COMP5423 Natural Language Processing

Neural Language Models (Part 2)

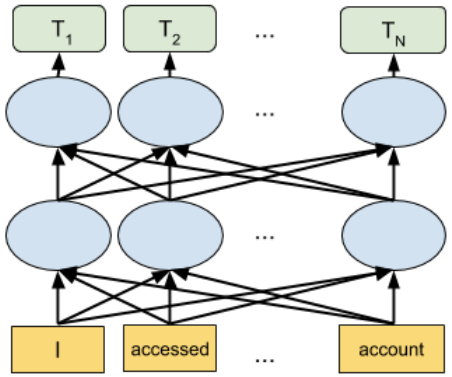
Outline

- **Learning Objectives**
  - Neural Language Models
    - Feedforward Neural Networks, Recurrent Neural Networks
    - Transformers, BERT, GPT, and BART
    - Attention Mechanisms
  - Neural Language Model Architectures
    - Encoder Only, Decoder Only, Encoder-Decoder (Sequence to Sequence Models)
  - Frequently Used Terms for Large Language Models
    - Pre-Training and Fine-Tuning, In Context Learning, Instruction Tuning, Alignment, and Prompting

Pre-Trained Language Models

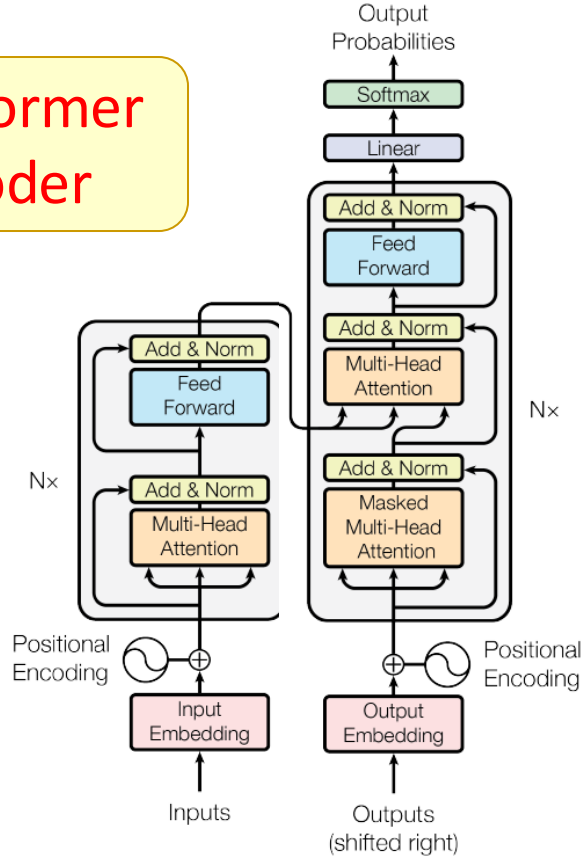
- **Pre-Trained Language Models (PLM)**

Transformer Encoder

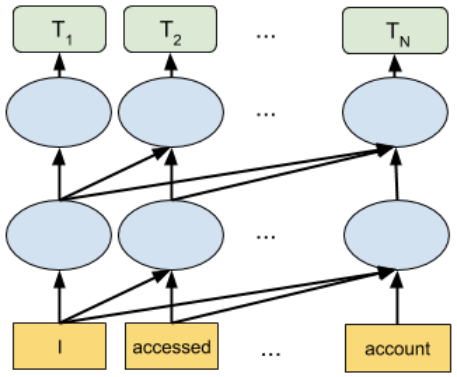


BERT

Bidirectional Encoder Representations from Transformers



Transformer Decoder

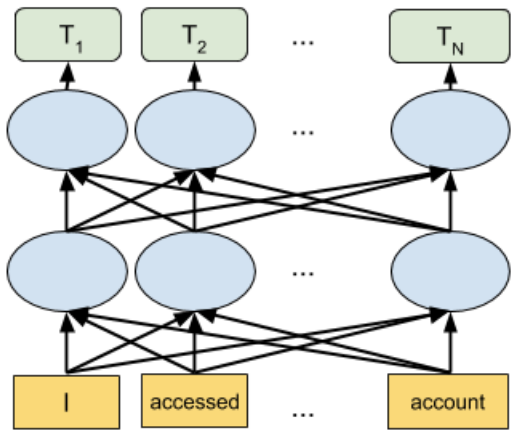


GPT

Generative Pre-trained Transformer

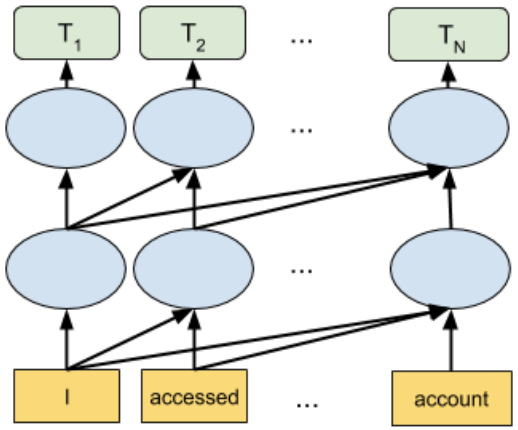
Pre-Trained Language Models

- **BERT: Bi-Directional Encoder Representation from Transformers**



BERT

Bidirectional Self-attention Model



GPT

Backward-Looking Self-attention Model

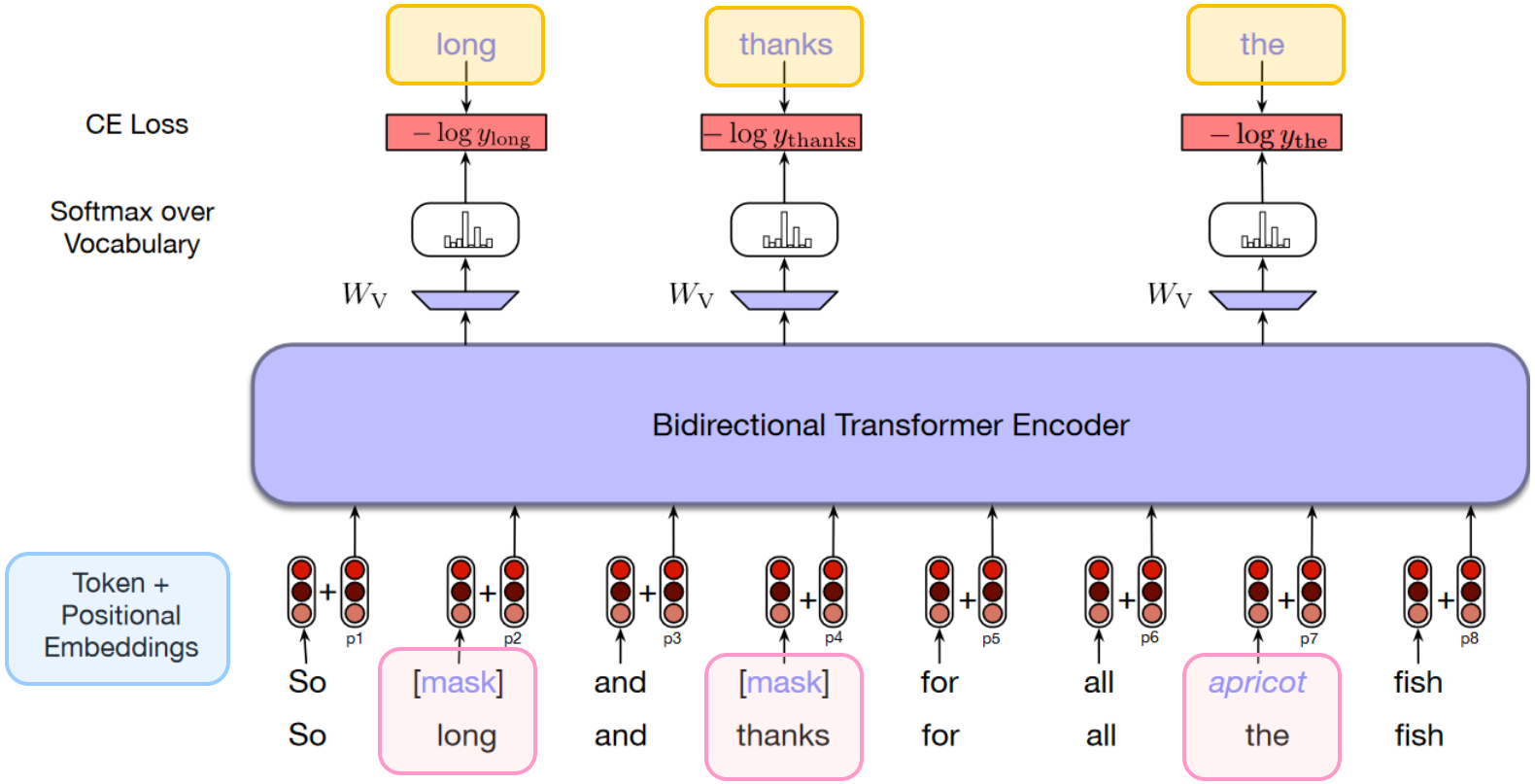
Natural Language Understanding

Pre-Trained Language Models

- **BERT: Bi-Directional Encoder Representation from Transformers**
  - Bi-Directional Transformer Encoder: Bidirectional encoders use self-attention to map sequences of input embeddings to sequences of output embeddings of the same length, where the output vectors have been contextualized using information from the entire input sequence.
  - Masked Language Model: To train a bi-directional encoder, instead of predicting the following word, we mask a word in the middle and ask the model to guess the word given the words on both sides.

Pre-Trained Language Models

- **BERT: Masked Language Models**
  - Training Tasks: (1) Fill-in-the-Blank (Cloze)

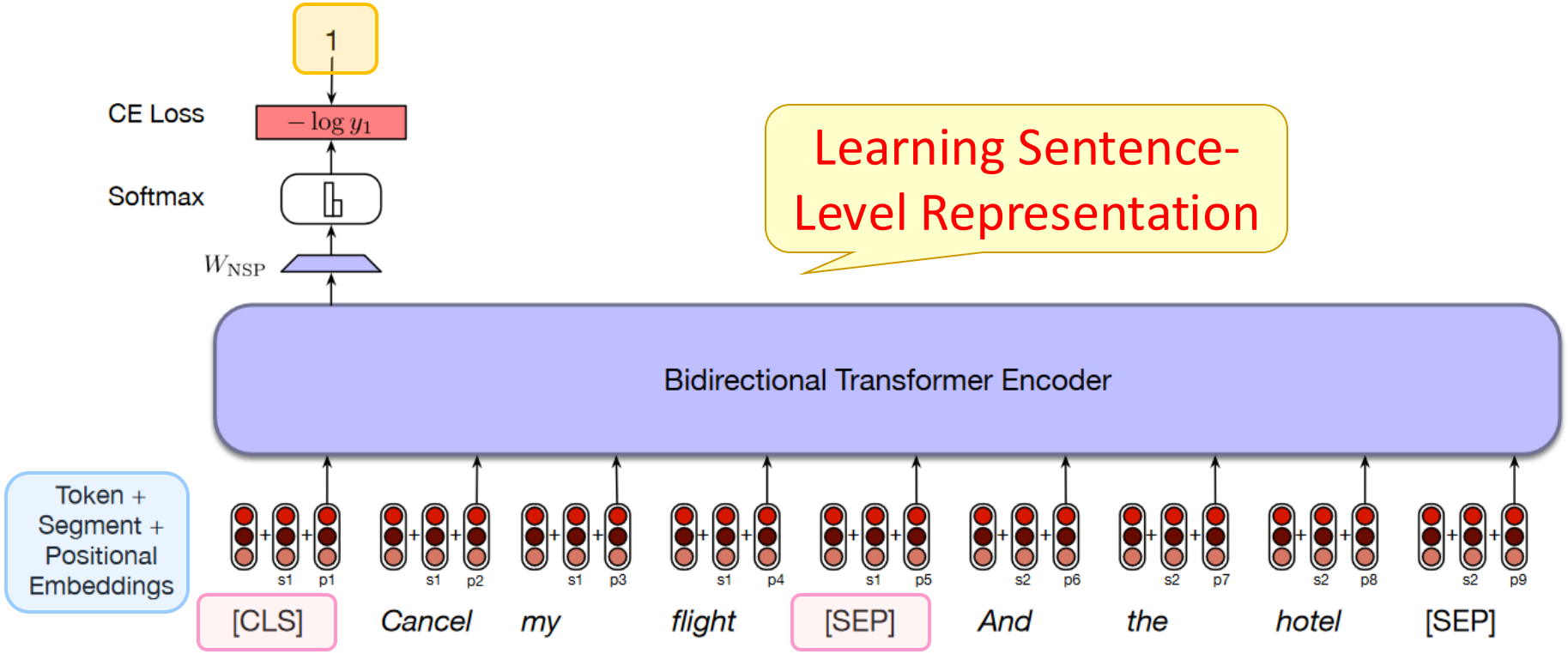


In BERT, 15% of the input tokens in a training sequence are sampled for learning. Of these, **80%** are replaced with [MASK], **10%** are replaced with randomly selected tokens, and the remaining **10%** are left unchanged.

Learning Word-Level Representation

Pre-Trained Language Models

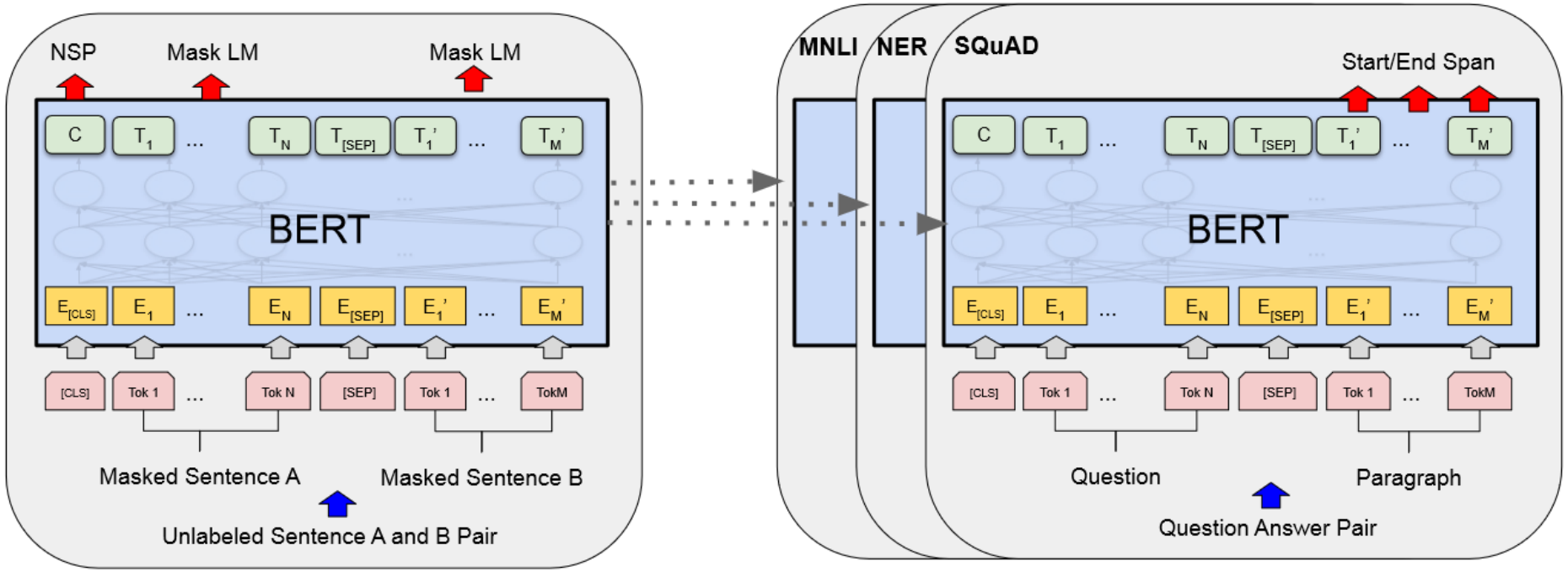
- **BERT: Masked Language Models**
  - Training Tasks: (2) Next Sentence Prediction (NSP)



