11/8

Quantization

$(-1)^{sign} \times (1+fraction) \times 2^{Exponent-b}$

Message

↳ It turns out that FP8 representation has many advantages over INT8 for inference, but we need to carefully tune.

$x^{(q)} = s \cdot \left\lfloor \frac{x}{s} \right\rfloor$; express the quantized input.

<17>

scale factor 's'

HRNet, DeepLab V3, ViT, BERT-base, ResNet 18

E4M3, E5M2

E4M3

$NaN = S.1111.111_2$

GPT3.int8()

「model = GPT2, PPL = 33.5, outliers = countil, 1-sided = 1.」 (↓)

+) Iteration 바꾸기.

11/8 reproduct challenge 할까

① LDPC (64/120) 과제 BER 구해서 sum product algorithm / min-sum algorithm 과 비교. / 추석 전까지 (MSA, SA)

② sparity 비율 「self-attention」

③ FLOPS 계산 가능?

「Model size VS quantization」

↳ quantized model의 성능 = model 크기 증가 (tend to degrade as model get lager) 성능↓↓.

∴ why? emergence of outliers.

「emergence of 'outliers features'」

$1.01 = 1 + \frac{1}{2} \times 0 + \frac{1}{2^2} \times 1$

$= \frac{5}{4}$

---

quartiles

= (-8, -7, -6) : GPT2

w/outliers

「with」

LLM.int8()

① 8-bit vector-wise quantization
② 16-bit decomposition.

---

3) zero quant = 「3번째 주제」

↳ 이전) ① Not really consider GPT -3 style models (decoder-only).

② Focusing only on high-precision quantization. (FP16/INT8)

③ KD = knowledge distillation.

Who : teacher model.

---

1) fine-grained hardware friendly quantization.

↳ INT8 cannot fully capture different numerical ranges.

(multiplicand precision).
real(2)/ 16-bit.

↳ architecture · Volta, Turing, NVIDIA, Ampere.

---

sequential layer-by-layer KD

$$\min_{L_i^q} \| L_i(X_{i-1}) - L_i^q(X_{i-1}^q) \|_2^2.$$

<2> $\| W \cdot X - \hat{W} \cdot X \|_F^2$

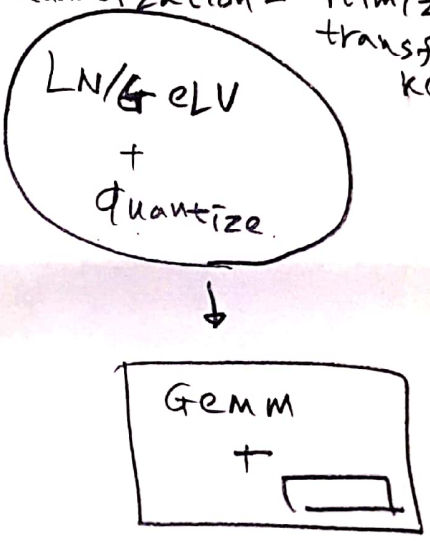
「layer by layer」 optimization.

KD = knowledge distillation.

↳ low precision quantization, KD requires you to load both teacher and student,

---

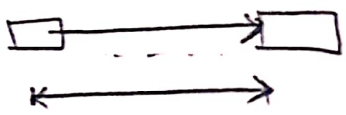
Quantization-optimized transformer kernels.



LKD
「Large knowledge distillation」

---

end-to-end average latency



time = latency.

VS

W8A8 ⟷ W16A16

5배↑.

256

↳ 128

= 28.13

128

↳ 140

↳ 「milliseconds」

BERT model →

다양한 모델 비교 ↵