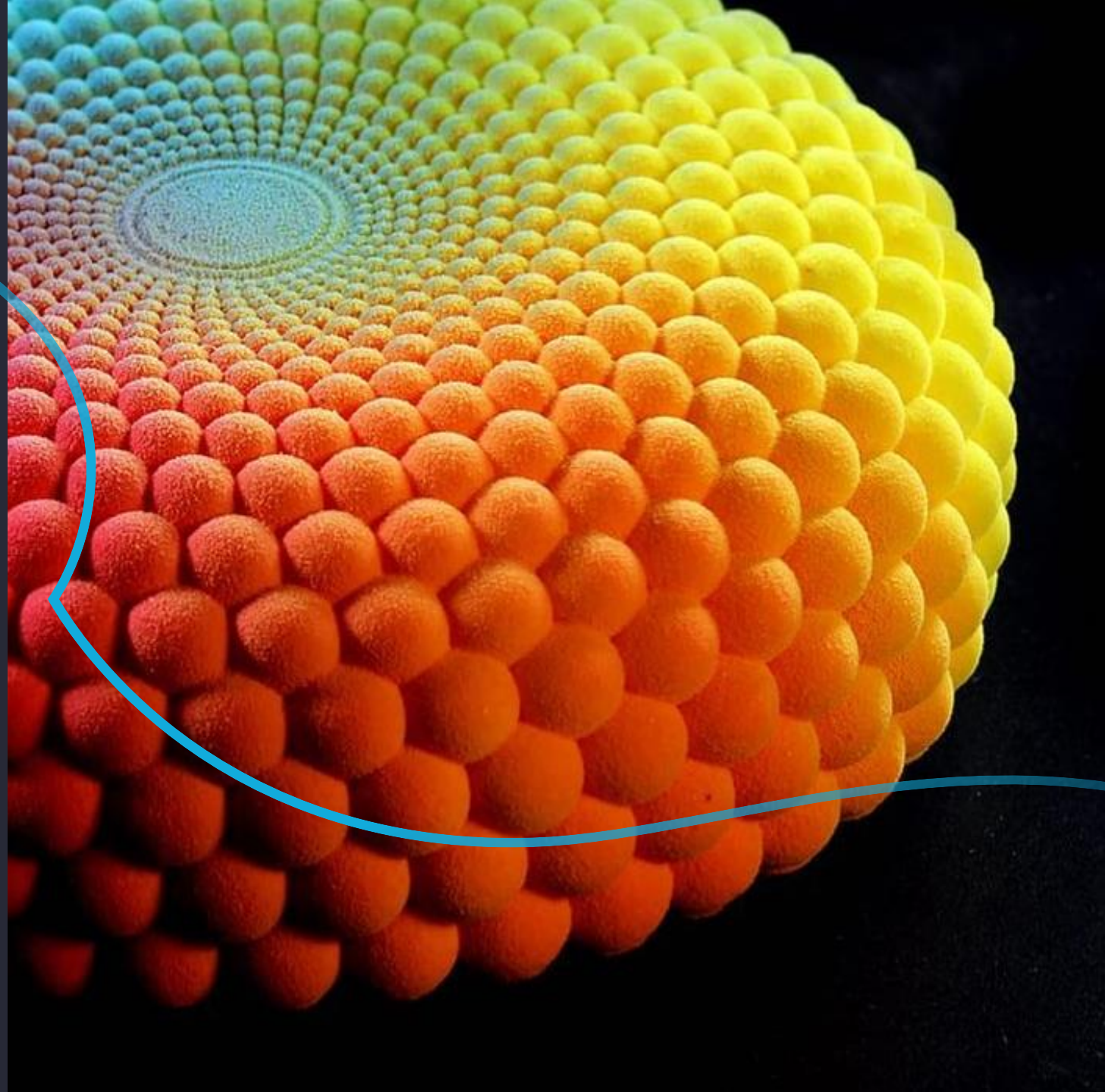
Capgemini UNIVERSITY

SMALL LANGUAGE MODELS

May 2024



Generative language models ar

capgemini.com/insights/expert-perspectives/generative-language-models-and-the-future-of-ai/

Finish update

Contact us Investors Global | EN

Capgemini

Insights Industries Services Careers News About us

Search

Home / Insights / Expert perspectives / Generative language models and the future of AI

GENERATIVE LANGUAGE MODELS AND THE FUTURE OF AI

Rajeswaran Viswanathan

2021-09-15

What Makes Small Language Models so Attractive?

Accessible and Affordable

They can be run (in inference mode) on limited resource regimes (such as laptops and/or small GPUs).

Easier to Customize

Small models can typically be fine-tuned on just a single GPU.

More Energy Efficient

Small language models require fewer computational resources making them more energy-efficient.

Cheaper to Develop

These models only require a relatively small number of GPUs.

Valuable for Educational Purposes

They are more manageable and thus easier to understand and tweak.

Moore's Law of Language Models

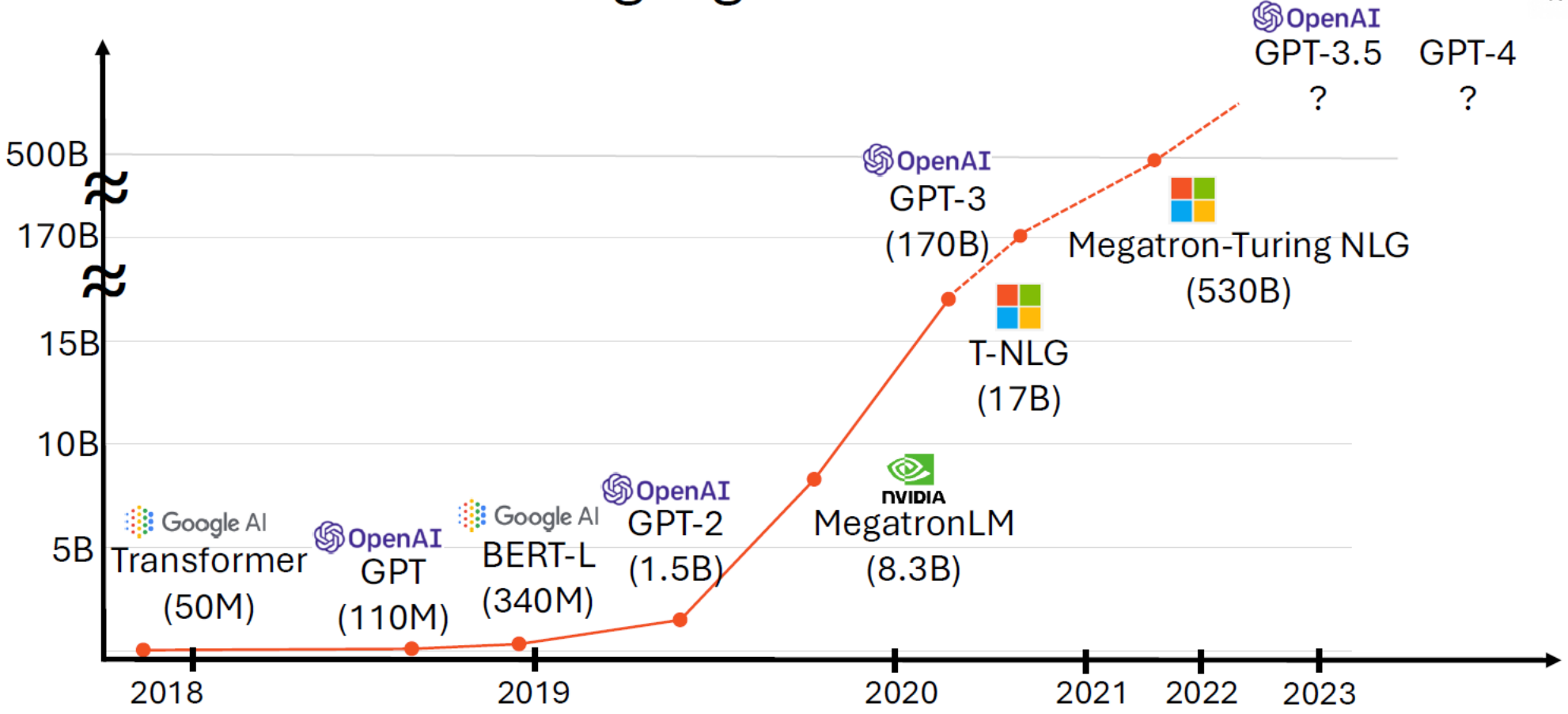
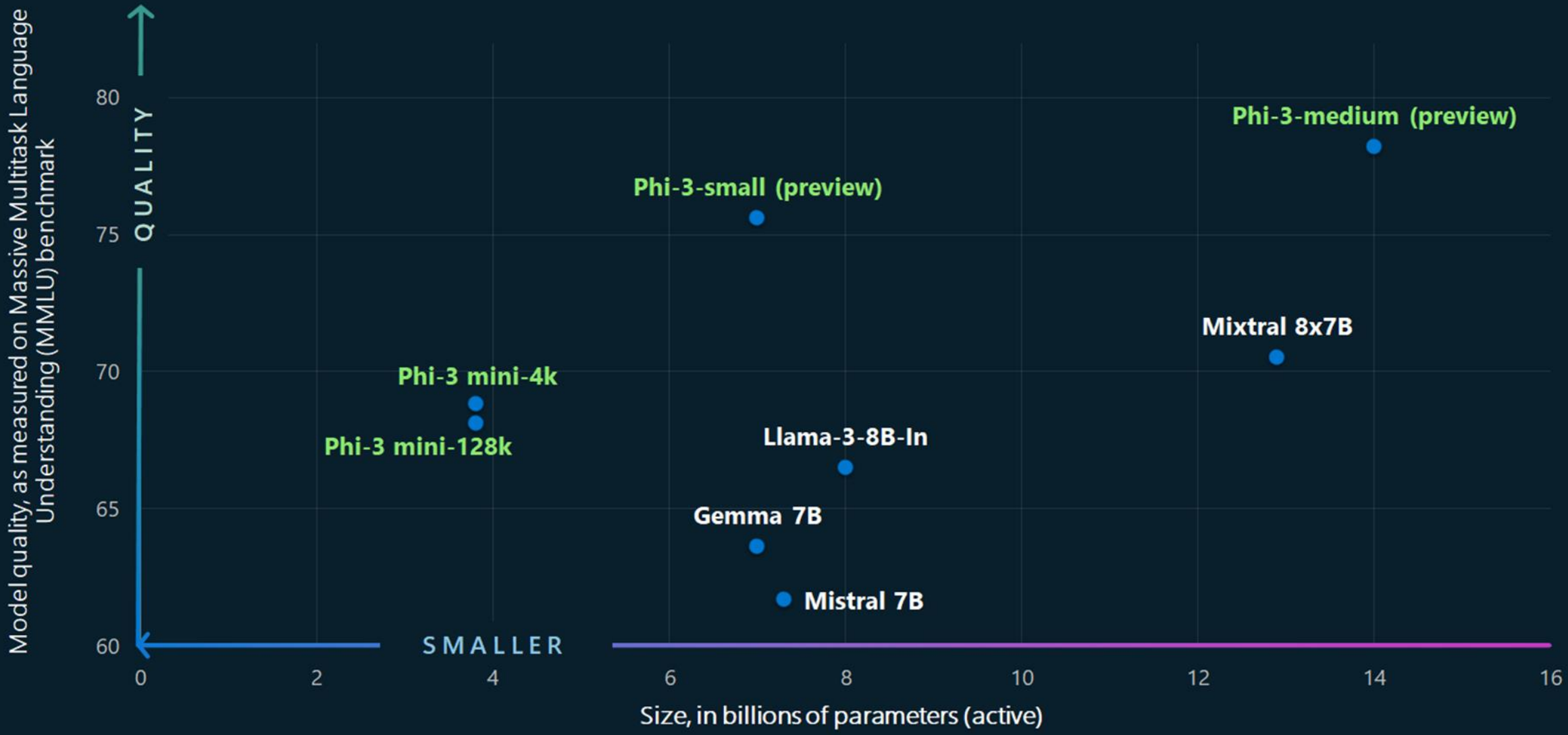


