Large Language Models

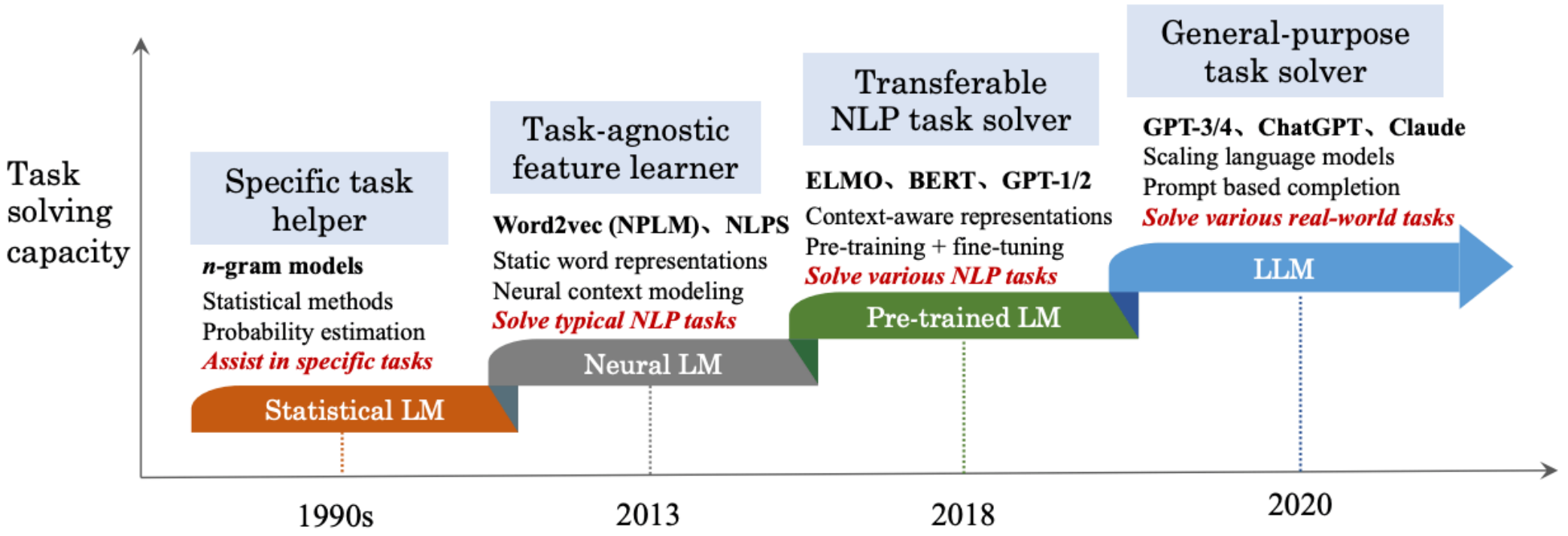
Jian Tang

HEC Montreal

Mila-Quebec AI Institute

Email: jian.tang@hec.ca

History of NLP Techniques



Natural Language Modeling

- Given a text sequence $X = x_1 x_2 \ldots x_T$, we want to model the joint distribution of

$$P(X)$$

- X can be a sentence or a document

Two ways of Modeling $P(X)$

- Masked Language Modeling: predicting the missing word(s) conditioning on the rest of the words, i.e. $P(x_i|X_{-i})$
  - Filling in the blank

$$x_1 \quad x_2 \quad x_3 \quad \ldots \quad x_{i-1} \quad ? \quad x_{i+1} \quad \ldots \quad x_T$$

- Bidirectional modeling

Two ways of Modeling $P(X)$

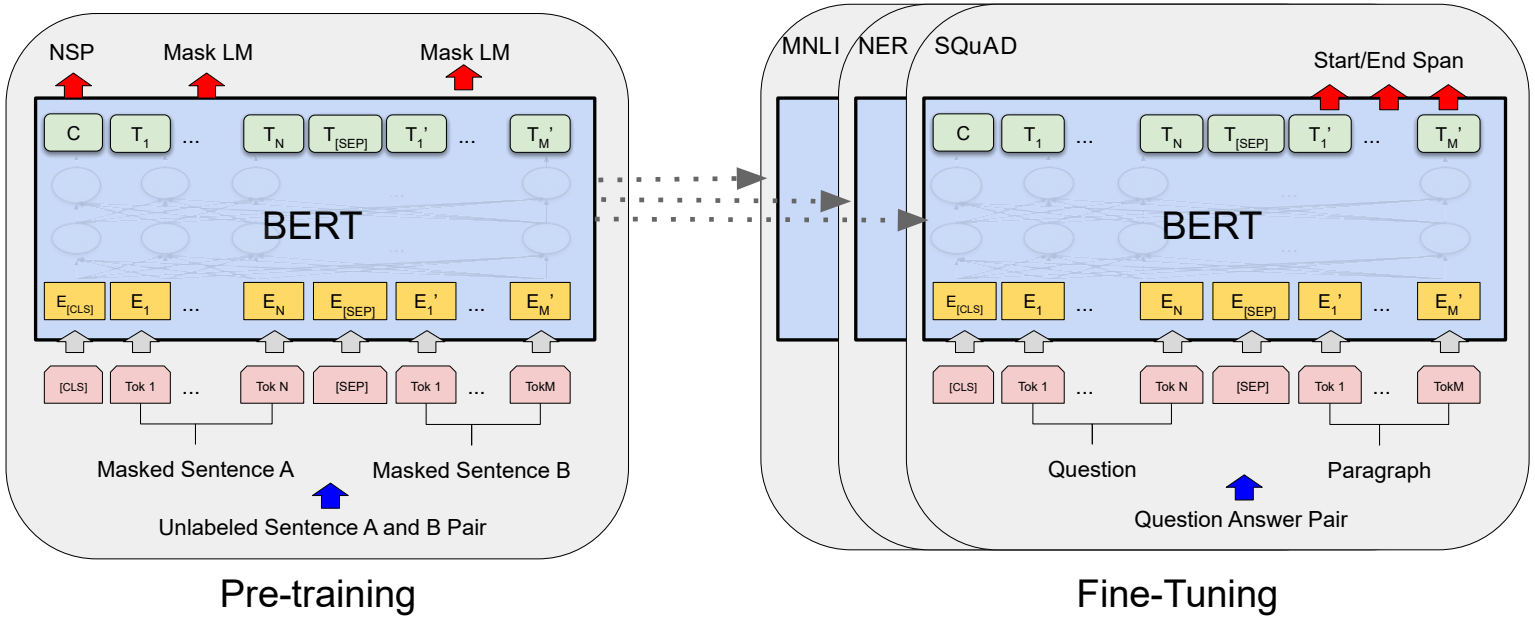
- Next Token Prediction: predicting the next token/word conditioning on the preceding tokens, i.e. $P(x_i|X_{<i})$

