Questions

- when pretraining and fine tuning, does it retrain the whole model or do they freeze the weights
    - pretty sure they retrain the whole thing

Problem Statement

-

Weight Updates

- M = 100
- N= 100
- Conventional (MxN): 100*100
    - 10000
- D = 10
- Low rank ([MxD]x[DxN]) = (100*10)x(10*100)
    - 100*10 + 10*10 + 100*10 = 2100

only train BA which are matrixes that they propose that they ensure are smaller than original rank of original weight matrixes (that you would have to train if retraining the whole thing)

W0 is frozen and does not receive gradient updates, while A and B contain trainable parameters

Ideal rank

-

Implementation

- https://github.com/microsoft/LoRA/blob/main/examples/NLG/src/model.py

over-parametrized models in fact reside on a low intrinsic dimension ⇒

- models with many params in fact need a lot less params

https://arxiv.org/pdf/1902.00751.pdf

- another way of tackling the problem lora is tackling
- however they introduce inference latency
- but edwards question is why does it introduce inference latency because wouldn't lora do the same since it also introduces extra parameters
    - lora does it sequentially, adapter modules do it after training certain modules (like feed forward)
    - answered by appendix b in the paper

Can you just apply lora to everything? why did authors only apply it to attention module

confusing figures

- figure 3
- figure 4

What is subspace similarity?

feedback for future meetings

- general format for these meeting notes
- eod wednesday find a new paper for next paper