Principais contribuições do artigo “QLoRA: Efficient Finetuning of Quantized LLMs”, de Tim Dettmers et al

Aluno: Leandro Carísio Fernandes

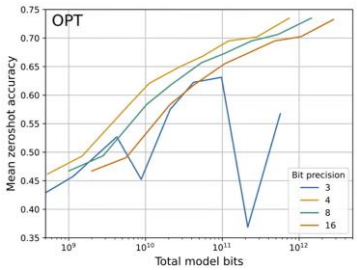
- É um artigo muito denso, difícil de ler, que foca na representação interna de números.
- Quantização: a ideia é usar menos bits para representar os números. Pode-se usar um inteiro para representar float. Exemplo do vídeo https://www.youtube.com/watch?v=y9PHWGOa8HA:

Map: {Index: 0, 1, 2, 3 -> Values: -1.0, 0.3, 0.5, 1.0}

Input tensor: [10, -3, 5, 4]

1. Normalize with absmax: [10, -3, 5, 4] -> [1, -0.3, 0.5, 0.4]
2. Find closest value: [1, -0.3, 0.5, 0.4] -> [1.0, 0.3, 0.5, 0.5]
3. Find the associated index: [1.0, 0.3, 0.5, 0.5] -> [3, 1, 2, 2] -> store
4. Dequantization: load -> [3, 1, 2, 2] -> lookup -> [1.0, 0.3, 0.5, 0.5] -> denormalize -> [10, 3, 5, 5]

- Na quantização sempre há perdas. Tradicionalmente não se preocupa com outliers na quantização, mas no caso de machine learning são valores que tendem a ser muito importantes, não podem ser descartados.
- Mas não descartar tem um efeito de desperdiçar bits. Por exemplo, suponha um caso onde os dados estão todos concentrados em -3 a +3, exceto por um único parâmetro igual a -10. A quantização é feita considerando o intervalo de -10 a +3. Se estamos usando 4 bits (16 intervalos), cada intervalo vai ter tamanho ~0.81. Para representar o intervalo de -3 a +3 será usado 8 níveis e, para representar o -10, será usado mais um nível. Os outros 7 níveis ficarão vazios, sem nenhum dado.
- Uma das coisas que o artigo propõe é usar a quantização em bloco. Onde tem o outlier, vai continuar assim. Entretanto, onde não tem (a regra), a quantização será feita de -3 a +3 em vez de -10 a +3. O efeito é o de melhorar a representação numérica para a maior parte dos casos. Blocos menores é melhor.
- Qualidade: menos bits e mais parâmetros é melhor que mais bits e menos parâmetros. Quantização com 3 bits é instável:



- Double quantization: quantização da quantização (faz a quantização duas vezes)
- Paged optimizers: faz o gerenciamento dos dados automaticamente entre CPU/GPU
- N4-bit Normal Float (NF4) => é um modelo para quantização de dados de distribuição normal. Em vez de quantizar com intervalos iguais, NF4 vai quantizar com quantis iguais. Por exemplo, supondo 20 intervalos, vai ter um número representando os quantis 50% a 55% e outro representando os quantis 95% a 100%. Entretanto, como há na normal muito mais dado ao redor da média/mediana e esse esquema de quantização foca no quantil (quantizando aproximadamente a mesma quantidade de números em cada bin), o intervalo associado ao primeiro caso será bem mais curto do que o segundo.