Mini-batch Coresets for Memory-efficient Language Model Training on Data Mixtures

Dang Nguyen[1], Wenhan Yang[1], Rathul Anand[1], Yu Yang[2], Baharan Mirzasoleiman[1]

[1] Department of Computer Science, UCLA
[2] OpenAI

BigML

UCLA

Training LLMs requires massive GPU memory

- Training an LLM with N billion parameters using mixed precision [1] requires 18N GB of GPU memory [2].

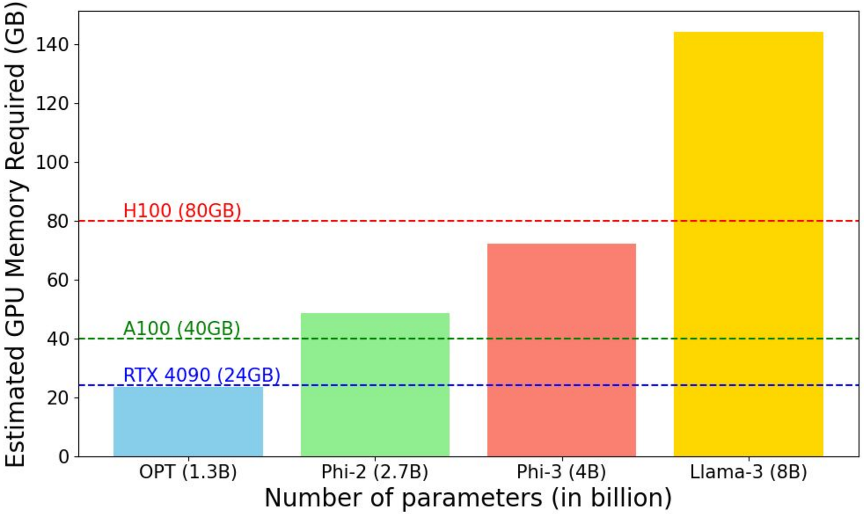
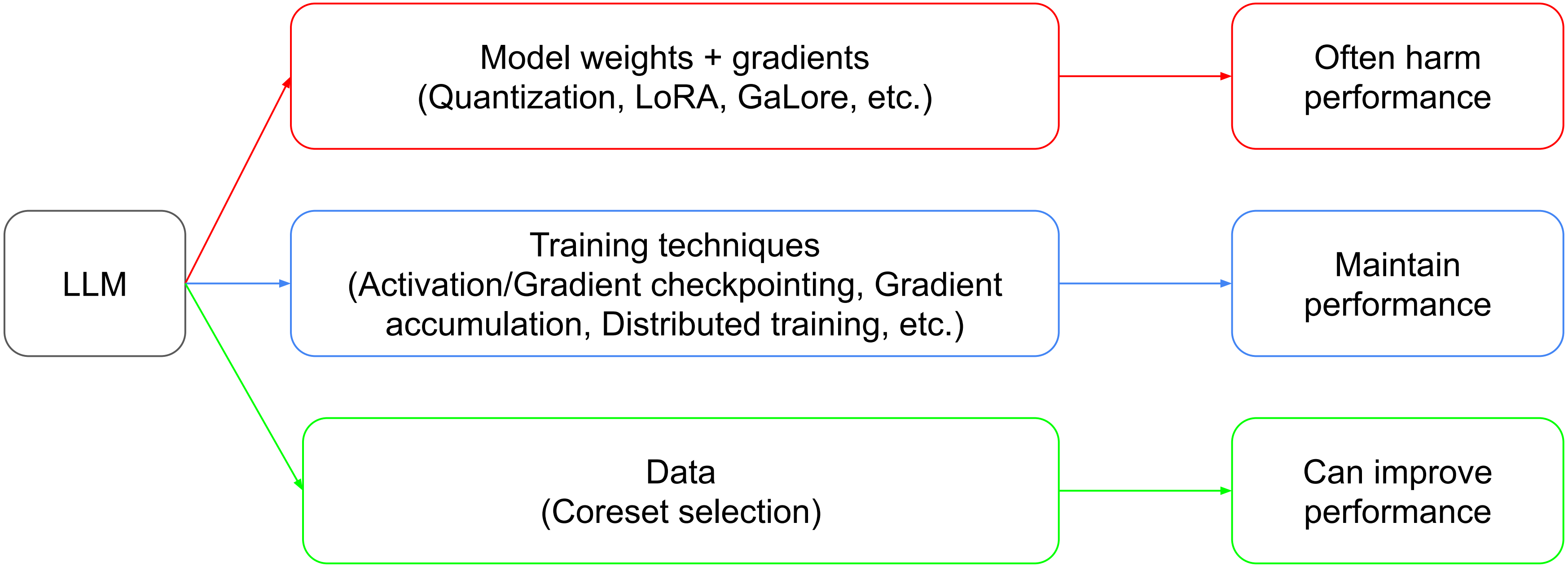


