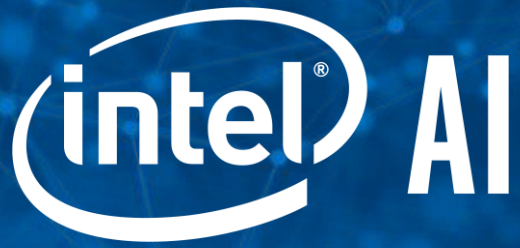


Q8BERT: QUANTIZED 8BIT BERT

Ofir Zafrir, Guy Boudoukh, Peter Izsak, Moshe Wasserblat (Intel AI Lab)

EMC² Workshop @ NeurIPS 2019

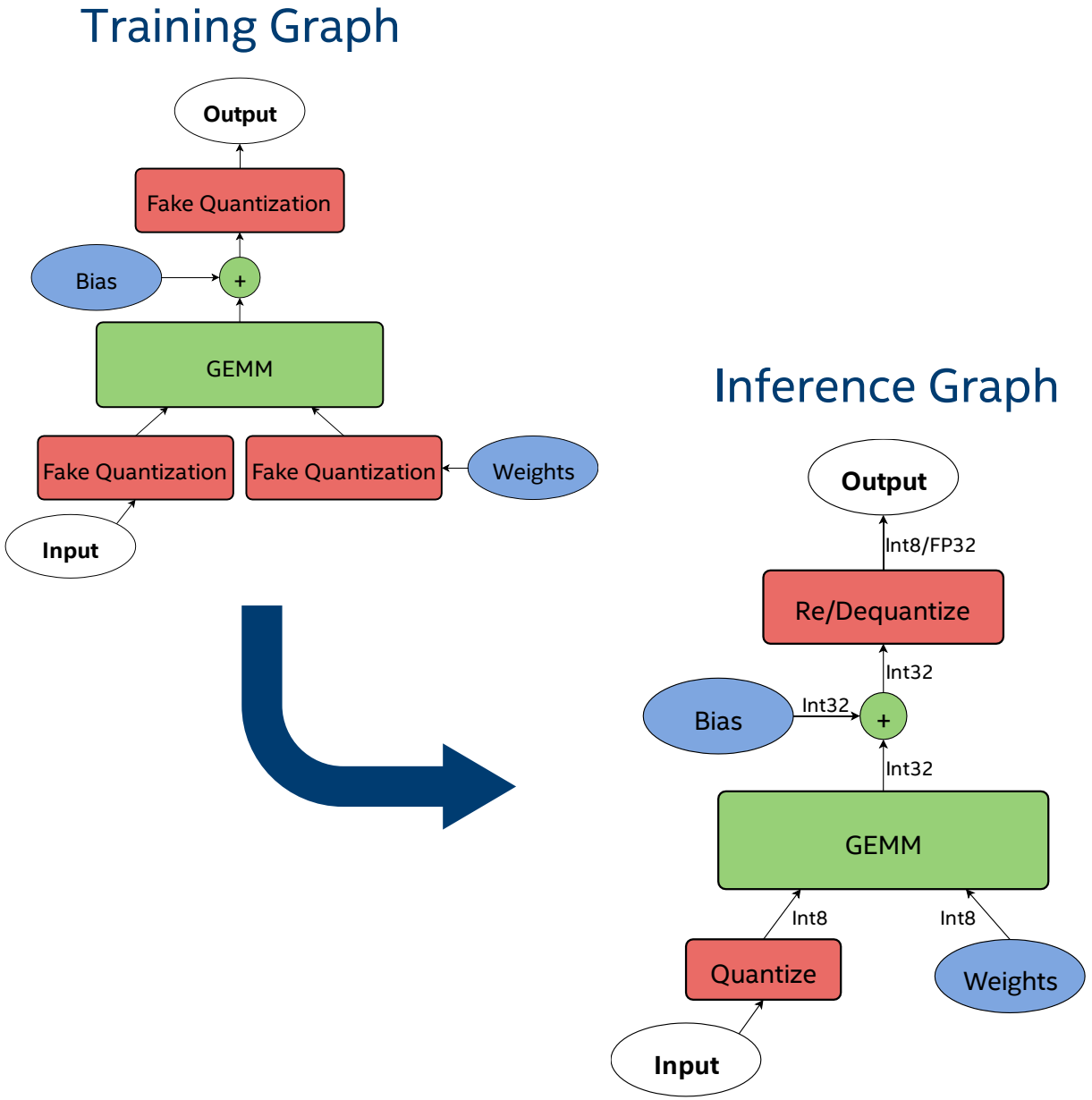
Motivation

- Pre-trained Transformer language models such as BERT, have demonstrated State-of-the-Art results for a variety of NLP tasks
- BERT poses a challenge to deployment in production environments
  - Google: “Some of the models we can build with BERT are so complex that they push the limits of what we can do using traditional hardware”*
- Hardware supporting 8bit Integer quantization is emerging
- Quantization to 8bit Integers can accelerate inference using only 25% of the memory footprint

* https://www.blog.google/products/search/search-language-understanding-bert/

Quantization-Aware Training (QAT)

- Train Neural Networks (NN) to be quantized at the inference stage
- Fake quantization is used to introduce the quantization error training
- We apply Fake Quantization on all GEMM & Word/Position Embedding layers
- Sensitive operations are kept in FP32 (Softmax, LN, GELU)



Experiments & Results

- GLUE benchmark and SQuADv1.1 Datasets
- QAT while Fine-tune pre-trained BERT-Base/Large
- Reported Mean and STD over five experiments
- Relative error induced by quantization is less than 1%

Task	Metric	Baseline Score (STD)	*Our* 8bit BERT Score (STD)	Relative Error
CoLA	MCC	**58.48 (1.54)**	**58.48 (1.32)**	0.00%
MRPC	F1	**90.00 (0.23)**	89.56 (0.18)	-0.49%
MRPC-*L**	F1	90.86 (0.55)	**90.90 (0.29)**	0.04%
QNLI	Acc.	90.30 (0.44)	**90.62 (0.29)**	0.35%
QNLI-*L**	Acc.	91.66 (0.15)	**91.74 (0.36)**	0.09%
QQP	F1	87.84 (0.19)	**87.96 (0.35)**	0.14%