$$x_1 \quad x_2 \quad x_3 \quad \ldots \quad x_{\mathrm{i}-1} \quad ?$$

- Unidirectional modeling

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

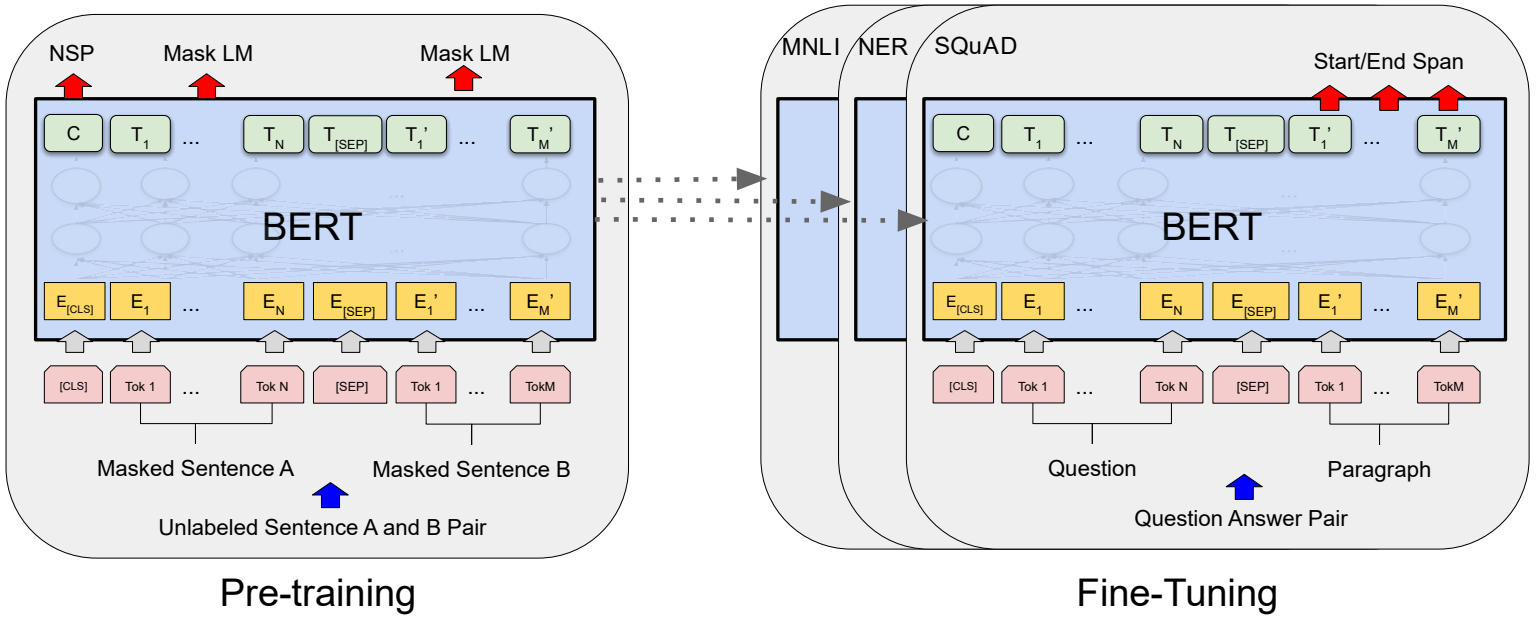
BERT: Pre-training and Fine-Tuning Framework



- **Pre-train** the model on unlabeled data over pre-training tasks
- Initializing the model with the pre-trained parameters, and **fine-tune** all the parameters using labeled data from downstream tasks
- Unified architecture across different tasks

Model Architecture

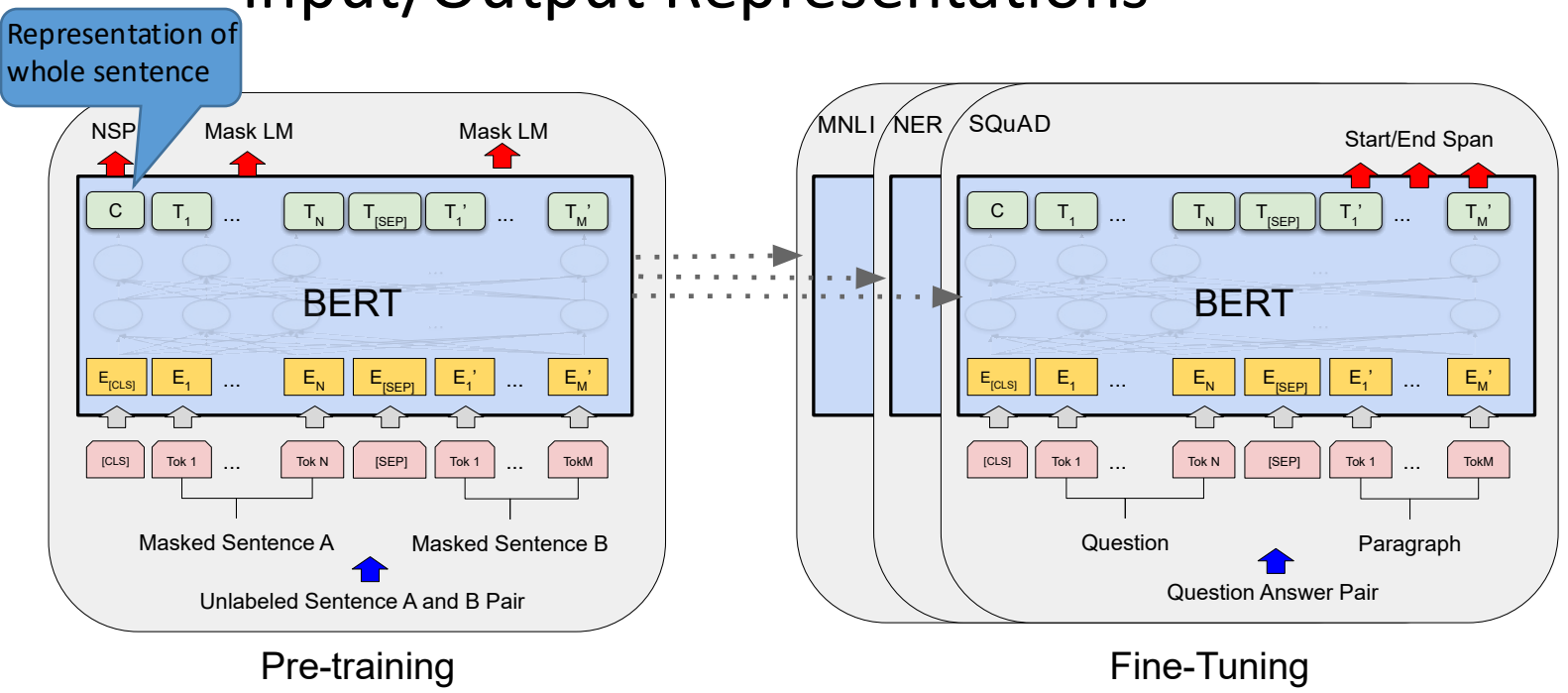
- A multi-layer bidirectional Transformer encoder



- **BERT_BASE**: Number of layers: 12, Hidden Size: 768, the number of self-attention heads: 12, total number of parameters: 110M
- **BERT_LARGE**: Number of layers: 23, Hidden Size: 1024, the number of self-attention heads: 16, total number of parameters: 340M

Model Architecture

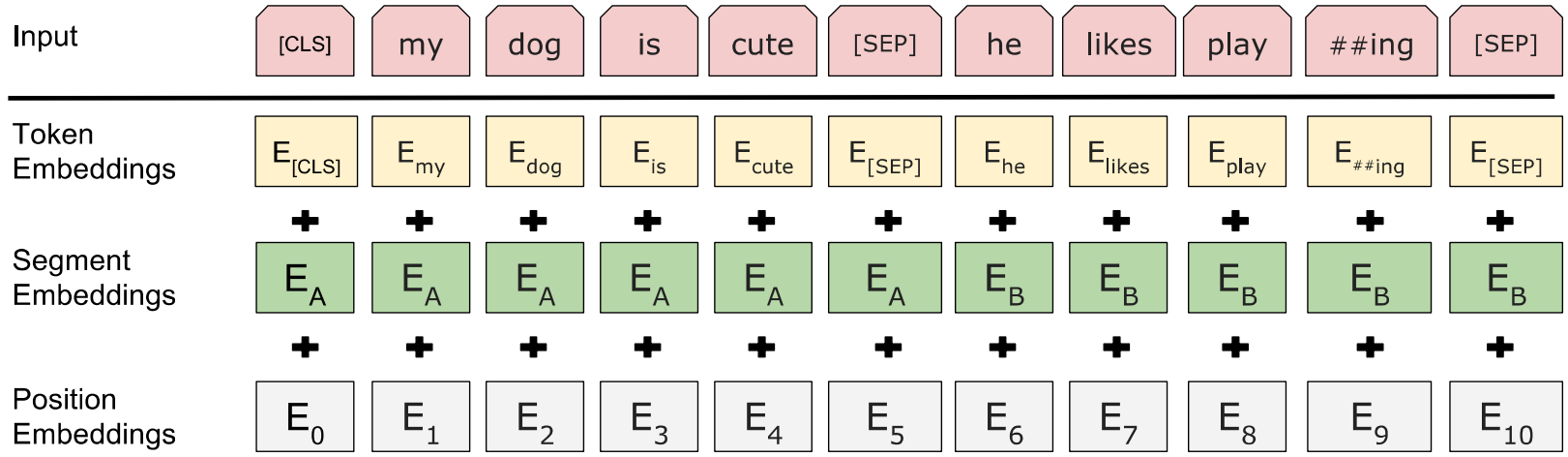
- Input/Output Representations



- Both a single sentence and a pair of sentences are represented in one token sequence
  - Start with a special token [CLS]
  - Separate sentences with a special token [SEP]
  - Add a learned embeddings to every token indicating whether it belongs to sentence A or sentence B

Input Representation

- Each token embedding is the summation of **token embedding**, **segment embedding**, and **position embeddings.**



Pre-training BERT

- **Task #1: Masked LM**
- Mask some percentages of the input tokens at random, and then predict those masked tokens.
  - 15% of the tokens in each sequence are masked out at random