  - Training Data: BooksCorpus + English Wikipedia

Pre-Trained Language Models

- **BERT: Masked Language Models**
  - Pre-Training and Fine-Turning

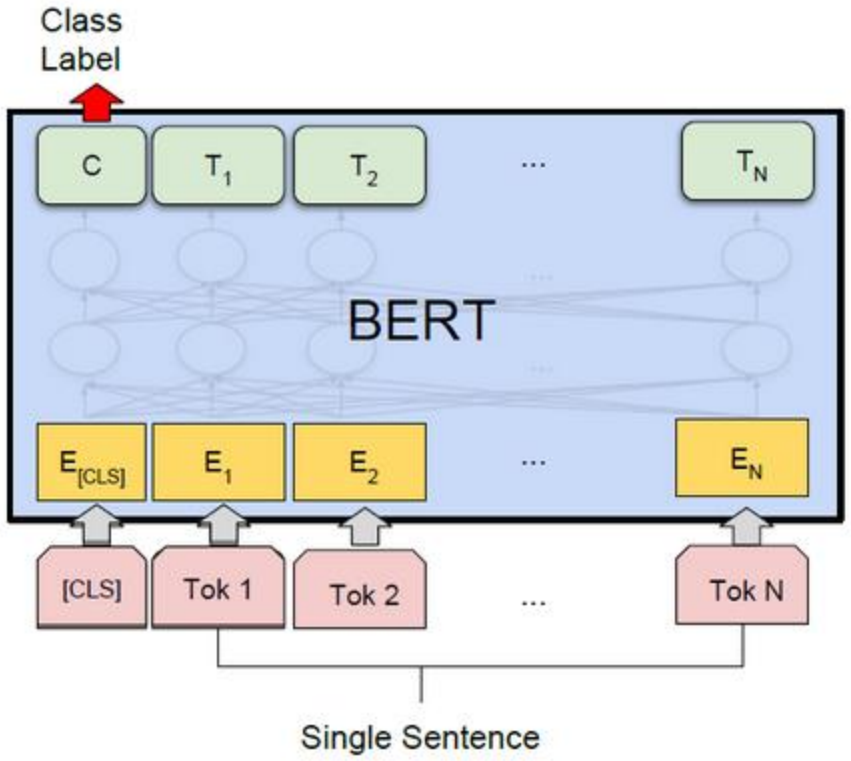


Self-supervised Pre-training

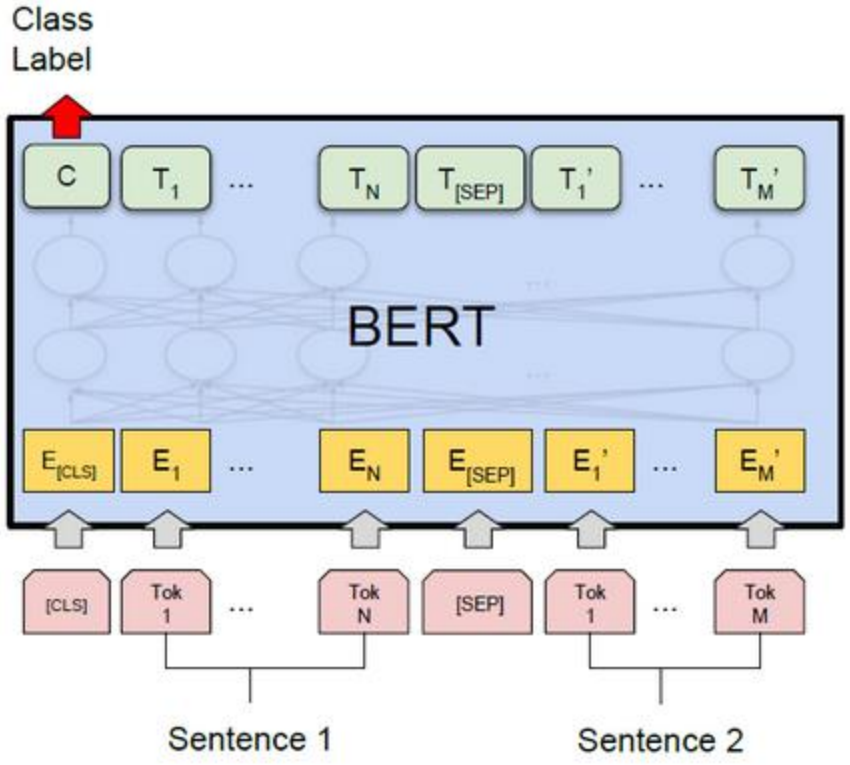
Supervised Fine-Turning

Pre-Trained Language Models

- **BERT: Masked Language Models**
  - Pre-Training and Fine-Turning



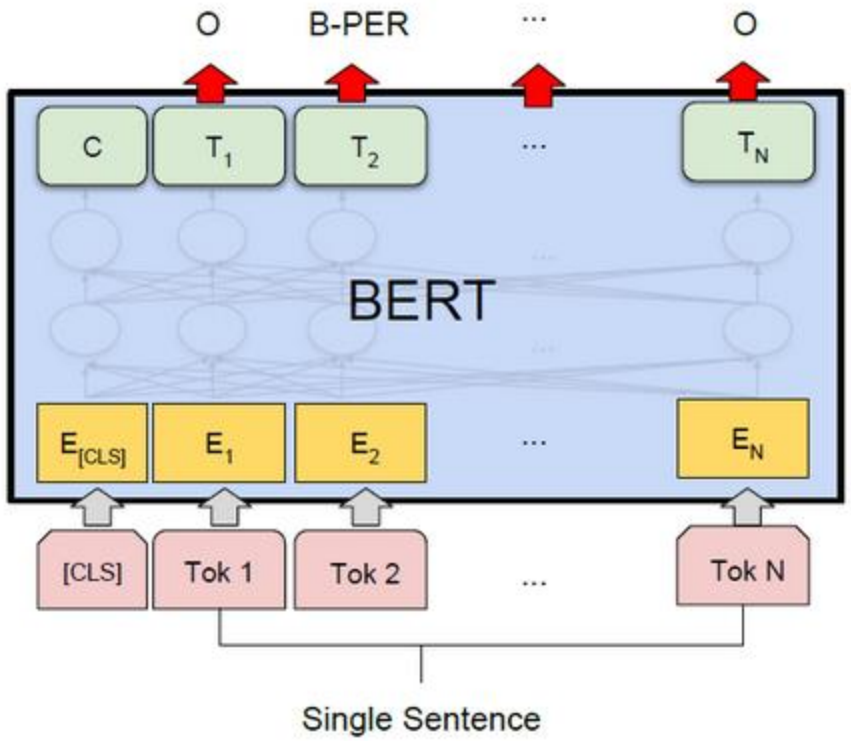
Single Sentence Classification



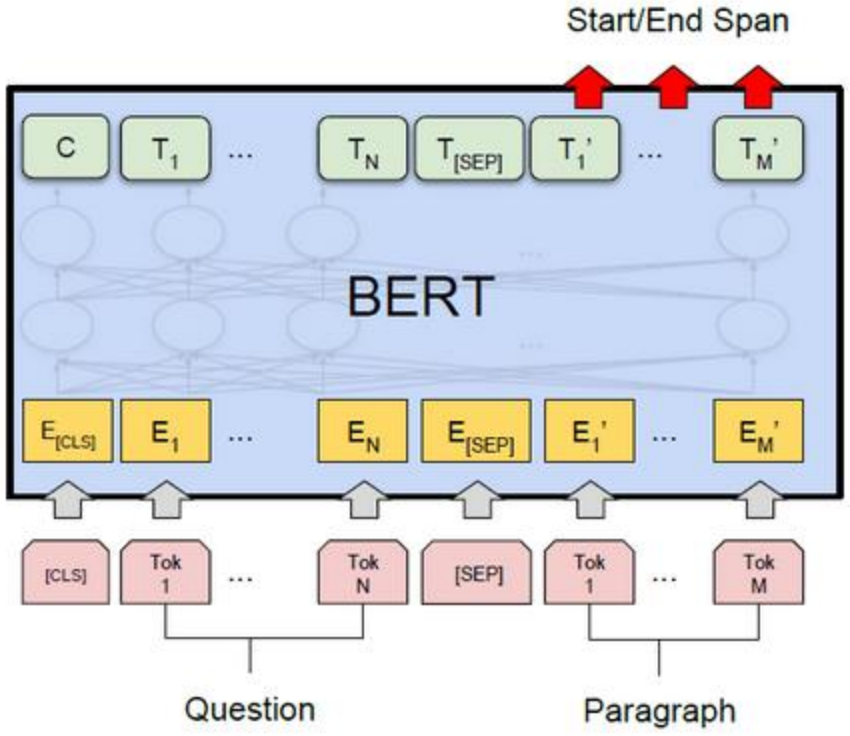
Sentence Pair Classification

Pre-Trained Language Models

- **BERT: Masked Language Models**
  - Pre-Training and Fine-Turning



Named Entity Tagging

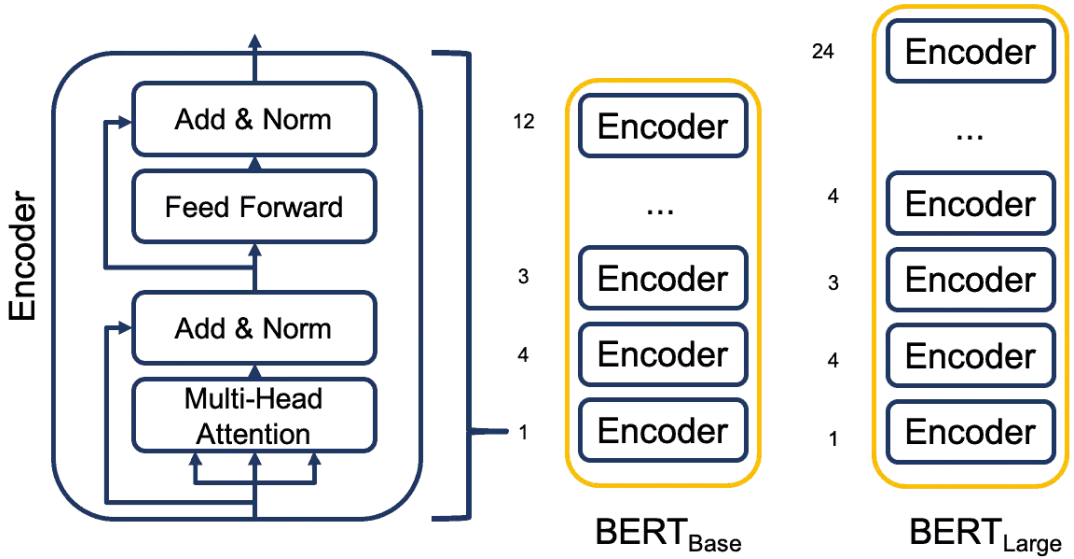


Answer Extraction

Pre-Trained Language Models

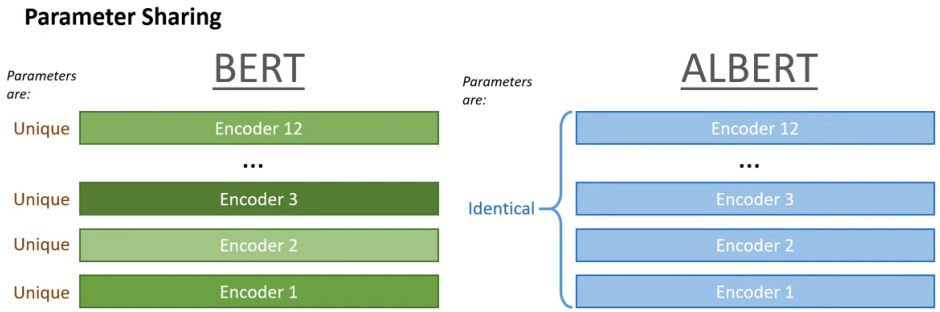
- **BERT: Masked Language Models**
  - BERT-Base, BERT-Large

Model		Parameters	Layers	Hidden
BERT	base	108M	12	768
	large	334M	24	1024



Pre-Trained Language Models

- **BERT Variations**
  - RoBERTa: BoBERTa: A Robustly Optimized BERT Pre-Training Approach, arXiv'2019.
    - Employs dynamic masking and remove next sentence prediction.
  - ALBERT: ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, ICLR'2020.
    - Proposes parameter reduction techniques.



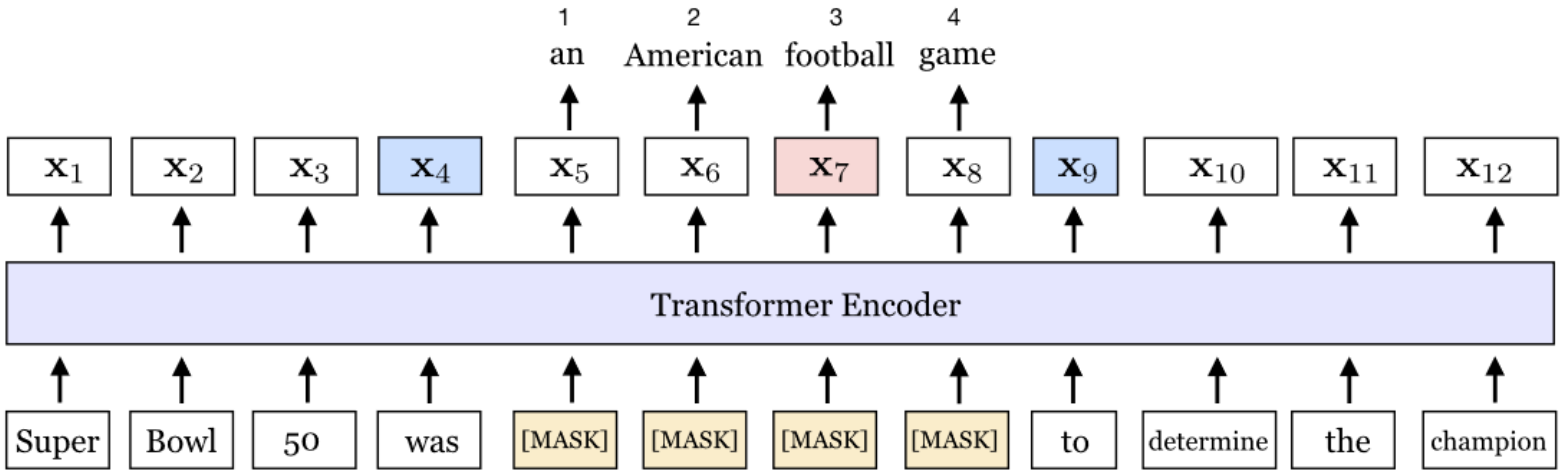
    - NSP (Next Sentence Prediction) → SOP (Sentence Order Prediction)

Pre-Trained Language Models

- **BERT Variations**
  - SpanBERT: Improving Pre-training by Representing and Predicting Spans, TACL'2020.
    - Masks random contiguous spans, rather than random individual tokens.
    - Trains the model based on tokens at the start and end of the boundary of span to predict the entire marked spans.

Pre-Trained Language Models

$$\mathcal{L}(\text{football}) = \mathcal{L}_{\text{MLM}}(\text{football}) + \mathcal{L}_{\text{SBO}}(\text{football})$$
$$= -\log P(\text{football} \mid \mathbf{x}_7) - \log P(\text{football} \mid \mathbf{x}_4, \mathbf{x}_9, \mathbf{p}_3)$$



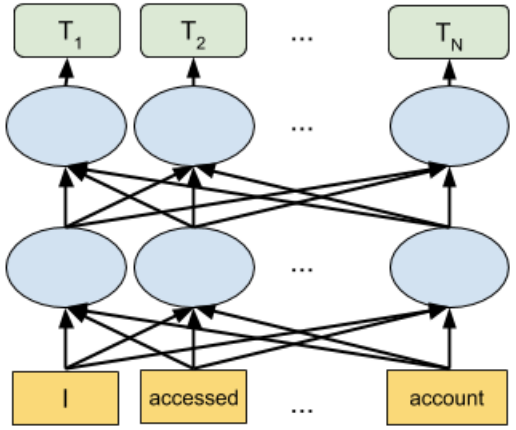
The *span an American football game* is masked. The Span-Boundary Objective (SBO) uses the output representations of the boundary tokens, x4 and x9 (in blue), to predict each token in the masked span. The equation shows the MLM and SBO loss terms for predicting the token, football (in pink), which as marked by the position embedding p3, is the third token from x4.

Pre-Trained Language Models

- **Homework**
  - Both RNN and BERT are effective for sequence labeling tasks like Part-of-Speech Tagging and Name Entity Recognition. (1) Can these models also be applied to Chinese word segmentation? If yes, how should we prepare the training data, including label annotation, and how can we train the RNN and BERT models for this task? (2) In addition, are there traditional non-neural-network sequence models that are suitable for Chinese word segmentation? If so, what are they and what features would you like to consider as input? (3) Furthermore, is it possible to use an encoder-decoder model for Chinese word segmentation? If so, how can this be achieved? If you answer to the above questions are no, please explain why.