Figure 1. Estimated GPU memory requirement for training LLMs using mixed precision.

[1] Micikevicius, Paulius, et al. "Mixed precision training." arXiv preprint arXiv:1710.03740 (2017).
[2] https://huggingface.co/docs/transformers/model_memory_anatomy

How to reduce GPU memory requirement?



How to reduce GPU memory requirement?

It's possible to stack three different approaches together.

LLM

Model weights + gradients
(Quantization, **LoRA**, GaLore, etc.) → Often harm performance

+

Training techniques
(**Activation/Gradient checkpointing**, **Gradient accumulation**, **Distributed training**, etc.) → Maintain performance

+

We propose:

Data
(**Coreset selection**) → Can improve performance

(Mini-batch) coreset selection

- Training with larger mini-batch has a small variance, thus convergences faster [3].
- A mini-batch coreset is a subset that has the similar gradient to the original mini-batch. Thus, it can be found by solving the gradient matching problem [4].

Coreset size   Big constant

$$S_t^* \in \underset{\boxed{S} \subset \boxed{\mathcal{M}_t^L}, |S| \le \boxed{b}}{\arg\max} \sum_{i \in \mathcal{M}_t^L} \max_{s \in S} [\boxed{C} - \| \boxed{\mathbf{g}_{i,t}} - \boxed{\mathbf{g}_{s,t}} \|],$$

Mini-batch coreset   Original mini-batch   Gradient of original mini-batch   Gradient of mini-batch coreset

[3] Ghadimi, Saeed, and Guanghui Lan. "Stochastic first-and zeroth-order methods for nonconvex stochastic programming." SIAM journal on optimization 23.4 (2013): 2341-2368.
[4] Mirzasoleiman, Baharan, Jeff Bilmes, and Jure Leskovec. "Coresets for data-efficient training of machine learning models." International Conference on Machine Learning. PMLR, 2020.

Challenges of coreset selection for training LLMs

1. Highly Imbalanced Language Data
2. Adam optimizer
3. Very Large Gradient Dimensionality

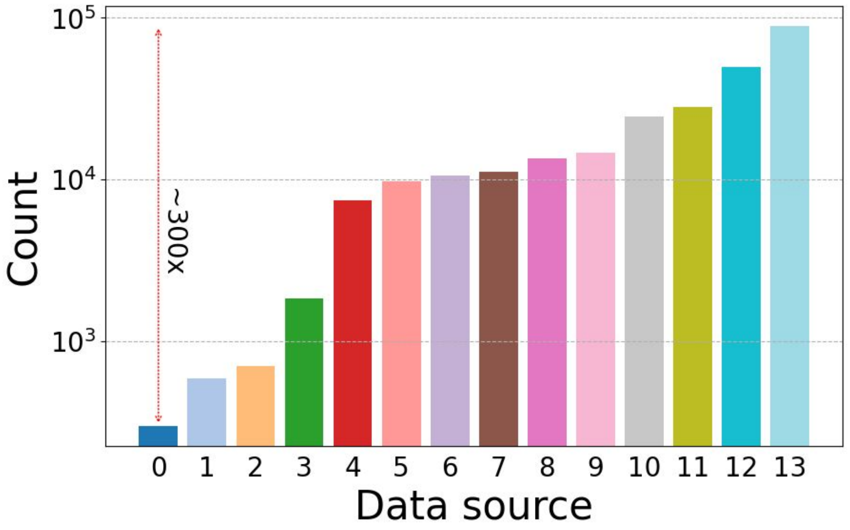


Figure 3. The number of samples from different data sources in MathInstruct.