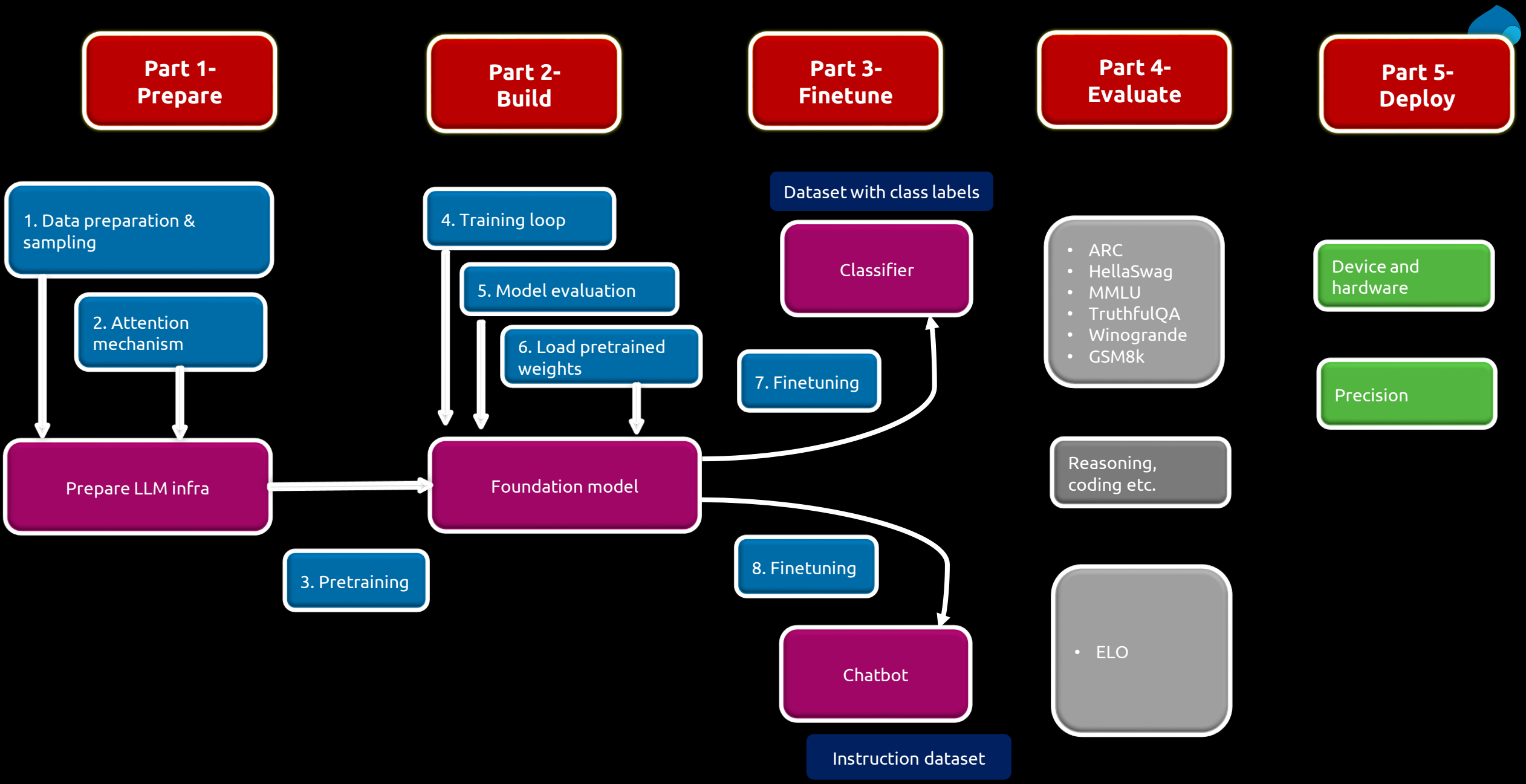
Figure inspired by Microsoft Research Blog: https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft/

Quality vs Size in Small Language Models (SLMs)

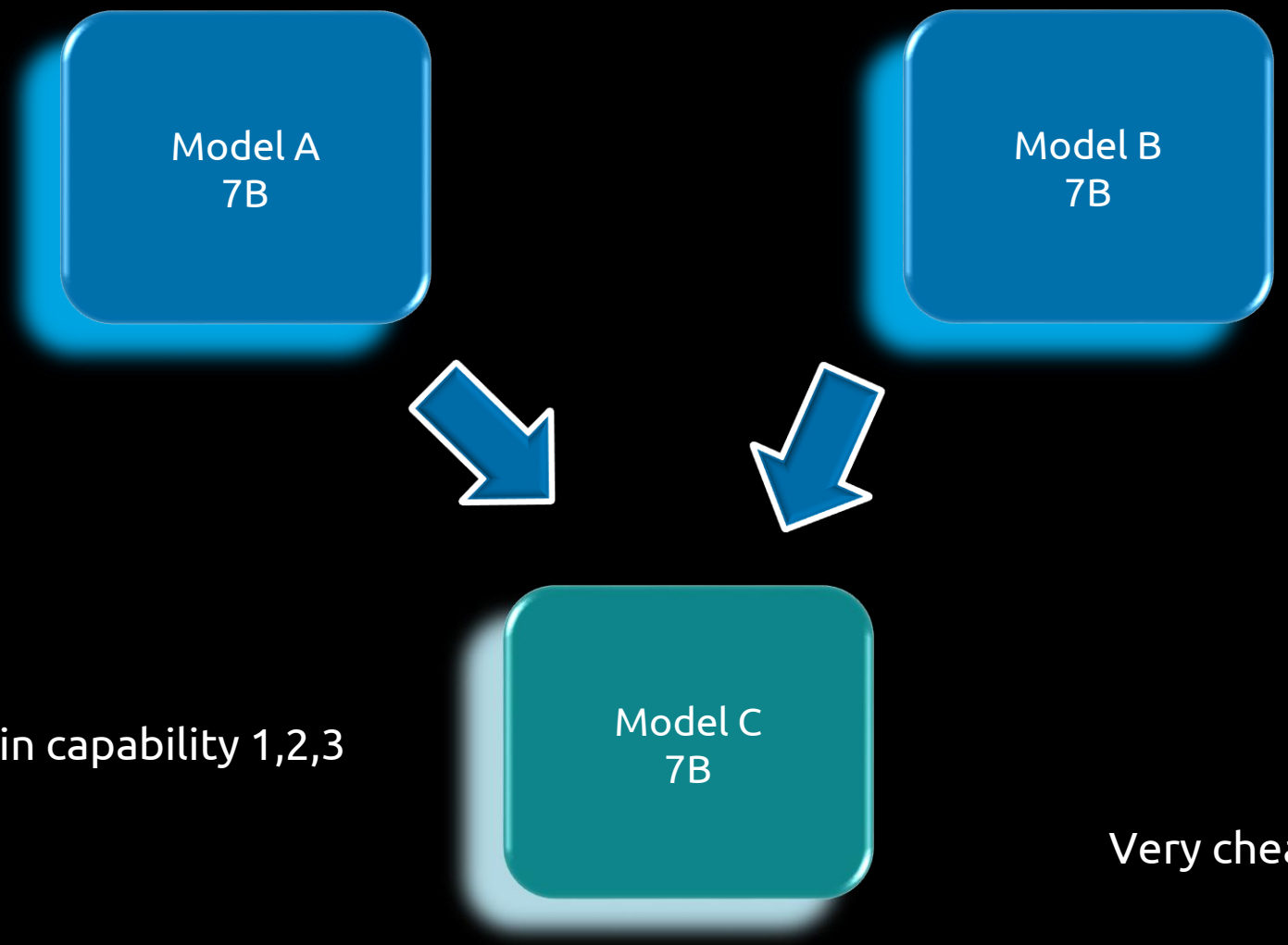


5 Leading Small Language Models of 2024

Llama 3	Meta	8 billion parameters
Phi-3	Microsoft	3.8 billion - 7 billion parameters
Gemma	Google	2 billion - 7 billion parameters
Mixtral 8x7B	Mistral AI	7 billion parameters
OpenELM	Apple	0.27 billion - 3 billion parameters