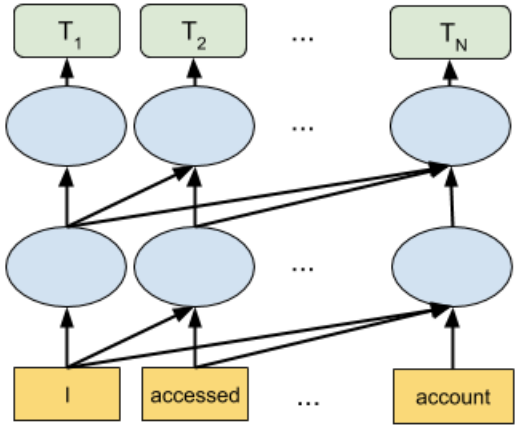
Pre-Trained Language Models

- **GPT: Auto-Regressive Language Models**



BERT

Bidirectional Self-attention Model

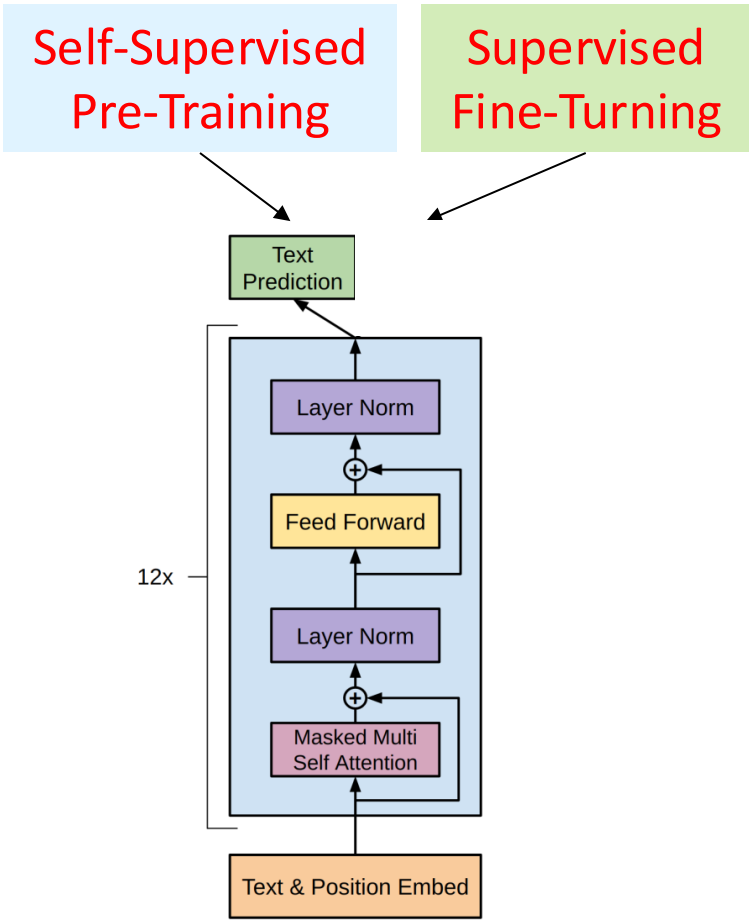


GPT

Backward-Looking Self-attention Model

Natural Language Generation

Pre-Trained Language Models

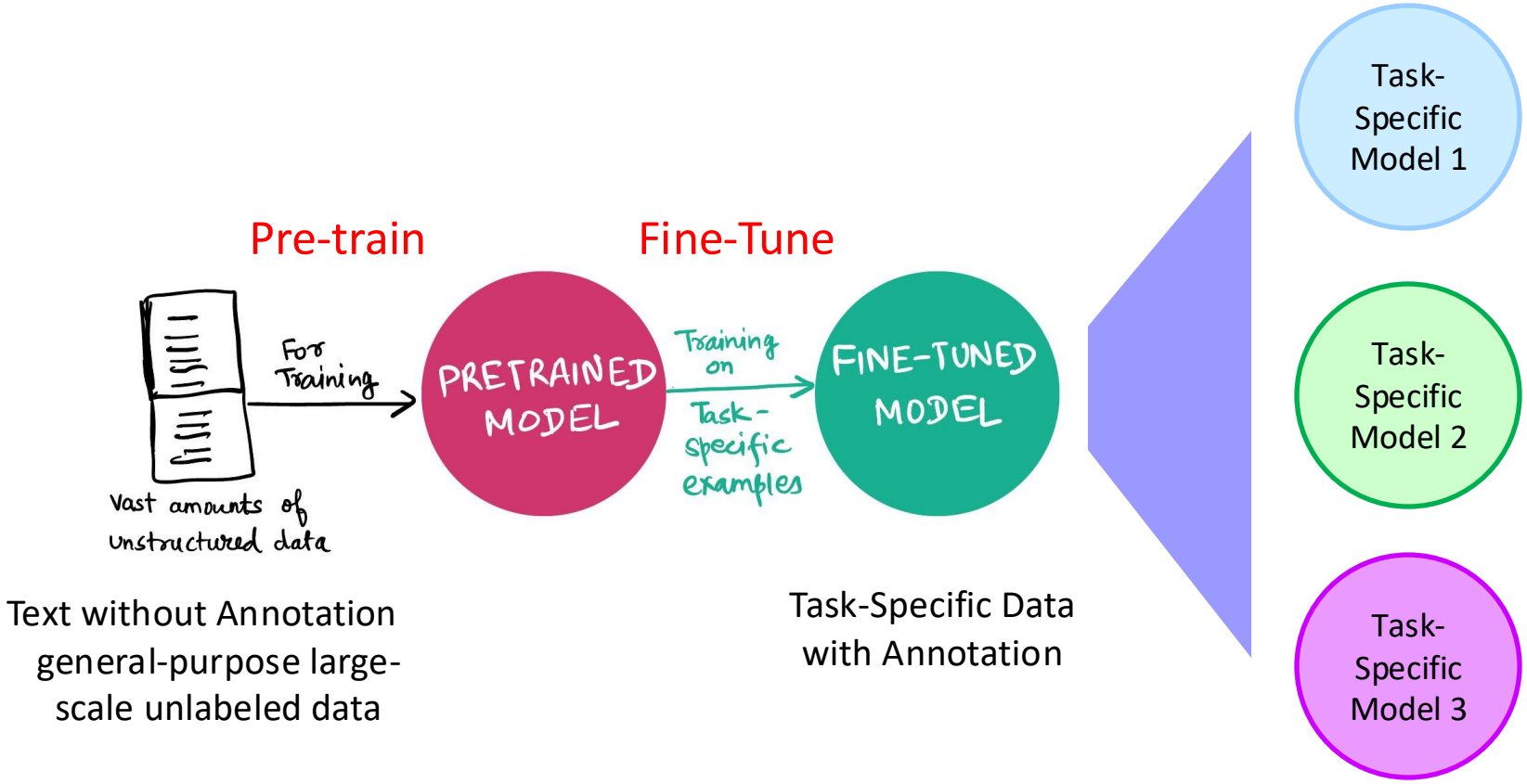


Pre-Trained Language Models

- **GPT: Auto-Regressive Language Models**
  - Training Task: Next Word Prediction
  - Task Adaptation
    - Utilizes Task-Specific Input Adaptation
    - Trains Parameters of an Added Linear Output Layer
    - It is found that including language modeling as an auxiliary objective to the fine-tuning helps learning (fine-tuning as continuous pretraining).
  - Training Data: BooksCorpus

Pre-Trained Language Models

- **Pre-Training and Fine-Tuning Paradigm**
  - Pre-trained language models are independent of any particular task.



Pre-Trained Language Models

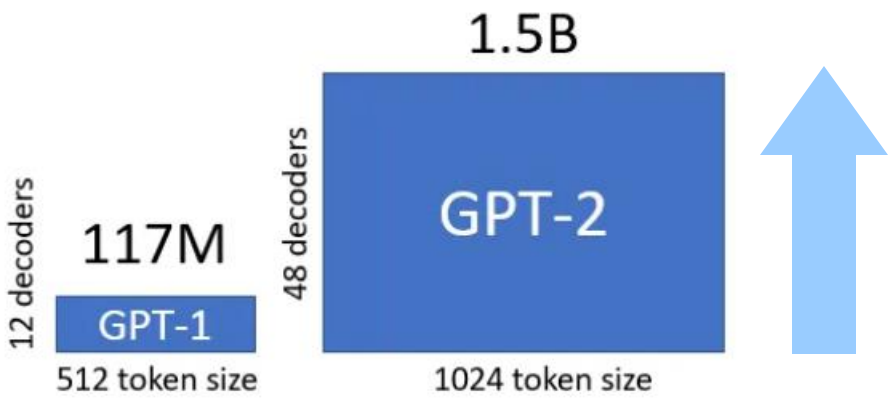
- **Pre-Training and Fine-Tuning Paradigm**
  - Un-Supervised Pre-Training and Supervised Fine-Turning
    - Pre-training is the process of learning some sort of representation of meaning for words or sentences by processing very large amounts of (raw) text.
    - Fine-tuning is the process of taking the representations from a pre-trained model and further training the model on task-specific data to perform a downstream task, like text classification or sequence labeling, using task-specific optimizations.

Pre-Trained Language Models

- **GPT-2: Auto-Regressive Language Models**

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

The GPT-2 model largely follows the details of GPT-1 with a few modifications.

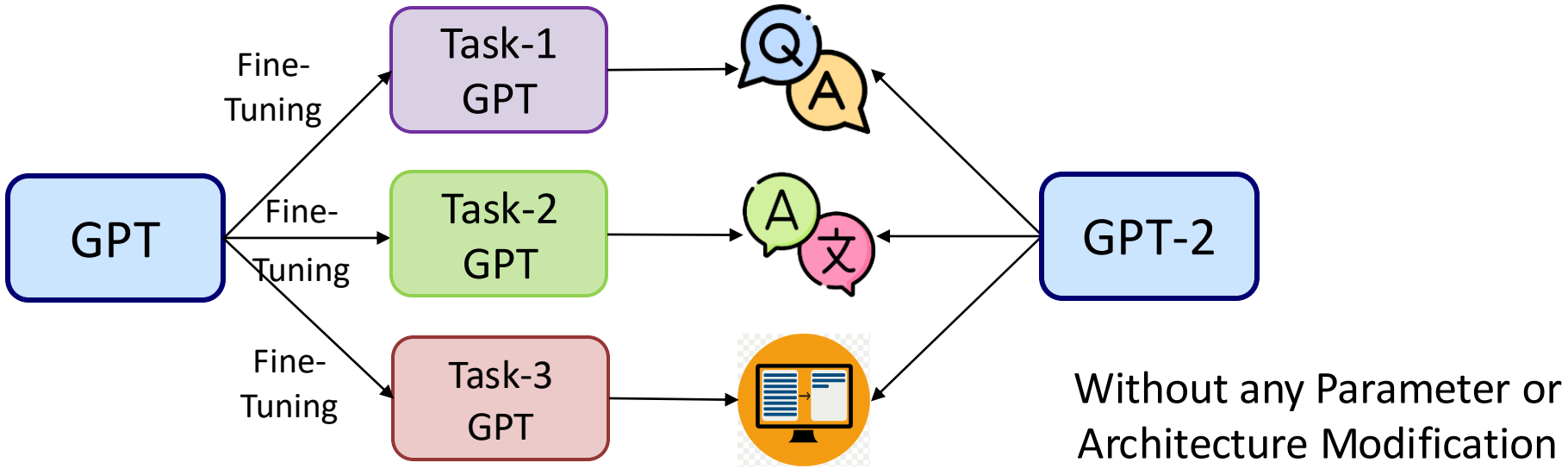


GPT-2 is 10 times more than GPT-1.

- Training Data: WebText

Pre-Trained Language Models

- **GPT-2: Auto-Regressive Language Models**
  - Un-supervised Multi-Task Learning: One Unsupervised Model for Multi-Tasks (Remove Task-Specific Fine-Tuning)



GPT-2 aims at learning multiple tasks using the same unsupervised model (by designing a series of heuristic solutions).

Pre-Trained Language Models

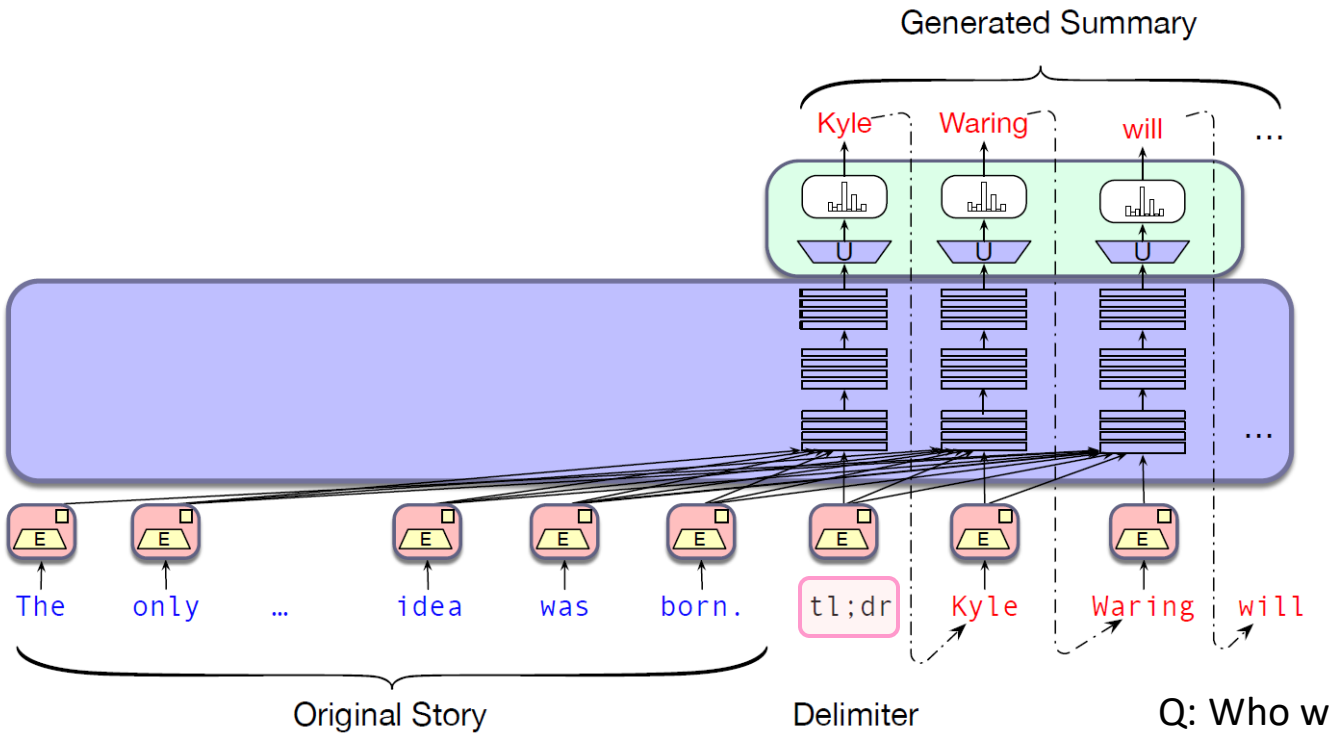
- **GPT-2: Auto-Regressive Language Model**
  - Multi-Task Learning
    - Trains a single model to perform many different tasks, even for the same input, by conditioning not only on the input but also on the task to be performed.

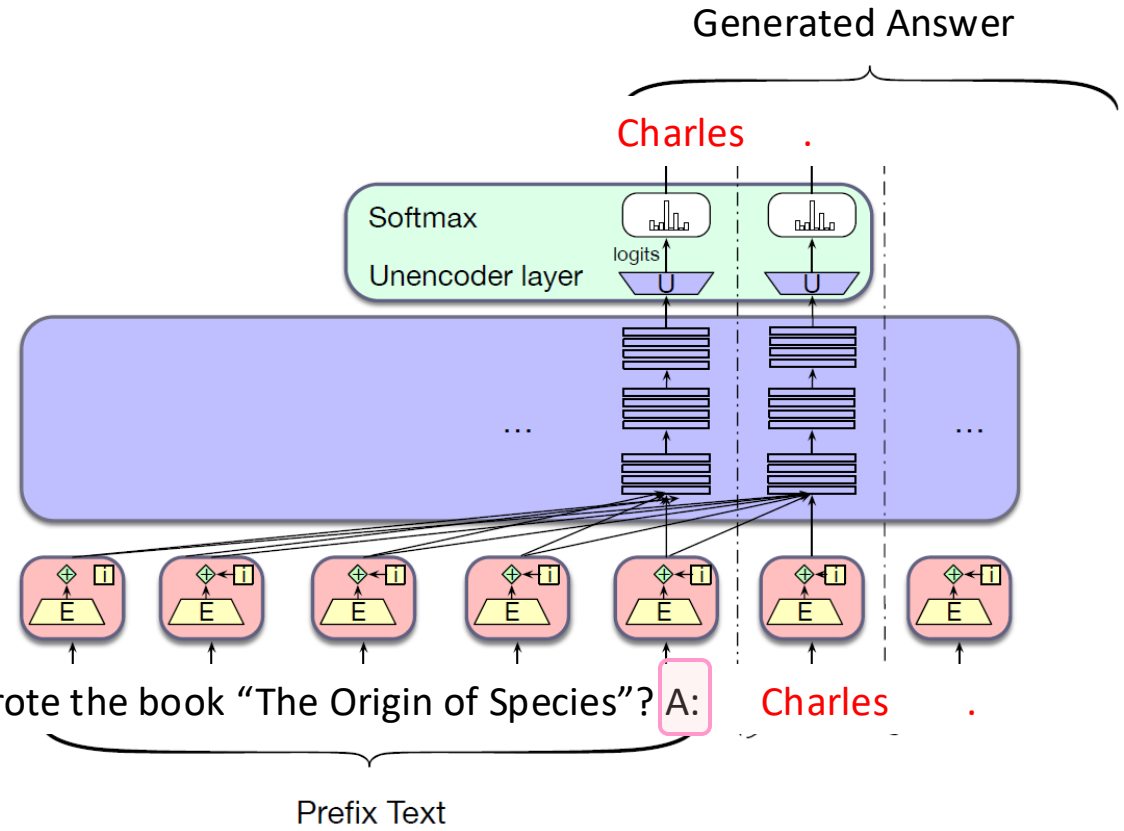
$$p(output|input) \Rightarrow p(output|input, \boxed{task})$$

For example, a translation training example can be written as the sequence (English text, translate to French, French text), and a reading comprehension training example can be written as (document, question, answer the question, answer).