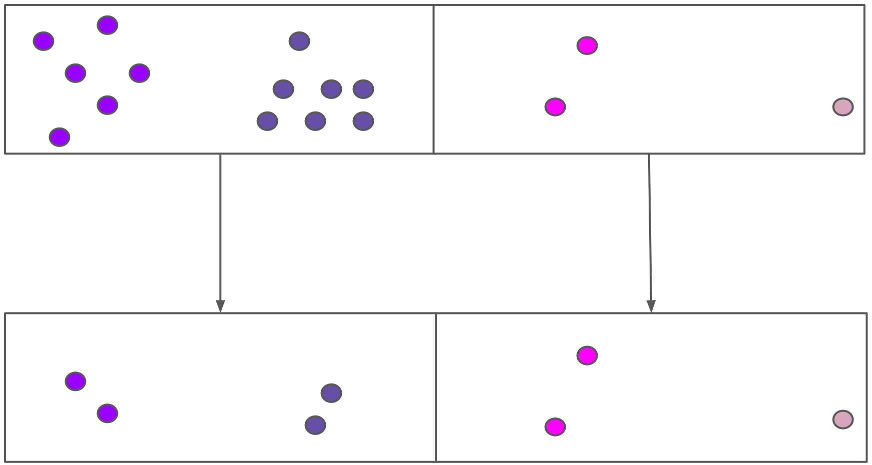
Coresets for Training LLMs (CoLM)

Coresets for Training LLMs (CoLM)

1. A balanced sampling strategy
   a. Keep all data from small sources

   b. Select data from large sources

Original mini-batch



Mini-batch coreset

Coresets for Training LLMs (CoLM)

1. A balanced sampling strategy
    a. Keep all data from small sources
    b. Select data from large sources
2. Adam-like gradient normalization
    a. Normalize gradient to mimic Adam updates

Coresets for Training LLMs (CoLM)

1. A balanced sampling strategy
    a. Keep all data from small sources
    b. Select data from large sources
2. Adam-like gradient normalization
    a. Normalize gradient to mimic Adam updates
3. Sparsified zeroth-order gradient estimation
    a. Use zeroth-order to estimate gradient for only the last V-projection matrix
    b. Sparisfy the gradient further by keeping only ~1% largest entities

Experimental results

CoLM yields the best of both worlds, increasing accuracy while reducing time & memory consumption.

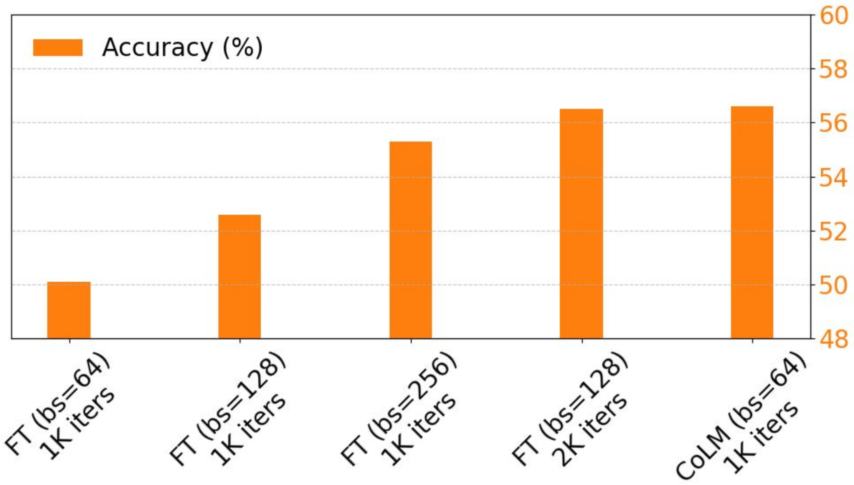


Figure 4. Time vs Accuracy vs Memory of fine-tuning Phi-2 with LoRA on MathInstruct.

