Seoul National University
Computing Memory Architecture Laboratory[1]
Design Automation Laboratory[2]

Weighted-Entropy-based Quantization for Deep Neural Network

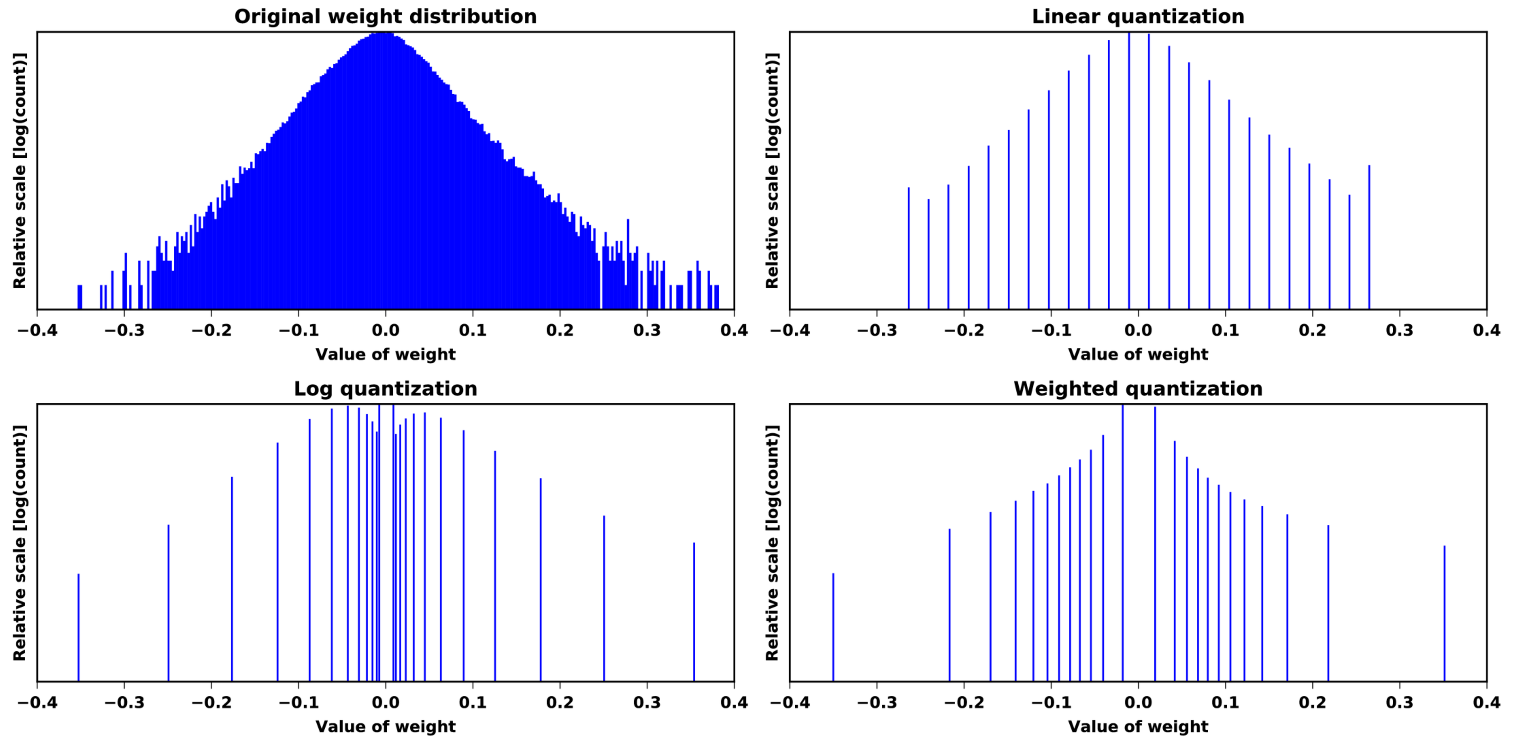
Eunhyeok Park[1], Junwhan Ahn[2], Sungjoo Yoo[1]

IEEE 2017 Conference on Computer Vision and Pattern Recognition
CVPR July 21-26 HONOLULU 2017

Abstract

- We propose a new multi-bit quantization method for both weights and activations. Our scheme is applicable for any number of bits per weight / activation.
- Our scheme facilitates automated quantization of the entire neural network. It does not require any modifications to the network, thus it can be easily integrated into conventional training algorithms for neural network.
- We demonstrate the effectiveness of our method based on various practical neural network designs including image classification, object detection, and language modeling.