Pre-Trained Language Models

Conditional generation is the task of generating text conditioned on an input piece of text.





$$\hat{w}_t = argmax_{w \in V} P(w|w_{<t})$$

Pre-Trained Language Models

- **GPT-2: Auto-Regressive Language Models**
  - Zero-Shot NLP Task Generalization: Reading Comprehension, Summarization, Translation, and Question Answering

Reading Comprehension

Conditioned on a document, the history of the associated conversation and a final token **A:**, GPT-2 achieves **55** F1, comparable to 3 out of 4 supervised baseline systems. The supervised SOTA, a BERT based system is nearing the **89** F1 performance of humans.

Summarization

Generate 3 sentences conditioned on a document and **TL;DR:**. GPT-2 often focuses on recent content from the article or confuses specific details and just barely outperforms selecting 3 random sentences.

Pre-Trained Language Models

- **GPT-2: Auto-Regressive Language Models**

Translation

GPT-2 is conditioned on a context of example pairs of the format "English sentence = French sentence". And then, after a final prompt of "English sentence =", the translation is generated. GPT-2 gets **5** BLEU, slightly worse than a word-by-word substitution with a bilingual lexicon and much worse than the **33.5** BLEU of the current best unsupervised machine translation approach.

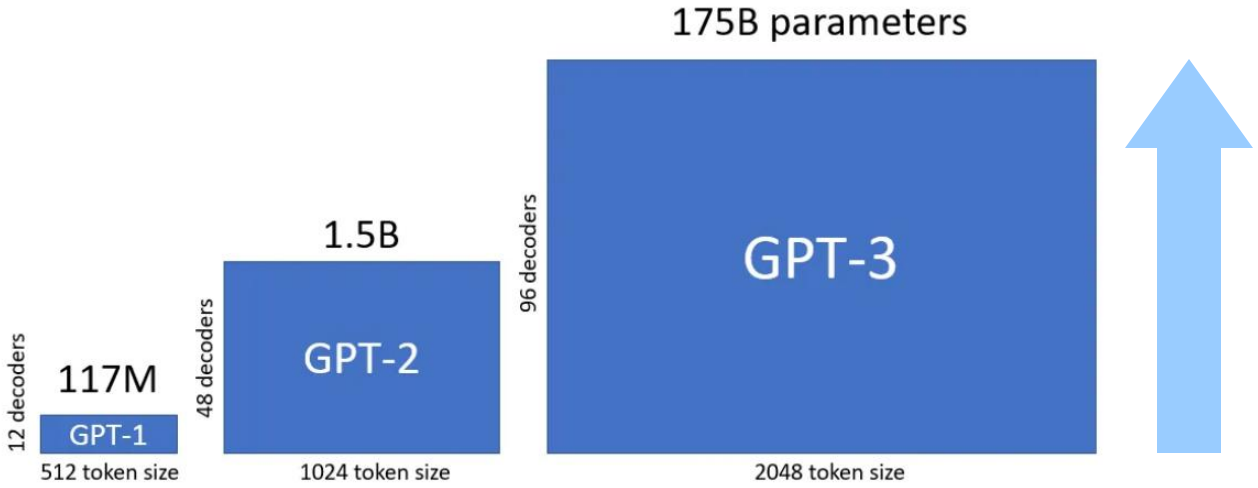
Question Answering

Similar to translation, the context of the language model is seeded with example question answer pairs. GPT-2 answers **4.1%** of questions correctly, much, much, worse than the 30 to 50% range of open domain question answering systems. It also suggests that **model capacity** has been a major factor in the poor performance of neural systems on this kind of task.

Pre-Trained Language Models

GPT-3: Auto-Regressive Language Models

Model Name	n_{params}	n_{layers}	d_{model}
GPT-3 Small	125M	12	768
GPT-3 Medium	350M	24	1024
GPT-3 Large	760M	24	1536
GPT-3 XL	1.3B	24	2048
GPT-3 2.7B	2.7B	32	2560
GPT-3 6.7B	6.7B	32	4096
GPT-3 13B	13.0B	40	5140
GPT-3 175B or "GPT-3"	175.0B	96	12288



The GPT-3 architecture is mostly the same as GPT-2 one.
The largest GPT-3 model size is 100 times larger than the largest GPT-2 model.
GPT-3 achieves strong performance on many NLP datasets, including translation, question-answering, and cloze tasks, as well as several tasks that require on-the-fly reasoning or domain adaptation.

Learning within the Context

The three settings we explore for in-context learning

Zero-shot

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

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

One-shot

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

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

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
```

Traditional fine-tuning (not used for GPT-3)

Fine-tuning

The model is trained via repeated gradient updates using a large corpus of example tasks.