Pre-training BERT

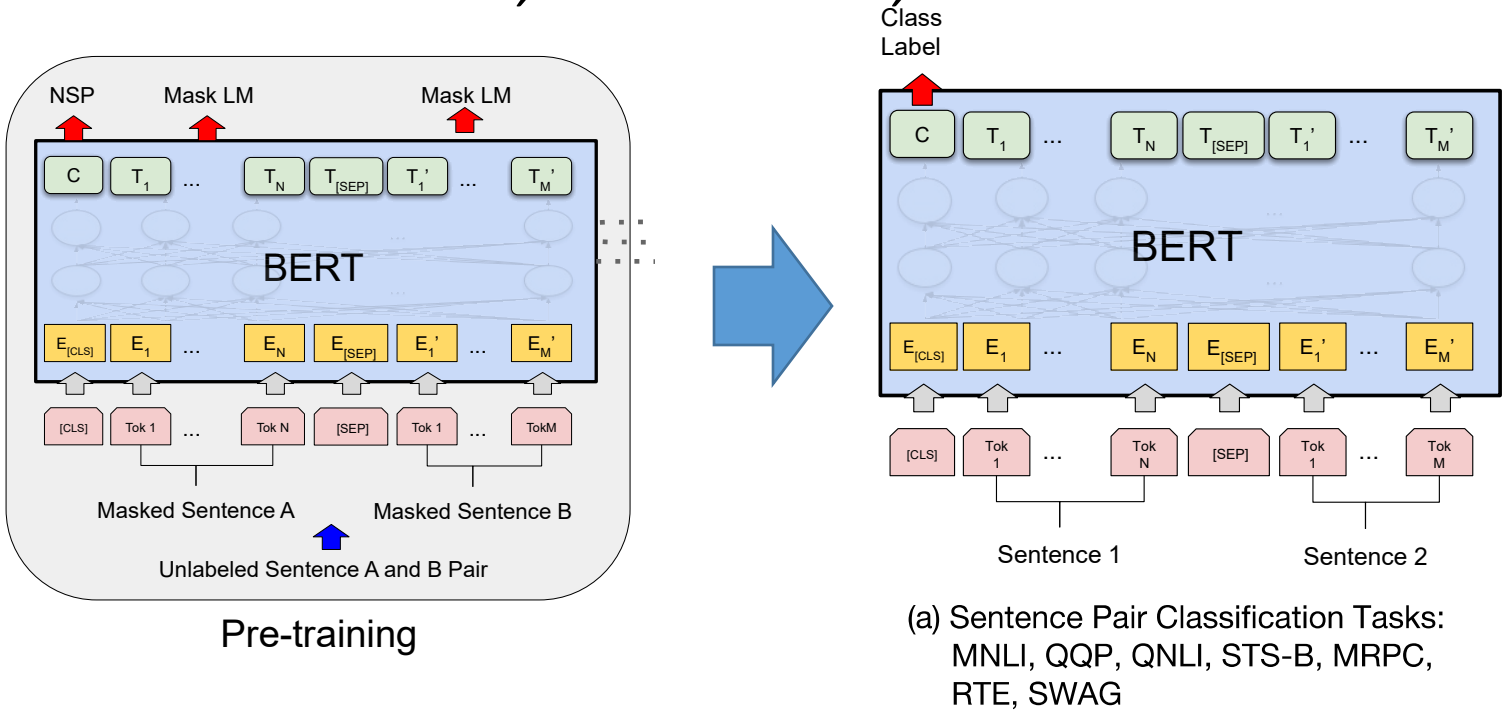
- **Task #2: Next Sentence Prediction (NSP)**
  - Important in many downstream tasks such as question answering (QA) and natural language inference (NLI)
- 50% sentence pairs (A,B) are positive
  - Actually sentence B that follows A
- 50% sentence pairs (A, B) are negative
  - Randomly select a sentence B from the training corpus

Pre-training Data

- BookCorpus (800M words)
- English Wikipedia(2,500M words)

Fine-Tuning BERT

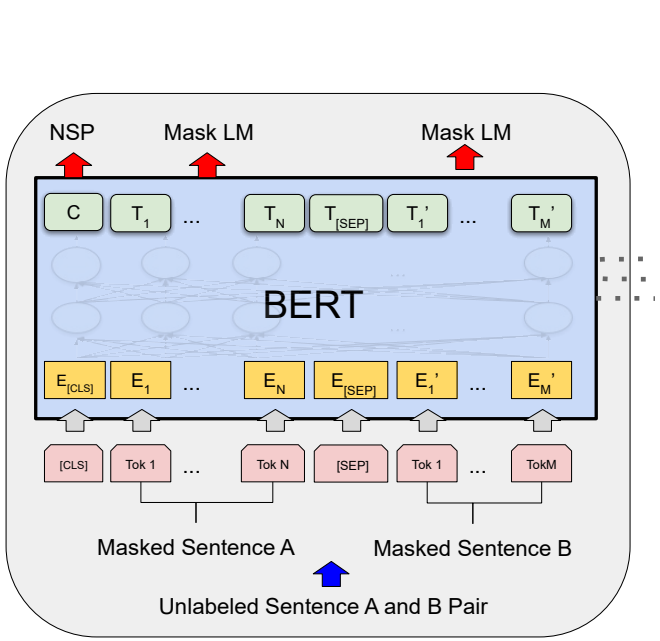
- Sentence Pair Classification Tasks, e.g., nature language inference
  - Given a pair of sentences, the goal is to predict whether the second sentence is an *entailment, contradiction, or neutral* w.r.t. the first one



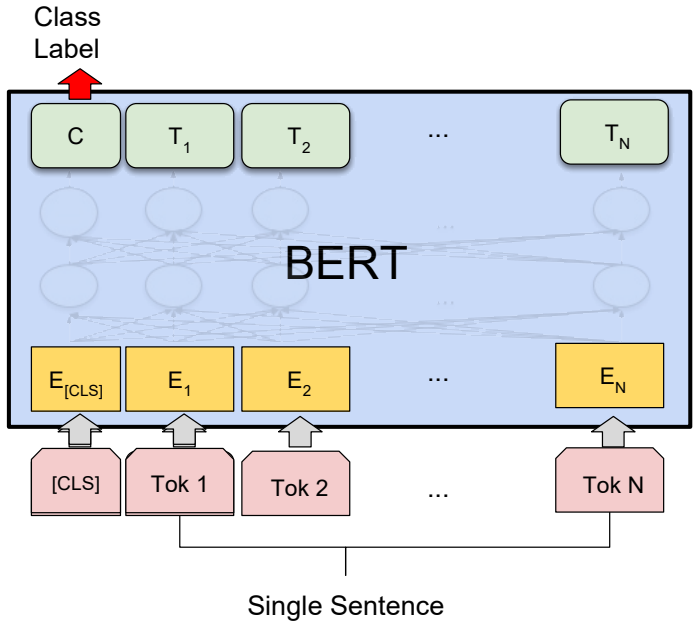
(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

Fine-Tuning BERT

- Sentence Classification



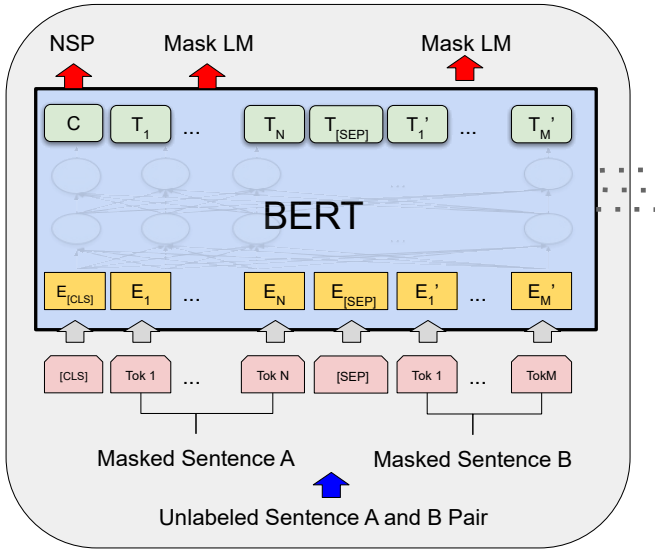
Pre-training



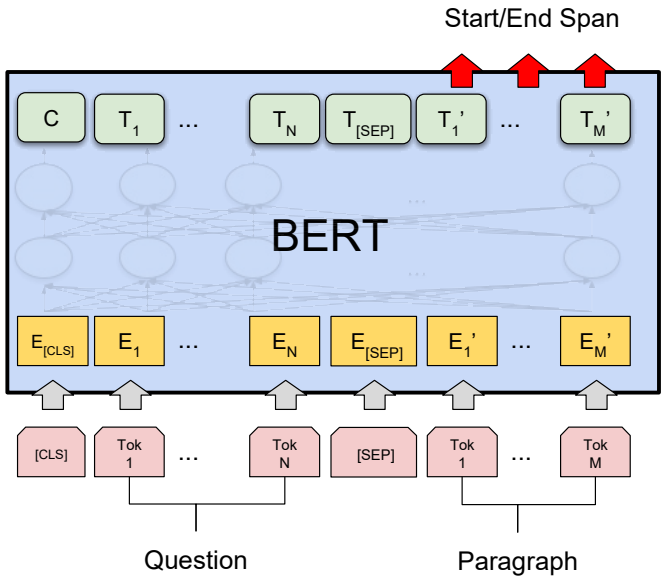
(b) Single Sentence Classification Tasks:
SST-2, CoLA

