ZeroQuant: Efficient and Affordable Post-Training Quantization for Large-Scale Transformers

Zhewei Yao, Reza Yazdani Aminabadi, Minjia Zhang, Xiaoxia Wu, Conglong Li, Yuxiong He

Microsoft

Presenter: Xinyu Lian
ZeroQuant: Efficient and Affordable Post-Training Quantization for Large-Scale Transformers
Background

Bridges the Gap between the Supply and Demand of AI Computing

Model compression bridges the gap.

Assume data are FP16.
Background

Bridges the Gap between the Supply and Demand of AI Computing

Model compression:
Pruning, sparsity, quantization, etc
# Motivation: Save Energy

## Less Bit-Width → Less Energy

| Operation        | Energy [pJ] |
|------------------|-------------|
| 8 bit int ADD    | 0.03        |
| 32 bit int ADD   | 0.1         |
| 16 bit float ADD | 0.4         |
| 32 bit float ADD | 0.9         |
| 8 bit int MULT   | 0.2         |
| 32 bit int MULT  | 3.1         |
| 16 bit float MULT| 1.1         |
| 32 bit float MULT| 3.7         |

Rough Energy Cost For Various Operations in 45nm 0.9V
Key Concepts: What is Quantization

Quantization is the process of constraining an input from a continuous or otherwise large set of values to a discrete set.

The difference between an input value and its quantized value is referred to as quantization error.
An affine mapping of integers to real numbers $r = S(q - Z)$

Key Concepts: Linear Quantization
**Key Concepts: Symmetric Linear Quantization**

**Full range mode**

\[ r = \begin{cases} \text{Floating-point range} & \text{for } r_{\min} \leq r \leq r_{\max} \\ \times S & \text{for } r_{\min} > r \text{ or } r > r_{\max} \end{cases} \]

\[ S = \frac{r_{\max} - r_{\min}}{q_{\max} - q_{\min}} \]

- use full range of quantized integers
- example: PyTorch’s native quantization, ONNX

| Bit Width | \( q_{\min} \) | \( q_{\max} \) |
|-----------|----------------|----------------|
| 2         | -2             | 1              |
| 3         | -4             | 3              |
| 4         | -8             | 7              |
| \( N \)   | \(-2^{N-1}\)   | \(2^{N-1}-1\)  |
Key Concepts: Quantization Granularity

- Per-Tensor Quantization
- Per-Channel Quantization
- Group Quantization
**Challenge**

Table 1: Post training quantization results of GPT-3\textsubscript{350M} on 20 zero-shot evaluation datasets. Here WxAy means x-/y-bit for weight/activation. Particularly, for W4/8, we quantize the MHSA’s weight to INT8 and FFC’s weight to INT4. Please see Table I.1 for the results of all 20 tasks.

| Precision | Lambada (↑) | PIQA (↑) | OpenBookQA (↑) | RTE (↑) | ReCoRd (↑) | Ave. 19 Tasks (↑) | Wikitext-2 (↓) |
|-----------|-------------|----------|----------------|---------|------------|-------------------|----------------|
| W16A16    | 49.3        | 66.3     | 29.4           | 53.8    | 75.1       | 38.9              | 21.5           |
| W8A16     | 49.3        | 66.1     | 29.6           | 54.2    | 74.8       | 38.5              | 22.1           |
| **W16A8** | **44.7**    | **64.8** | **28.2**       | **52.7**| **69.2**   | **37.8**          | **24.6**       |
| W8A8      | 42.6        | 64.1     | 28.0           | 53.1    | 67.5       | 37.8              | 26.2           |
| W4/8A16   | 0.00        | 51.4     | 30.2           | 52.7    | 16.1       | 28.9              | 1.76e5         |

- INT8 activation quantization causes the primary accuracy loss.
Challenge

Activation Range of Each Token for Different Layers

Range of Each Row for Different Attention Output Matrices
Key ideas: Fine-grained Quantization