```
sea otter => loutre de mer        ← example #1
↓
gradient update
↓
peppermint => menthe poivrée      ← example #2
↓
gradient update
↓
...
↓
plush giraffe => girafe peluche   ← example #N
gradient update
cheese =>                         ← prompt
```

Demonstrations of Task

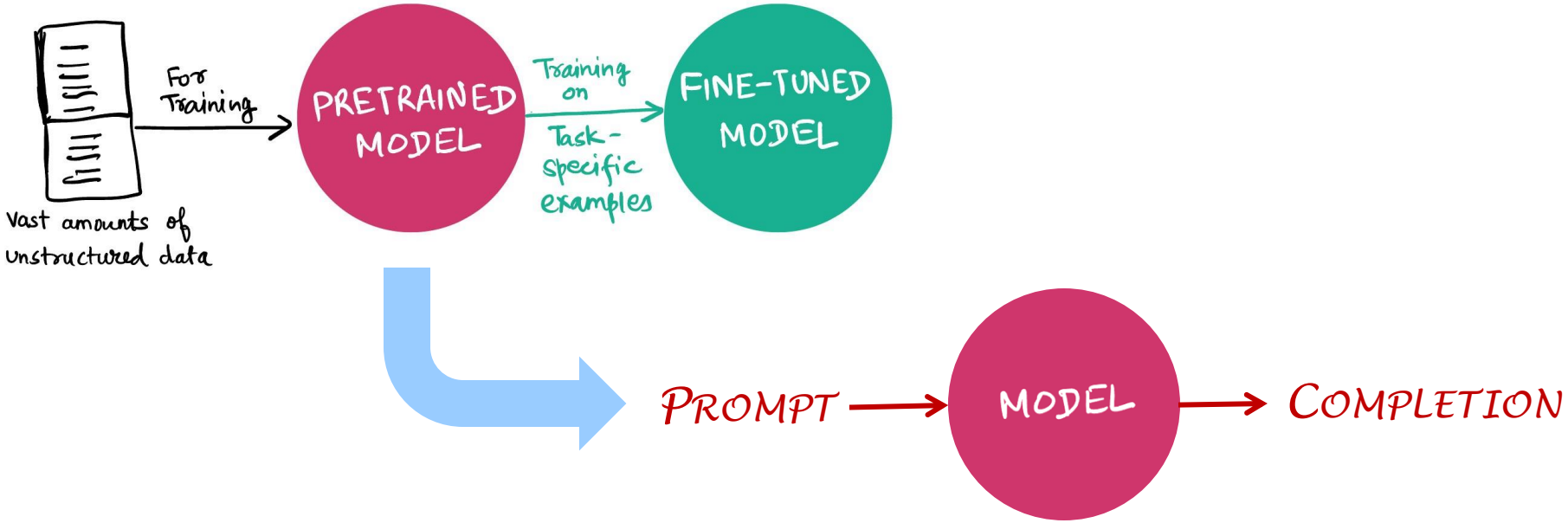
Pre-Trained Language Models

- **GPT-3: Auto-Regressive Language Models**
  - Prompt-based, In-Context Learning (ICL): Text Interaction with the Model, without any Gradient Updates or Fine-Tuning
    - Zero-Shot, One-Shot and Few-Shot Learning
    - Conclusion: Larger models make increasingly efficient use of in-context information.

In-context learning (ICL) is a specific method of prompt engineering where demonstrations of the task are provided to the model as part of the prompt (in natural language). With ICL, you can use off-the-shelf large language models (LLMs) to solve novel tasks without the need for fine-tuning.

Pre-Trained Language Models

- **GPT-3: Auto-Regressive Language Models**



Basic Prompt Templates

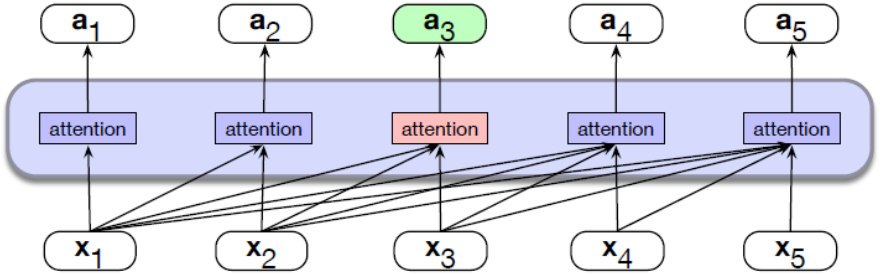
Summarization	{input} ; tldr;
Translation	{input} ; translate to French:
Sentiment	{input}; Overall, it was
Fine-Grained-Sentiment	{input}; What aspects were important in this review?

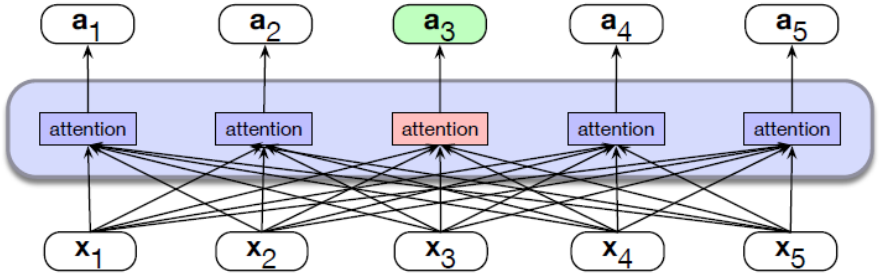
Sentiment	**Prompt**: $INPUT + In short, our stay was **Output**: not enjoyable
Fine-grained Sentiment	**Prompt**: $INPUT + These aspects were important to the reviewer: **Output**: 1. Poor service 2. Unpleasant location 3. Noisy and busy area
Summarization	**Prompt**: $INPUT + tl;dr **Output**: I had a bad experience with the hotel's service and the location was loud and busy.
Translation	**Prompt**: $INPUT + Translate this to French **Output**: Je n'ai pas aimé le service qui m'a été offert lorsque je suis entré dans l'hôtel. Je n'ai également pas aimé la zone dans laquelle se trouvait l'hôtel. Trop de bruit et d'événements pour que je me sente détendu et loin de la vie citadine.

Pre-Trained Language Models

- **GPT vs. BERT: A Comparison**

OpenAI	Google
Decoder-only Model	Encoder-only Model
Uni-Directional Attention: Left Context Only	Bi-Directional Attention, Left and Right Context
Causal Language Model: Autoregressive Text Generation	Masked Language Model: Language Understanding, Contextual Embedding

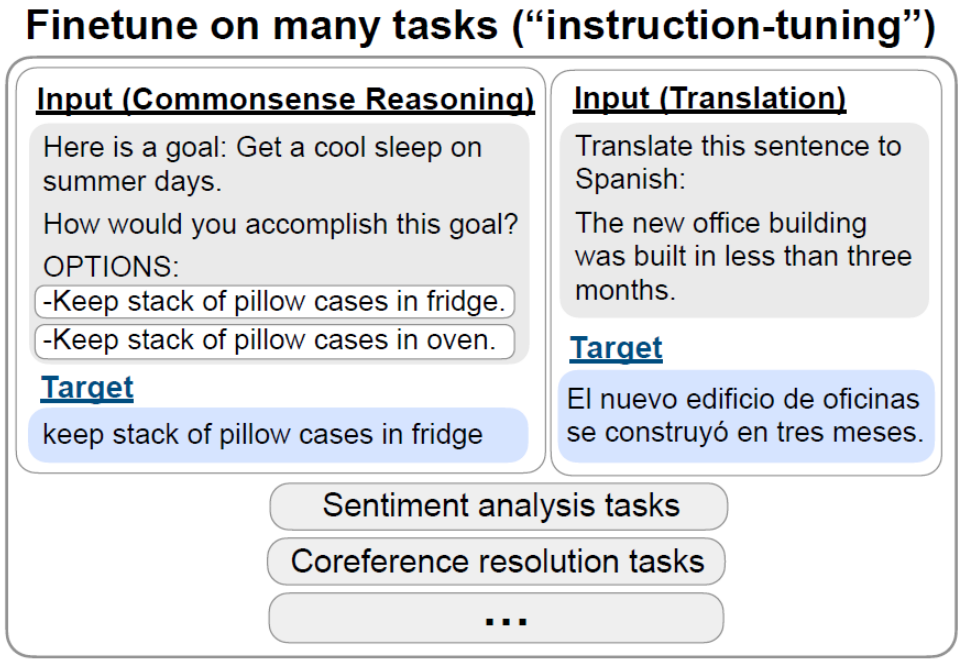


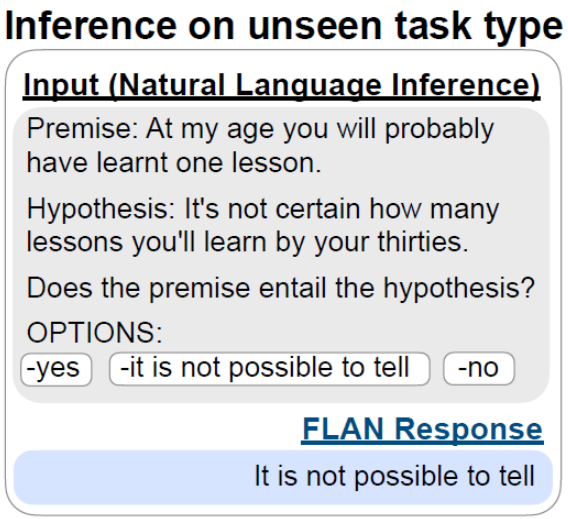


Pre-Trained Language Models

■ Instruction Tuning

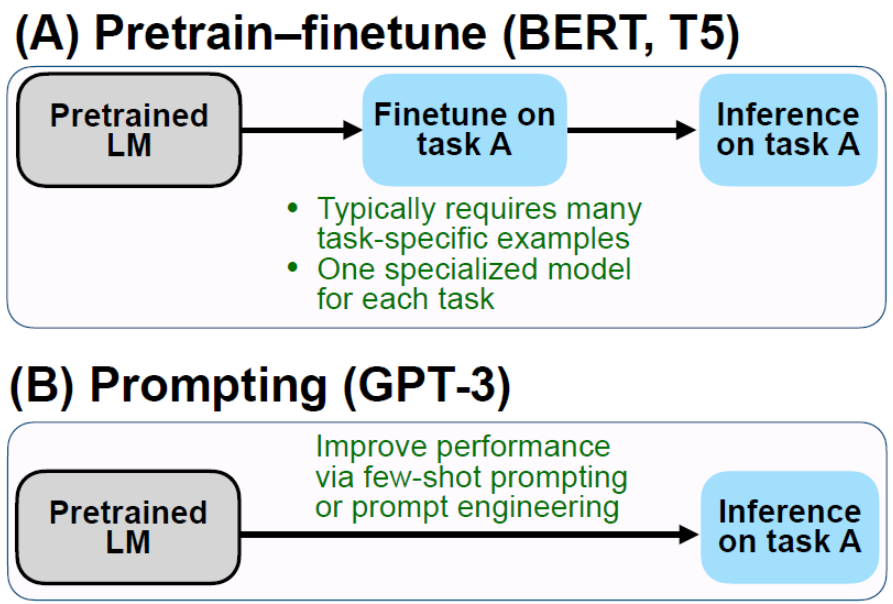
FLAN (Finetuned Language Net) is a decoder-only transformer language model of 137B parameters



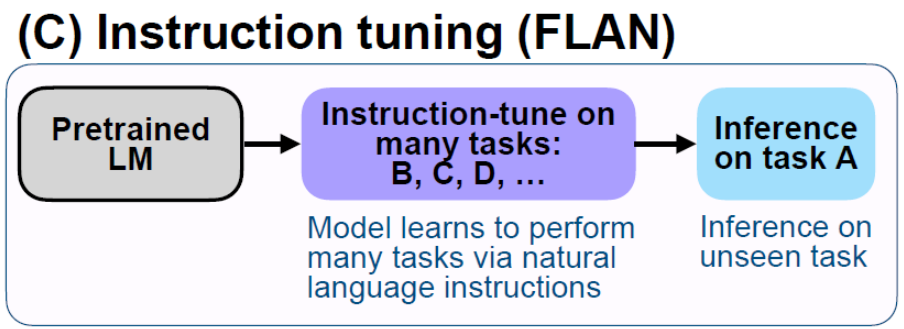


Pre-Trained Language Models

Instruction Tuning

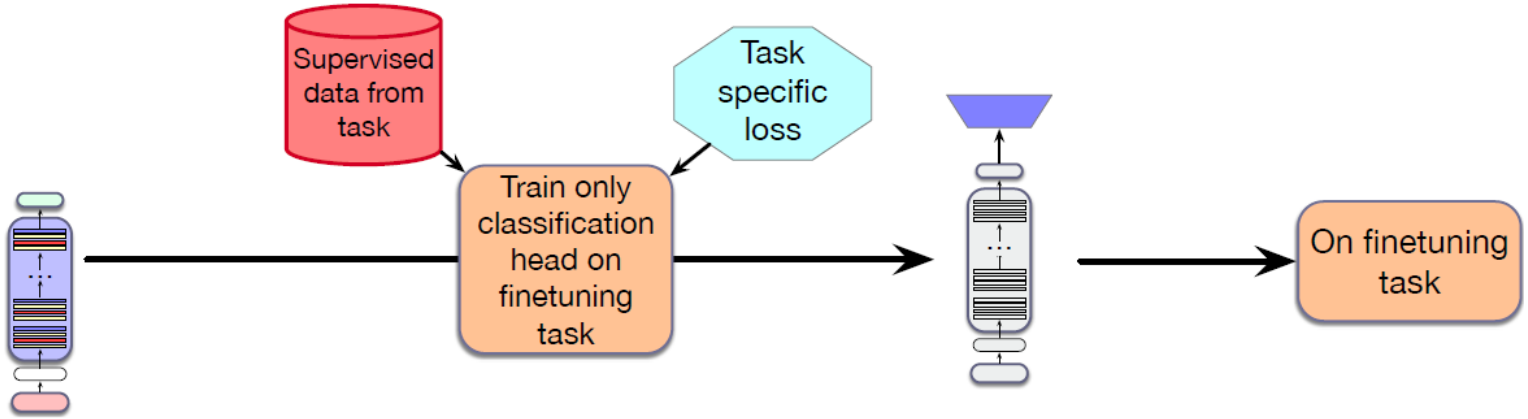


Comparing Instruction Tuning with Pretrain/Finetune and Prompting

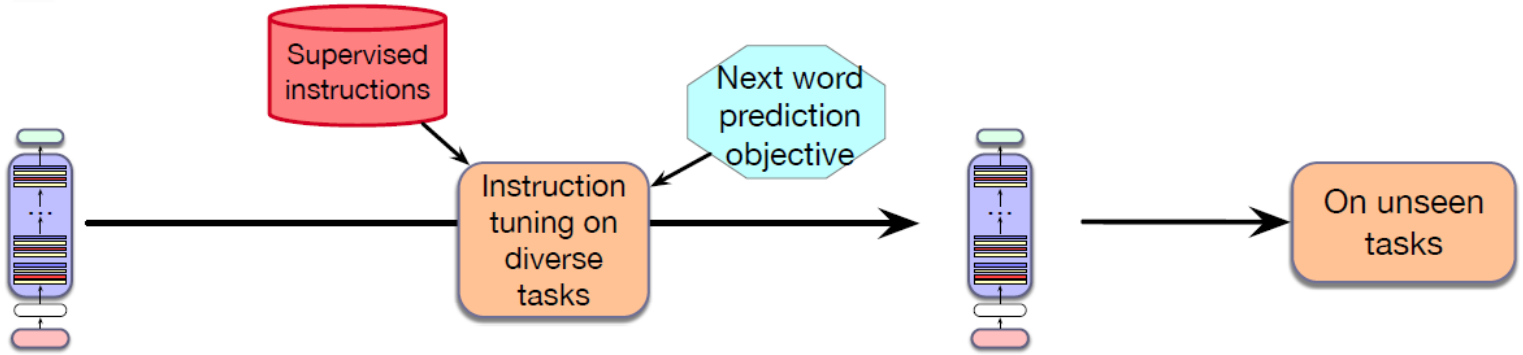


Instruction tuning combines appealing aspects of both the pretrain/finetune and prompting paradigms by using supervision via finetuning. The empirical results demonstrate promising abilities of language models to perform tasks described purely via instructions.

Pre-Trained Language Models



Task-based Fine-Tuning



Instruction Tuning

Pre-Trained Language Models

- **Instruction Tuning**
  - Instruction tuning (short for instruction finetuning, and sometimes even short-ended to instruct tuning) is a method for making an LLM better at following instructions. It involves taking a base pretrained LLM and continue training it to follow instructions for a range of tasks, by finetuning it on a corpus of instructions and responses, by using the same language modeling objective used to train the original model.
  - The resulting model not only learns those tasks, but also engages in a form of meta-learning, i.e., it improves its ability to follow instructions generally.

Pre-Trained Language Models

- **InstructGPT: Trained with Humans in the Loop**

Aligning Language Models to Follow Instructions

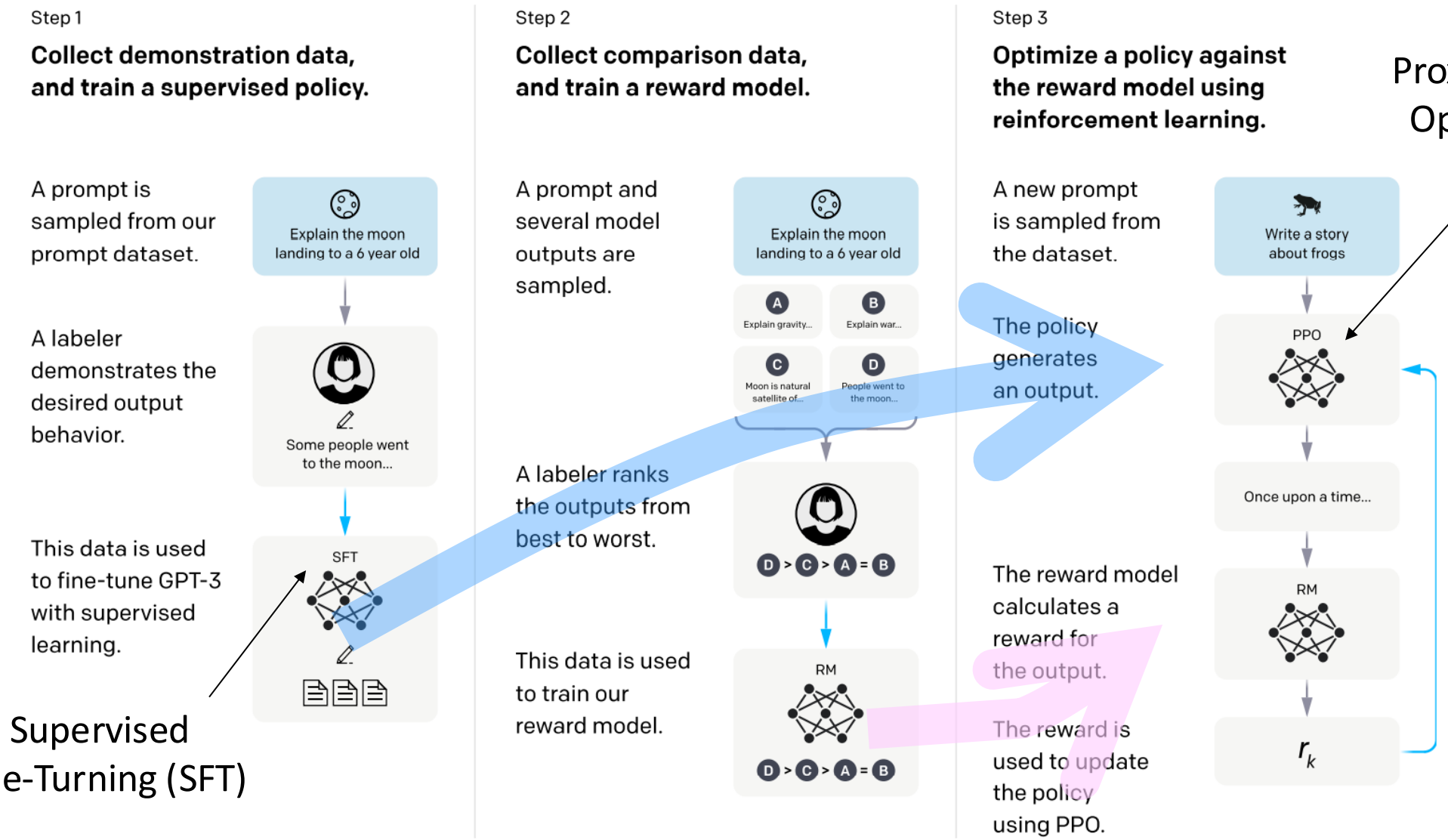
