同济大学
TONGJI UNIVERSITY

Quantization-based Deep Neural Network Compression

Jiaxin Hu,

Prof. Weixiong Rao

School of Software Engineering, Tongji University

16/08/2019

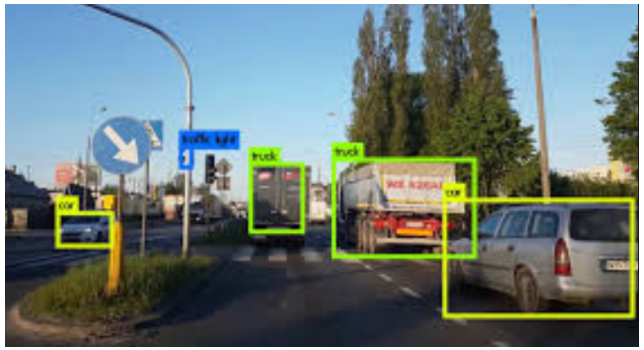
Outline

- Motivation
- Related Work
- Our Approach
- Conclusion

Neural Networks perform well in image related tasks

- Image Classification:
  - VGG, ResNet, Inception

- Image Detection:
  - Mask R-CNN, YOLO



- Image Segmentation
  - U-Net

Neural Network can be compressed

- Network is **large**
  - difficult to implement in mobile device.
    - ResNet-50: 243MB.
    - U-Net: 229MB.
  - enlarge inference time.
  - unfriendly for energy consumption.

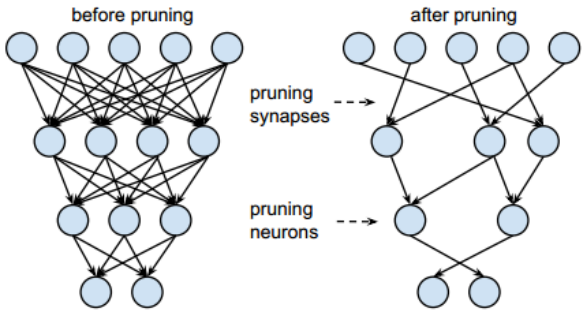
- Many parameters are redundant.
  - few of parameters are meaningful for network inference.
  - 32 bits is too large for storing weight.

Outline

- Motivation
- Related Work
- Our Approach
- Conclusion

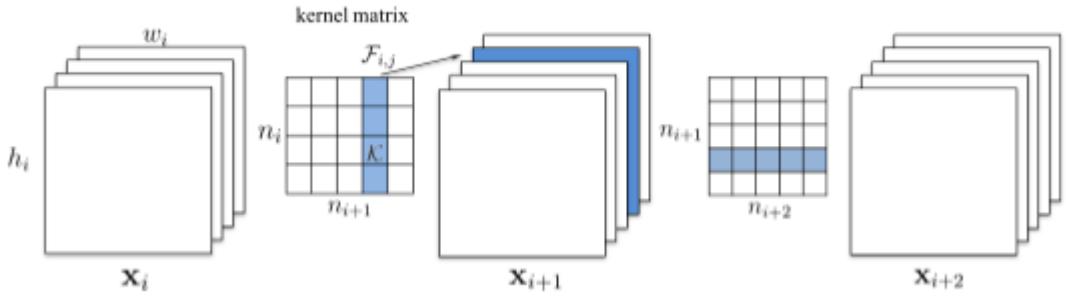
Network Pruning

- unstructured pruning
  - *Learning both Weights and Connections for Efficient Neural Networks –* ***Song Han***



Storing Method：Sparse Matrix (CSR, CSC)

- Structured pruning



Network Quantization

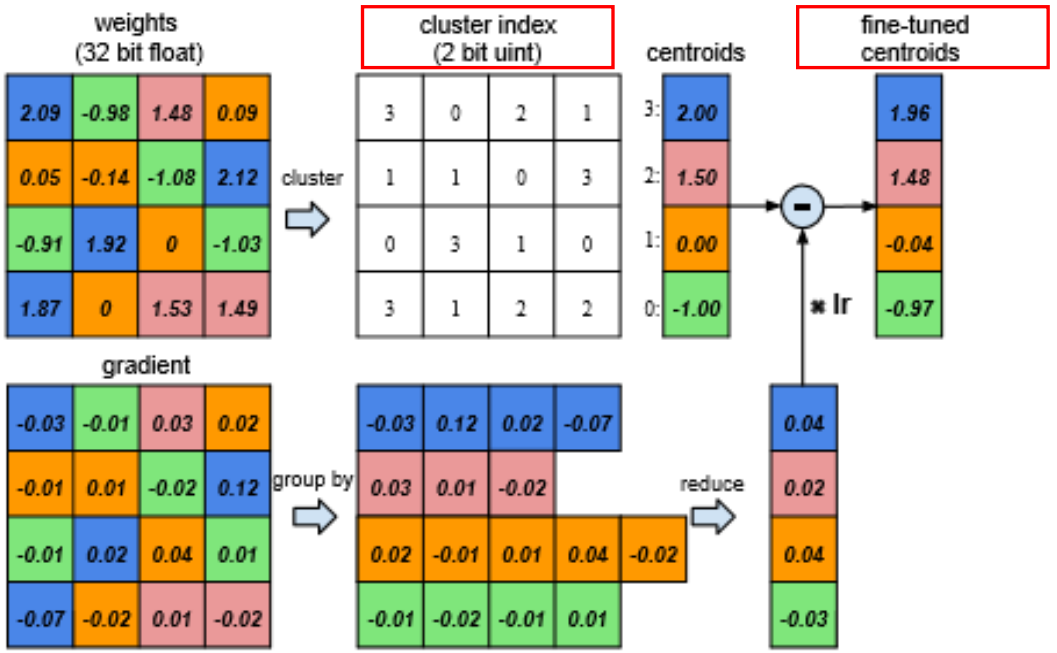
- Scalar Quantization
  - parameters sharing
  - fix floating point quantization (16bits, 8bits)
- Vector Quantization
  - vector based
  - product based
- Binary & Ternary Network