MERGE OF MODELS



Good in capability 1 and 2

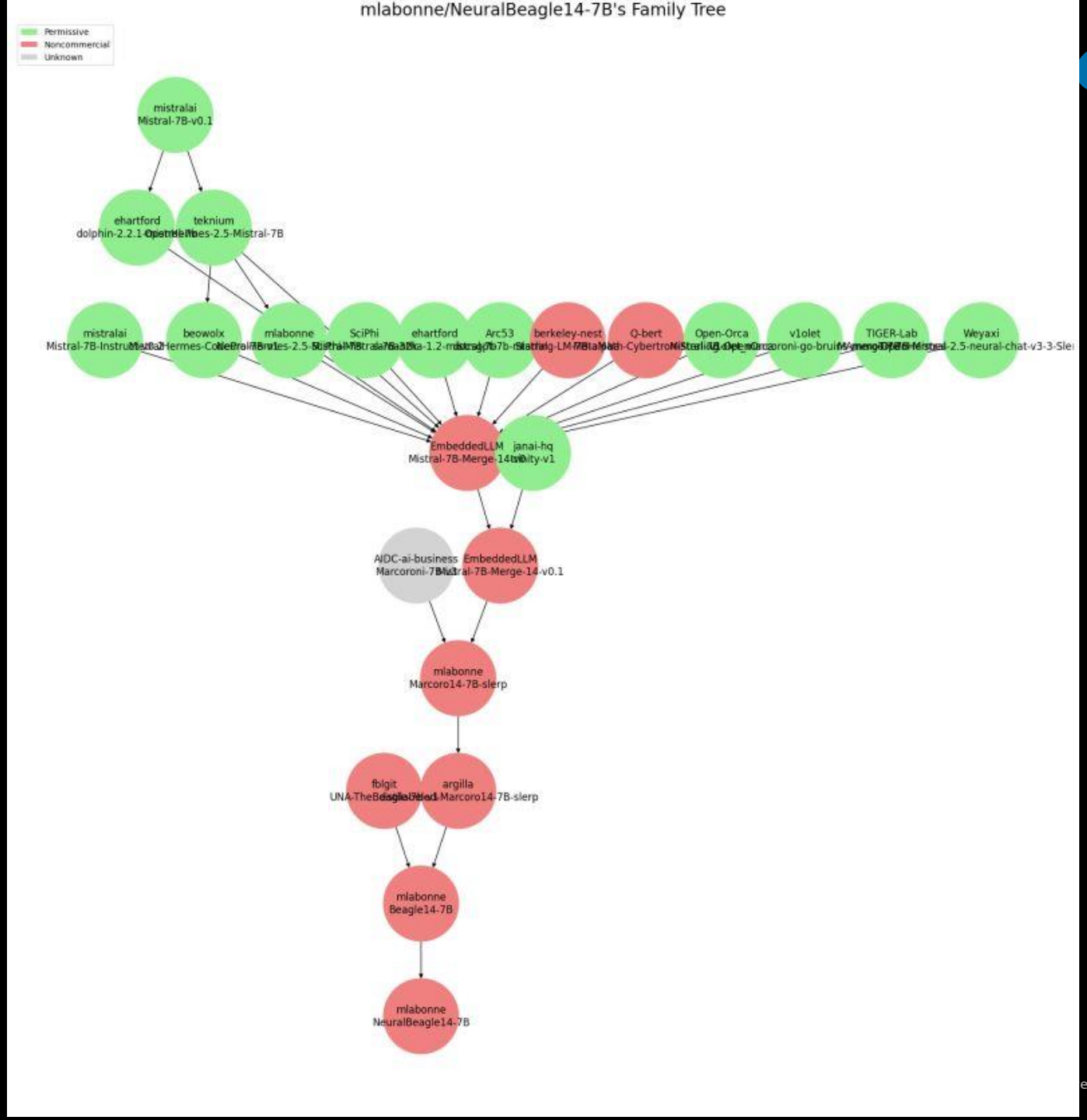
Good in capability 3 and 4

Good in capability 1,2,3 and 4

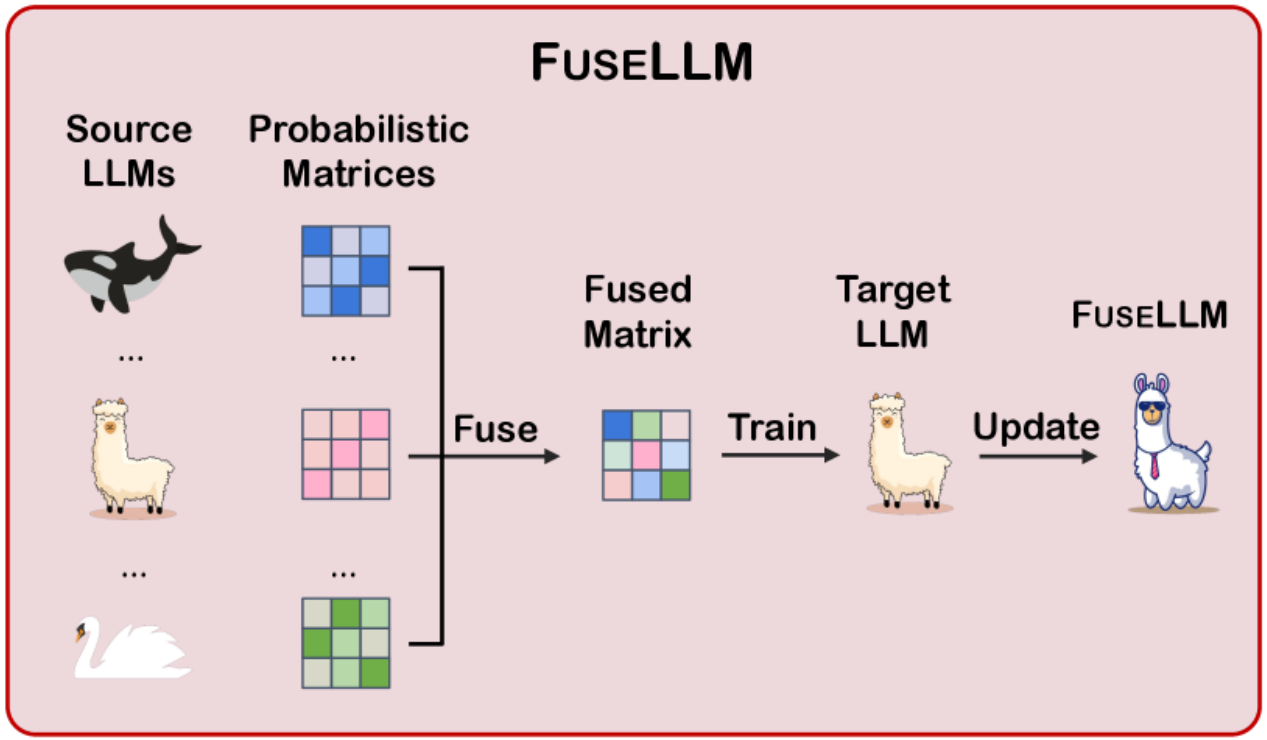
Very cheap (<$100)

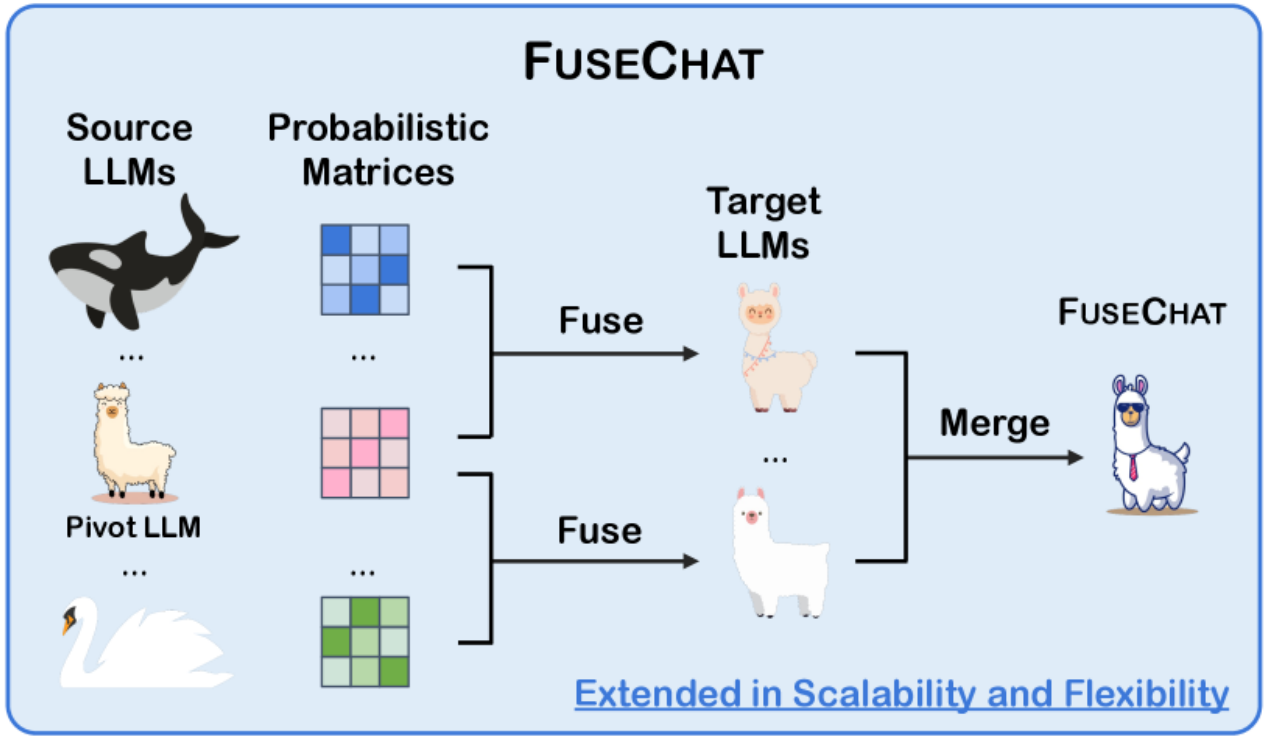
Single new model

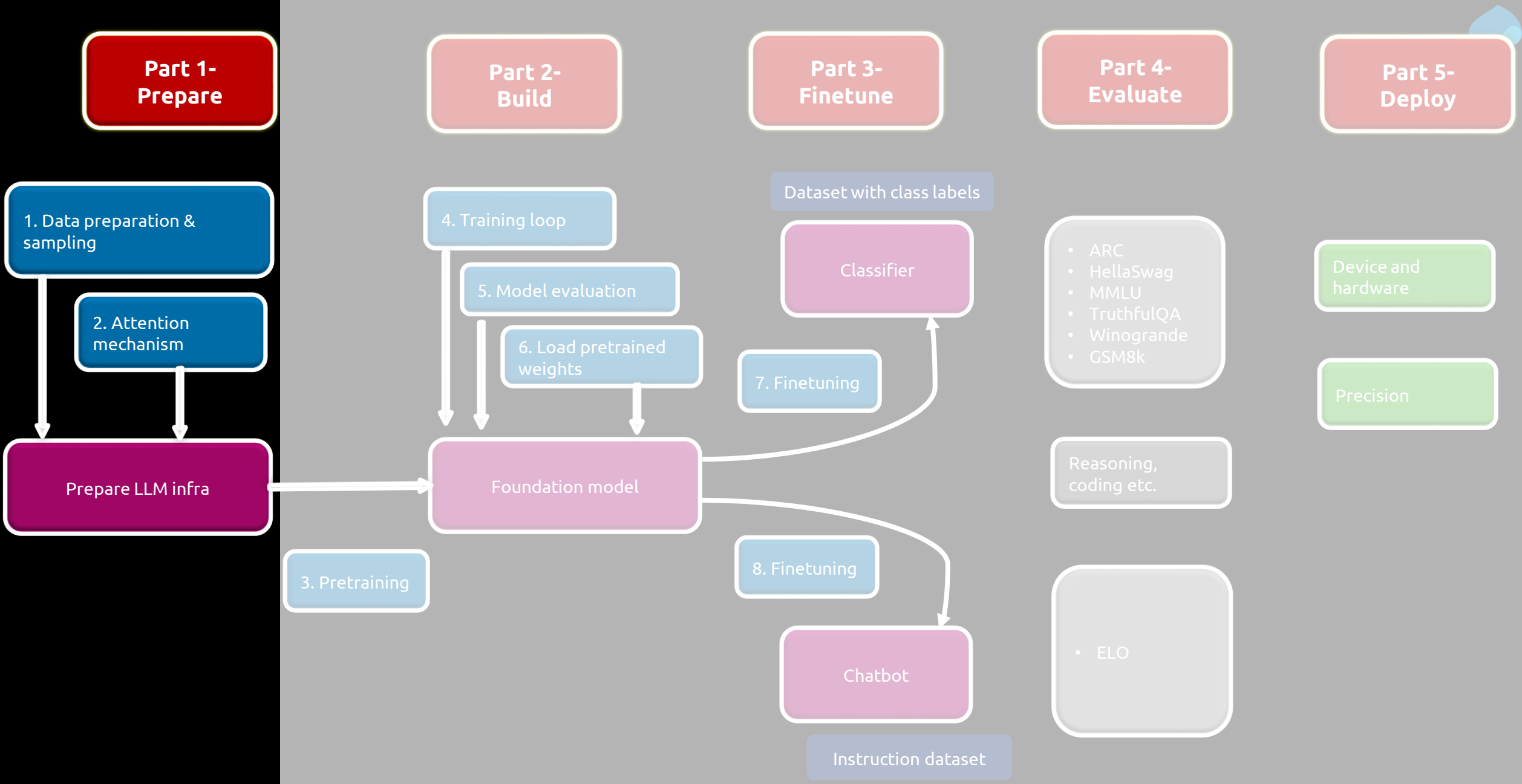
SAMPLE MERGE TREE



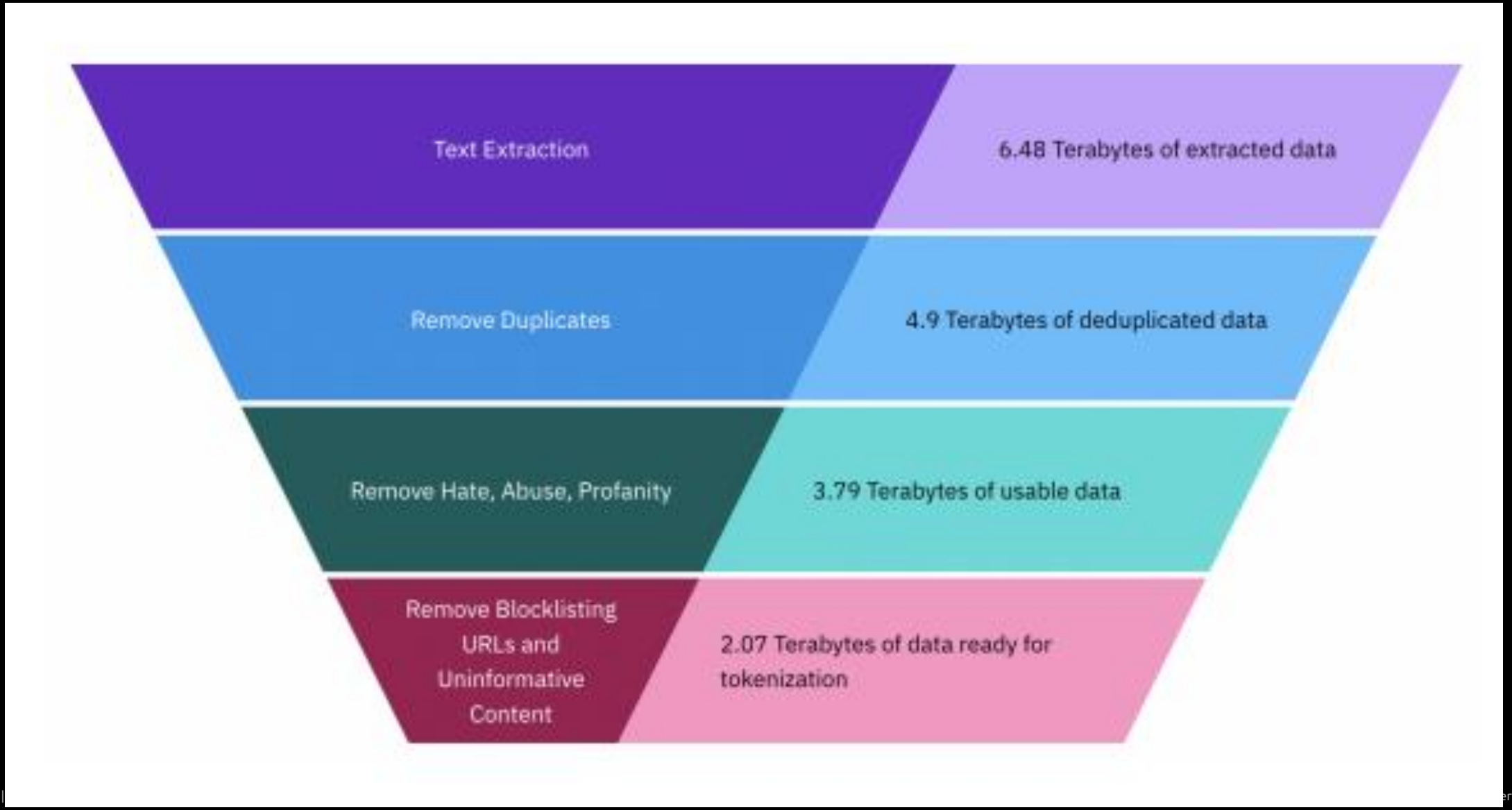
MERGE IS DIFFERENT FOR LLM OR CHAT MODELS







IBM GRANITE LLM



KEY INSIGHT – “TEXTBOOKS ARE ALL YOU NEED”

***phi-1* (1.3B)**

microsoft/**phi-1** 🤗

Python **coding** model with perf. comparable to models 10x larger trained on 100x more data

***phi-1.5* (1.3B)**

microsoft/**phi-1_5** 🤗

Natural language model with NL comparable to models 10x larger trained on 30x more data and reasoning comparable to models 50x larger.