Observation



- **Near-zero values** dominate the total frequency of values, but their impact on the output is small. It is desirable to assign fewer quantization levels.
- **Large values** have significant impact on the quality of output, but they are infrequent. It is also desirable to assign a small number of levels to them.
- **Intermediate values** constitute a relatively large number of population with noticeable impacts on the output quality. We must assign more levels to those values than in conventional quantization methods.

Motivation & Idea

- Quantization levels should be assigned judiciously by taking into account the values and frequencies of weight and activation.
- We figured out that that the above conditions can be accomplished by maximizing weighted entropy of quantization.

Weighted-Entropy-based Quantization

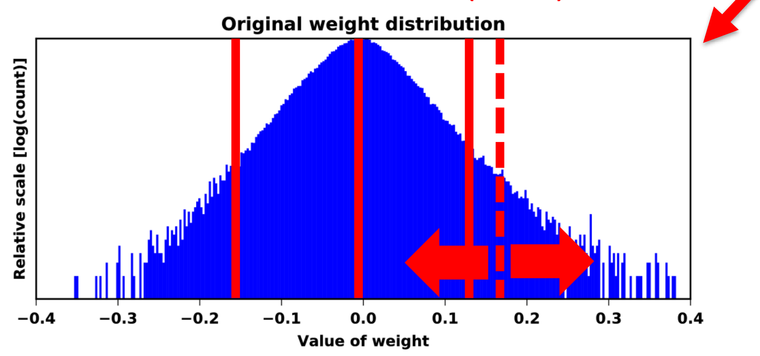
$$S = -\Sigma_n I_n P_n \log P_n,$$

where

$$P_n = \frac{|C_n|}{\Sigma_k |C_k|} \quad \text{(relative frequency)},$$

$$I_n = \frac{\Sigma_m i_{(n,m)}}{|C_n|} \quad \text{(representative importance)},$$

$$i_{(n,m)} = w^2_{(n,m)} \quad \text{(importance mapping)}.$$



Algorithm 1 Weight Quantization

```
function OptSearch(N, w)
    for k = 0 to N_w − 1 do
        i_k ← f_i(w_k)
    s ← sort([i_0, ··· , i_{N_w−1}])
    c_0, ··· , c_N ← initial cluster boundary
    while S is increased do
        for k = 1 to N − 1 do
            for c'_k ∈ [c_{k−1}, c_{k+1}] do
                S' ← S with c_0, ··· , c'_k, ··· , c_N
                if S' > S then
                    c_k ← c'_k
    for k = 0 to N − 1 do
        I_k ← Σ_{i=c_k}^{c_{k+1}−1} s[i]/(c_{k+1} − c_k)
        r_k ← f_i^{−1}(I_k)
        b_k ← f_i^{−1}(s[c_k])
    b_N ← ∞
    return [r_0 : r_{N−1}], [b_0 : b_N]
function Quantize(w_n, [r_0 : r_{N−1}], [b_0 : b_N])
    return r_k for k s.t. b_k ≤ w_n < b_{k+1}
```

- N: The number of levels
- N_w: The number of weights
- w_n: Value of n-th weight
- i_n: Importance of n-th weight
- f_i: Importance mapping function
- c_i: Cluster boundary index
- S: Overall weighted entropy