- Weights Quantization: Group-Wise
Key ideas: Fine-grained Quantization

- **Weights Quantization**: Group-Wise
  - First work on Group-Wise Quantization for Post-Training Quantization
Key ideas: Fine-grained Quantization

- **Weights Quantization**: Group-Wise
  - First work on Group-Wise Quantization for Post-Training Quantization
  - Optimize for Ampere Architecture (A100)
    - Warp Matrix Multiply and Accumulate tiling size
Key ideas: Fine-grained Quantization

- **Weights Quantization:** Group-Wise
  - First work on Group-Wise Quantization for Post-Training Quantization
  - Optimize for Ampere Architecture (A100)
    - Warp Matrix Multiply and Accumulate tiling size

No details provided on it
Key ideas: Fine-grained Quantization

- **Weights Quantization:** Group-Wise ✓

- **Activations:** Token-wise Quantization
  - Finer-grained
  - Dynamically calculate the min/max range
  - Kernel Fusion
Key ideas: Knowledge Distillation

- Layer-by-layer distillation (LKD) algorithm
  - Teacher Model: Original (i.e., unquantized) version
    - Use the output of the $L_{k-1}$ as the input of $L_k$

$$
\mathcal{L}_{LKD,k} = MSE \left( L_k \cdot L_{k-1} \cdot L_{k-2} \cdot ... \cdot L_1(\mathbf{X}) - \hat{L}_k \cdot L_{k-1} \cdot L_{k-2} \cdot ... \cdot L_1(\mathbf{X}) \right),
$$
Key ideas: Knowledge Distillation

● Layer-by-layer distillation (LKD) algorithm

○ Benefit:
  ■ No need to hold a separate teacher
  ■ Reduce the memory overhead of optimized states
  ■ The training does not depend on the label or even original training data
Key ideas: Optimized Transformer Kernels

- CUTLASS INT8 GeMM
- Fusing Token-wise Activation Quantization
Evaluation Methodology

- **Models:**
  - Bert
    - $Bert_{base}$ and $Bert_{large}$ on GLUE benchmark
  - GPT3
    - $GPT - 3_{350m}$ and $GPT - 3_{1.3B}$ on 20 zero-shot evaluation tasks
## Experimental Results

### Accuracy

Table 3: Result of BERT\textsubscript{large} on the development set of GLUE benchmark (except WNLI). \(^+\)We extensively tuned the learning rate for QAT (see Appendix F for more details).

| Precision (Method) | CoLA  | MNLI-m | MNLI-mm | MRPC  | QNLI  | QQP   | RTE   | SST-2 | STS-B | Ave.   | Ave. Time (s) |
|--------------------|-------|--------|---------|-------|-------|-------|-------|-------|-------|-------|---------------|
| W16A16 (Baseline)  | 63.35 | 86.65  | 85.91   | 87.99 | 91.62 | 92.24 | 91.08 | 88.08 | 74.01 | 93.46 | 90.34/90.11   |
| W8A8 [76] (QAT)    | —     | —      | —       | —     | —/90.9| 91.74 | 90.12 | —     | —     | —    | —             |
| W8A8 (QAT)\(^+\)   | 59.85 | 86.65  | 86.35   | 85.29 | 89.43 | 92.55 | 91.60 | 88.60 | 61.37 | 93.23 | 87.55/87.65   |
| W8A8 (PTQ)         | 60.57 | 75.69  | 76.94   | 81.13 | 84.93 | 88.49 | 84.04 | 74.35 | 46.93 | 91.74 | 62.75/55.77   |
| W8A8 (ZeroQuant)   | 63.38 | 86.52  | 85.64   | 87.75 | 91.50 | 92.31 | 91.09 | 88.05 | 72.56 | 93.35 | 90.45/90.19   |
| W4/8A16 (PTQ)      | 0.00  | 16.85  | 33.24   | 68.38 | 80.89 | 51.25 | 63.18 | 0.00  | 52.71 | 52.41 | -5.74/-8.51   |
| W4/8A16 (ZeroQuant)| 62.99 | 84.77  | 84.42   | 87.50 | 91.16 | 91.63 | 90.03 | 86.41 | 48.01 | 92.16 | 89.49/89.28   |
| W4/8A16 (ZeroQuant-LKD)| 63.72 | 84.90  | 84.81   | 87.99 | 91.39 | 91.45 | 90.34 | 86.92 | 51.62 | 92.43 | 89.46/89.29   |
| W4/8A8 (ZeroQuant) | 62.34 | 84.62  | 84.25   | 87.75 | 91.38 | 91.87 | 89.86 | 86.09 | 47.65 | 91.97 | 89.39/89.17   |
| W4/8A8 (ZeroQuant-LKD)| 63.51 | 84.70  | 84.71   | 88.73 | 91.99 | 91.73 | 90.25 | 86.74 | 49.82 | 92.09 | 89.34/89.08   |

