Low-bit Quantization for Deep Learning Models

PD Dr. Haojin Yang

Multimedia and Machine Learning Group

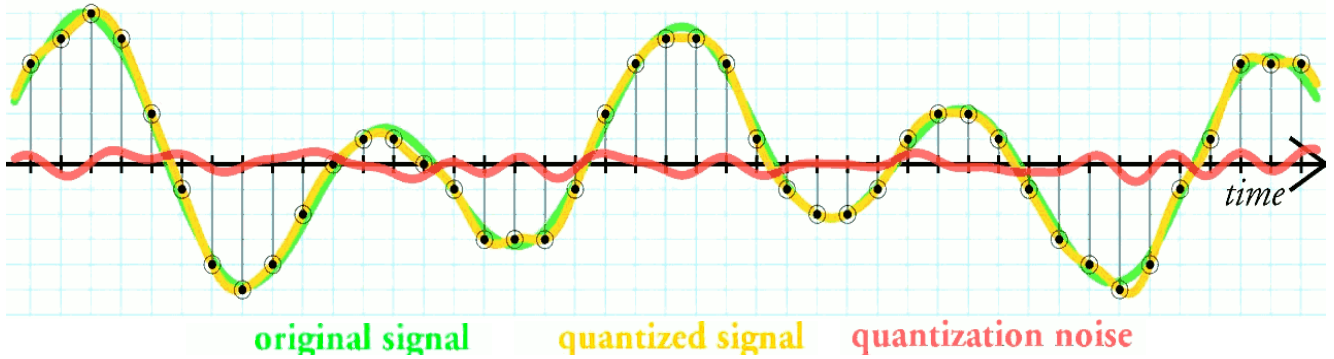
Hasso Plattner Institute

Design IT.
Create Knowledge.

www.hpi.de

Low-bit Quantization

- Neural network consists of floating-point operations and parameters
  - E.g., FP32 (32-bit) with the range $[(2-2^{-23}) \times 2^{127}, (2^{23}-2) \times 2^{127}]$, **the number of possible values is approximately 2^{32}**.
- **Quantization** in digital signal processing refers to approximating the continuous value of the signal to a finite number of discrete values.
- **Neural network quantization** refers to the use of low-bit values and operations instead of full-precision counterparts.
  - E.g., A fixed-point expression e.g., INT8 (8-bit) with the range [-128, 127], the number of possible values is approximately 2^8



Gregory Maxwell at English Wikipedia, CC BY-SA 3.0

Neural Network Quantization

- Why does quantization work for deep neural networks?
  - Deep neural networks are likely overparameterized.
  - The neural network's weights have a narrow distribution range and are close to zero.
- Advantages of neural network quantization
  - Significantly save memory and improve inference speed
  - Support more applications of edge devices
- Type of quantization methods
  - Post-training quantization (**PTQ**)
  - Quantization aware training (**QAT**)

Low-bit Model Architectures

- Binary Neural Networks (BNNs): Using 1-bit weights and activations
- Ternary Weight Networks (TWNs): Using -1, 0, +1 weight values
- Quantized Neural Networks (QNNs): 2-8 bit precision models
- Mixed-precision architectures: Different bit-widths for different layers