We've trained language models that are much better at following user intentions than GPT-3 while also making them more truthful and less toxic, using techniques developed through our alignment research. These *InstructGPT* models, which are trained with humans in the loop, are now deployed as the default language models on our API.

January 27, 2022
16 minute read

READ PAPER | VIEW MODEL CARD

- Instruction Tuning (aka. Instruction Finetuning and Instruct Tuning)
  - Model Alignment: Align language models to follow instructions.
- Reinforcement Learning from Human Feedback (RLHF) or DPO

Pre-Trained Language Models



Proximal Policy Optimization

Supervised Fine-Turning (SFT)

Pre-Trained Language Models

- **ChatGPT**

Methods

We trained this model using Reinforcement Learning from Human Feedback (RLHF), using the same methods as InstructGPT, but with slight differences in the data collection setup. We trained an initial model using supervised fine-tuning: human AI trainers provided conversations in which they played both sides—the user and an AI assistant. We gave the trainers access to model-written suggestions to help them compose their responses. We mixed this new dialogue dataset with the InstructGPT dataset, which we transformed into a dialogue format.

ChatGPT: Optimizing Language Models for Dialogue

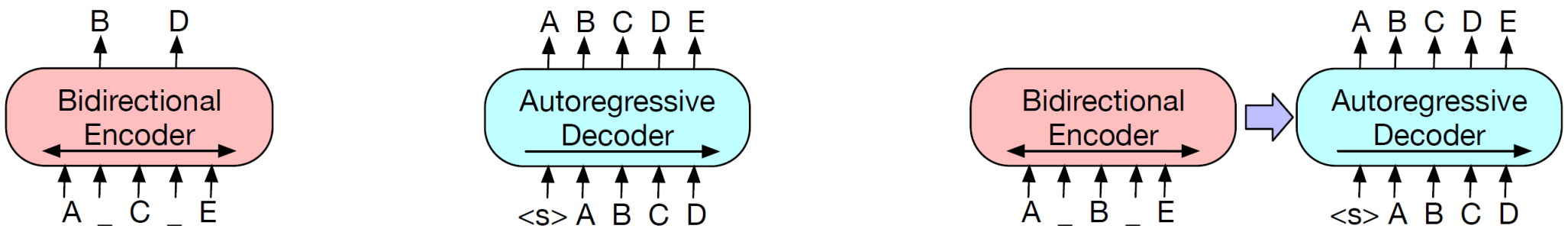
We've trained a model called ChatGPT which interacts in a conversational way. The dialogue format makes it possible for ChatGPT to answer followup questions, admit its mistakes, challenge incorrect premises, and reject inappropriate requests. ChatGPT is a sibling model to InstructGPT, which is trained to follow an instruction in a prompt and provide a detailed response.

TRY CHATGPT ↗

November 30, 2022
13 minute read

Pre-Trained Language Models

■ BART: Pre-Trained Sequence-to-Sequence Language Models



(a) BERT: Random tokens are replaced with masks, and the document is encoded bidirectionally. Missing tokens are predicted independently, so BERT cannot easily be used for generation.

(b) GPT: Tokens are predicted auto-regressively, meaning GPT can be used for generation. However words can only condition on leftward context, so it cannot learn bidirectional interactions.

(c) BART: Inputs to the encoder need not be aligned with decoder outputs, allowing arbitrary noise transformations. Here, a document has been corrupted by replacing spans of text with a mask symbols. The corrupted document (left) is encoded with a bidirectional model, and then the likelihood of the original document (right) is calculated with an autoregressive decoder.

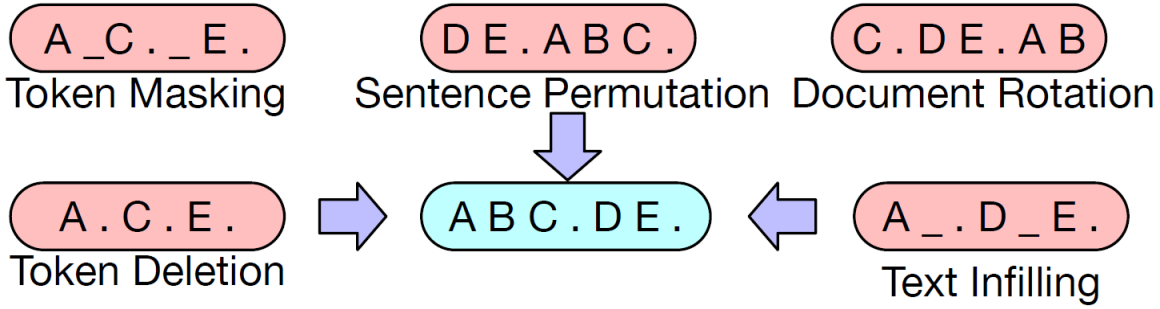
Pre-Trained Language Models

- **BART: Pre-Trained Sequence-to-Sequence Language Models**
  - Pre-Training

A document is divided into sentences based on full stops, and these sentences are shuffled in a random order.

Random tokens are sampled and replaced with [MASK] elements.

Random tokens are deleted from the input.



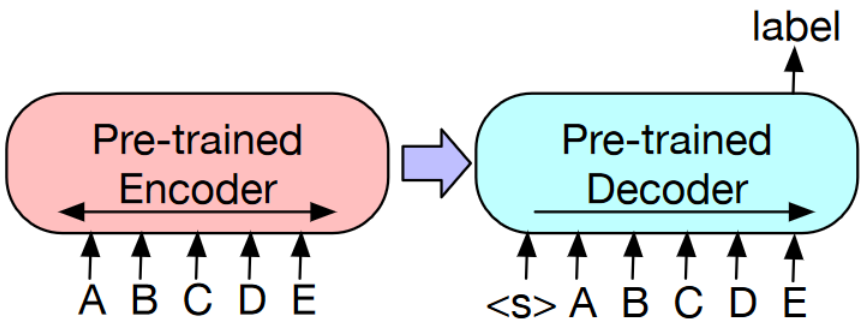
A token is chosen uniformly at random, and the document is rotated so that it begins with that token. This task trains the model to identify the start of the document.

A number of text spans are sampled, with span lengths drawn from a Poisson distribution. Each span is replaced with a single [MASK] token.

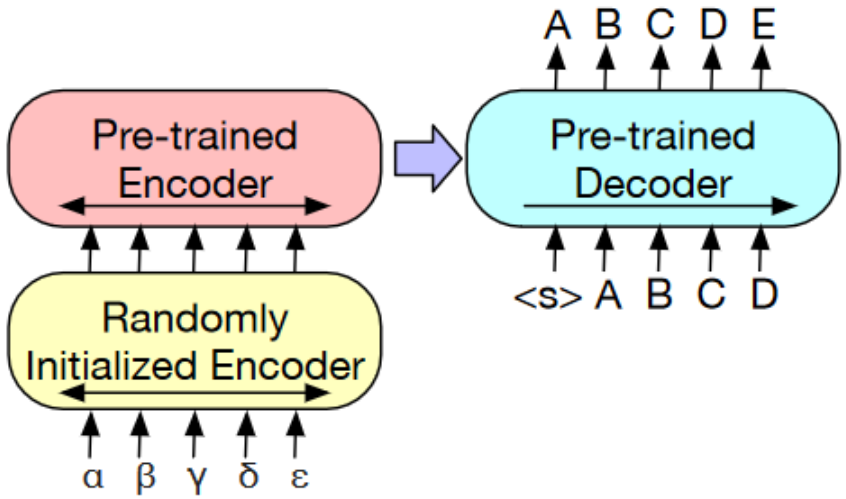
BART is a denoising autoencoder for pretraining sequence-to-sequence models. It is trained by corrupting documents and then optimizing a reconstruction loss, i.e., the cross-entropy between the decoder's output and the original document.

Pre-Trained Language Models

- **BART: Pre-Trained Sequence-to-Sequence Language Models**
  - Fine-Tuning



(a) To use BART for classification problems, the same input is fed into the encoder and decoder, and the representation from the final output is used.

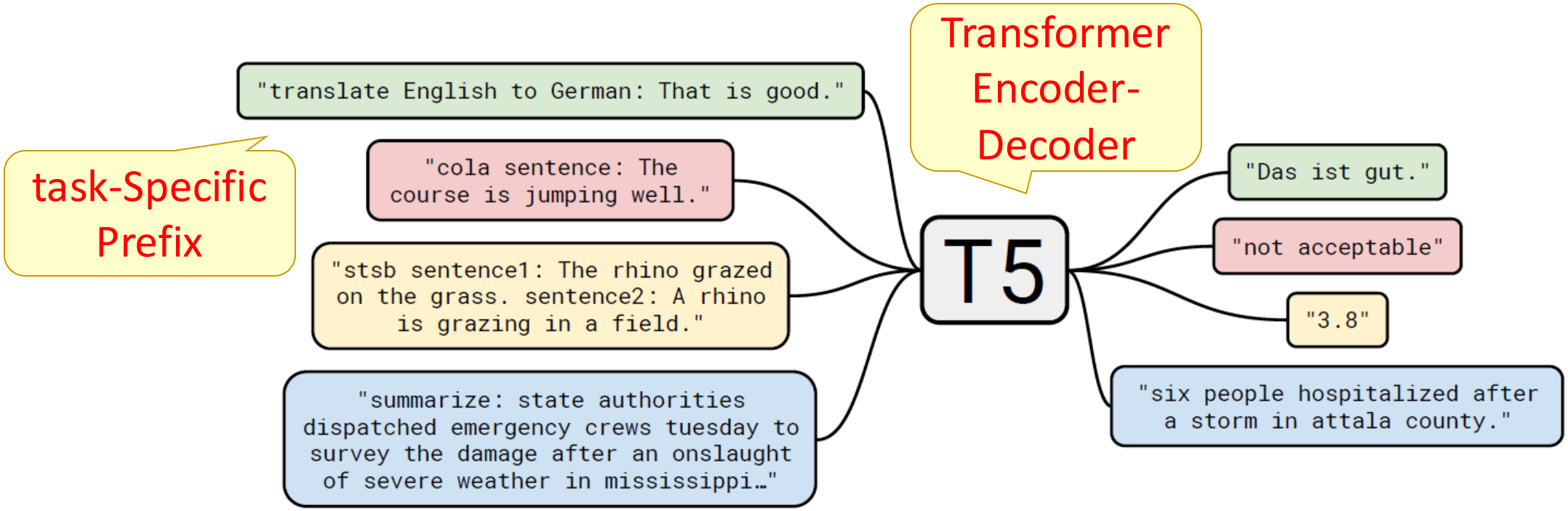


(b) For machine translation, it learns a small additional encoder that replaces the word embeddings in BART. The new encoder can use a disjoint vocabulary.

Fine-Tuning BART for Classification and Translation