\[ \text{Ave.} = \frac{\text{Ave. Time}}{\text{Ave.}} \]
### Experimental Results

#### Accuracy

Table 3: Result of $\text{BERT}_\text{large}$ on the development set of GLUE benchmark (except WNLI). †We extensively tuned the learning rate for QAT (see Appendix F for more details).

| Precision (Method)          | CoLA | MNLI-m | MNLI-mm | MRPC | QNLI | QQP | RTE | SST-2 | STS-B | Ave.  | Ave. Time (s) |
|-----------------------------|------|--------|---------|------|------|-----|-----|-------|-------|-------|---------------|
| W16A16 (Baseline)           | 63.35| 86.65  | 85.91   | 87.99/91.62 | 92.24 | 91.08/88.08 | 74.01 | 93.46 | 90.34/90.11 | 85.03 | N/A           |
| W8A8 [76] (QAT)             | —    | —      | —       | —    | —/90.9| 91.74 | —   | 90.12/— | —     | —    | —             |
| W8A8 (QAT)†                 | 59.85| 86.65  | 86.35   | 85.29/89.43 | 92.55 | 91.60/88.60 | 61.37 | 93.23 | 87.55/87.65 | 82.78 | 7181          |
| W8A8 (PTQ)                  | 60.57| 75.69  | 76.94   | 81.13/84.93 | 88.49 | 84.04/74.35 | 46.93 | 91.74 | 62.75/55.77 | 73.54 | 31            |
| W8A8 (ZeroQuant)            | 63.38| 86.52  | 85.64   | 87.75/91.50 | 92.31 | 91.09/88.05 | 72.56 | 93.35 | 90.45/90.19 | 84.81 | 0             |
| W4/8A16 (PTQ)               | 0.00 | 16.85  | 33.24   | 68.38/80.89 | 51.25 | 63.18/0.00 | 52.71 | 52.41 | -5.74/-5.51 | 35.73 | 31            |
| W4/8A16 (ZeroQuant)         | 62.99| 84.77  | 84.42   | 87.50/91.16 | 91.63 | 90.03/86.41 | 48.01 | 92.16 | 89.49/89.28 | 81.23 | 0             |
| W4/8A16 (ZeroQuant-LKD)     | 63.72| 84.90  | 84.81   | 87.99/91.39 | 91.45 | 90.34/86.92 | 51.62 | 92.43 | 89.46/89.29 | 81.85 | 550           |
| W4/8A8 (ZeroQuant)          | 62.34| 84.62  | 84.25   | 87.75/91.38 | 91.87 | 89.86/86.09 | 47.65 | 91.97 | 89.39/89.17 | 81.06 | 0             |
| W4/8A8 (ZeroQuant-LKD)      | 63.51| 84.70  | 84.71   | 88.73/91.99 | 91.73 | 90.25/86.74 | 49.82 | 89.08 | 81.62/550   |     |               |

The LKD seems not help a lot to Bert.
Experimental Results

Table 4: Post training quantization result of GPT-3_{350M} on 20 zero-shot evaluation datasets. Please see Table H.1 for the results of all 20 tasks.