RTE	Acc.	**69.70 (1.50)**	68.78 (3.52)	-1.32%
SST-2	Acc.	**92.36 (0.59)**	92.24 (0.27)	-0.13%
STS-B	PCC	**89.62 (0.31)**	89.04 (0.17)	-0.65%
STS-B-*L**	PCC	**90.34 (0.21)**	90.12 (0.13)	-0.24%
SQuADv1.1	F1	**88.46 (0.15)**	87.74 (0.15)	-0.81%

* -*L* means BERT-Large was used

Comparison with Dynamic Quantization

- Compare QAT to post training quantization
- Dynamic Quantization (DQ)
- Applied DQ on baseline models for each dataset
- DQ method produces significantly worse results over all tasks

Task	Metric	Baseline Score (STD)	DQ 8bit BERT Score (STD)	Relative Error
CoLA	MCC	**58.48 (1.54)**	56.74 (0.61)	-2.98%
MRPC	F1	**90.00 (0.23)**	87.88 (2.03)	-2.36%
MRPC-*L**	F1	**90.86 (0.55)**	88.18 (2.19)	-2.95%
QNLI	Acc.	**90.30 (0.44)**	89.34 (0.61)	-1.06%
QNLI-*L**	Acc.	**91.66 (0.15)**	88.38 (2.22)	-3.58%
QQP	F1	**87.84 (0.19)**	84.98 (0.97)	-3.26%
RTE	Acc.	**69.70 (1.50)**	63.32 (4.58)	-9.15%
SST-2	Acc.	**92.36 (0.59)**	91.04 (0.43)	-1.43%
STS-B	PCC	**89.62 (0.31)**	87.66 (0.41)	-2.19%
STS-B-*L**	PCC	**90.34 (0.21)**	83.04 (5.71)	-8.08%
SQuADv1.1	F1	**88.46 (0.15)**	80.02 (2.38)	-9.54%

*-*L* means BERT-Large was used

Conclusions

- We have presented a method for quantizing BERT to 8bit for a variety of NLP tasks with minimum loss in accuracy
- We compared our Quantization-Aware Training method to a Post-Training Quantization method and shown our method produces significantly better results
- Future directions are to run Q8BERT with supporting hardware and apply other compression methods to BERT and combine them.
- We made our work available for the community in our open-source library NLP Architect

NLP ARCHITECT

NervanaSystems/nlp-architect