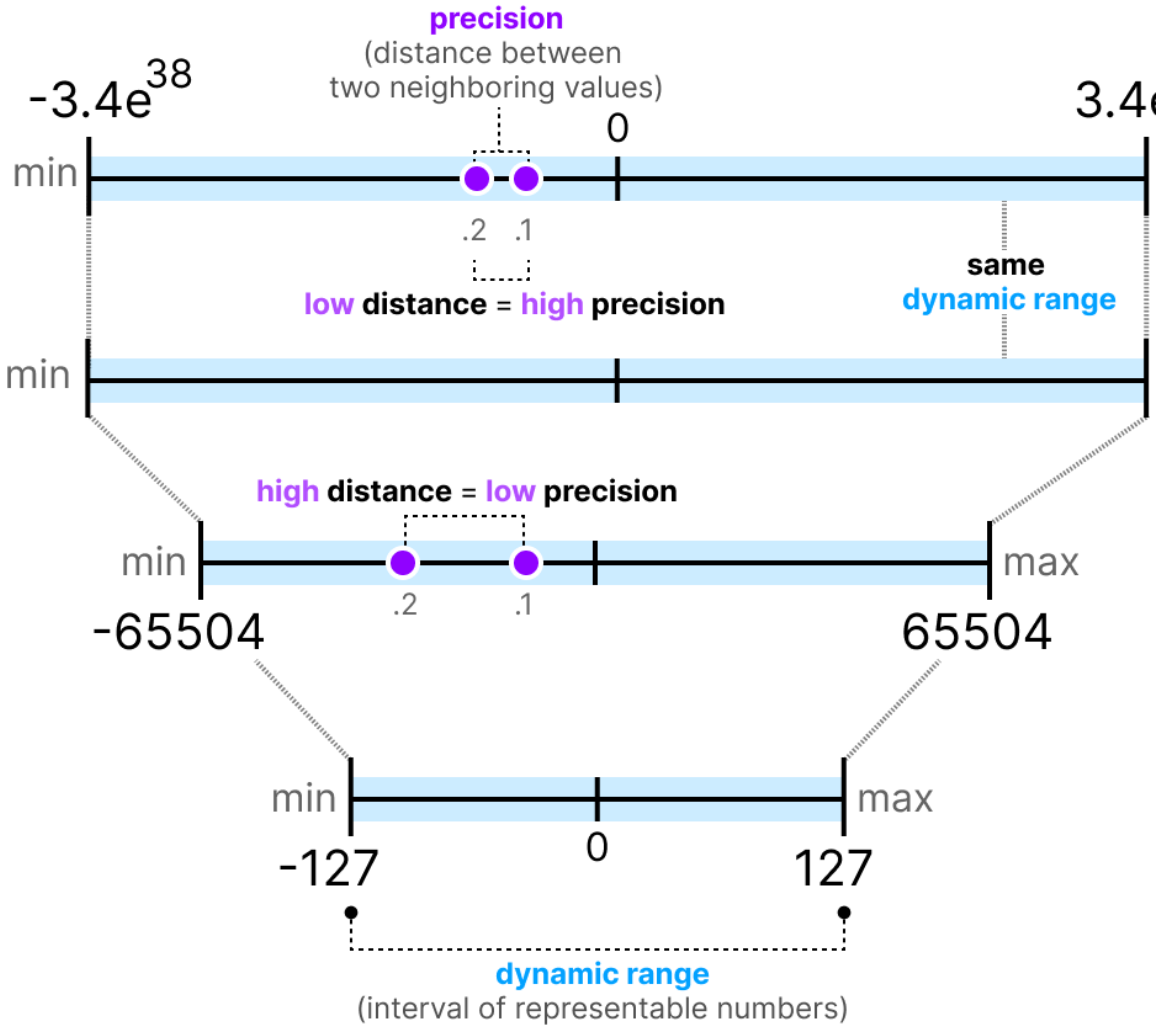


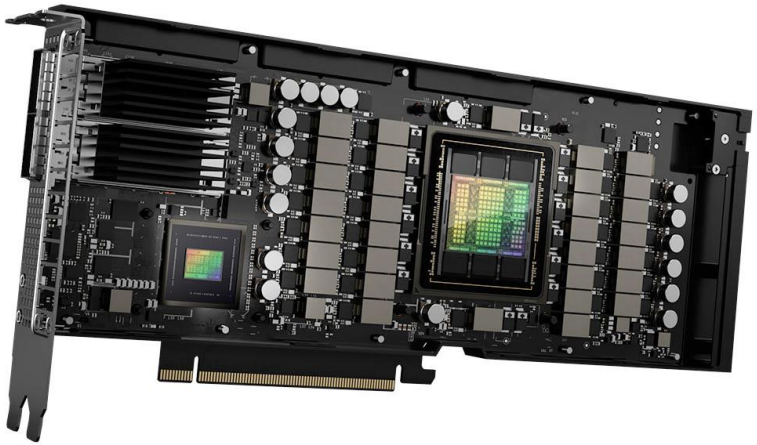
Image source: https://www.maartengrootendorst.com/blog/quantization/

Computing Engines and Optimizers

- Specialized hardware accelerators for low-bit operations (TPUs, NPUs)
- Software frameworks optimized for quantized computations
- Bit-serial computation techniques for flexible precision
- Energy efficiency gains through custom computing engines
- Training optimizers designed for low-precision gradients
- Memory bandwidth reduction through computation-in-memory approaches

Google’s Cloud TPU

Nvidia GPU



Quantization Aware Training

- Ultra-low bit quantization (< 8-bit) will cause significant precision drop.
- Train a neural network using quantized weights and activations
- Upcoming video: We will explain how do we train binary neural networks (1-bit).

$$\text{Forward: } r_o = \frac{\text{round}((2^k - 1) \cdot x)}{2^k - 1}$$

$$\text{Backward: } \frac{\partial c}{\partial r_i} = \frac{\partial c}{\partial r_o}$$