***phi-2* (2.7B)**

microsoft/**phi-2** 🤗

Building High-Quality Datasets

1. Filtering web data:
   - GPT-4 can reliably classify documents based on “high educational value”.
   - **Challenge:** Stack (Python) is 26B tokens (around $1M cost in 2023).
   - **Solution:** label small fraction, then train a random forest classifier on it and use the classifier to filter the rest.

Educational values deemed by the filter

High educational value

```python
import torch
import torch.nn.functional as F

def normalize(x, axis=-1):
    """Performs L2-Norm."""
    num = x
    denom = torch.norm(x, 2, axis, keepdim=True)
    .expand_as(x) + 1e-12
    return num / denom

def euclidean_dist(x, y):
    """Computes Euclidean distance."""
    m, n = x.size(0), y.size(0)
    xx = torch.pow(x, 2).sum(1, keepdim=True).
    expand(m, n)
    yy = torch.pow(x, 2).sum(1, keepdim=True).
    expand(m, m).t()
    dist = xx + yy - 2 * torch.matmul(x, y.t())
    dist = dist.clamp(min=1e-12).sqrt()
    return dist

def cosine_dist(x, y):
    """Computes Cosine Distance."""
    x = F.normalize(x, dim=1)
    y = F.normalize(y, dim=1)
    dist = 2 - 2 * torch.mm(x, y.t())
    return dist
```

Low educational value

```python
import re
import typing
...

class Default(object):
    def __init__(self, vim: Nvim) -> None:
        self._vim = vim
        self._denite: typing.Optional[SyncParent]
    = None
        self._selected_candidates: typing.List[int
    ] = []
        self._candidates: Candidates = []
        self._cursor = 0
        self._entire_len = 0
        self._result: typing.List[typing.Any] = []
        self._context: UserContext = {}
        self._bufnr = -1
        self._winid = -1
        self._winrestcmd = ''
        self._initialized = False
        self._winheight = 0
        self._winwidth = 0
        self._winminheight = -1
        self._is_multi = False
        self._is_async = False
        self._matched_pattern = ''
        ...
```

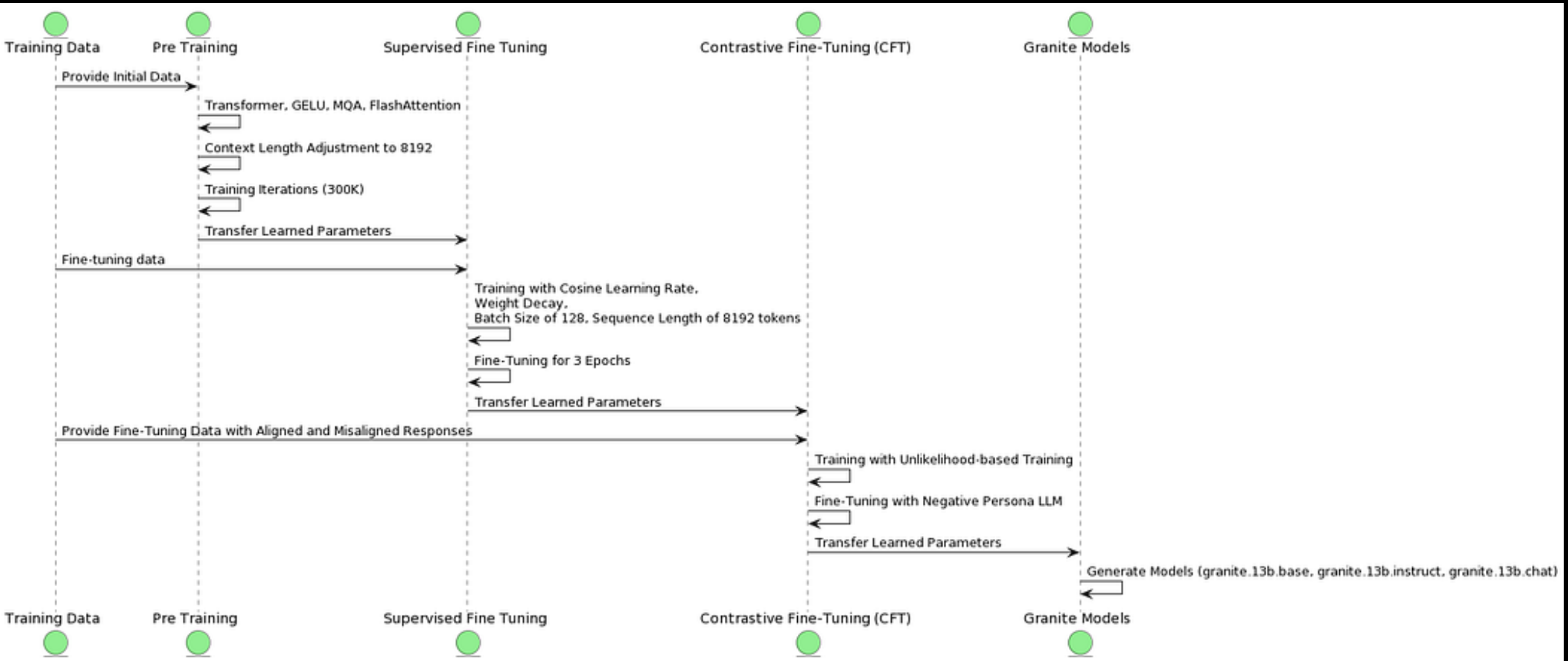
Building High-Quality Datasets

2. Create synthetic data:
   - Synthetic textbooks: teach the model coding with natural language
   - 1B tokens generated with GPT-3.5
   - **Challenge:** achieving high diversity (coding concepts, skills, level of difficulty, etc.) and low repetition
   - **Solution:** inject creative randomness into the prompt [1]

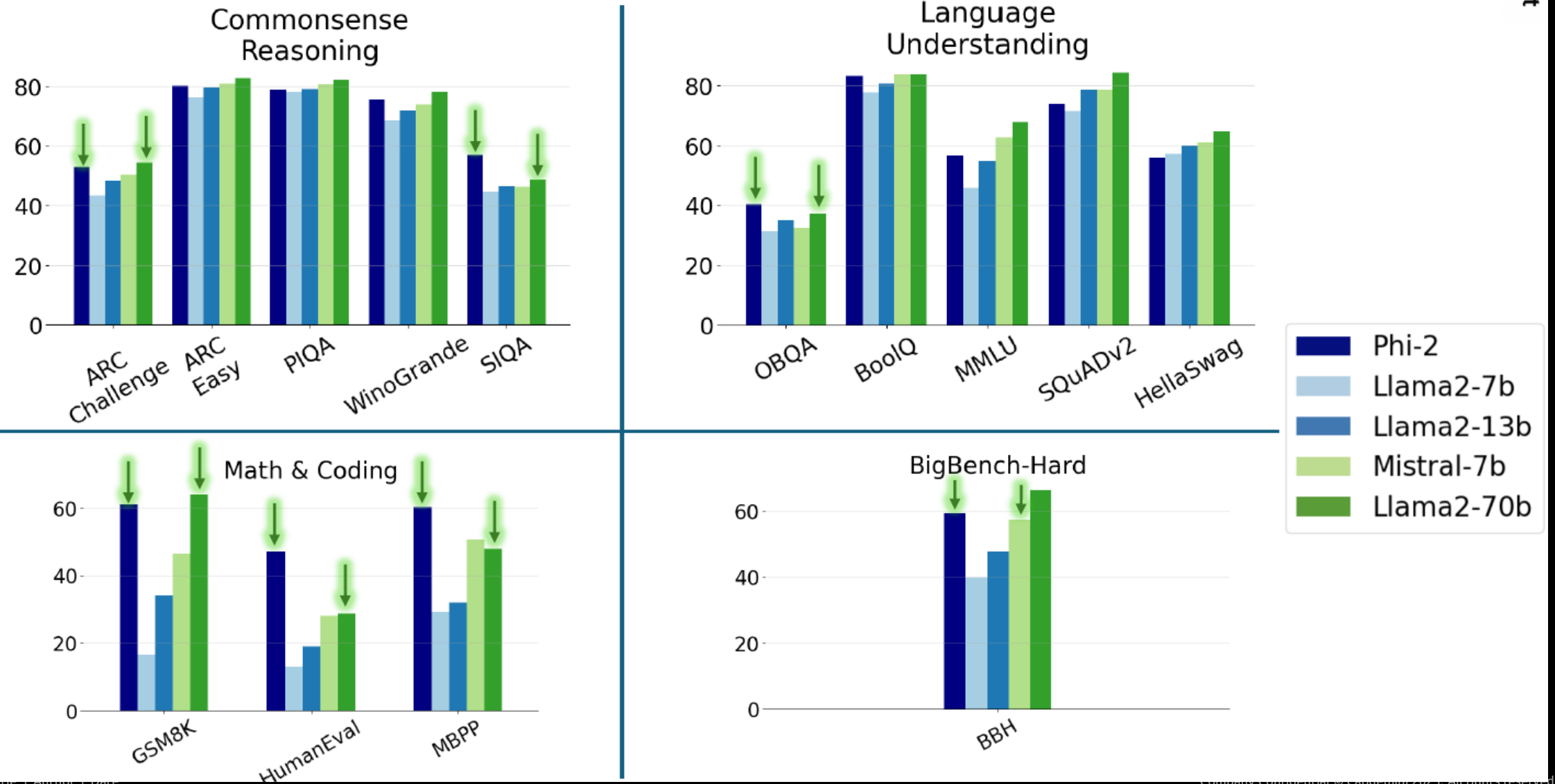
```
To begin, let us define singular and nonsingular matrices. A matrix is said to be singular if its
determinant is zero. On the other hand, a matrix is said to be nonsingular if its determinant is not
 zero. Now, let's explore these concepts through examples.

Example 1: Consider the matrix A = np.array([[1, 2], [2, 4]]). We can check if this matrix is
singular or nonsingular using the determinant function. We can define a Python function, `
is_singular(A)`, which  returns true if the determinant of A is zero, and false otherwise.

import numpy as np
def is_singular(A):
    det = np.linalg.det(A)
    if det == 0:
        return True
    else:
        return False

A = np.array([[1, 2], [2, 4]])
print(is_singular(A)) # True
```

