Optimizing Keyword Spotting on Microcontrollers

Adithya Lakshminarayanan, Professor Brett Meyer
Department of Electrical and Computer Engineering, McGill University

Introduction and Motivation

- Deep learning models are able to attain incredibly high accuracies in fields such as image and speech recognition [1].
- Research has traditionally focused on optimizing for accuracy; models are computationally complex, and run on high-performance hardware.
- Microcontrollers (MCUs) and other resource constrained devices are proliferating. Use cases include IoT devices, robotics, and wearables. [2].
- Neural networks must also be optimized for model complexity and inference latency before they are deployed on MCUs.

Convolutional Neural Nets and Keyword Spotting

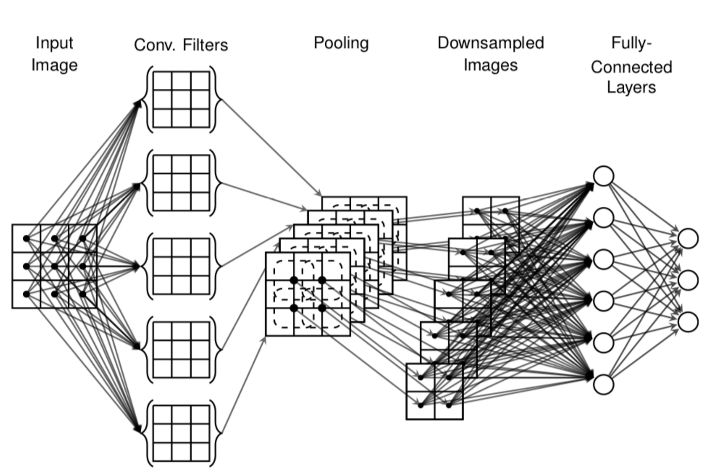


Figure: Typical CNN structure [3]

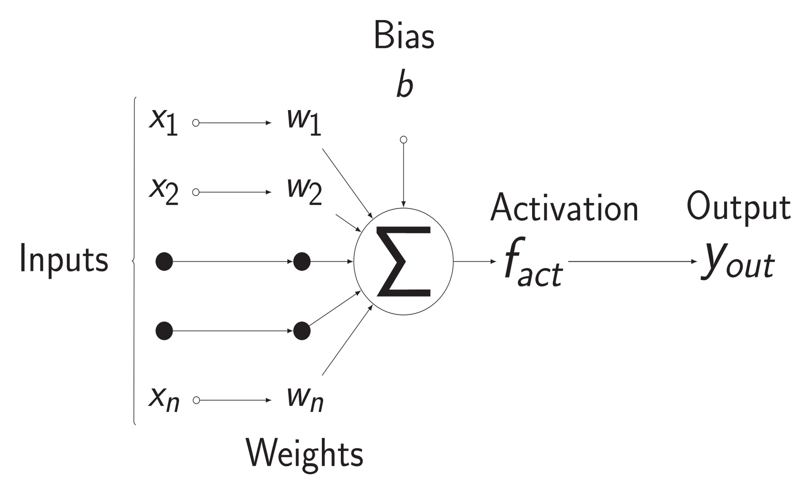


Figure: Artificial Neuron Structure

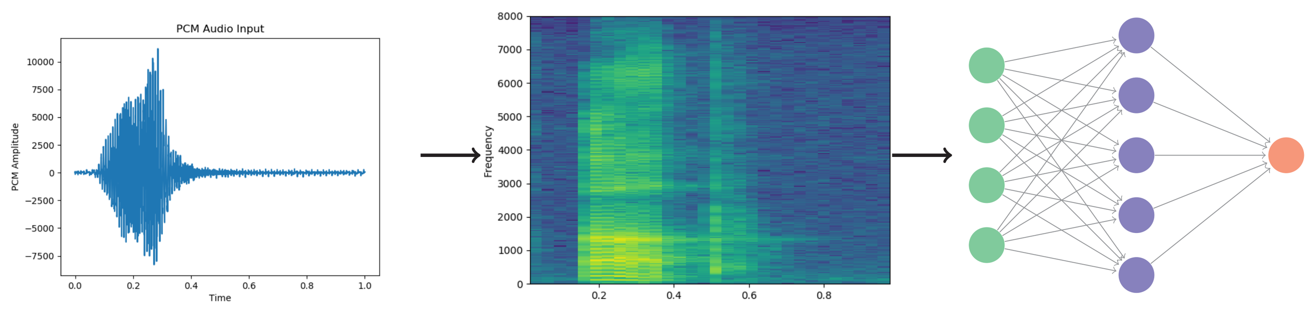


Figure: CNN Implementation of Keyword Spotting

Quantization

- Affine Transformation of floating-point values to integers of lower bit-width.
- $r = S(q - Z)$ where r is a real value, S is the floating point scale factor, q is a quantized integer, and Z is the quantized zero point.
- $S = 2^{-n}$ allows for simplified arithmetic using bitwise shifts.
- Fixed point representation: integer with n fractional, m integer bits.

