Contribution ID: **24** Type: **Standard Talk**

High Granularity Quantization for Ultra-Fast ML Applications on FPGAs

Wednesday 27 September 2023 14:00 (15 minutes)

For many deep learning applications, model size and inference speed at deployment time become a major challenge. To tackle these issues, a promising strategy is quantization.
A straightforward uniform quantization to very low precision often results in considerable accuracy loss. A solution to this predicament is the usage of mixed-precision quantization, founded on the idea that certain sections of the network can accommodate lower precision without compromising performance compared to other sections.

In this work, we present "High Granularity Quantization (HGQ)", an innovative quantization-aware training (QAT) method designed to fine-tune the per-weight and per-activation precision for ultra-low latency neural networks which are to be deployed on FPGAs.

In contrast to what is done in the popular QAT library \texttt{QKeras}, where weights and activations are processed in blocks, HGQ enables each weight and activation to have its unique bitwidth. By optimizing these individual bitwidths alongside the network using gradient descent, the need for training the network multiple times to optimize bitwidths for each block of the network is eliminated. Optimizing at the single-weight level also allows HGQ to find a better trade-off relation between model accuracy and resource consumption.

When multiplication operations in neural networks primarily involve low-bitwidth operands and are implemented with LUTs (in contrast to DSPs), HGQ could demonstrate a significant reduction in on-chip resource consumption by eliminating unnecessary computations without compromising performance. Depending on the specific task, we demonstrate that HGQ has the potential to outperform \texttt{AutoQKeras} by a substantial margin, achieving resource reduction by up to a factor of 10 and latency improvement by a factor of 5 while preserving accuracy. Even in more challenging tasks where the base model is under-fitted, HGQ can still yield considerable enhancements while maintaining the same resource usage.

A functional HGQ framework has been developed using \texttt{tensorflow.keras} has been released, and the Vivado FPGA backend is supported through integrating with \texttt{hl4ml}. The current implementation ensures a bit-to-bit match with the final firmware when there is no numerical overflow, with the added flexibility of adjusting the cover-factor to mitigate such risks.

Primary author: SUN, Chang (ETH Zurich (CH))

Presenter: SUN, Chang (ETH Zurich (CH))

Session Classification: Contributed Talks

Track Classification: Contributed Talks