Pre-Trained Language Models

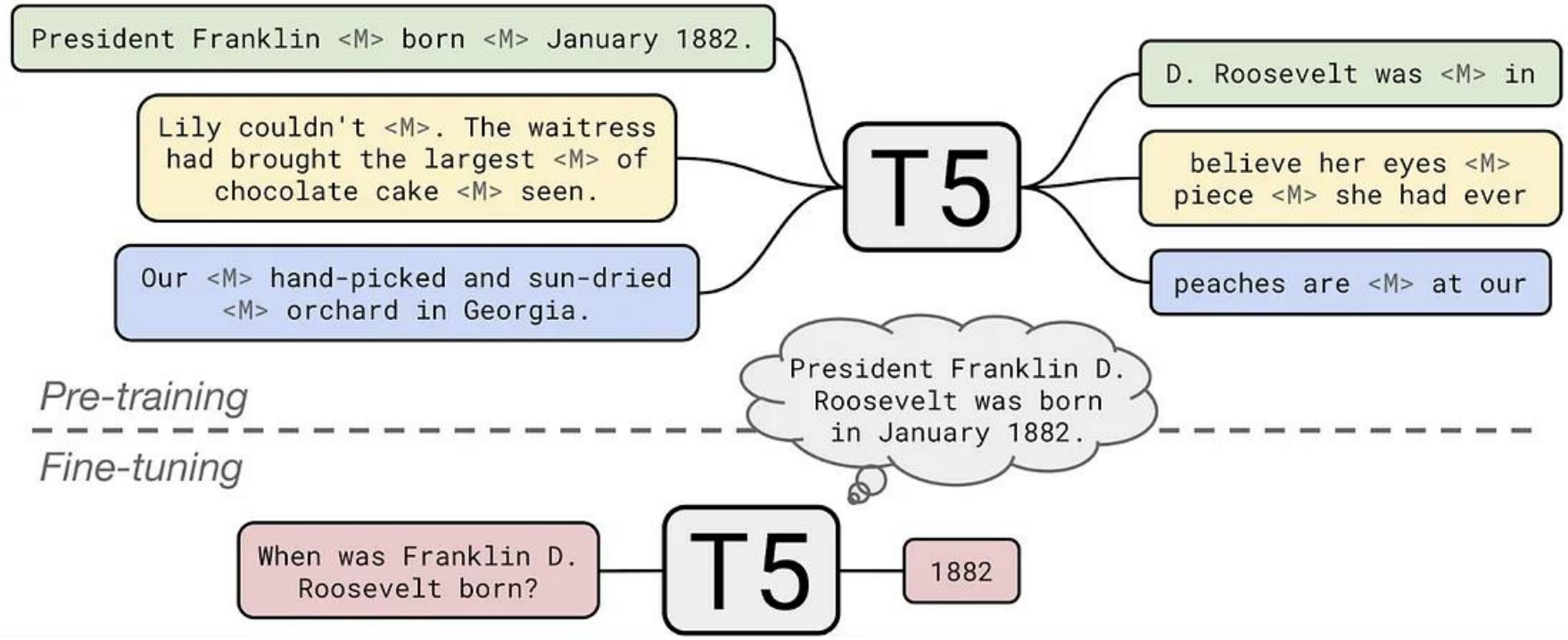
- **T5: Text-to-Text Transfer Transformer**



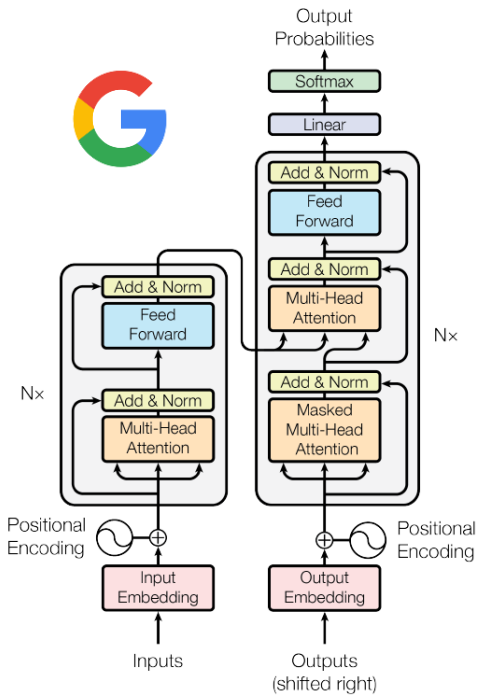
A Shared Text-To-Text Framework

Pre-Trained Language Models

- **T5: Text-to-Text Transfer Transformer**



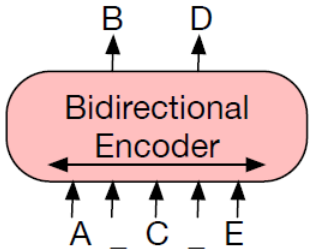
Large Language Models



Masked Language Model

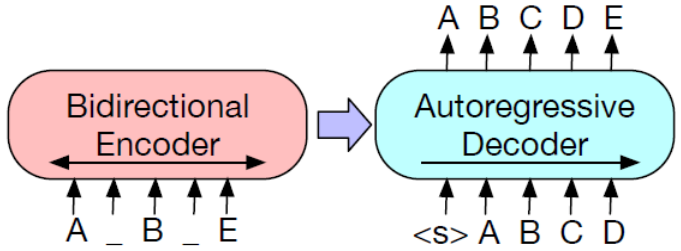
Google

Encoder-Only



Meta

Encoder-Decoder

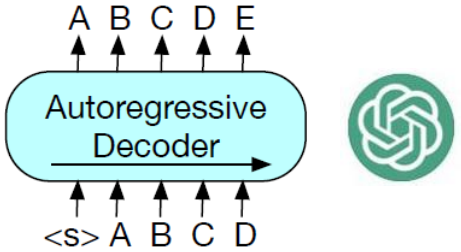


Meta

Decoder-Only

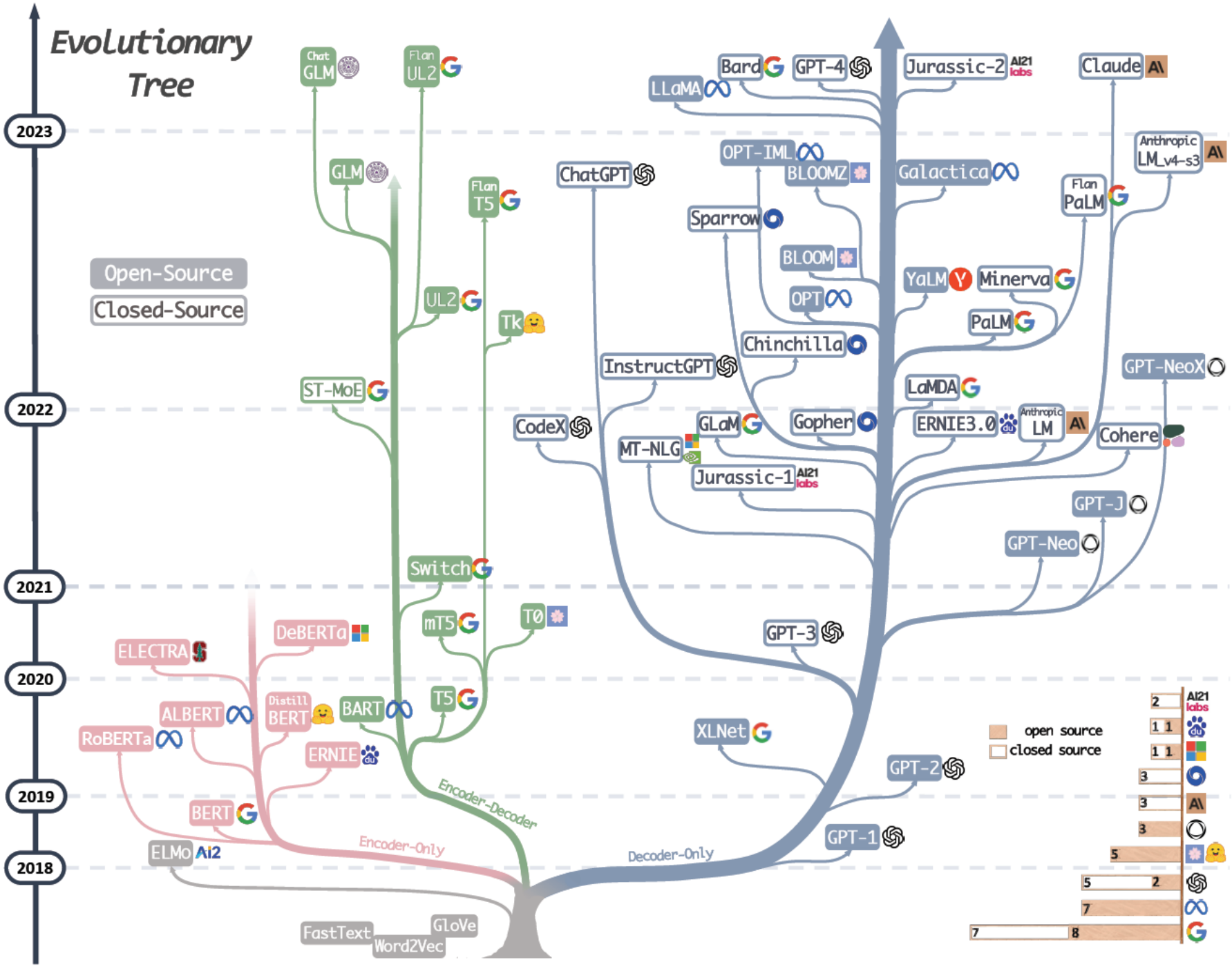
GPT

OpenAI



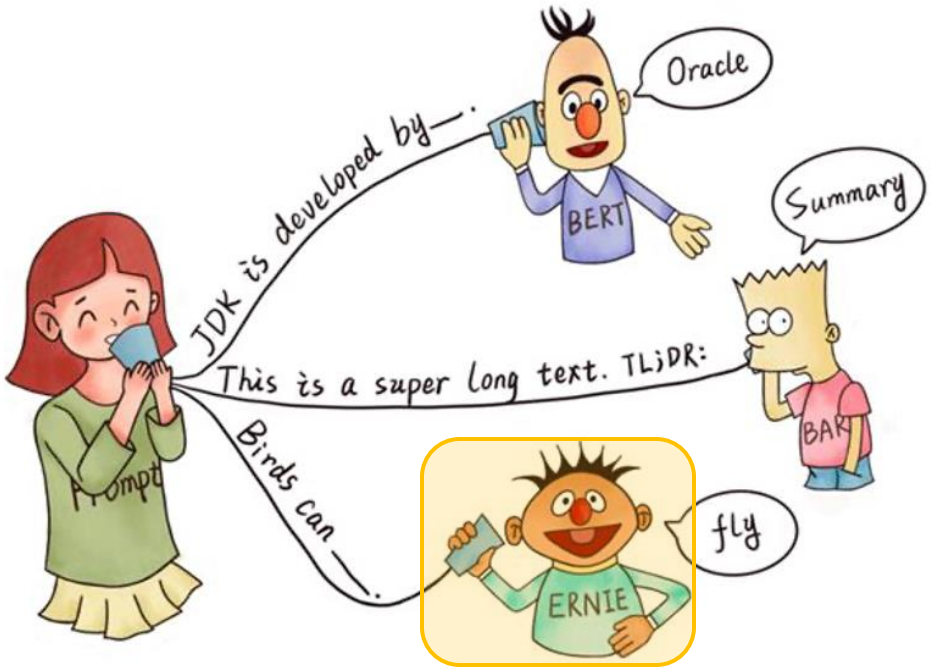
Auto-Regressive Language Model

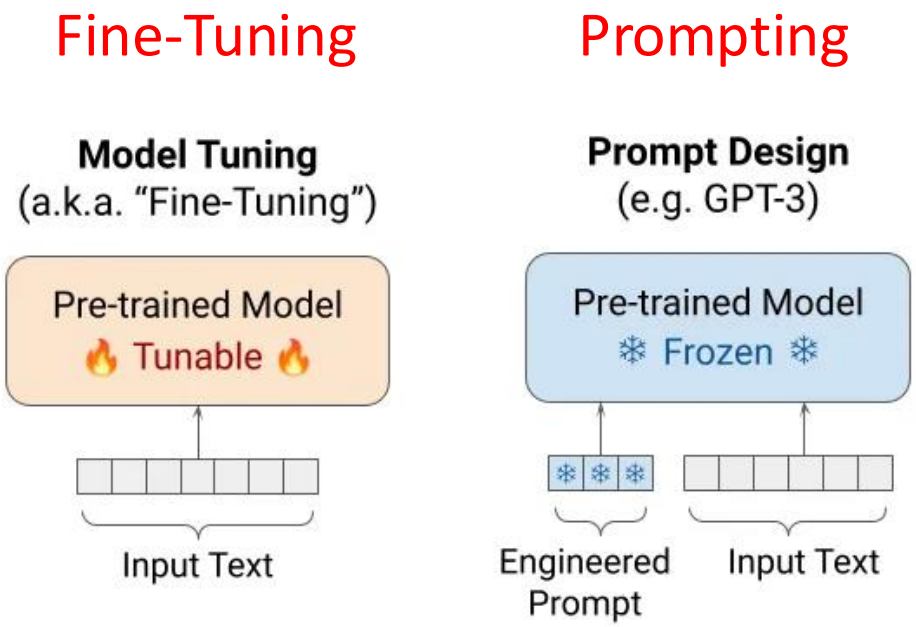
Transformers have epoch-making significance for the development of NLP.


Evolutionary Tree
2023
2022
2021
2020
2019
2018
Open-Source
Closed-Source
Chat GLM
Flan UL2
GLM
Flan T5
UL2
Tk
ST-MoE
Switch
mT5
T0
T5
BART
ELECTRA
DeBERTa
ALBERT
Distill BERT
RoBERTa
ERNIE
BERT
ELMo Ai2
FastText
Word2Vec
GloVe
Encoder-Only
Encoder-Decoder
Decoder-Only
LLaMA
Bard
GPT-4
Jurassic-2 AI21 labs
Claude
ChatGPT
OPT-IML
BLOOMZ
Galactica
Anthropic LM_v4-s3
Flan PaLM
Sparrow
BLOOM
YaLM
Minerva
OPT
PaLM
Chinchilla
InstructGPT
GPT-NeoX
LaMDA
CodeX
GLaM
Gopher
ERNIE3.0
Anthropic LM
Cohere
MT-NLG
Jurassic-1 AI21 labs
GPT-J
GPT-Neo
GPT-3
XLNet
GPT-2
GPT-1
open source
closed source
AI21 labs
2
1 1
1 1
3
3
3
5
5 2
7
7 8

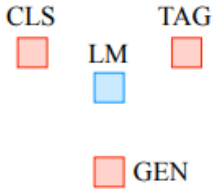
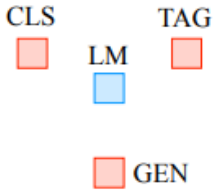
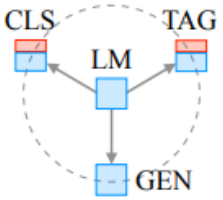
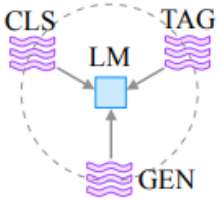
Large Language Models

- **From Pre-Training/Fine-Tuning to Pre-Training/Prompting**





Source: https://blog.research.google/2022/02/guiding-frozen-language-models-with.html

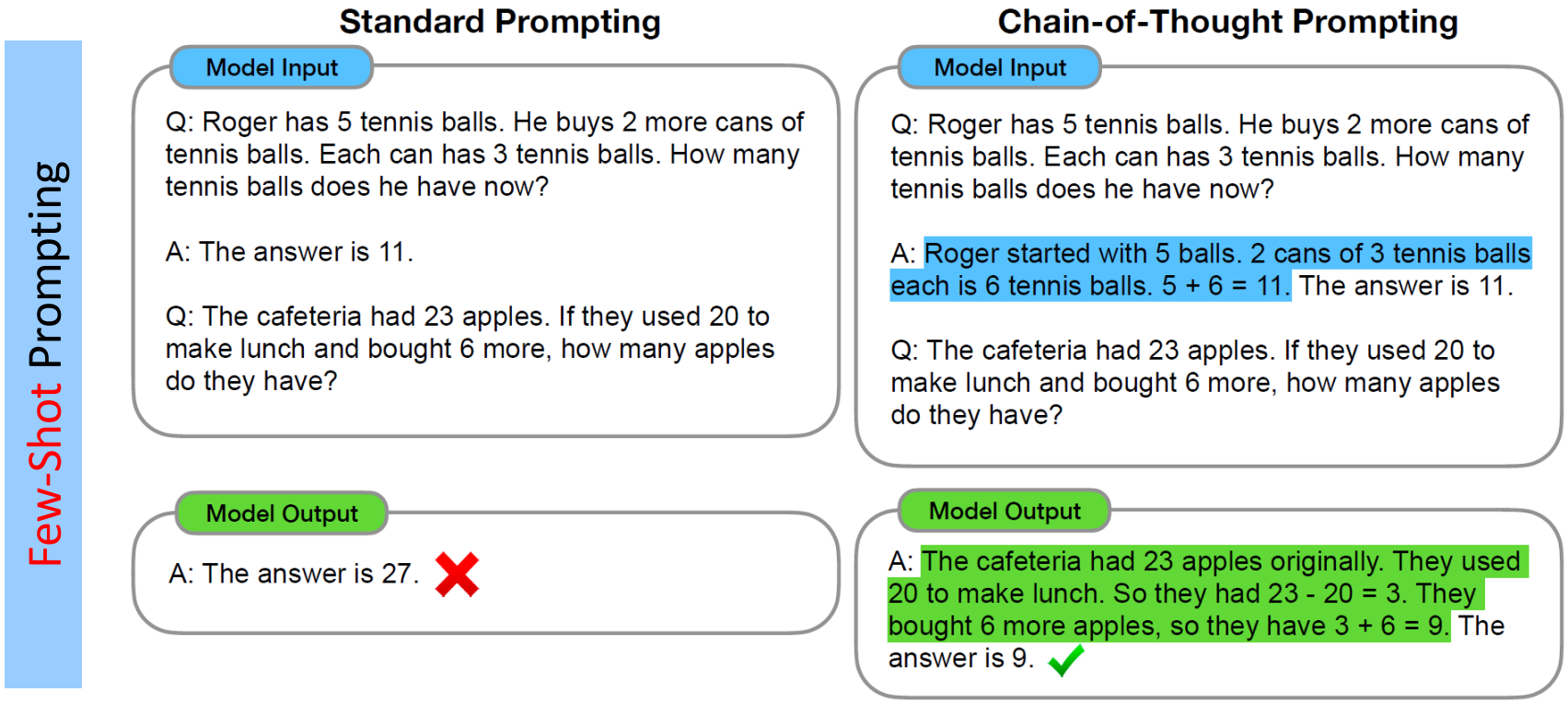
Paradigm	Engineering	Task Relation
a. Fully Supervised Learning (Non-Neural Network)	Features (e.g. word identity, part-of-speech, sentence length)	
b. Fully Supervised Learning (Neural Network)	Architecture (e.g. convolutional, recurrent, self-attentional)	
c. Pre-train, Fine-tune	Objective (e.g. masked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

?

Teaching LLMs to think

Large Language Models

Chain-of-Thought (Few-Shot) Prompting



Chain-of-Thought prompting elicits reasoning in large language models.

Large Language Models

Chain-of-Thought (Zero-Shot) Prompting

Zero-Shot CoT essentially involves adding "Let's think step by step" to the original prompt. This is particularly useful where you don't have too many examples to use in the prompt.

Zero-Shot Prompting

(i) Zero-Shot

Q: There were 10 friends playing a video game online when 7 players quit. If each player left had 8 lives, how many lives did they have total?

A: The answer is

(Output) 80. ✗

(ii) Zero-Shot-CoT

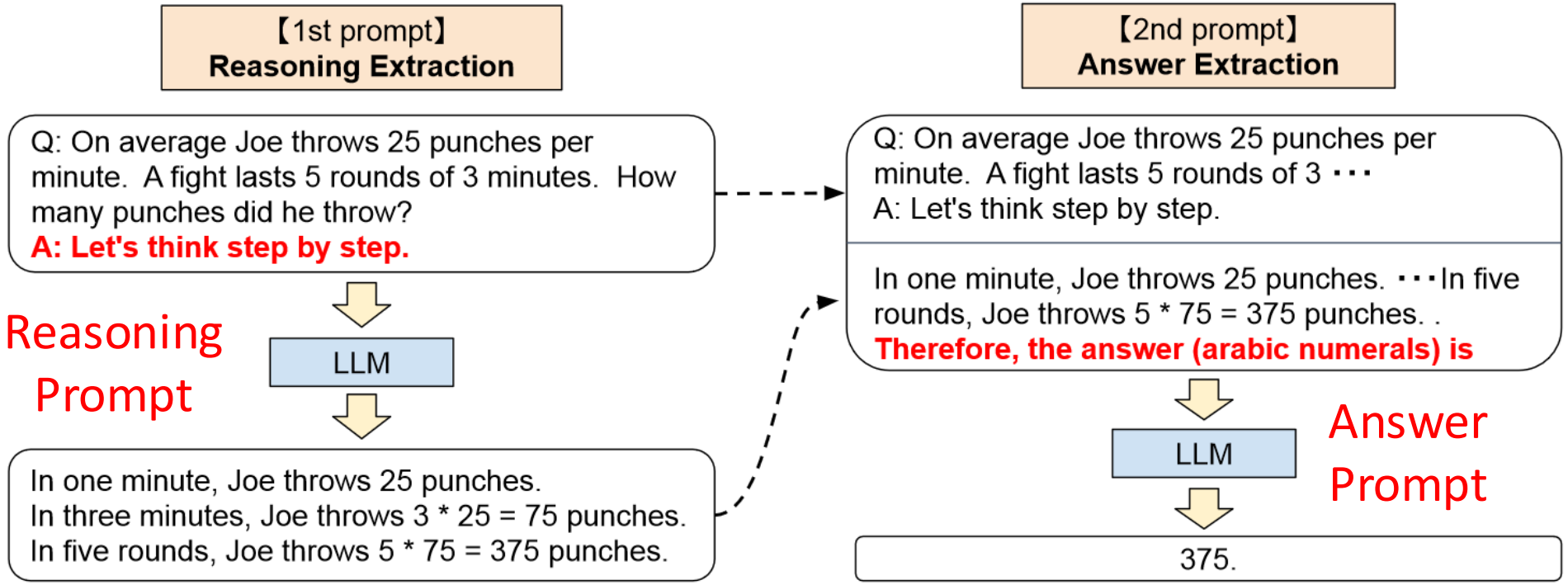
Q: There were 10 friends playing a video game online when 7 players quit. If each player left had 8 lives, how many lives did they have total?

A: Let's think step by step.

(Output) There were 10 friends playing a video game online. This means that, at the start, there were 10 x 8 = 80 lives in total. Then, 7 players quit. This means that 7 x 8 = 56 lives were lost. Therefore, the total number of lives remaining is 80 - 56 = 24. The answer is 24. ✓

Large Language Models

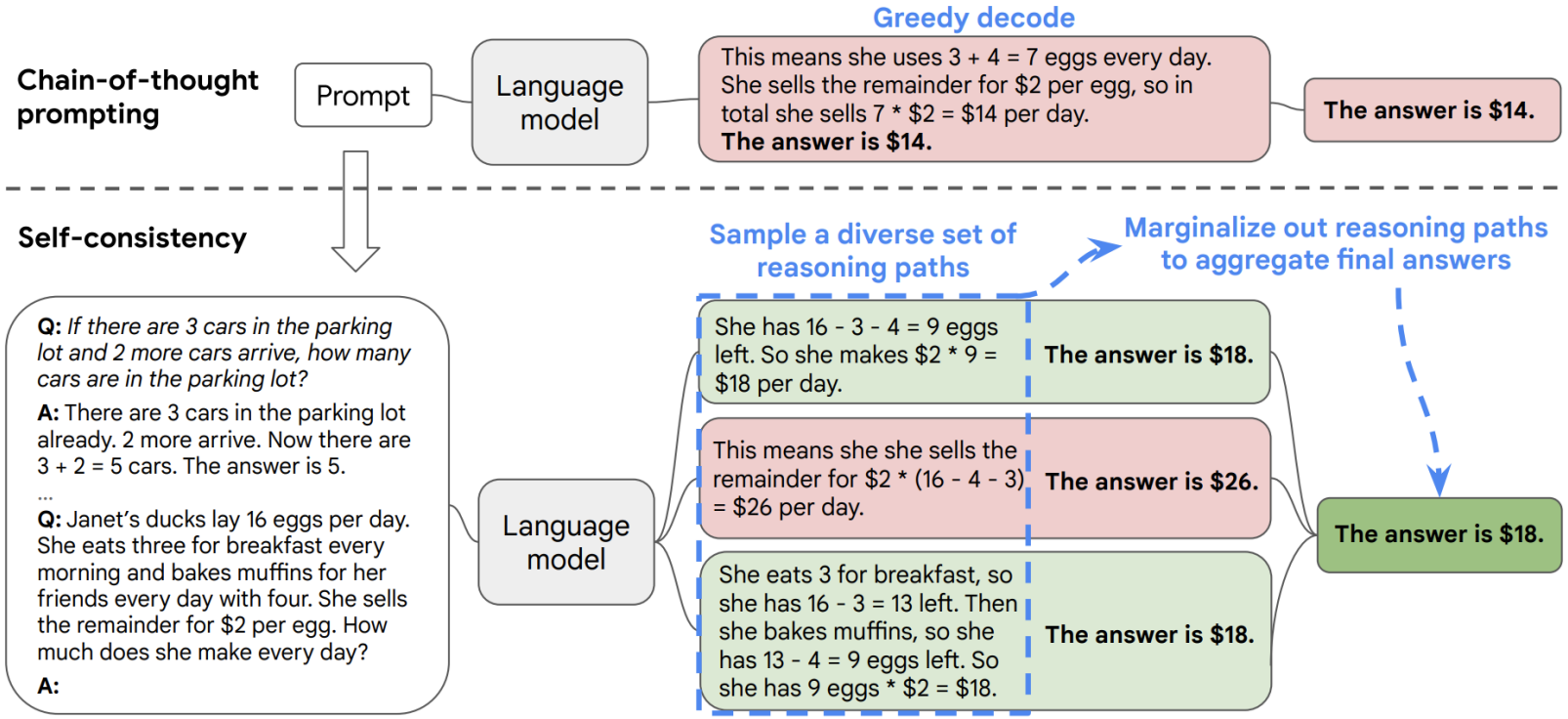
Chain-of-Thought (Zero-Shot) Prompting



Two-Stage Prompting

Large Language Models

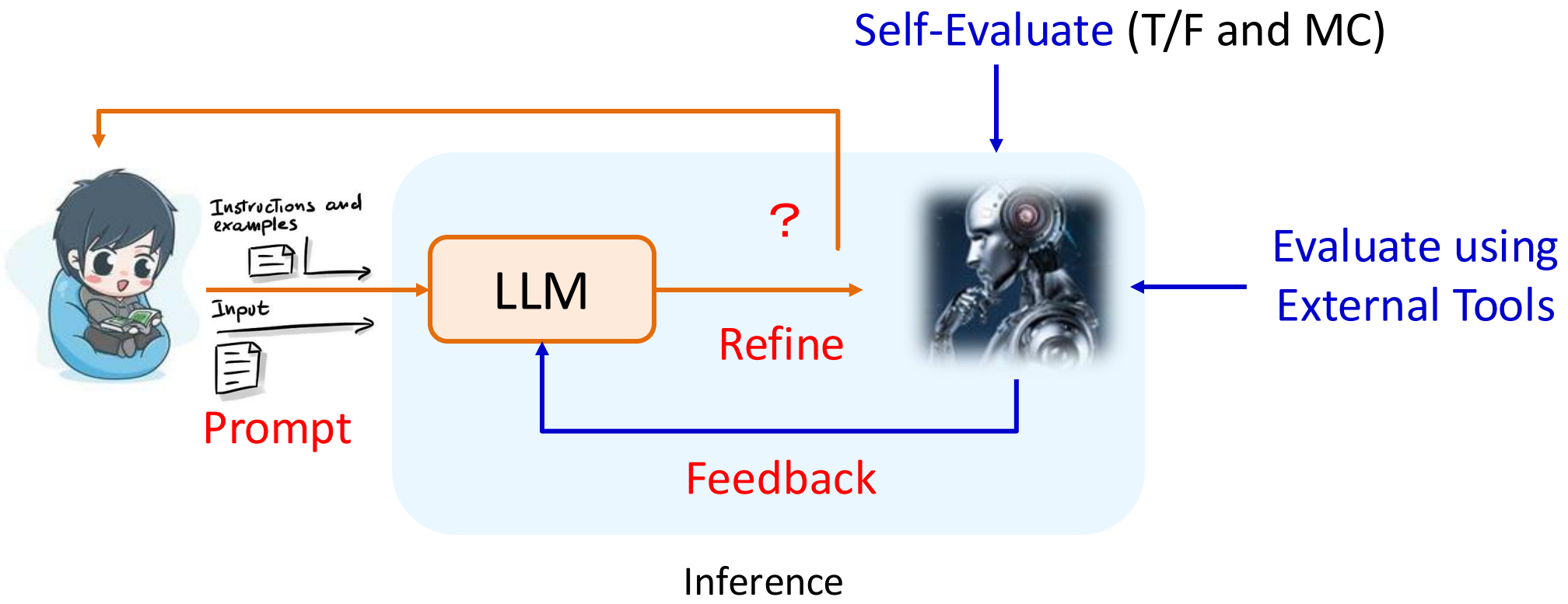
■ CoT with Self-Consistency



The idea is to sample multiple, diverse reasoning paths through few-shot CoT, and use the generations to select the most consistent answer. This helps to boost the performance of CoT prompting on tasks involving arithmetic and commonsense reasoning.

Large Language Models

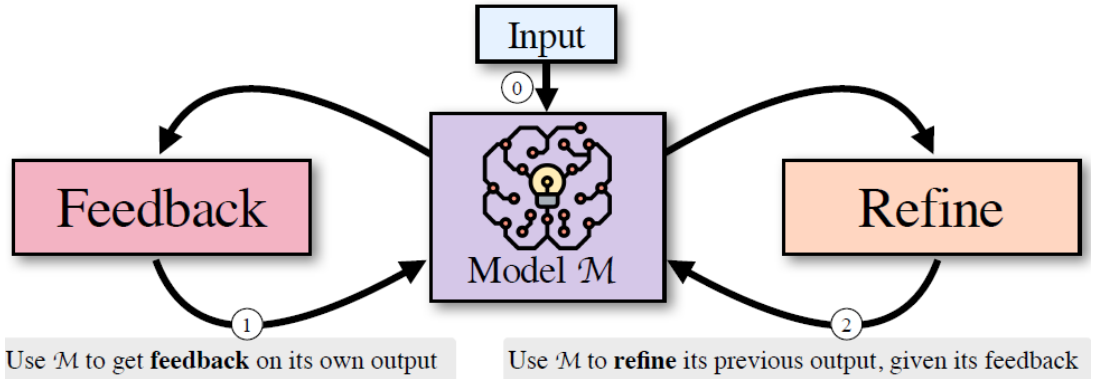
- **Refinement/Correction with Feedback**



Large Language Models

Refinement with Self-Feedback

Self-Refine allows LLMs to iteratively refine outputs and incorporate feedback to improve performance on diverse tasks.



Sentiment Transfer: x, $\mathbf{y_t}$