Fine-Tuning BERT

- Question Answering



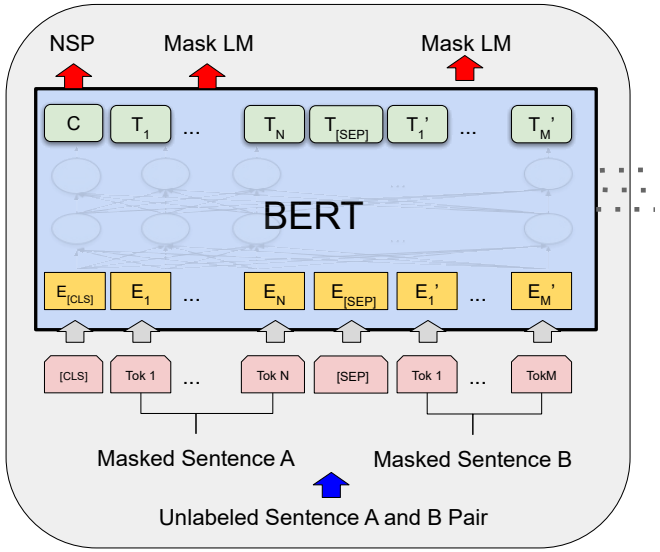
Pre-training



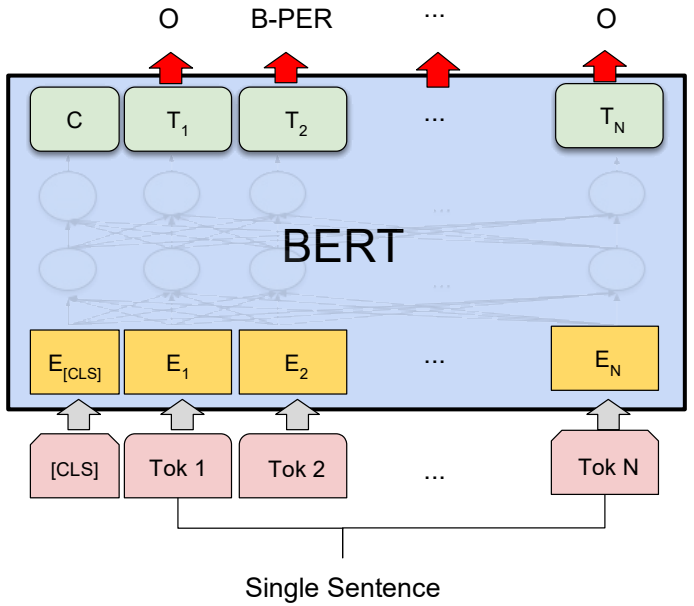
(c) Question Answering Tasks:
SQuAD v1.1

Fine-Tuning BERT

- Sentence Tagging



Pre-training



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

Experimental Results

- GLUE: General Language Understanding Evaluation Benchmark

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**

Ablation Study

- LTR: left-to-right, unidirectional language modeling

Tasks	Dev Set MNLI-m (Acc)	QNLI (Acc)	MRPC (Acc)	SST-2 (Acc)	SQuAD (F1)
$\text{BERT}_{\text{BASE}}$	84.4	88.4	86.7	92.7	88.5
No NSP	83.9	84.9	86.5	92.6	87.9
LTR & No NSP	82.1	84.3	77.5	92.1	77.8
+ BiLSTM	82.1	84.1	75.7	91.6	84.9

Effect of Model Size

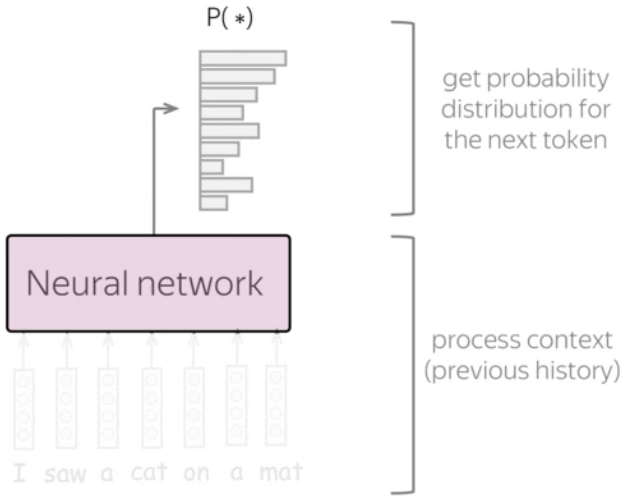
- L: number of layers
- H: hidden dimension
- A: number of self-attention heads

Hyperparams			Dev Set Accuracy			
#L	#H	#A	LM (ppl)	MNLI-m	MRPC	SST-2
3	768	12	5.84	77.9	79.8	88.4
6	768	3	5.24	80.6	82.2	90.7
6	768	12	4.68	81.9	84.8	91.3
12	768	12	3.99	84.4	86.7	92.9
12	1024	16	3.54	85.7	86.9	93.3
24	1024	16	3.23	86.6	87.8	93.7

Large Language Models

- Given a sequence $X = x_1 x_2 \ \ldots x_T$, Next Token Prediction: predicting the next token/word conditioning on the preceding tokens, i.e. $P(x_i | X_{<i})$

$$O = 1/T \sum_{i=1}^{T} \log P(x_i | X_{<i})$$



- In practice, we will specify a maximum context window

https://lena-voita.github.io/nlp_course/language_modeling.html

Next Token Prediction

- Essentially learning a mapping function f: context -> word
  - a classification problem