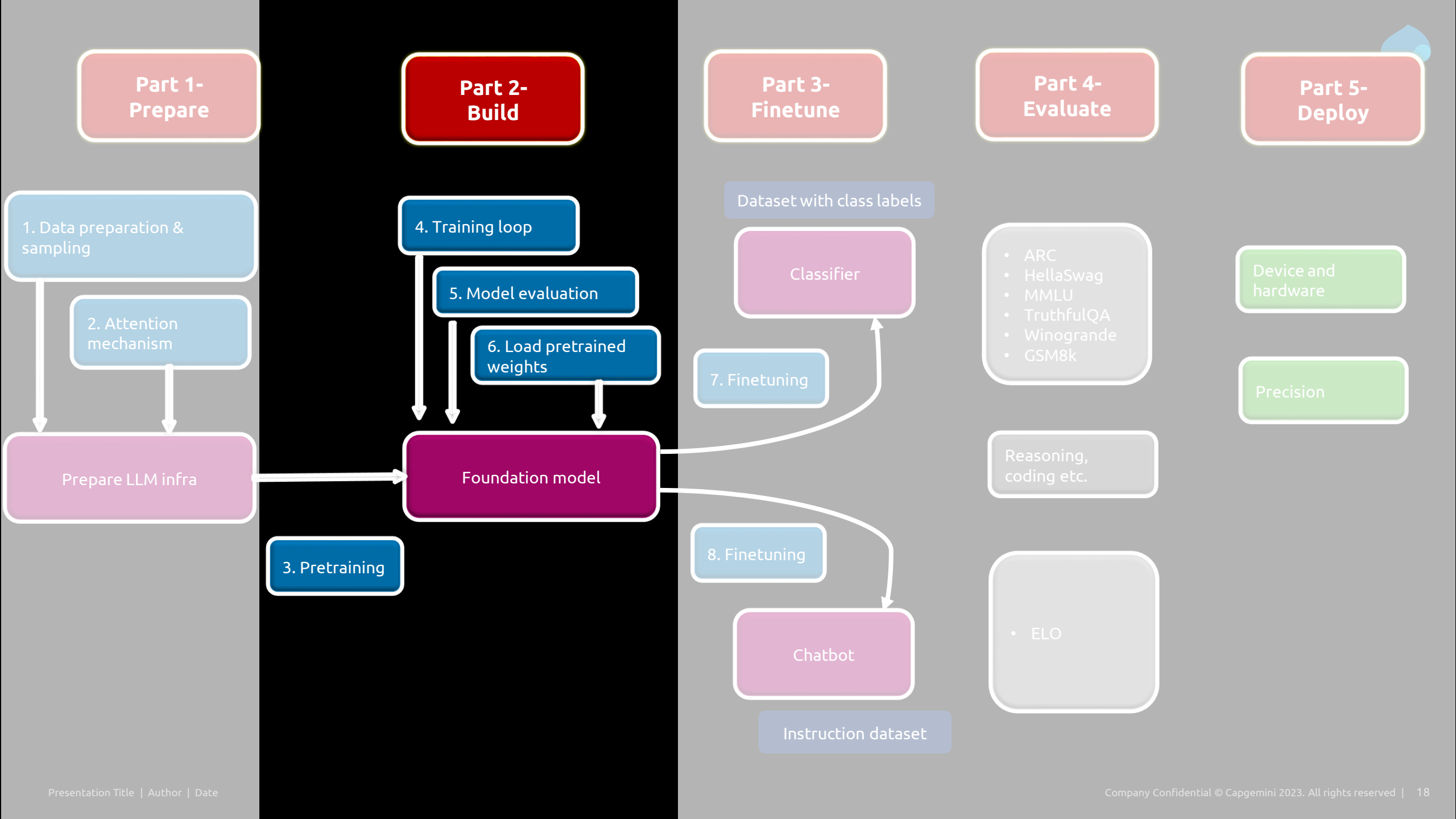
[1] Eldan, Ronen, and Yuanzhi Li. “TinyStories: How Small Can Language Models Be and Still Speak Coherent English?” *arXiv preprint arXiv:2305.07759* (2023).