Example with matrix multiplication:

$$S_3(q_3^{(i,k)} - Z_3) = \sum S_1(q_1^{(i,j)} - Z_1) \times S_2(q_2^{(j,k)} - Z_2) \tag{1}$$

$$\therefore q_3^{(i,k)} = Z_3 + \sum M(q_1^{(i,j)} - Z_1) \times (q_2^{(j,k)} - Z_2) \tag{2}$$

with $M = \frac{S_1 \times S_2}{S_3}$, which is a simple bitwise shift.

Simulated Quantization in Training and Quantized Inference

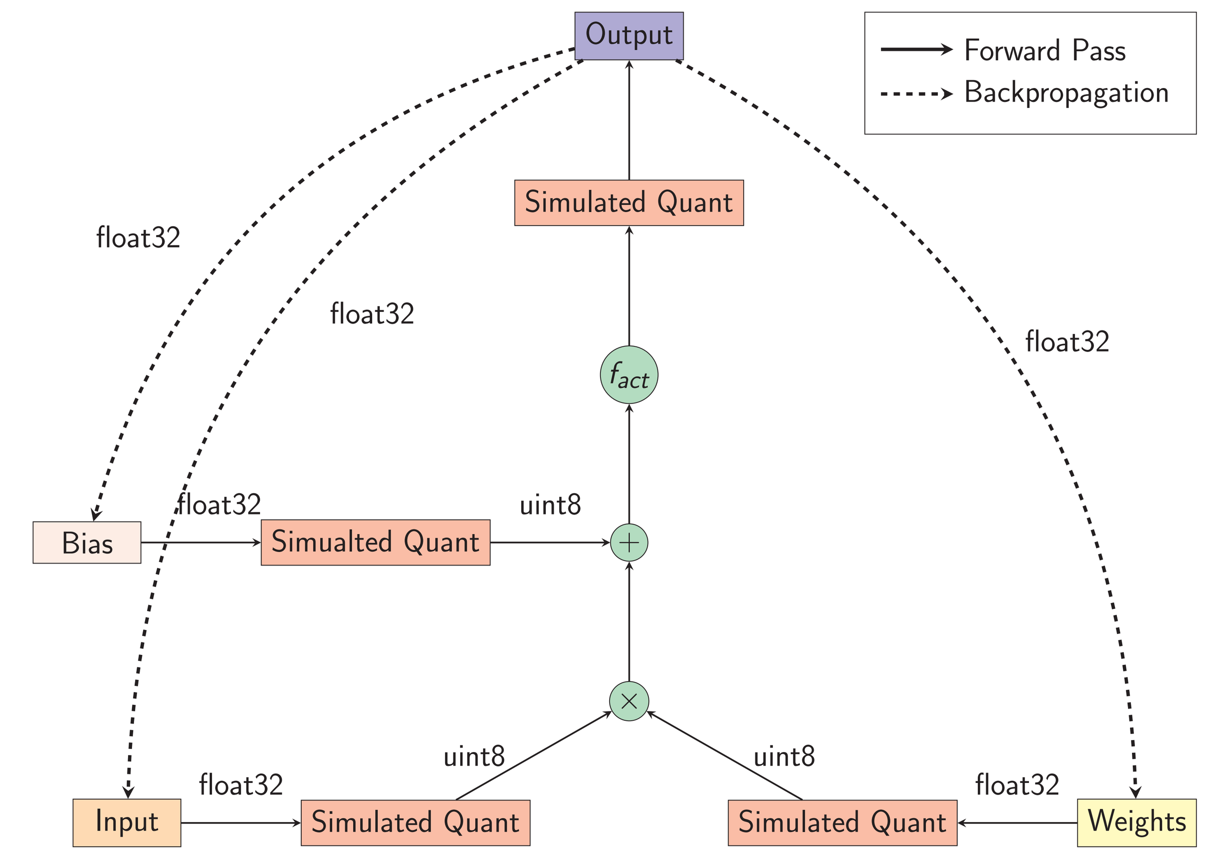


Figure: Simulated Quantization in a Typical Dense Layer

Design Space Exploration: Ordinary People Accelerating Learning (OPAL)