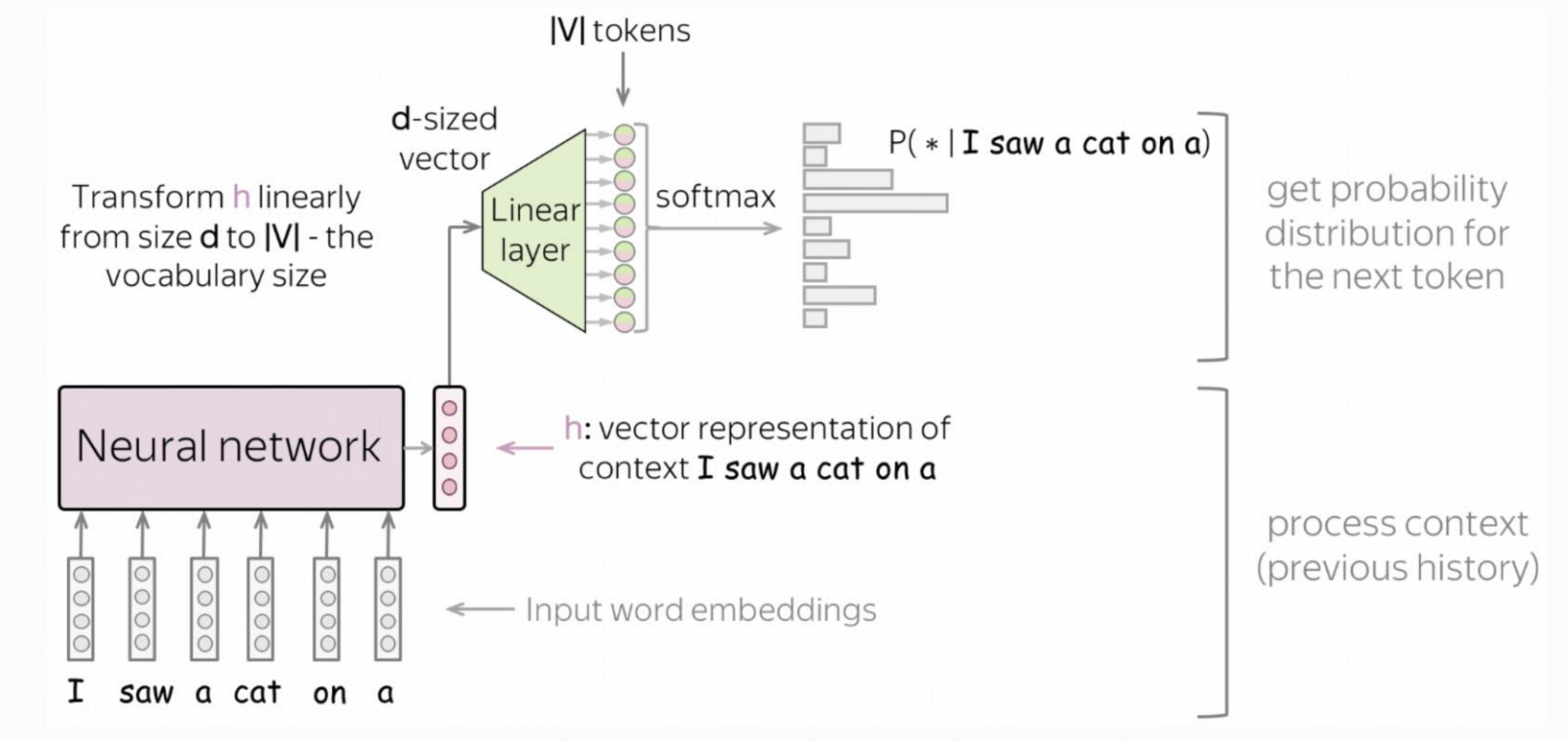


Image Credit: https://lena-voita.github.io/nlp_course/language_modeling.html

History of Generative LLMs

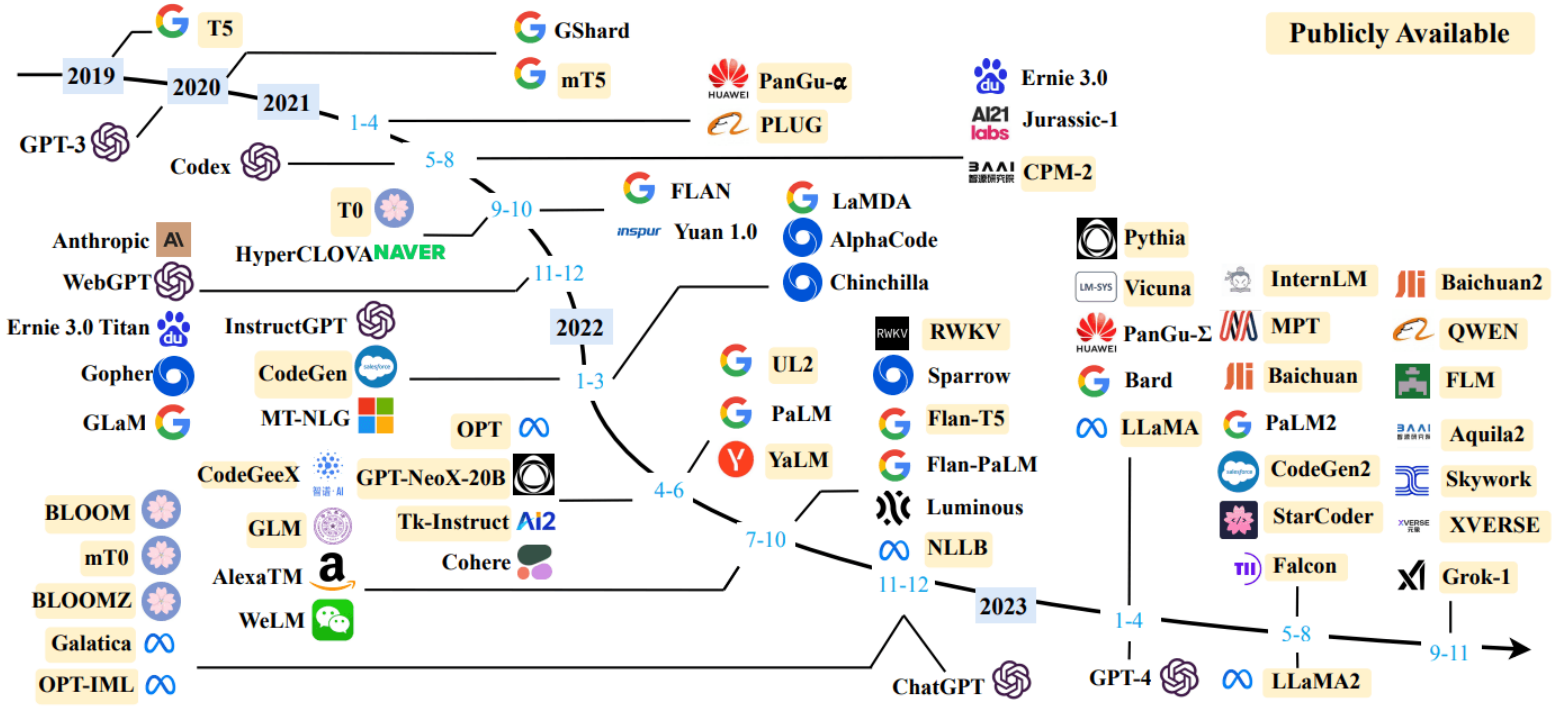
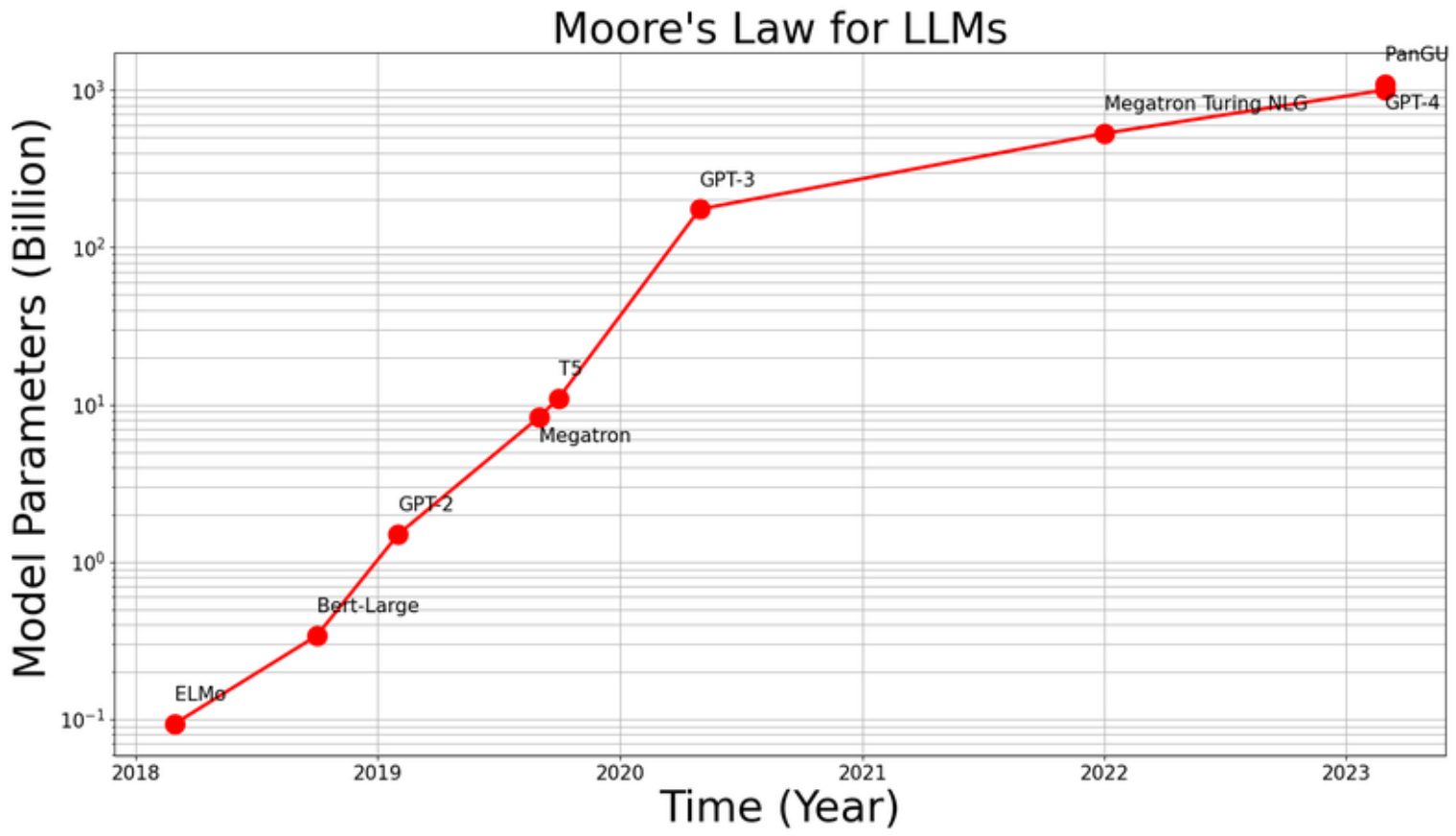