GRANITE MODELS

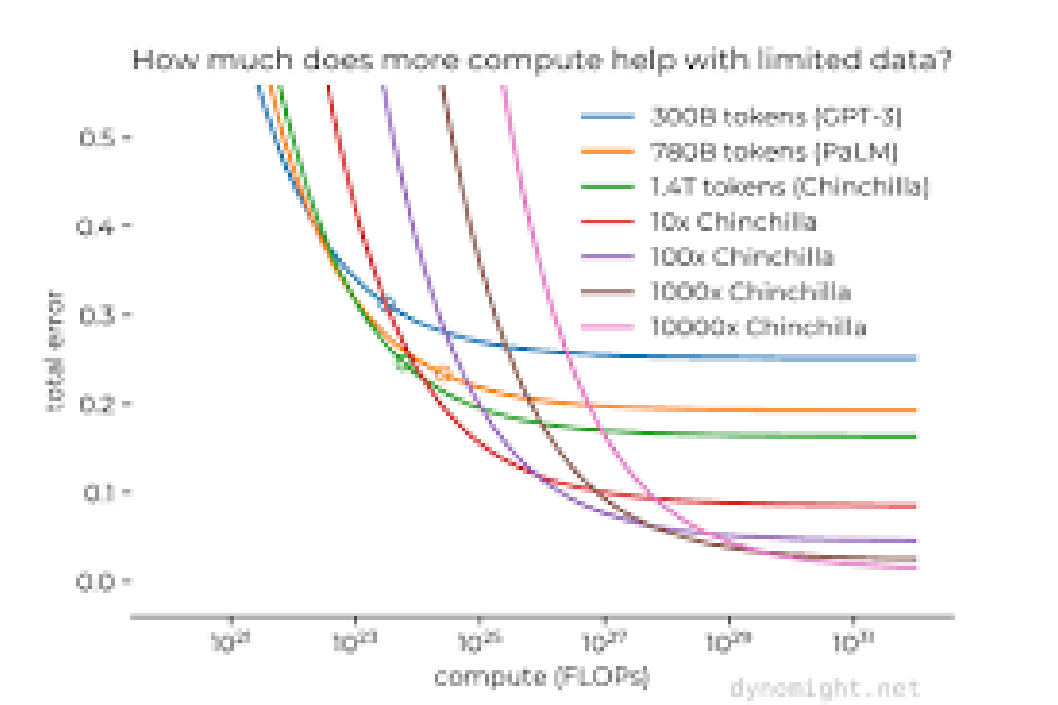


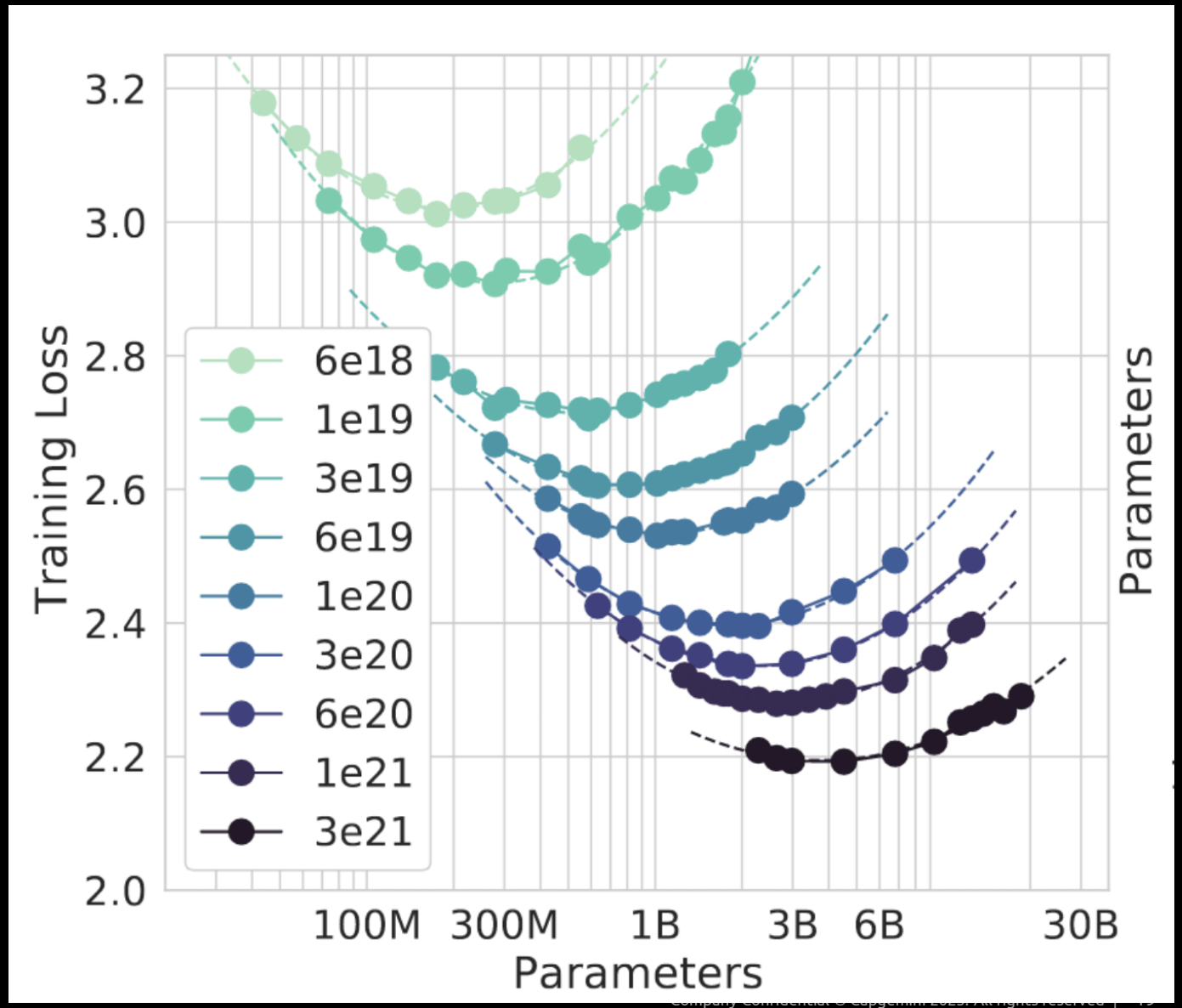
Phi-2 Performance





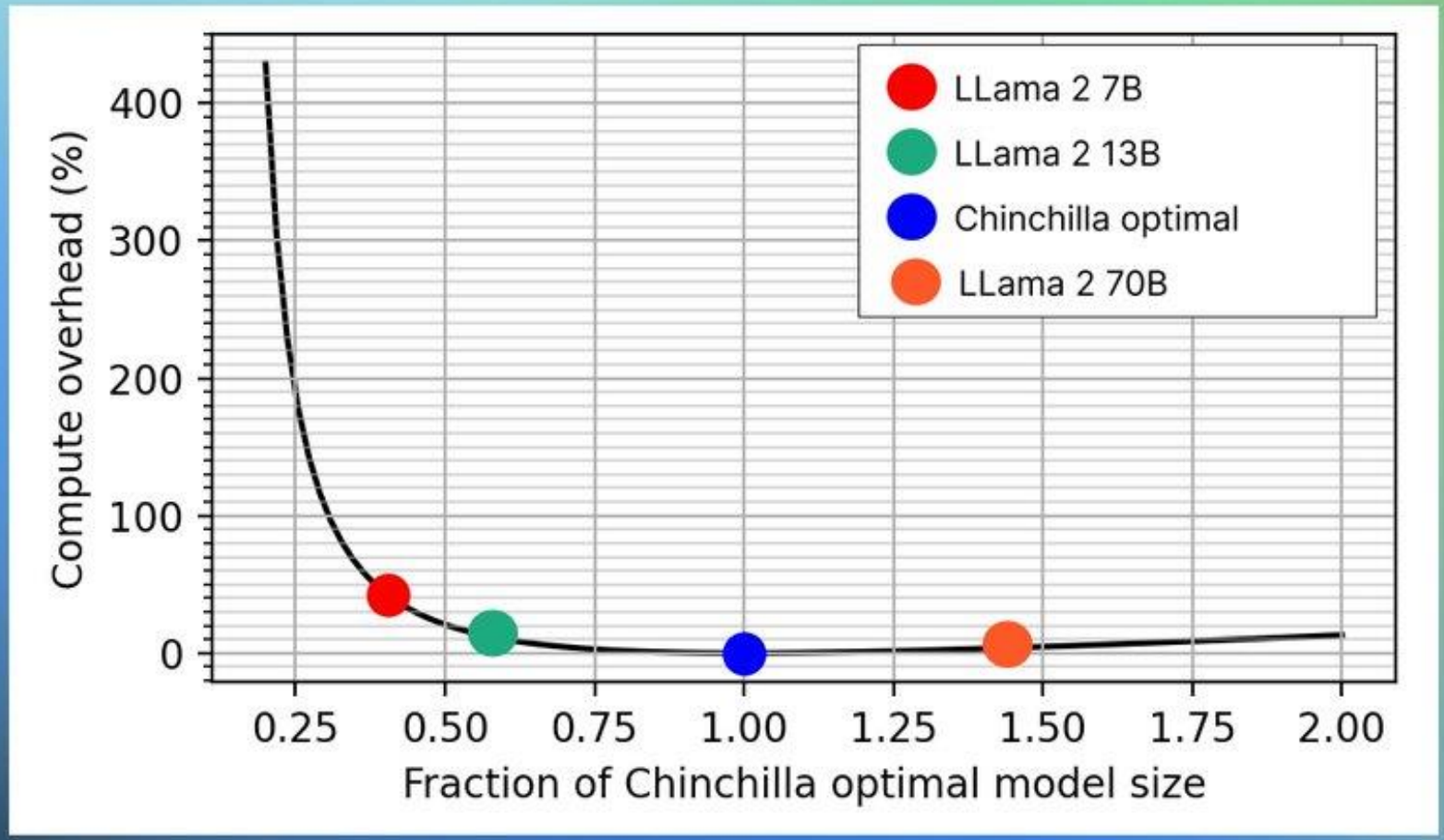
CHINCHILA

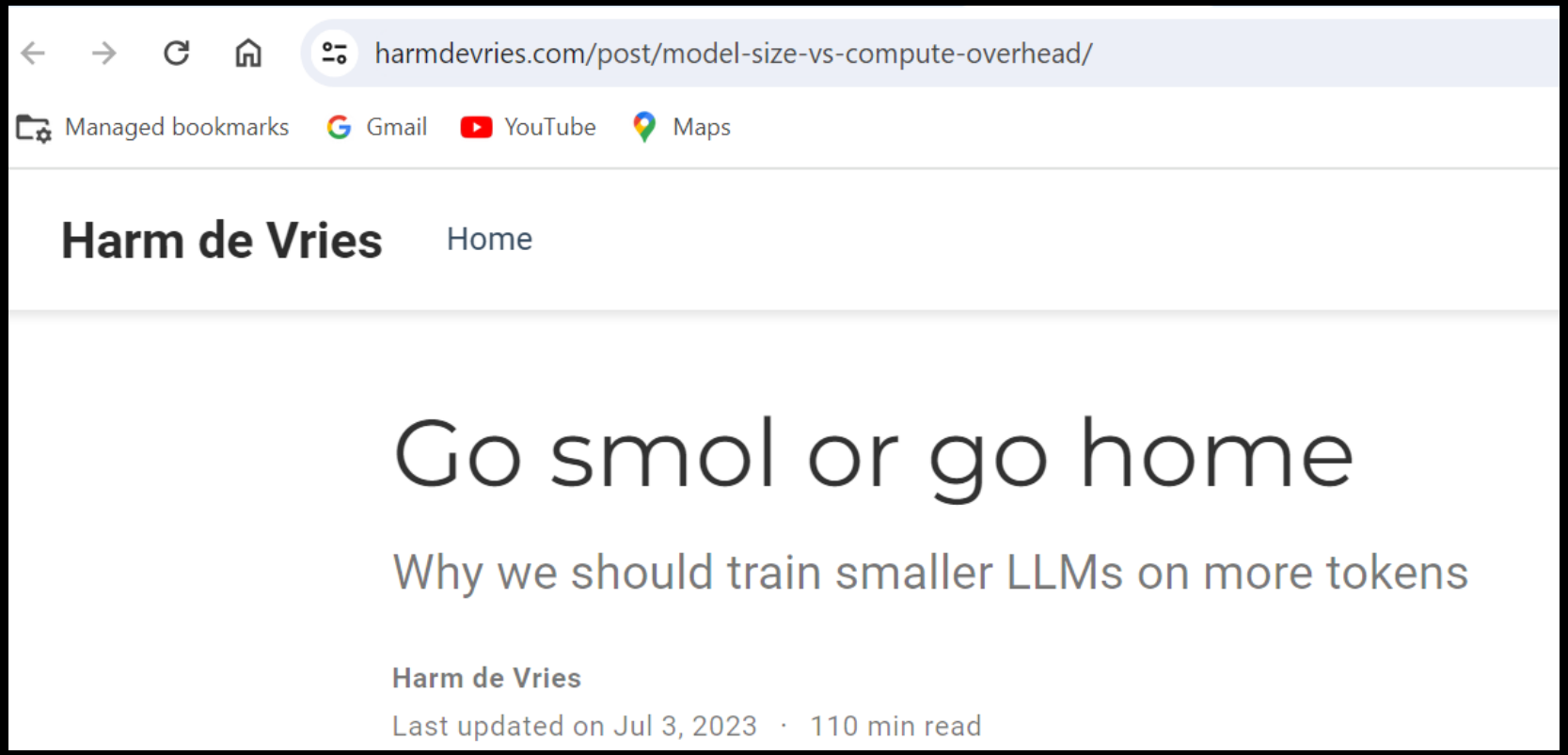

How much does more compute help with limited data?
300B tokens (GPT-3)
780B tokens (PaLM)
1.4T tokens (Chinchilla)
10x Chinchilla
100x Chinchilla
1000x Chinchilla
10000x Chinchilla
total error
0.5
0.4
0.3
0.2
0.1
0.0
compute (FLOPs)
dynomight.net


Training Loss
3.2
3.0
2.8
2.6
2.4
2.2
2.0
6e18
1e19
3e19
6e19
1e20
3e20
6e20
1e21
3e21
100M
300M
1B
3B
6B
30B
Parameters
Parameters

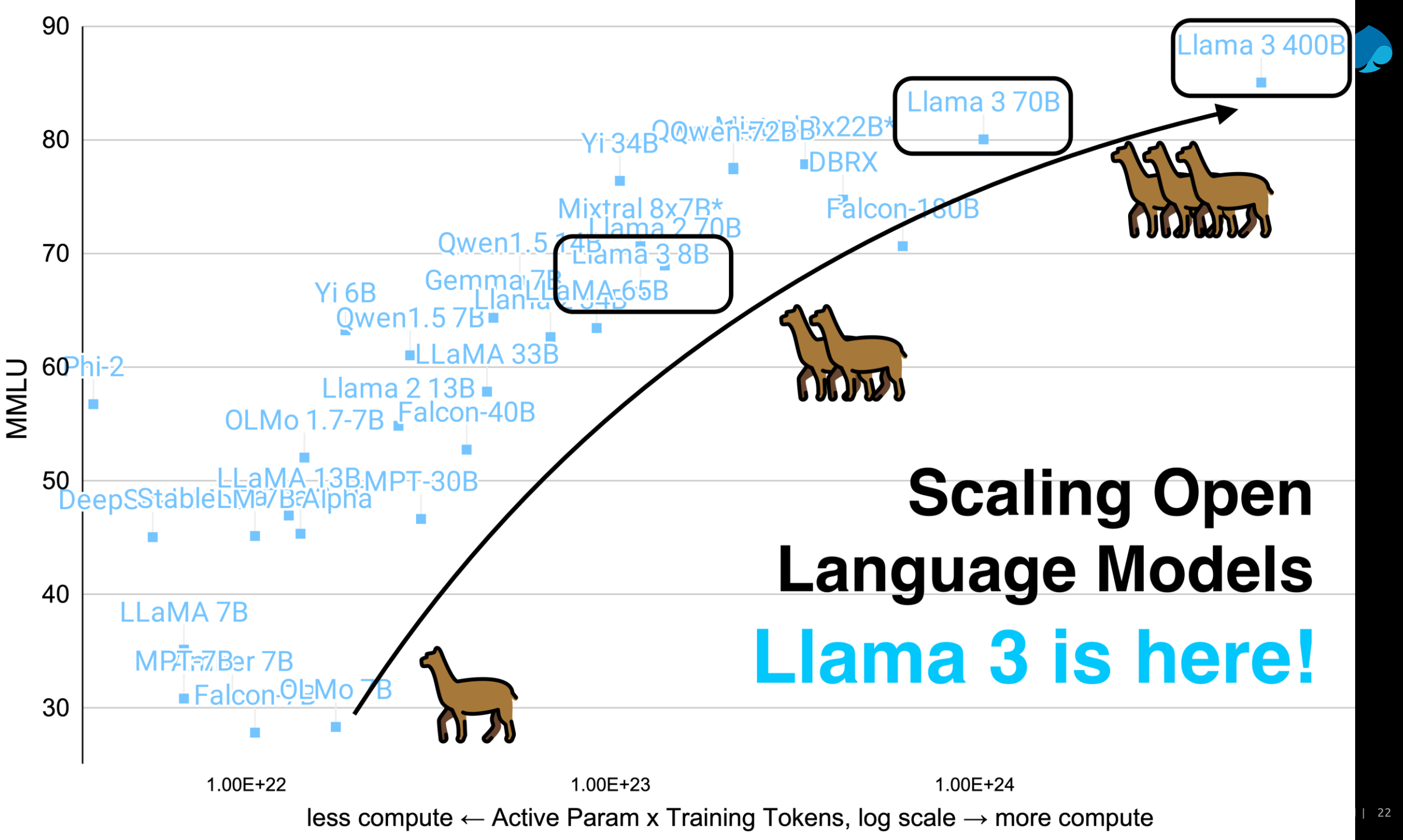
Are the scaling Laws for LLMs shifting?

Train smaller models on more data and longer, they can perform as well as larger models based on the Chinchilla scaling.



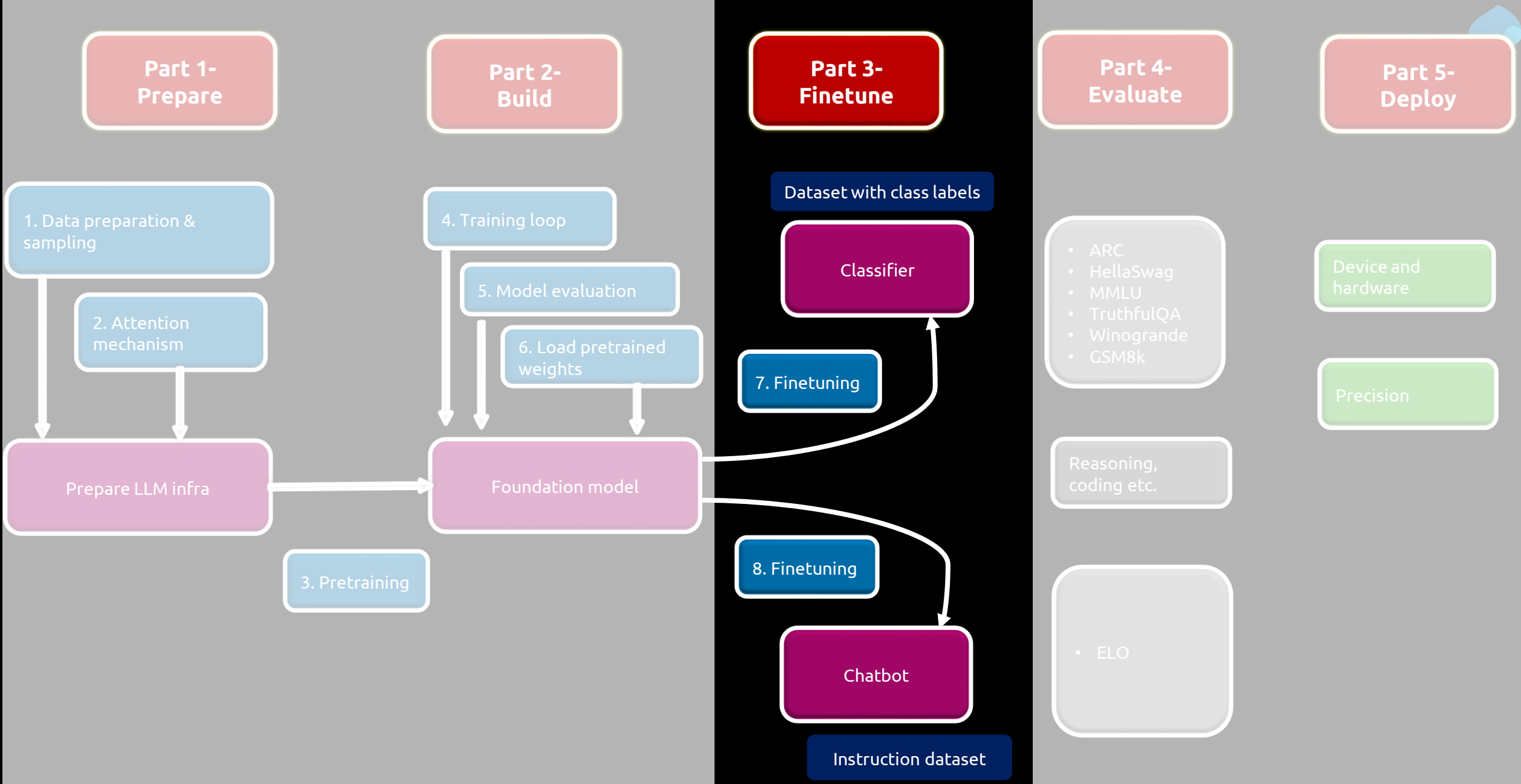


HTTPS://HUGGINGFACE.CO/SPACES/LVWERRA/HARMS-LAW


90
80
70
60
50
40
30
MMLU
Llama 3 400B
Llama 3 70B
Yi 34B
Qwen-72B
DBRX
Mixtral 8x7B*
Falcon-180B
Llama 2 70B
Qwen1.5 14B
Llama 3 8B
Gemma 7B
Yi 6B
LLaMA-65B
Qwen1.5 7B
LLaMA 33B
Phi-2
Llama 2 13B
Falcon-40B
OLMo 1.7-7B
LLaMA 13B
MPT-30B
DeepSeek LLM 7B Alpha
LLaMA 7B
MPT-7B
Mistral 7B
Falcon-7B
OLMo 7B
Scaling Open Language Models
Llama 3 is here!
1.00E+22
1.00E+23
1.00E+24
less compute ← Active Param x Training Tokens, log scale → more compute