- Use an NN to explore candidate NN solutions [3].
- DSE NN takes hyper-parameter ranges as inputs.
- Predicts accuracy of a candidate NN, and computes cost in terms of weights and multiply-accumulates.
- Trains candidate solutions predicted to be pareto-optimal, and a small portion of those that are not.
- Actual accuracy of trained candidates are added to the DSE NN's training set.
- Returns a set of pareto-optimal candidate NNs.

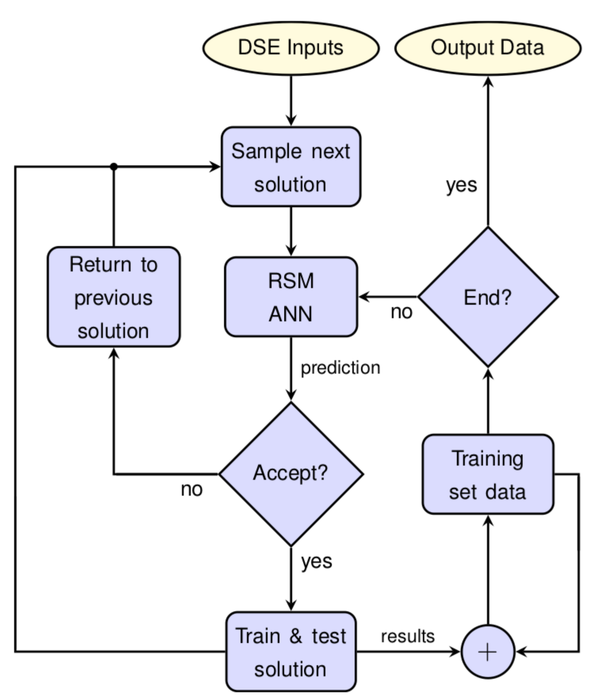
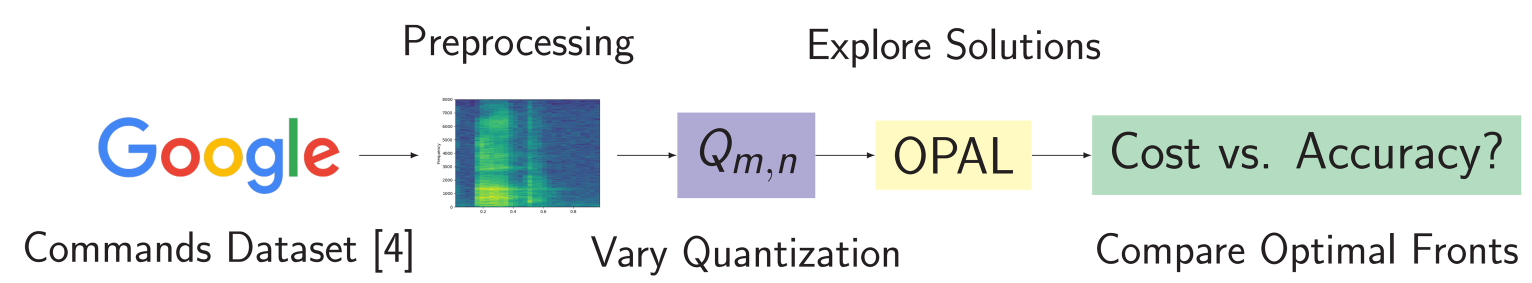
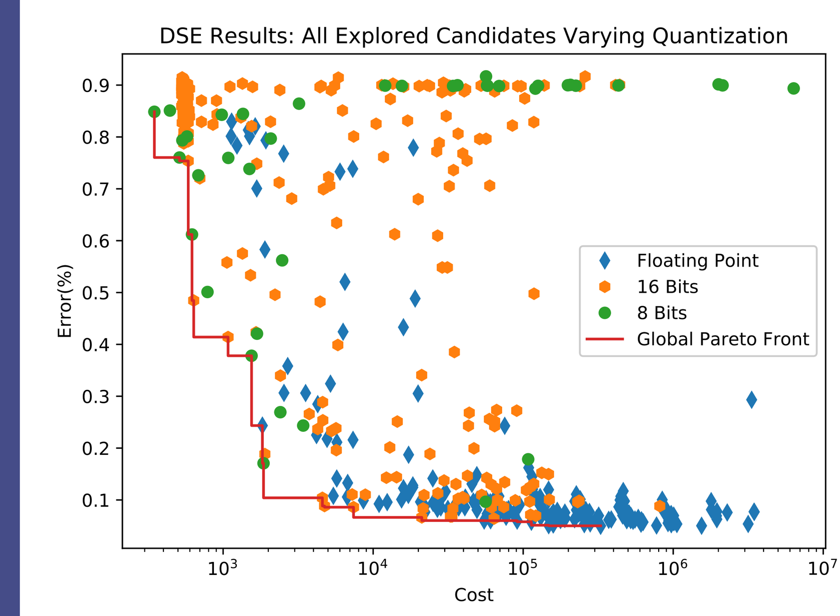