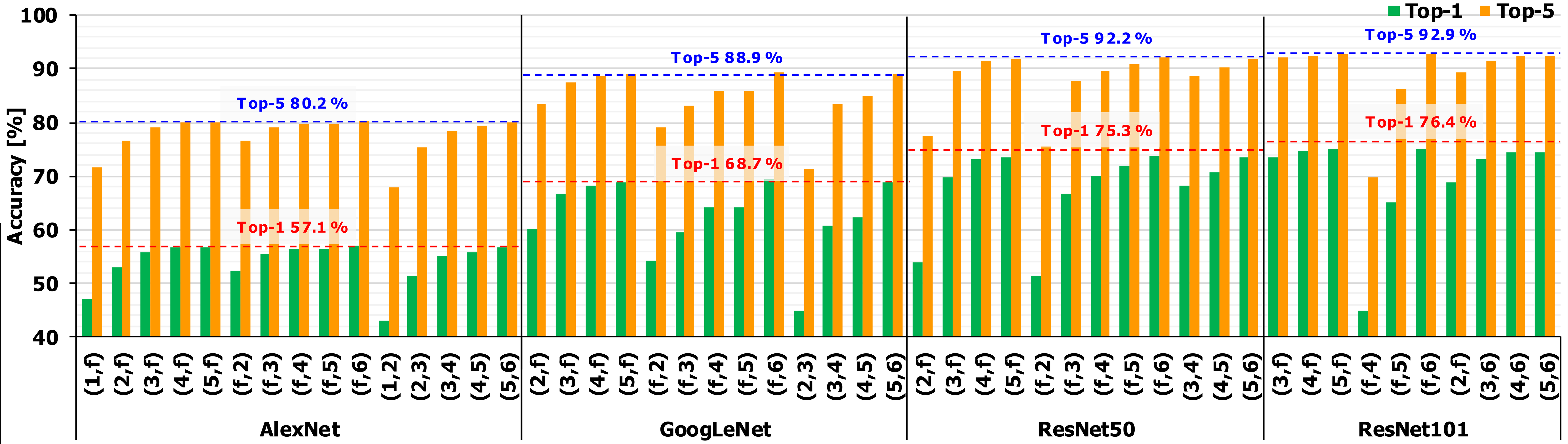
- Weights are clustered while maximizing weighted entropy
- Activations are under logarithm-based quantization. Their hyper-parameters, e.g. log base and offset, are obtained by maximizing weighted entropy.