Closed-source vs. open-weight models

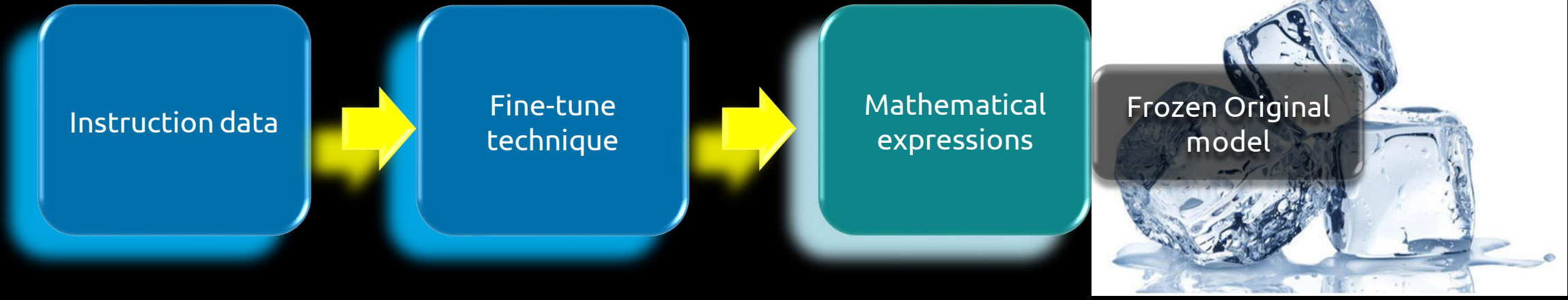
Llama 3 405B from Meta closes the gap between closed-source and open-weight models.



HIGH LEVEL FINE-TUNE OF MODELS

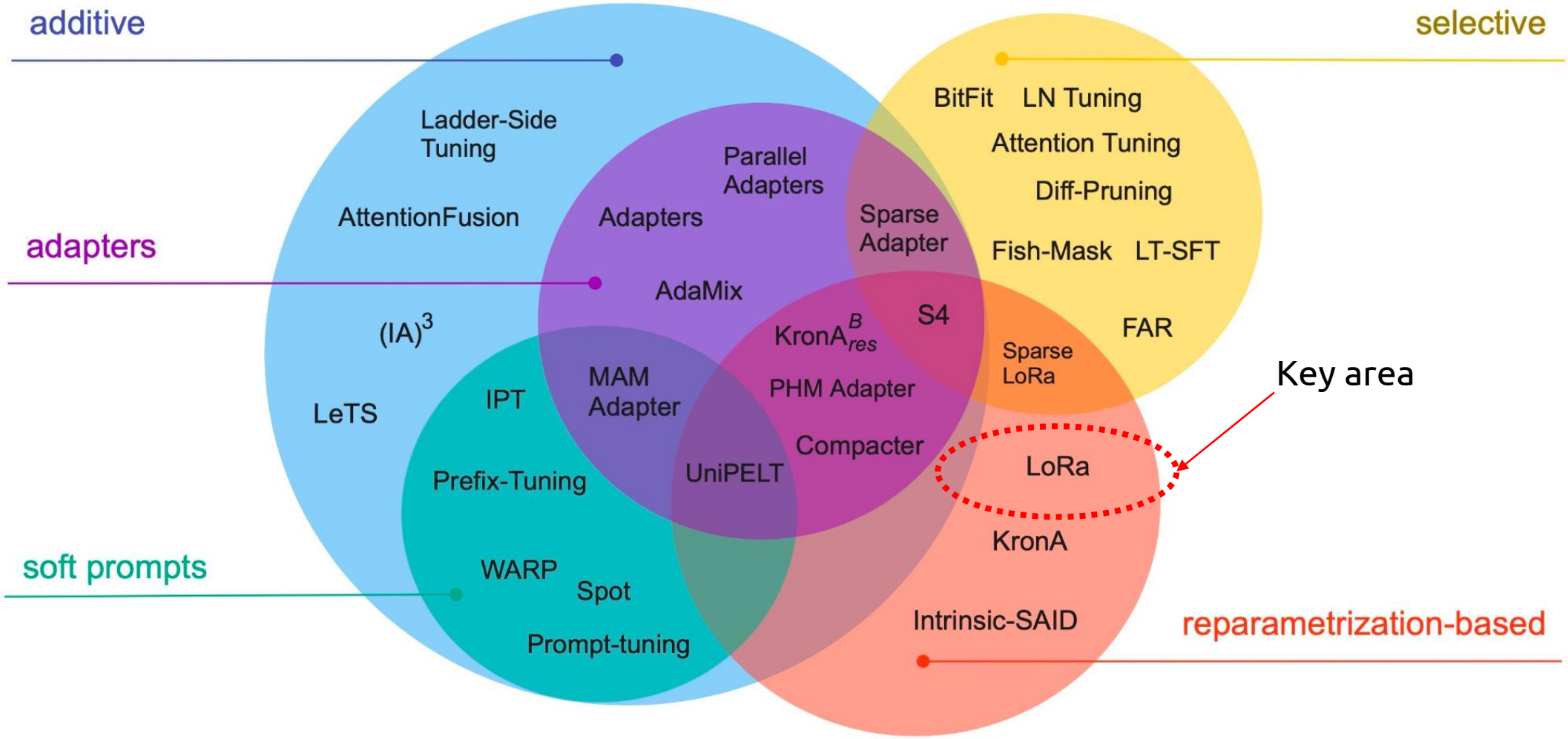
Very small calculations

Billions of parameters



The most important point is that the original model DOES NOT CHANGE

MANY WAYS OF IMPROVING LARGE MODELS

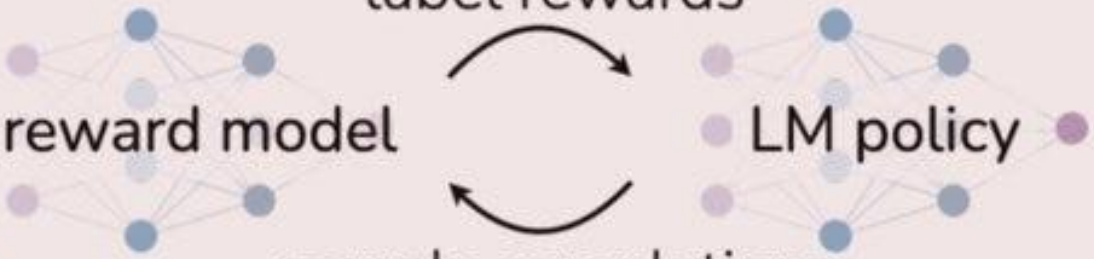


Reinforcement Learning from Human Feedback (RLHF)

x: "write me a poem about the history of jazz"

preference data

maximum likelihood



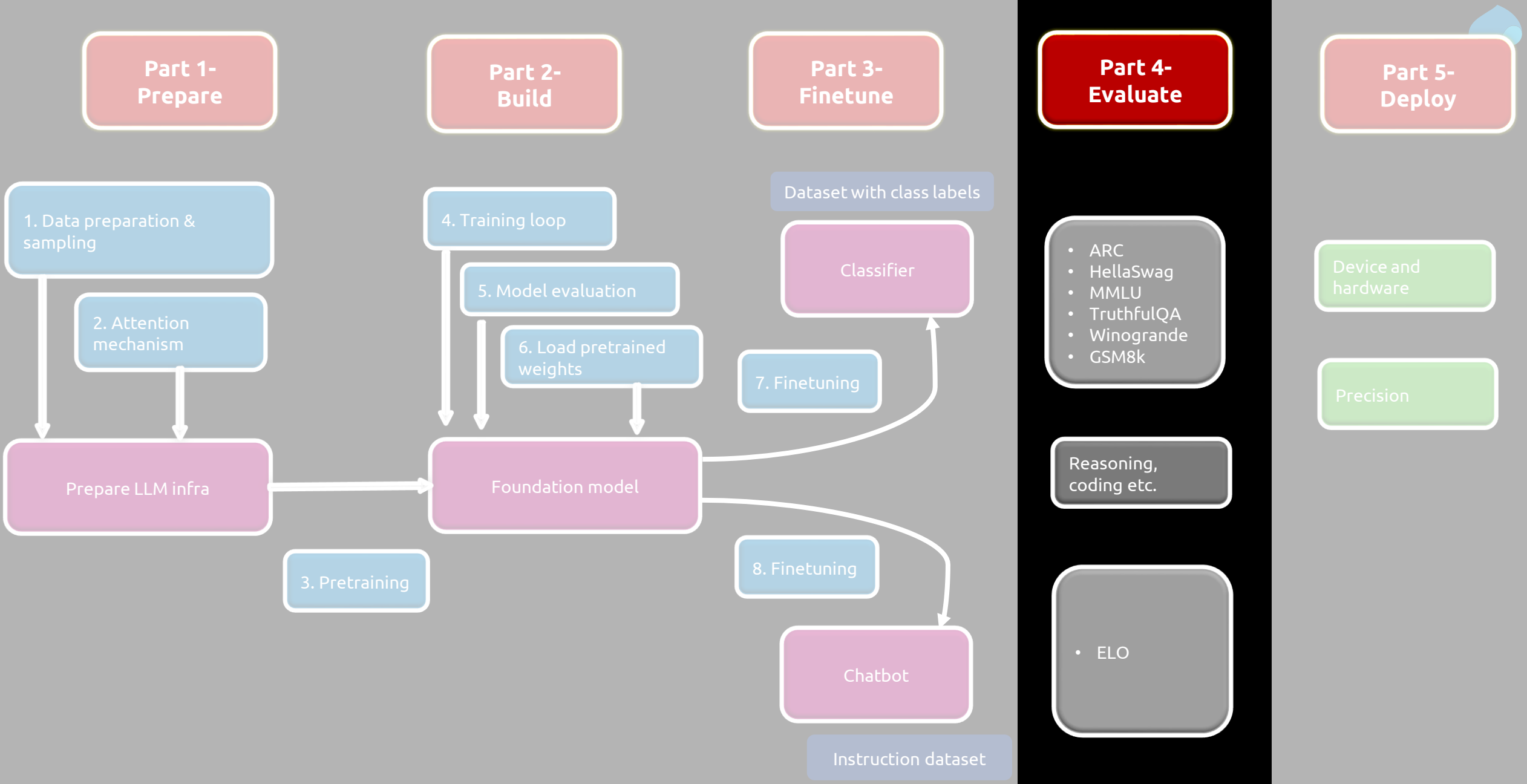
reinforcement learning

Direct Preference Optimization (DPO)

x: "write me a poem about the history of jazz"