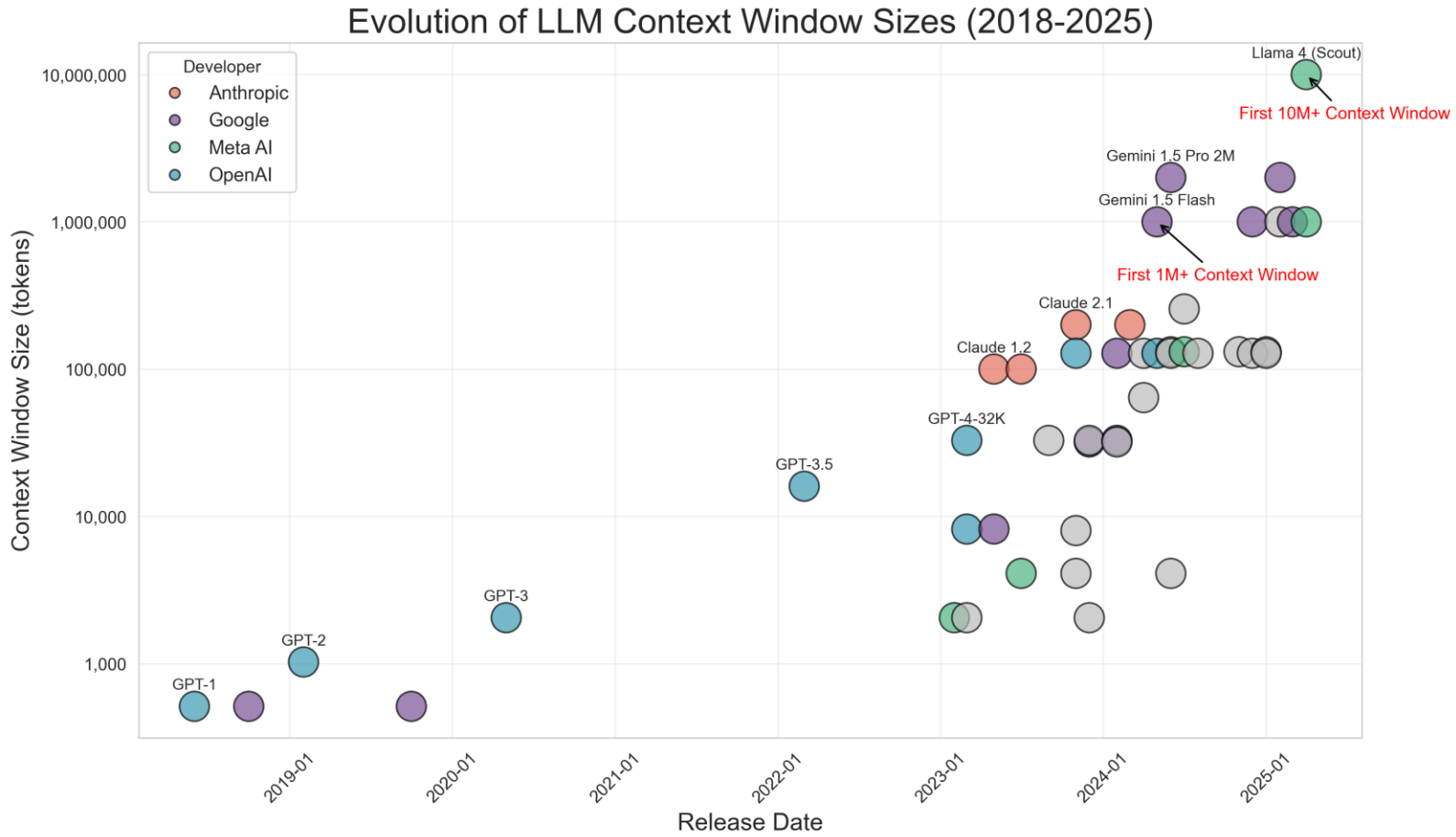
Fig. 3: A timeline of existing large language models (having a size larger than 10B) in recent years. The timeline was established mainly according to the release date (*e.g.*, the submission date to arXiv) of the technical paper for a model. If there was not a corresponding paper, we set the date of a model as the earliest time of its public release or announcement. We mark the LLMs with publicly available model checkpoints in yellow color. Due to the space limit of the figure, we only include the LLMs with publicly reported evaluation results.

Figure from https://klu.ai/glossary/large-language-model

Mode Size over Time



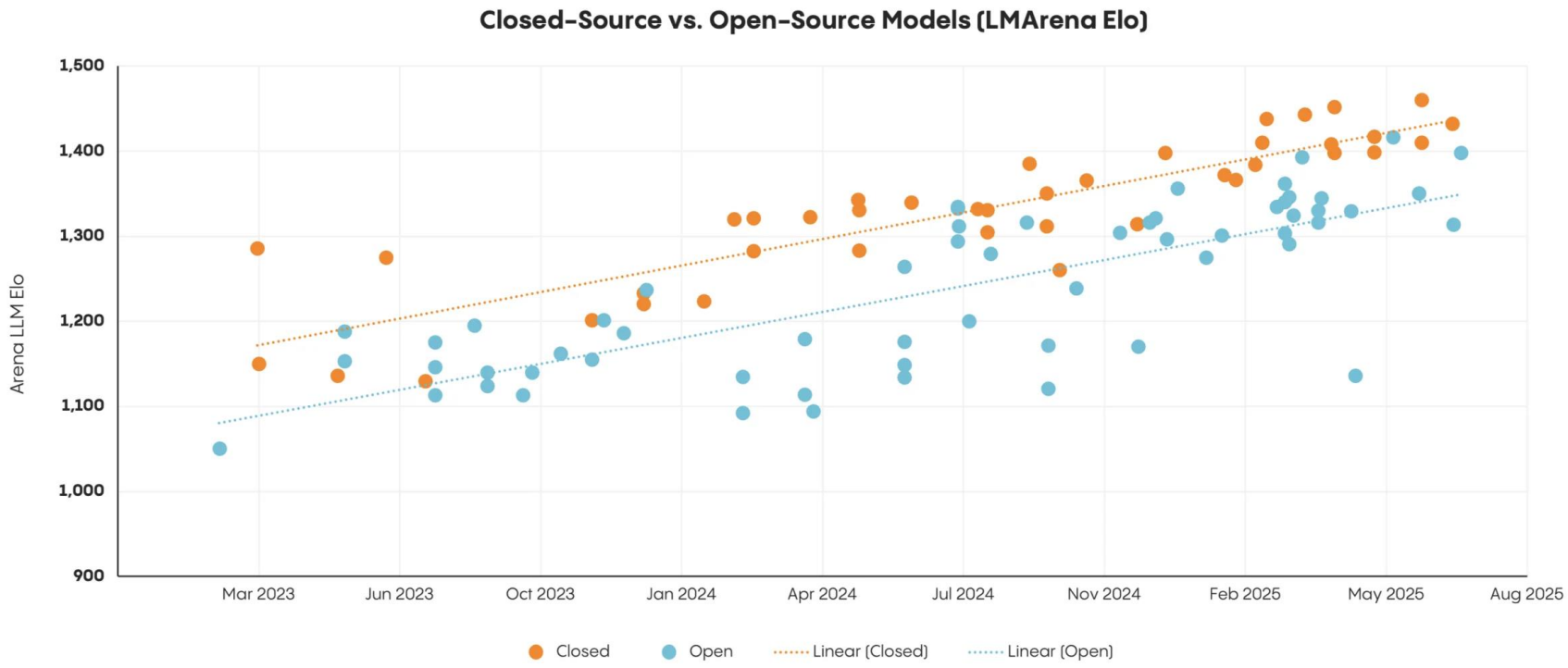
Context Size over Time



https://www.meibel.ai/post/understanding-the-impact-of-increasing-llm-context-windows

Open vs Closed LLMs

Closed-Source vs. Open-Source Models



Source: LMArena

https://menlovc.com/perspective/2025-mid-year-llm-market-update/

Closed-Source LLMs

- GPT
- Gemini
- Claude

Text Arena

View rankings across various LLMs on their versatility, linguistic precision, and cultural context across text.