Integration of quantization into training

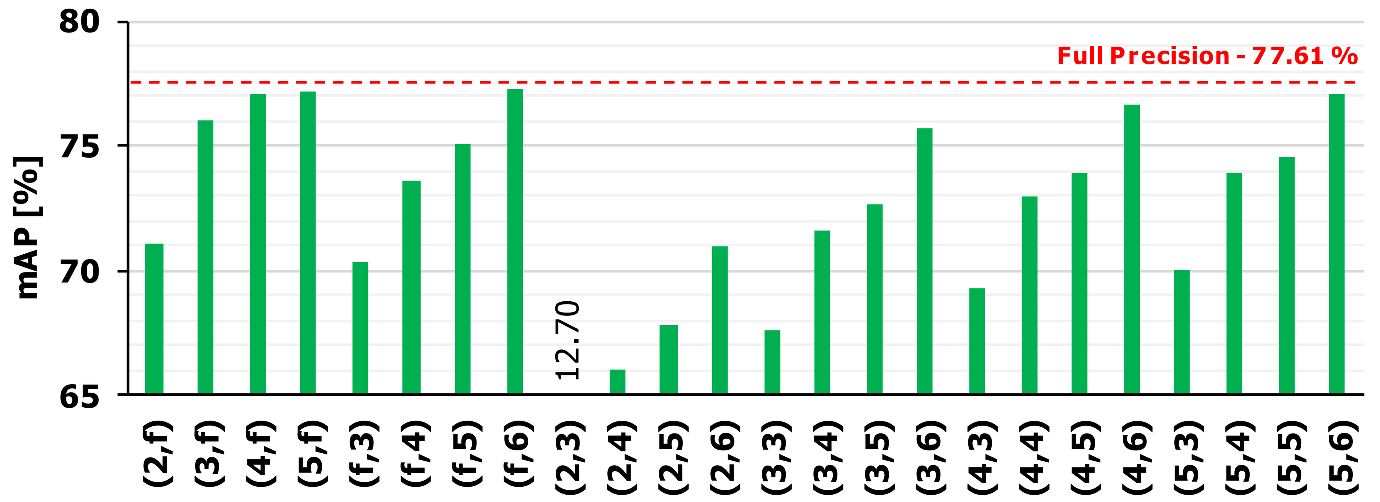
For each mini-batch,

- Forward pass to calculate P_n for activation
- Activation quantization to adjust base and offset for **LogQuant** to maximize S (weighted entropy for activation)
- Backward pass and weight (32b) update
- Weight quantization to maximize weighted entropy

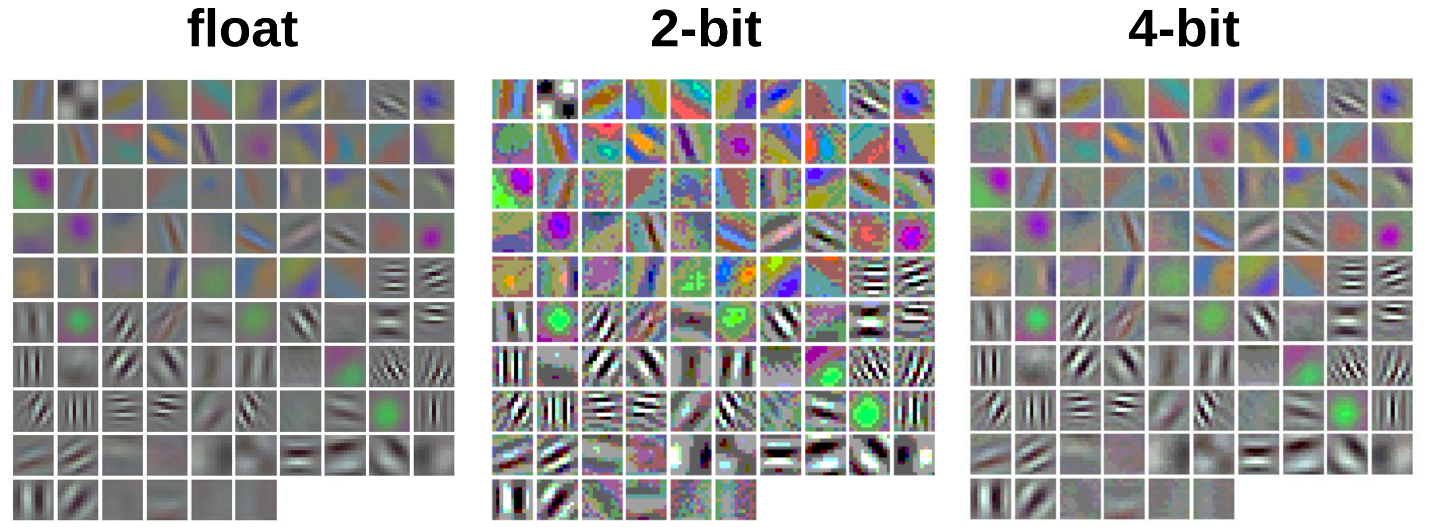
Image Classifier Quantization



Object Detector Quantization (R-FCN)



Visualization of Feature Maps



LSTM for Language Modeling

	Large		Medium		Small	
	Valid	Test	Valid	Test	Valid	Test
float	82.77	78.63	87.69	83.54	119.19	114.46
1-bit	92.20	88.48	104.0	100.7	147.19	141.07
2-bit	86.73	82.90	92.49	89.24	137.34	131.15
3-bit	85.59	81.57	86.73	83.50	121.21	117.00
4-bit	81.83	78.09	88.01	83.84	121.84	114.95

Conclusion

- We proposed a novel weight / activation quantization method based on the concept of weighted entropy.
- The key benefits of our approach are as follows.
  - Flexible multi-bit quantization, which allows us to optimize the neural network design under the tight accuracy loss constraint.
  - Automated quantization, which does not require modifications to the input networks.