```
If you ever wondered where
the magic of Vegas crawled
into a hole to rot, look no
further than the Trop. Write
with positive sentiment.

If you're looking for budget
friendly option in Vegas,
Trop maybe worth considering
```

FEEDBACK $\mathbf{fb}$

```
Is the sentiment of
this review Positive?
If not, how can it be
improved?

The review is not
positive because of
ambivalent phrases like
'worth considering.'
```

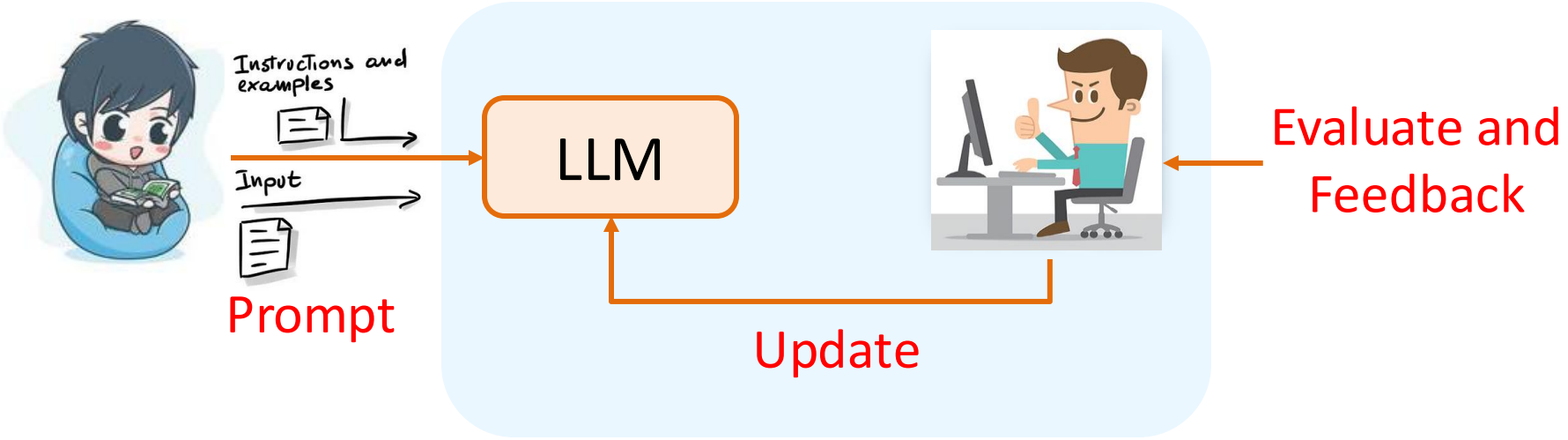
REFINE $\mathbf{y_{t+1}}$

```
If you're looking
for a unique and
affordable
experience in
Vegas, the Trop
may be the perfect
place for you.
```

Few-Shot Prompting

Large Language Models

- **From Reinforcement Learning from Human Feedback (RLHF) to Reinforcement Learning from AI Feedback (RLAIF)**
  - Automating RLHF with AI Feedback

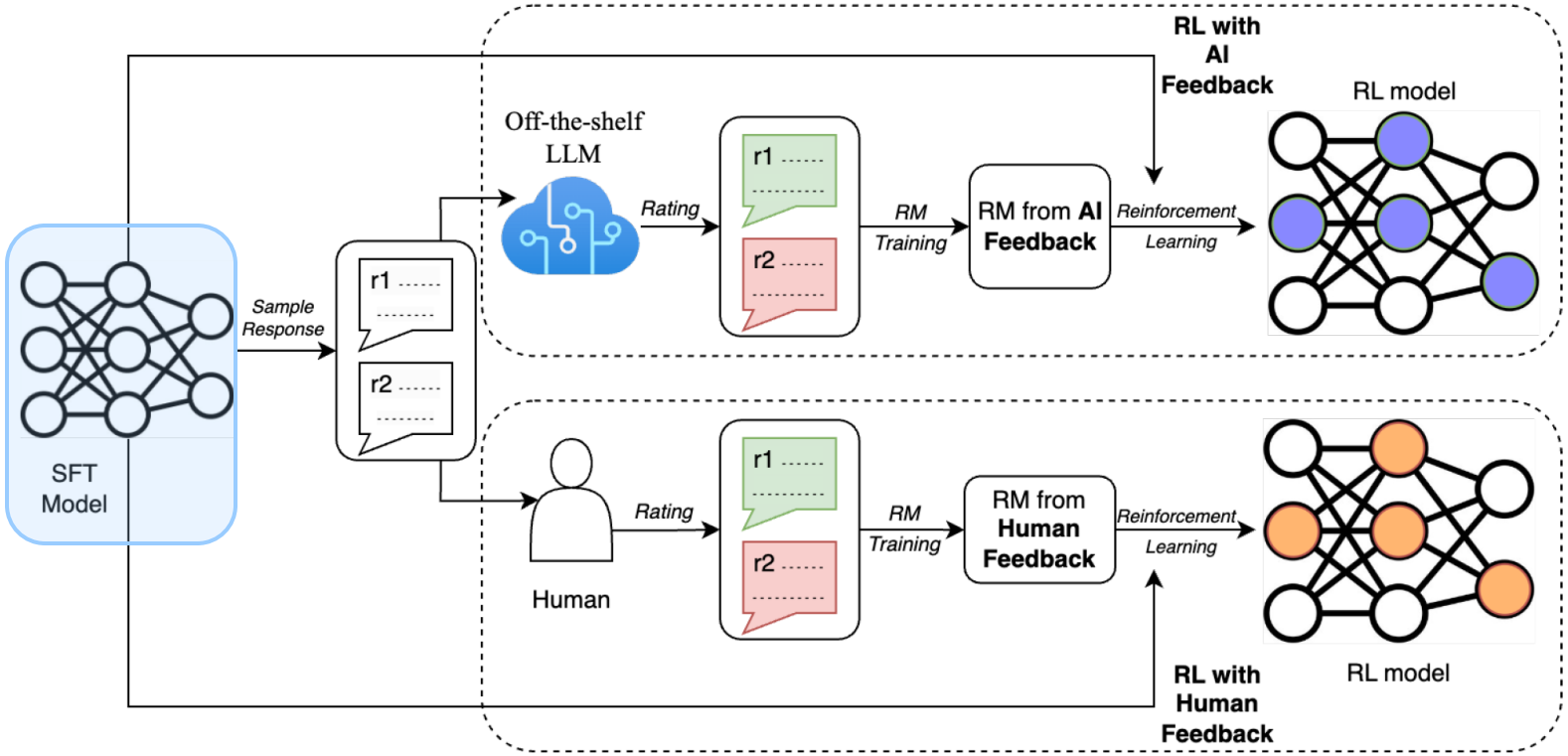


Training

As AI systems become more capable, we would like to enlist their help to supervise other AIs.

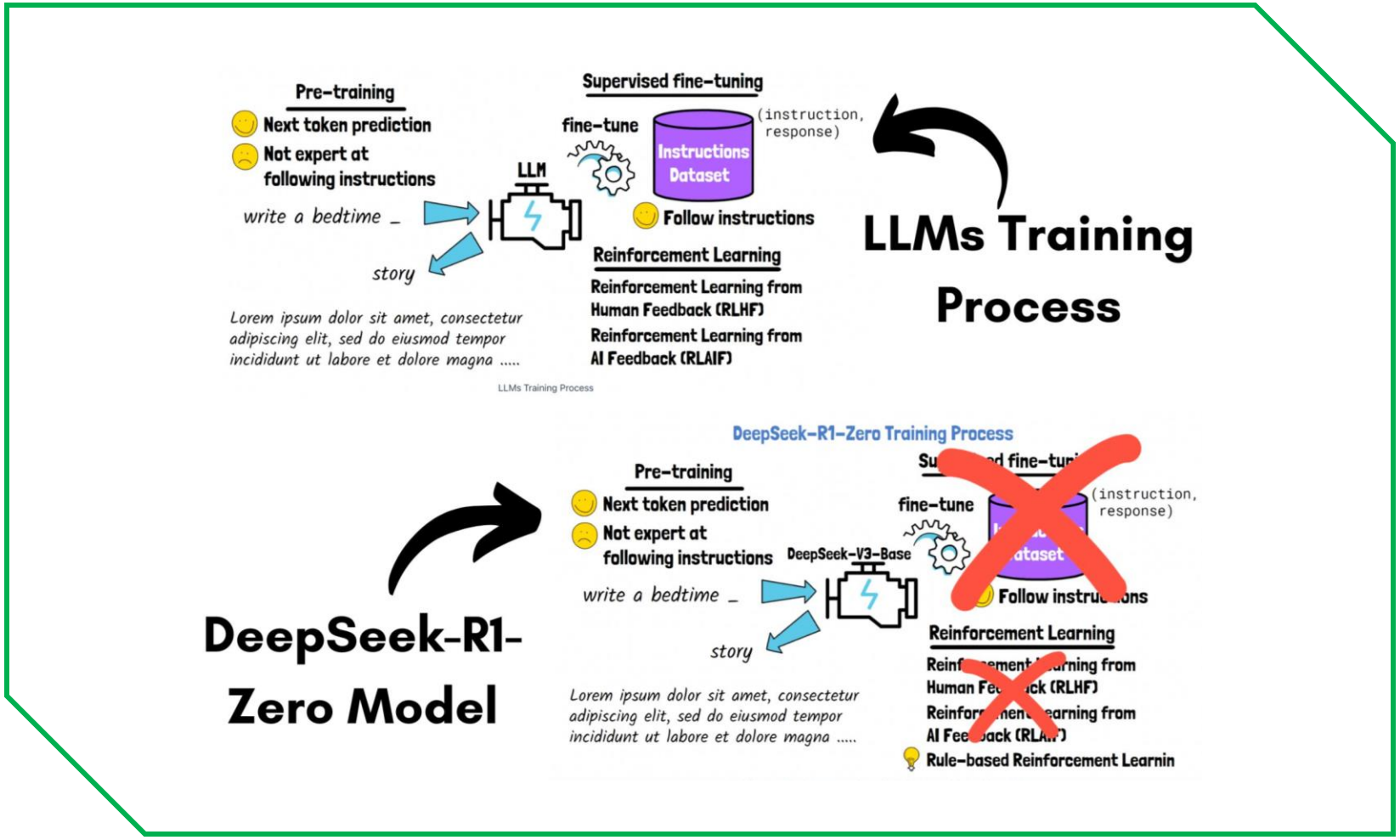
Large Language Models

- **From Reinforcement Learning from Human Feedback (RLHF) to Reinforcement Learning from AI Feedback (RLAIF)**



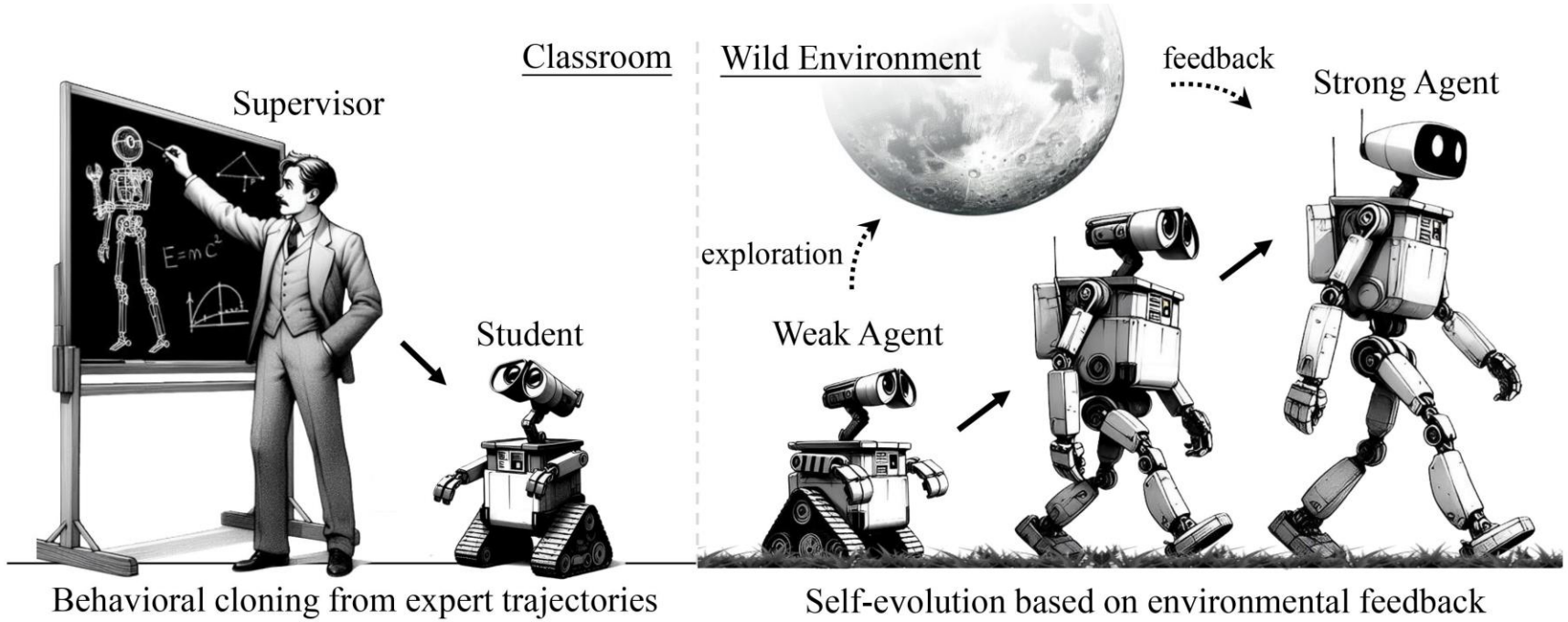
RM: Reward Model

Large Language Models



Large Language Models

- **Self-Improvement/Self-Evolution of Large Language Models**



References

- **YouTube Videos**

References

- **Book Chapters**
  - Speech and Language Processing
    - Chapter 10: Large Language Models
    - Chapter 11: Masked Language Models
    - Chapter 12: Model Alignment, Prompting, and In-Context Learning

References

- **Reference Papers**
  - BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, NAACL-HLT'2019. (BERT)
  - Improving Language Understanding by Generative Pre-Training, 2018. (GPT)
  - Language Models are Unsupervised Multitask Learners, 2020. (GPT-2)
  - Language Models are Few-Shot Learners, arXiv'2020. (GPT-3)
  - Training Language Models to Follow Instructions with Human Feedback, arXiv'2022. (InstructGPT)

References

- **Reference Papers**
  - BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, ACL'2020. (BART)
  - Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, NeurIPS'2022. (CoT)
  - Large Language Models are Zero-Shot Reasoners, NeurIPS'2022. (Zero-Shot CoT)

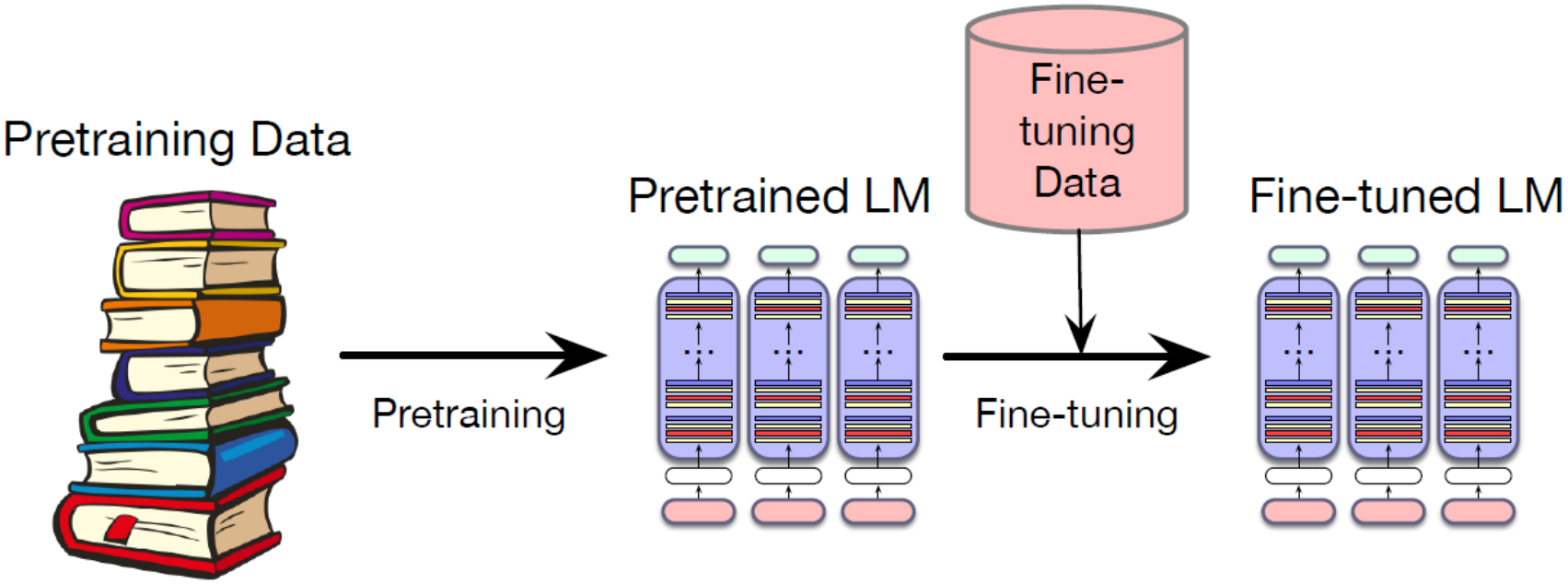
References

- **Pre-Trained Language Model Surveys**
  - Recent Advances in Natural Language Processing via Large Pre-Trained Language Models: A Survey, arXiv'2021.
  - Pre-trained Models for Natural Language Processing: A Survey, arXiv'2021.
  - Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing, ACM Computing Survey, 2022.

thank you

Pre-Trained Language Models

- **Pre-Training and Fine-Tuning Paradigm**



Self-supervised Pre-training Supervised Fine-Turning

Large Language Models

- **Frequently Used Terms for Large Language Models**
  - Pre-Training and Fine-Tuning
  - In Context Learning (ICL)
  - Prompting
  - Instruction Tuning (Instruction Finetuning, Supervised Finetuning)
  - (Instruction, Value, Safety, Fairness, Privacy, etc.) Alignment