Last Updated	Total Votes	Total Models
Oct 16, 2025	4,278,480	258

Overall | Search by model name... / | Default

Rank (UB)	Model	Score	95% CI (±)	Votes	Organization	License
1	gemini-2.5-pro	1451	±4	54,087	Google	Proprietary
1	claude-opus-4-1-20250805-thinking-16k	1447	±5	21,306	Anthropic	Proprietary
1	claude-sonnet-4-5-20250929-thinking-32k	1445	±8	6,287	Anthropic	Proprietary
1	gpt-4.5-preview-2025-02-27	1441	±6	14,644	OpenAI	Proprietary
2	chatgpt-4o-latest-20250326	1440	±4	40,013	OpenAI	Proprietary
2	o3-2025-04-16	1440	±4	51,293	OpenAI	Proprietary
2	claude-sonnet-4-5-20250929	1438	±8	6,144	Anthropic	Proprietary
2	gpt-5-high	1437	±5	23,580	OpenAI	Proprietary
2	claude-opus-4-1-20250805	1437	±5	33,298	Anthropic	Proprietary
3	qwen3-max-preview	1434	±6	18,078	Alibaba	Proprietary
10	gpt-5-chat	1425	±5	21,630	OpenAI	Proprietary
10	qwen3-max-2025-09-23	1423	±7	6,919	Alibaba	Proprietary
10	glm-4.6	1422	±9	4,401	Z.ai	MIT
11	grok-4-fast	1420	±8	7,104	xAI	Proprietary
11	claude-opus-4-20250514-thinking-16k	1419	±5	35,522	Anthropic	Proprietary
11	deepseek-v3.2-exp-thinking	1419	±9	4,320	DeepSeek AI	MIT
11	qwen3-vl-235b-a22b-instruct	1418	±8	6,312	Alibaba	Apache 2.0

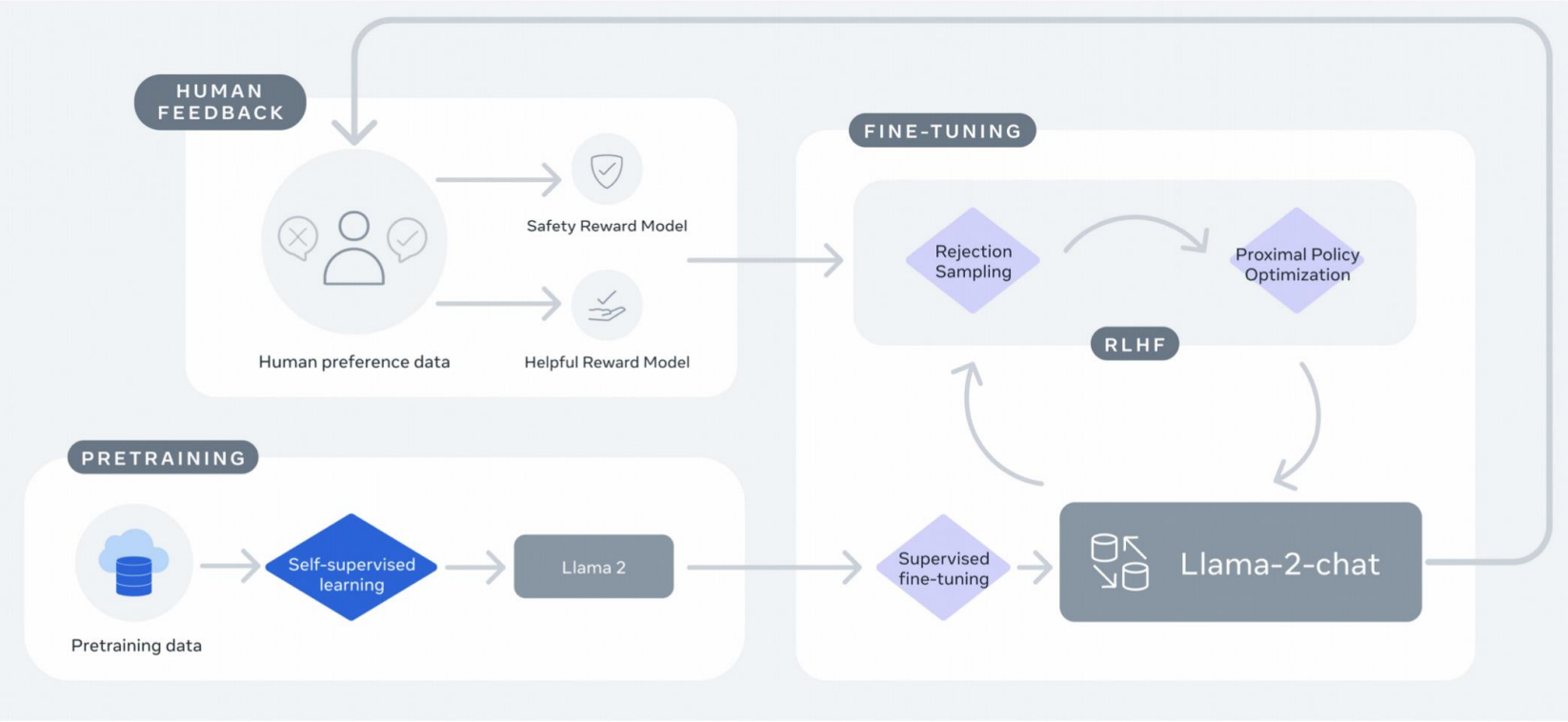
https://lmarena.ai/leaderboard/text

Open-source LLMs

10	glm-4.6	1422	±9	4,401	Z.ai	MIT
11	grok-4-fast	1420	±8	7,104	xAI	Proprietary
11	claude-opus-4-20250514-thinking-16k	1419	±5	35,522	Anthropic	Proprietary
11	deepseek-v3.2-exp-thinking	1419	±9	4,320	DeepSeek AI	MIT
11	qwen3-vl-235b-a22b-instruct	1418	±8	6,312	Alibaba	Apache 2.0
11	qwen3-235b-a22b-instruct-2507	1418	±5	29,343	Alibaba	Apache 2.0
11	deepseek-r1-0528	1417	±6	19,284	DeepSeek	MIT
11	kimi-k2-0905-preview	1417	±7	10,772	Moonshot	Modified MIT
11	deepseek-v3.1	1416	±6	15,380	DeepSeek	MIT
11	deepseek-v3.1-thinking	1415	±7	12,098	DeepSeek	MIT
11	kimi-k2-0711-preview	1415	±5	28,321	Moonshot	Modified MIT
11	deepseek-v3.1-terminus	1414	±10	3,775	DeepSeek AI	MIT
11	deepseek-v3.1-terminus-thinking	1413	±10	3,541	DeepSeek AI	MIT
12	grok-4-0709	1413	±5	29,264	xAI	Proprietary
12	claude-opus-4-20250514	1411	±4	43,310	Anthropic	Proprietary
12	deepseek-v3.2-exp	1408	±9	4,684	DeepSeek AI	MIT
13	gpt-4.1-2025-04-14	1411	±4	41,918	OpenAI	Proprietary
14	grok-3-preview-02-24	1409	±4	34,154	xAI	Proprietary
18	mistral-medium-2508	1406	±5	23,844	Mistral	Proprietary
18	glm-4.5	1406	±5	22,612	Z.ai	MIT

Overview of LLMs Training

- Pretraining -> Supervised Fine-tuning (SFT) -> Reinforcement Learning Human Feedback (RLHF)