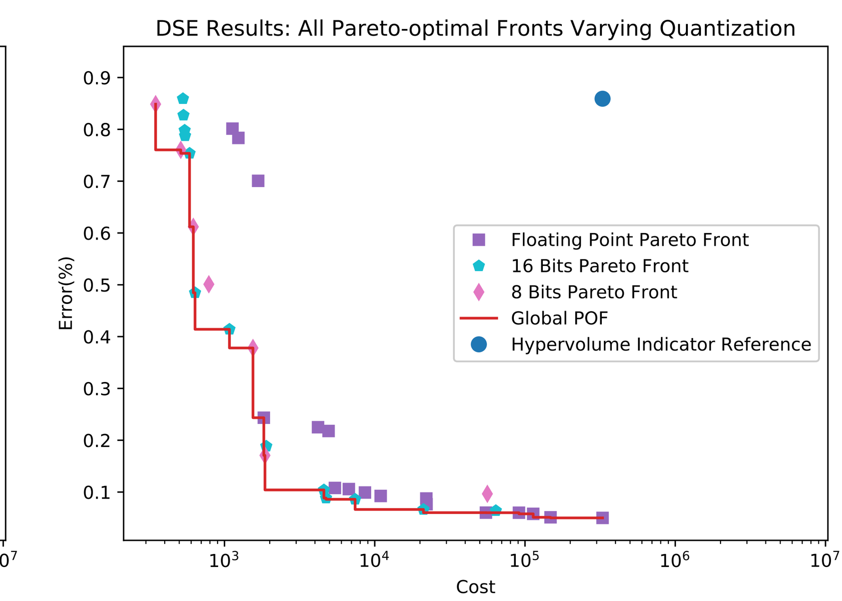
Figure: DSE Algorithm [3]

Project Workflow



Results





Analysis and Conclusions

Hypervolume Indicator

N-dimensional space contained by a pareto-optimal front and reference point.

Table: Hypervolume while varying quantization

Pareto-optimal Front	Hypervolume
Floating Point	260,815.94
16 Bits	259,245.62
8 Bits	245,394.43
Global Front	262,098.10
Reference Area	282,065.73

- Quantized models dominate at accuracies below 94%. Floating point models dominate at low-error regions.
- Fewer quantized models are trained in the same time frame.
- The hypervolume obtained using 16-bit quantization is comparable to that obtained using floating-point.
- Finding 'good' designs is objectively hard.

Future Work

- Convert high-level NN model framework for inference on MCU, using optimized CMSIS-NN libary [5].
- Investigate effects of quantization and other hyper-parameter choices, on other cost measures, such as inference delay and memory utilization.

References

- K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv e-prints*, p. arXiv:1512.03385, Dec 2015.
- P. Sparks, "White paper: The economics of a trillion connected devices."
- S. C. Smithson, G. Yang, W. J. Gross, and B. H. Meyer, "Neural Networks Designing Neural Networks: Multi-Objective Hyper-Parameter Optimization," *arXiv e-prints*, p. arXiv:1611.02120, Nov 2016.
- P. Warden, "Speech commands: A public dataset for single-word speech recognition.," 2017.
- L. Lai, N. Suda, and V. Chandra, "CMSIS-NN: Efficient Neural Network Kernels for Arm Cortex-M CPUs," *arXiv e-prints*, p. arXiv:1801.06601, Jan 2018.