preference data

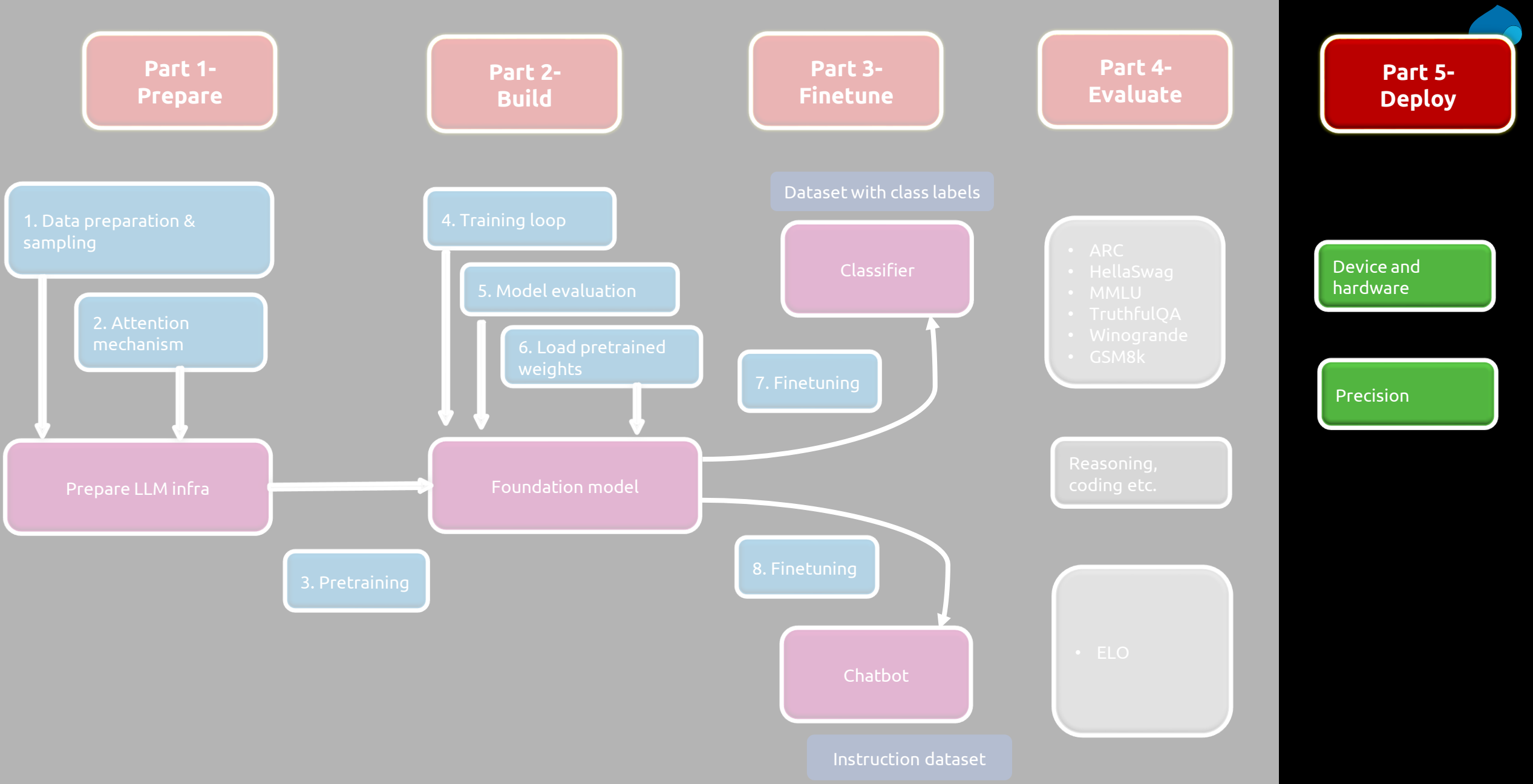
maximum likelihood



LEADER BOARDS

One view
https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard

Chatbot leader board
https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard



MODEL COMPRESSION METHODS

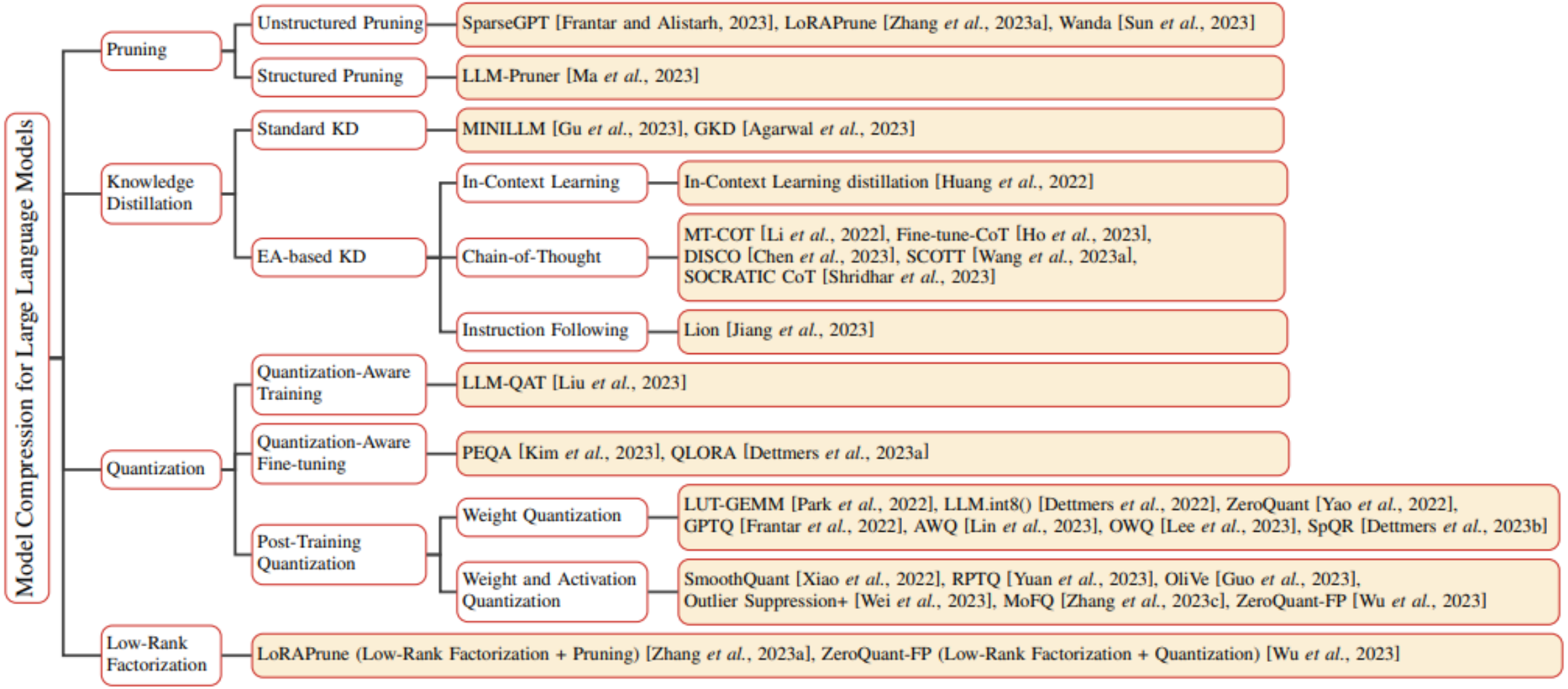


Figure 1: Taxonomy of Model Compression methods for Large Language Models.

https://arxiv.org/pdf/2308.07633.pdf

QUANTIZATION

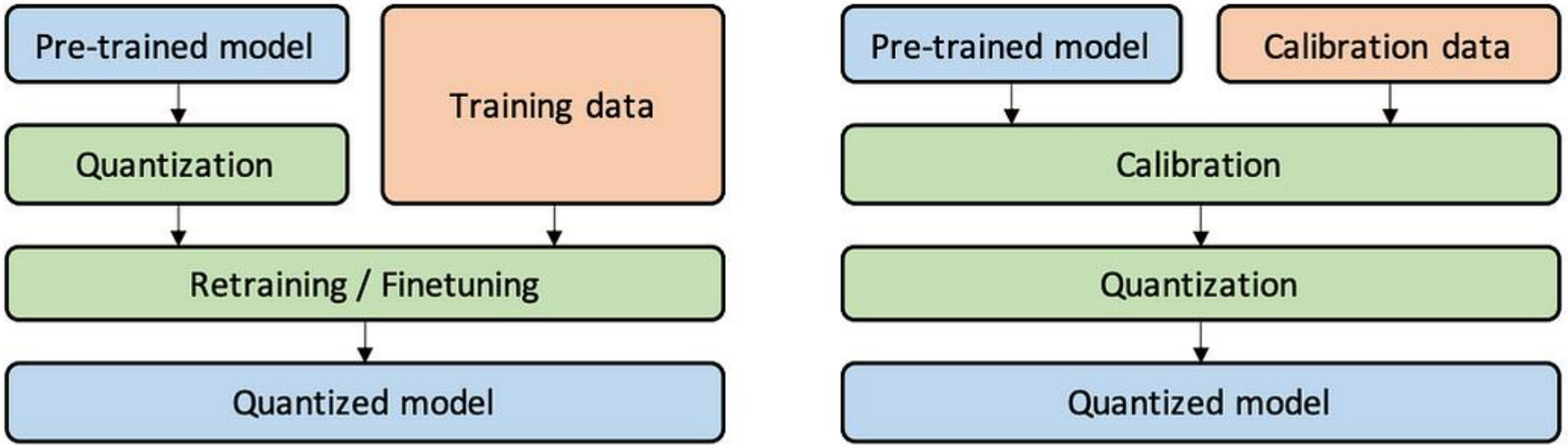


Figure 4: *Comparison between Quantization-Aware Training (QAT, Left) and Post-Training Quantization (PTQ, Right). In QAT, a pre-trained model is quantized and then finetuned using training data to adjust parameters and recover accuracy degradation. In PTQ, a pre-trained model is calibrated using calibration data (e.g., a small subset of training data) to compute the clipping ranges and the scaling factors. Then, the model is quantized based on the calibration result. Note that the calibration process is often conducted in parallel with the finetuning process for QAT.*

Apple's Open-Source SLM Family

Model	Model size	Pretraining tokens	ARC-c	ARC-e	BoolQ	HellaSwag	PIQA	SciQ	WinoGrande	Average	Average w/o SciQ
OpenELM	0.27 B	1.5 T	26.45	45.08	53.98	46.71	69.75	84.70	53.91	54.37	49.31
MobiLlama	0.50 B	1.3 T	26.62	46.04	55.72	51.06	71.11	83.60	53.20	55.34	50.63
OpenELM	0.45 B	1.5 T	**27.56**	**48.06**	**55.78**	**53.97**	**72.31**	**87.20**	**58.01**	**57.56**	**52.62**
TinyLlama	1.10 B	3.0 T	30.12	55.25	57.83	59.20	73.29	-	59.12		55.80
OpenLM	1.00 B	1.6 T	31.00	56.00	**65.00**	61.00	74.00	-	60.00		57.83
MobiLlama	0.80 B	1.3 T	28.84	49.62	60.03	52.45	73.18	85.90	55.96	58.00	53.35
MobiLlama	1.26 B	1.3 T	31.91	56.65	60.34	62.18	74.81	89.10	59.27	62.04	57.53
OLMo	1.18 B	3.0 T	31.06	**57.28**	61.74	62.92	75.14	87.00	59.98	62.16	58.02
OpenELM	1.08 B	1.5 T	**32.34**	55.43	63.58	**64.81**	**75.57**	**90.60**	**61.72**	**63.44**	**58.91**
OpenELM	3.04 B	1.5 T	35.58	59.89	67.40	72.44	78.24	92.70	65.51	67.39	63.18

LLM vs SLM

Aspect	LLM (Large Language Models)	SLM (Small Language Models)
Advantages 	• Deep language understanding • Versatility • Contextual relevance	• Efficient resource utilization • Suitable for smaller datasets • Faster training and inference
Drawbacks 	• Computationally intensive • Data requirements • Potential for bias	• Limited language understanding • Contextual limitations • Task-specific training


Q&A

CONTACT ME

Rajeswaran V (PhD)

Generative AI specialist. AI Futures and AI CoE head