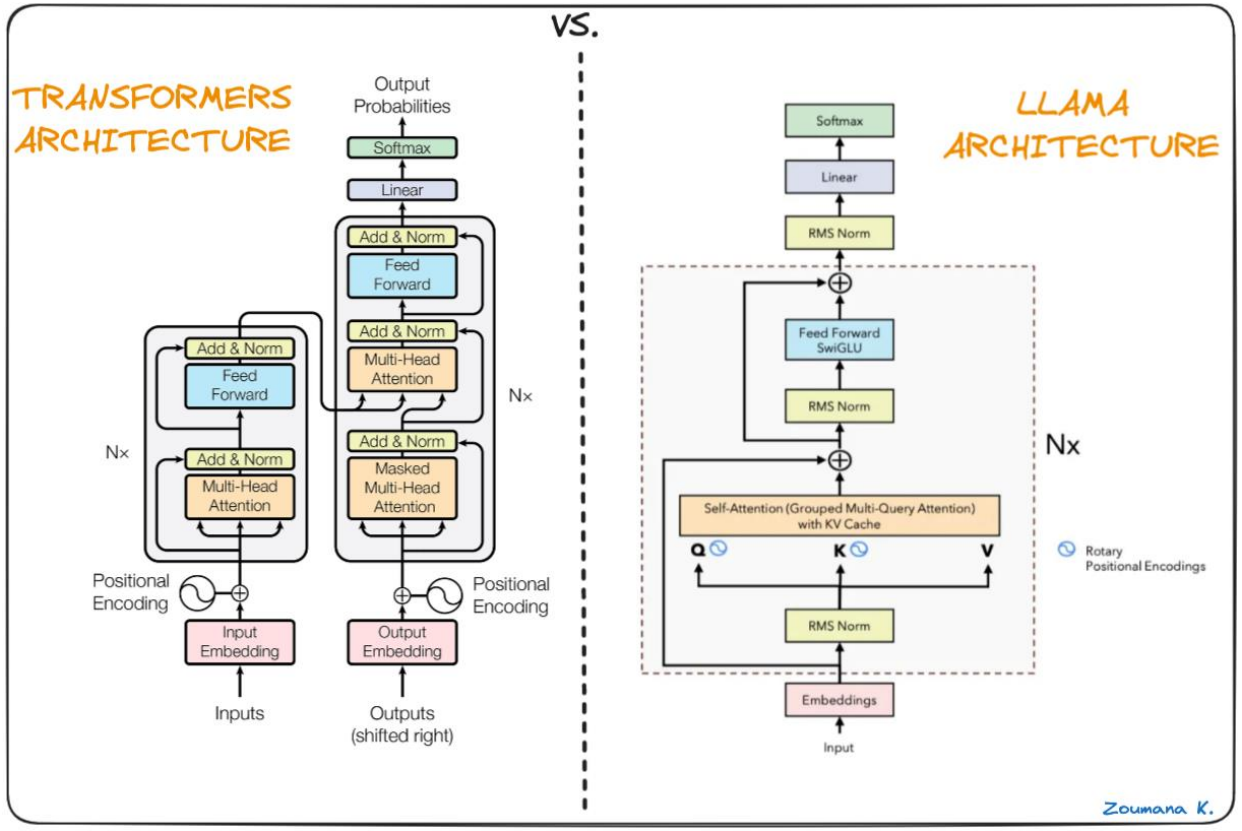
Pre-training Data

Example: Llama1

- 1.4 Trillion Tokens!!

Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

LlaMa Architecture



https://medium.com/@pranjalkhadka/llama-explained-a70e71e706e9

LLaMA:OpenandEfficient Foundation Language Models (https://arxiv.org/pdf/2302.13971)

Training Loss

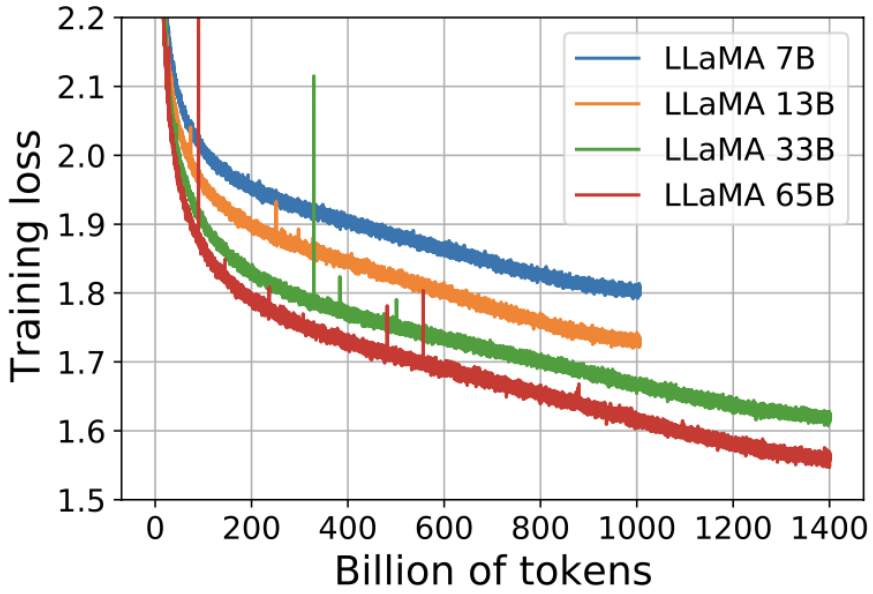


Figure 1: **Training loss over train tokens for the 7B, 13B, 33B, and 65 models.** LLaMA-33B and LLaMA-65B were trained on 1.4T tokens. The smaller models were trained on 1.0T tokens. All models are trained with a batch size of 4M tokens.

Llama (https://www.llama.com/)

Meta AI

Play with Llama:
https://www.meta.ai/?utm_source=llama_meta_site&utm_medium=web&utm_content=Llama_nav&utm_campaign=July_moment

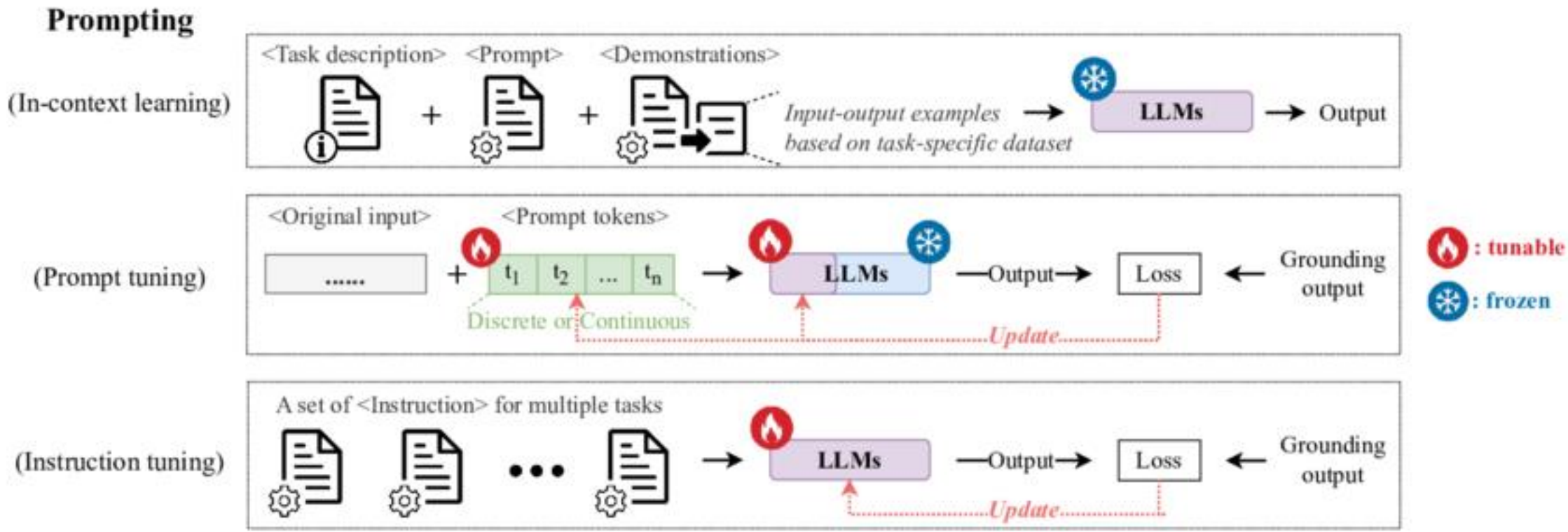
LoRa:

- Efficient fine-tuning

Fine-tuning Llama 2 in Google Colab

- https://colab.research.google.com/drive/1wbPpB3fY9YRzebrZq6WkP5HxMuHMQR72?usp=sharing

Different tasks in Natural Language Understanding


Prompting
(In-context learning)
<Task description>
<Prompt>
<Demonstrations>
Input-output examples
based on task-specific dataset
LLMs
Output
(Prompt tuning)
<Original input>
<Prompt tokens>
.....
t1 t2 ... tn
Discrete or Continuous
LLMs
Output
Loss
Grounding output
Update
: tunable
: frozen
(Instruction tuning)
A set of <Instruction> for multiple tasks
LLMs
Output
Loss
Grounding output
Update

Thanks!