Experimental results

CoLM yields the best of both worlds, increasing accuracy while reducing time & memory consumption.

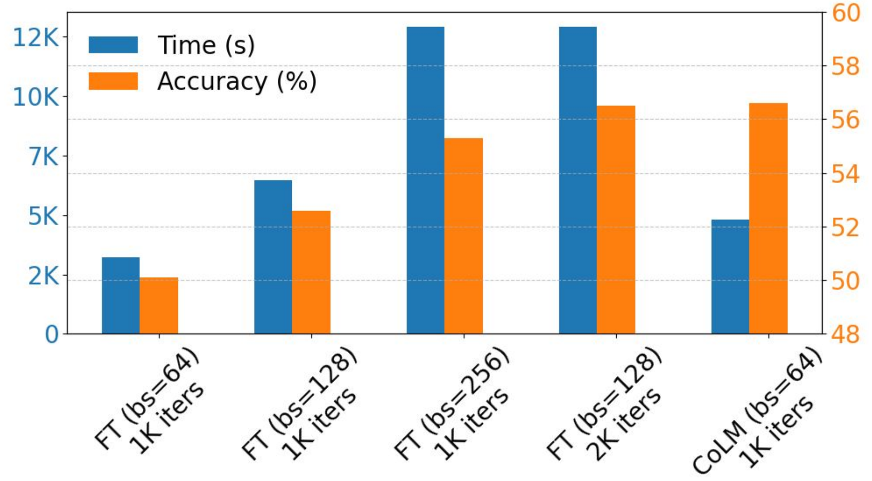


Figure 4. Time vs Accuracy vs Memory of fine-tuning Phi-2 with LoRA on MathInstruct.

Experimental results

CoLM yields the best of both worlds, increasing accuracy while reducing time & memory consumption.

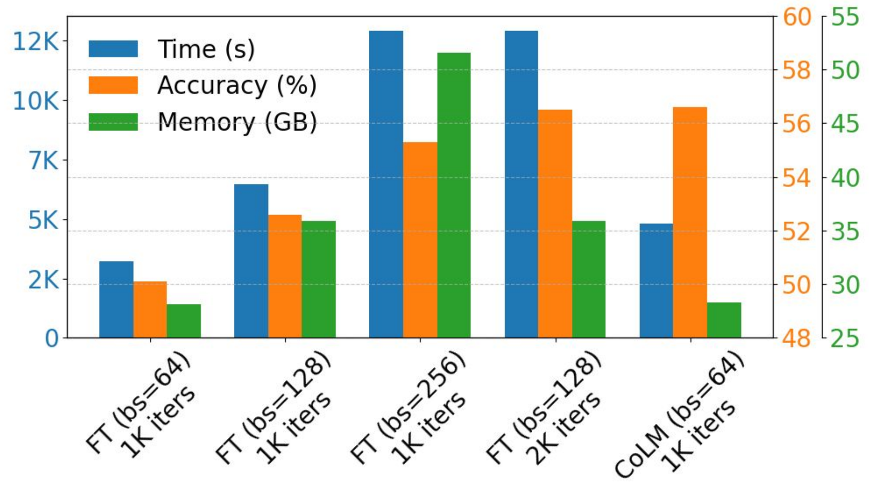


Figure 4. Time vs Accuracy vs Memory of fine-tuning Phi-2 with LoRA on MathInstruct.

Thank you!
Please come visit our poster at Session 3: Fri 25 April 9 AM - 11:30 AM GMT+7

Dang Nguyen nguyentuanhaidang@gmail.com @dangnth97 https://hsgser.github.io/