Google DeepMind

Efficient Sketches for Training Data Attribution and Studying the Loss Landscape

Andrea Schioppa

What are gradient and HVP sketches? Which problems can they solve?

Sketches

- A linear projection from a higher dimensional to a lower dimensional Euclidean space that approximately preserves distances and inner products

$$\Phi : \mathbb{R}^N \to \mathbb{R}^D$$

$$\text{Prob}\left(\left| \|\Phi(x)\|_2 - \|x\|_2 \right| \geq \varepsilon \|x\|_2 \right) \leq \delta.$$

- Many applications in Numerical Linear Algebra, e.g. approximate solutions to linear systems and approximating the spectrum of large matrices
- Matrices that are random wrt. some reasonable distributions will work

Applications to Deep Neural Networks

- Situations where we need to store numerous gradients / HVPs
- Training Data Attribution: store one gradient per training point
- Hessian analysis: store one HVP per iteration
- Intrinsic Dimension: do gradient descent in the image of the sketch map; find smallest D on which one achieves 90% of the full performance

Limitations of existing sketching methods

Dense Sketches lack scalability

- Memory grows like O(N D)
- Using layer selection (reduce N) gives incorrect TDA estimates
- Materializing random matrices in chunks [TRAK 2023] requires specialized implementations (CUDA, Triton) and run-time grows like O(D)

The FastFood Transform is not a valid sketching method

- FFD creates random features in O(N) memory and O(N log N) runtime
- Random perturbation + backprop gives a gradient sketch [Li 2018]

$$\mathcal{S}(\nabla_{\theta|\theta_0} L) = \nabla_{\omega|0} L(\theta_0 + \mathbf{FFD}(\omega));$$

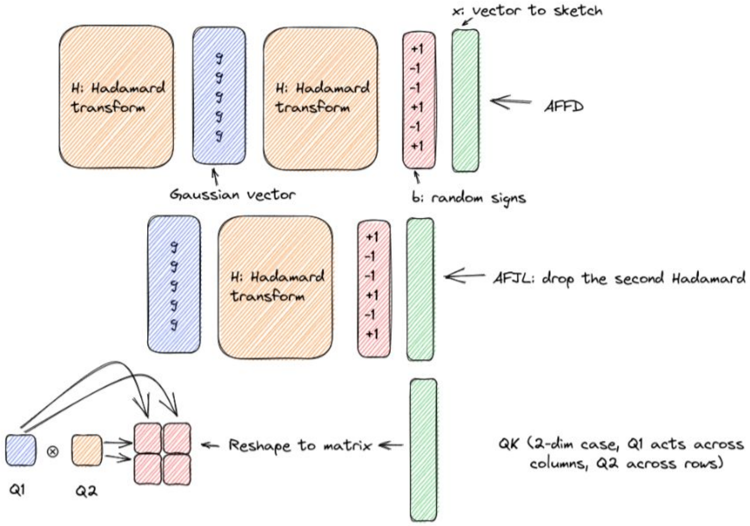
- We prove that using the FFD in this way does not satisfy the sketching definition
- It is more efficient to compute the gradient and sketch it

$$\mathcal{S}(\nabla_{\theta|\theta_0} L) = \Phi(\nabla_{\theta|\theta_0} L).$$

Proposed approach

Fast-Sketching

- We propose 3 methods AFFD, AFJL and QK which are O(N) in memory and O(N log N) in runtime
- Does not require dedicated implementation (JaX code works well on GPUs and TPUs)
- We prove that AFFD and QK are valid sketching algorithms; QK is faster but requires a larger target dimension
- We remove expensive memory bottlenecks from the FFD design that made it slow on modern accelerators; memory accesses are absorbed via Kronecker decompositions



Insights into LLMs

Tasks with large intrinsic dimension

- Our memory-efficient algorithms enable us to investigate scenarios where the intrinsic dimension approaches the model dimension
- We show that on a generative task the intrinsic dimension approaches the model dimension
- Previous studies had shown that the intrinsic dimension is much smaller than the model dimension, but they had focused on fine-tuning LLMs for classifications
- Our finding prevents to use a low value of the intrinsic dimension for generalization bounds on such generative tasks

LLM spectra with fine-tuning

- Two observations found on smaller networks **do not hold** for LLMs
- Obs1 [Positive Hessian]: Negative eigenvalues gradually disappear during training
- Obs2 [Small Subspace]: For classification, the gradient does not align with the eigenspaces corresponding to a few outlier eigenvalues.