| Precision (Method)       | Lambda (↑) | PIQA (↑) | OpenBookQA (↑) | RTE (↑) | ReCoRd (↑) | Ave. 19 Tasks (↑) | Wikitext-2 (↓) | Time Cost |
|--------------------------|------------|----------|----------------|---------|------------|-------------------|----------------|-----------|
| W16A16                   | 49.3       | 66.3     | 29.4           | 53.8    | 75.1       | 38.9              | 21.5           | N/A       |
| W8A8 (PTQ)               | 42.6       | 64.1     | 28.0           | 53.1    | 67.5       | 37.8              | 26.2           | 7 mins    |
| W8A8 (ZeroQuant)         | 51.0       | 66.5     | 29.2           | 53.4    | 74.9       | 38.7              | 21.7           | 0         |
| W4/8A16 (PTQ)            | 0.00       | 51.4     | 30.2           | 52.7    | 16.1       | 28.9              | 1.76e5         | 7 mins    |
| W4/8A16 (ZeroQuant)      | 10.1       | 58.5     | 27.2           | 52.0    | 56.5       | 33.5              | 88.6           | 0         |
| W4/8A16 (ZeroQuant-LKD)  | 39.8       | 63.8     | 29.4           | 53.1    | 70.1       | 37.0              | 30.6           | 1.1 hours |
| W4/8A8 (ZeroQuant)       | 10.5       | 57.7     | 28.0           | 52.7    | 55.3       | 33.4              | 92.1           | 0         |
| W4/8A8 (ZeroQuant-LKD)   | 37.4       | 61.8     | 28.2           | 53.1    | 68.5       | 36.6              | 31.1           | 1.1 hours |

The LKD seems help a lot to GPT3.
Experimental Results

Inference Speed

Table 6: The speedup of our W8A8 as compared to W16A16. We measure the end-to-end average latency for the entire BERT model, and the time reported is in milliseconds.

| Seq Len BS | Precision | 128  | 256  |
|------------|-----------|------|------|
|            |           | 1    | 2    | 4    | 8    | 16   | 16   | 64   | 128  | 1    | 2    | 4    | 8    | 16   | 16   | 64   | 128  |
| BERT<sub>base</sub> | W1A6      | 2.45 | 3.22 | 3.85 | 5.51 | 9.96 | 17.93| 34.25| 67.08| 3.13 | 4.05 | 5.70 | 10.55| 19.27| 36.69| 71.75| 140.0|
|            | W8A8      | 1.08 | 1.16 | 1.42 | 1.76 | 2.58 | 3.90 | 6.74 | 12.92| 1.22 | 1.44 | 2.08 | 2.88 | 4.10 | 7.80 | 14.66| 28.13|
|            | Speedup   | 2.27 | 2.78 | 2.71 | 3.13 | 3.86 | 4.60 | 5.08 | 5.19 | 2.57 | 2.81 | 2.74 | 3.66 | 4.70 | 4.70 | 4.89 | 4.98 |
| BERT<sub>large</sub> | W1A6      | 5.45 | 6.38 | 8.73 | 13.88| 26.34| 48.39| 92.49| 183.4| 6.39 | 8.94 | 14.66| 27.99| 51.94| 98.78| 195.9| 384.5|
|            | W8A8      | 2.08 | 2.58 | 2.84 | 3.79 | 6.21 | 10.28| 18.86| 36.62| 2.55 | 3.36 | 4.16 | 6.88 | 11.61| 21.20| 41.24| 79.90|
|            | Speedup   | 2.62 | 2.47 | 3.07 | 3.66 | 4.24 | 4.73 | 4.90 | 5.01 | 2.51 | 2.66 | 3.52 | 4.07 | 4.47 | 4.66 | 4.75 | 4.81 |
Own Thoughts

- Industry work
- Very solid work with extensive experiment
- Optimize the GPU kernel to demonstrate the real speedup.

- The ideas are not novel.

Questions:
- Can it scale to larger Models?
- H100 -> FP quantization?