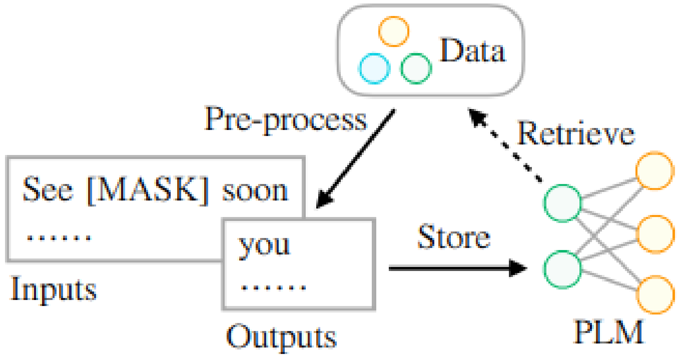
变化中寻找“不变”：**信息存储**和**访问的方式**越来越近



(a) Biological neural networks.

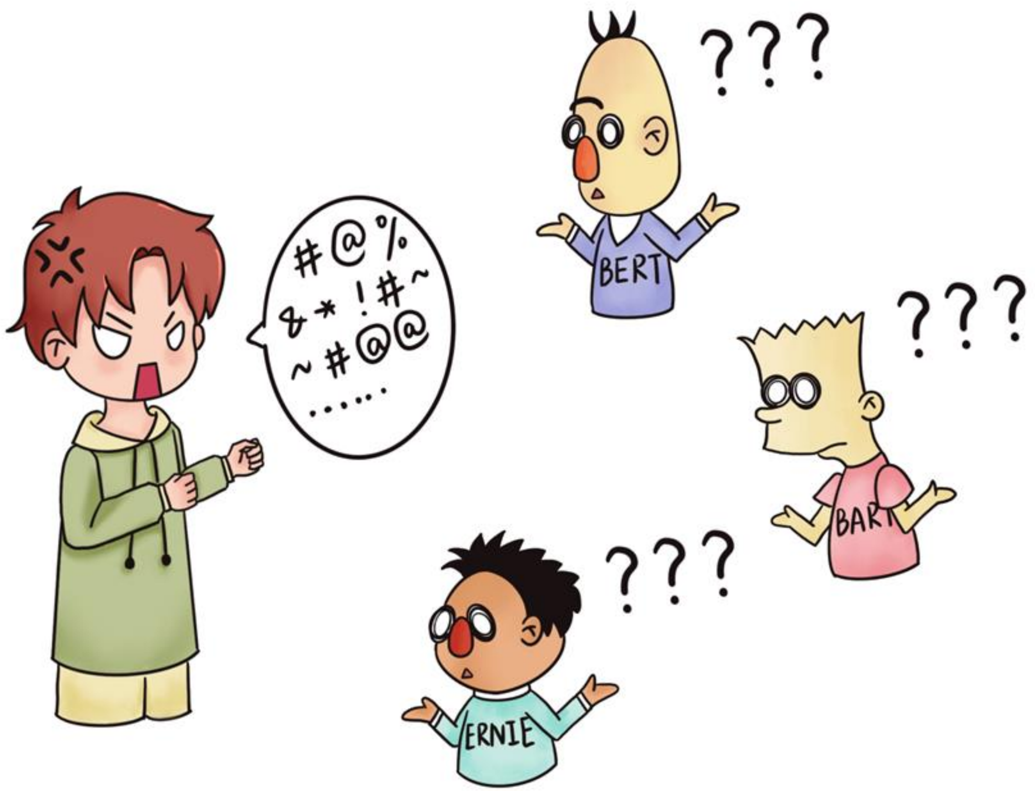


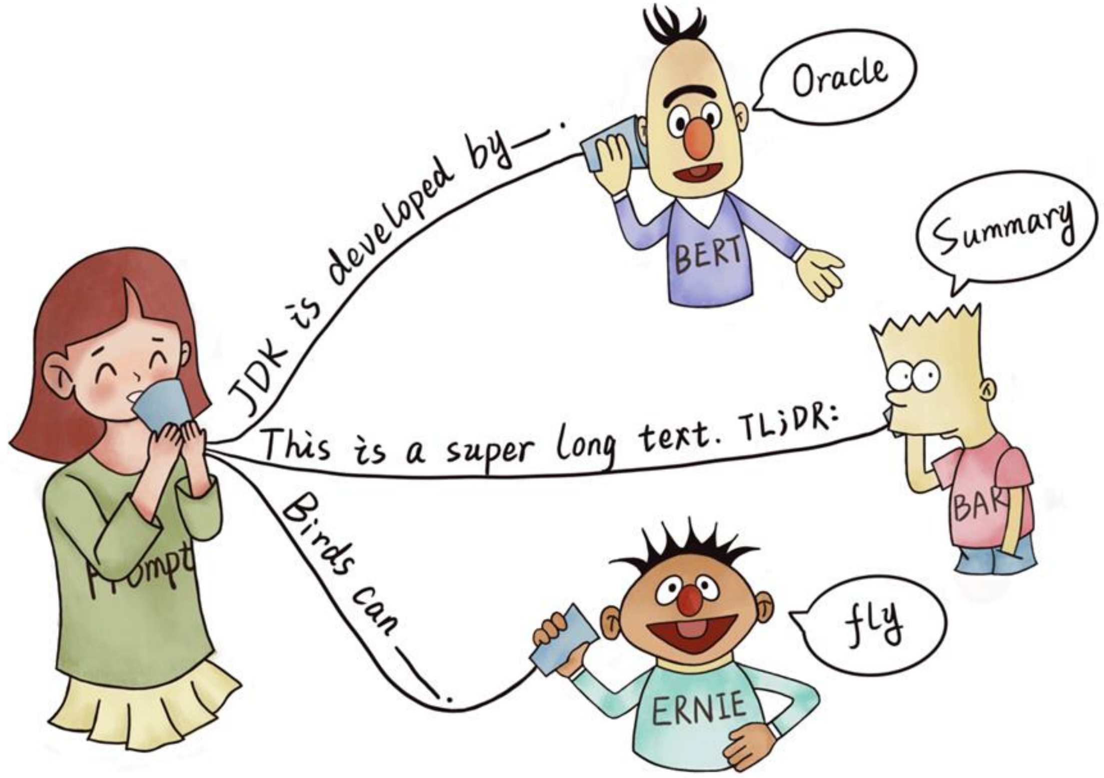
(b) Disk/Cloud storage.



(c) Artificial neural netwokrs.

变化中寻找“不变”：人与AI越来越近





如何写好prompt?

- Six strategies for getting better results (OpenAI)
- OpenAI Cook Book