Parameters Sharing

*DEEP COMPRESSION: COMPRESSING DEEP NEURAL NETWORKS WITH PRUNING, TRAINED QUANTIZATION AND HUFFMAN CODING -- **Song Han, et al**.*

Example:
Quantize weight matrix of size of 4 x 4 into 2 bits matrix

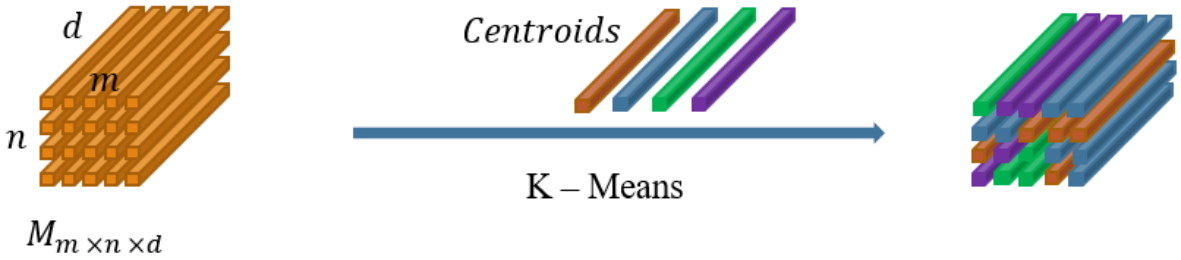
Generally 8 bits for conv layers and 5 bits for fc layers



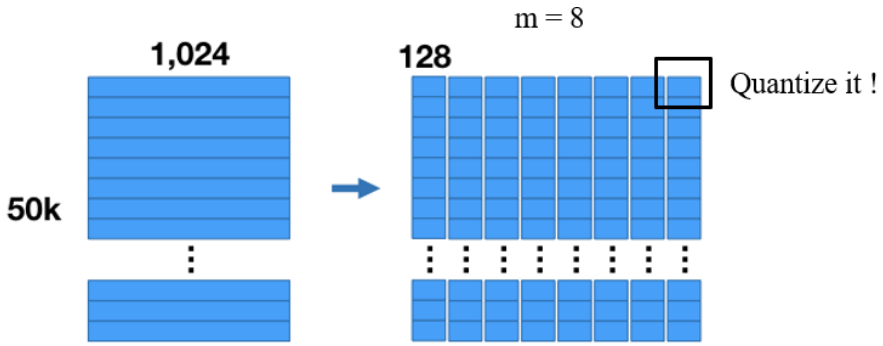
Vector / Product Quantization

And The Bits Goes Down: Revisiting the Quantization of Neural Networks -- ***Pierre Stock*** *(Facebook AI Research)*

- Vector Quantization：



- Product Quantization:



Other Techniques

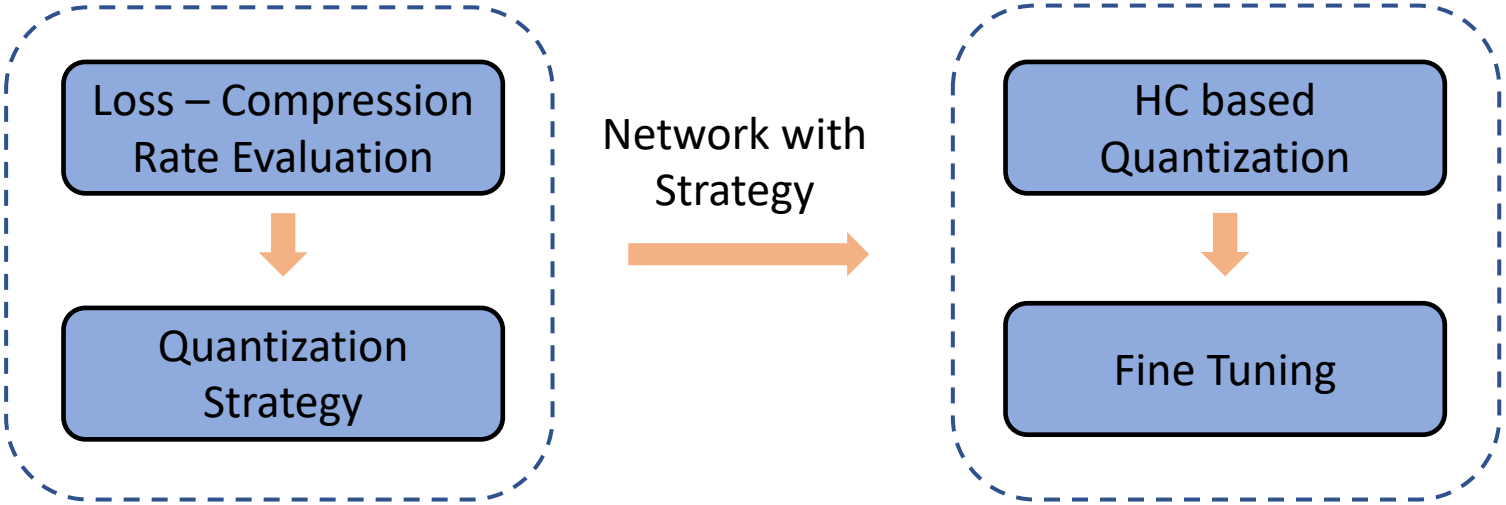
- Matrix Factorization
  - Reduce dimensionality of matrix by factorize it to multiplication of several sub-matrixs.
- Knowledge distillation
  - *And the Bit Goes Down: Revisiting the Quantization of Neural Networks* -- ***Pierre Stock.***
  - Main Idea: construct parent-children network, and fine tune children network by only parent network without help of data set。
- Framework Rebuild
  - MobileNet, ShuffleNet, SqueezeNet, etc.

Outline

- Motivation
- Related Work
- Our Approach
- Conclusion

Mix Floating Point Quantization Framework

- Motivation:
  - Quantization with high accuracy preserved requires long time fine-tuning (eg: K-Means Scalar Parameters Sharing).
  - Existing Quantization methods use uniform quantization settings to quantize all the kernels, but different kernels have their own quantization bits width limit.
- Framework:



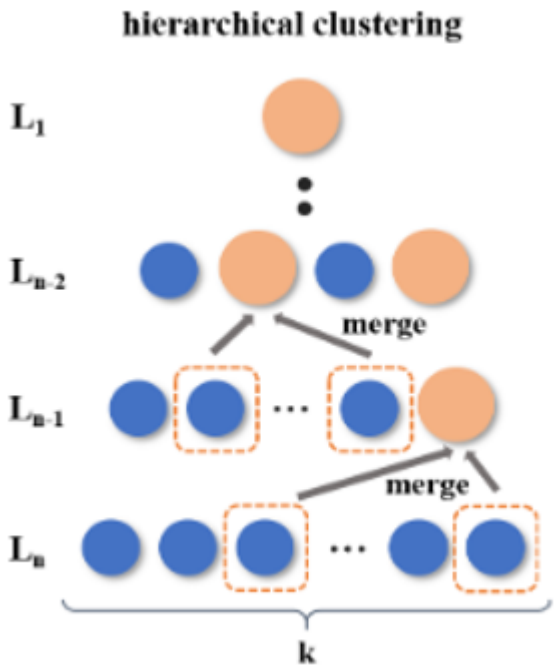
Traditional Hierarchical Clustering Algorithm

- Features: Construct clustering tree which is able to log situations of different clustering number.

- Time Complexity: $O(n^3)$

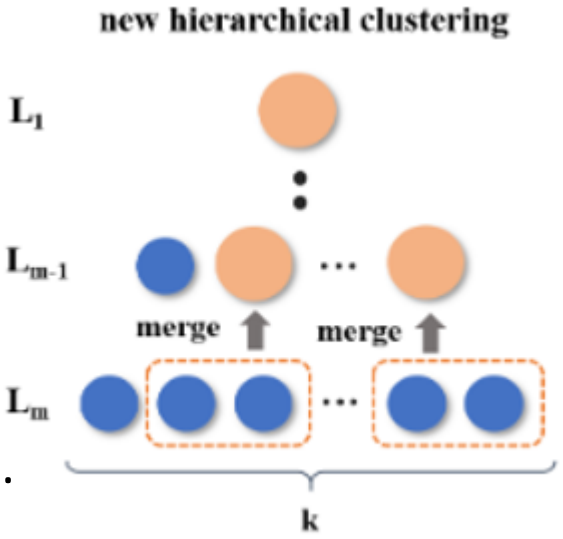
- Example: cluster k items by HC

We only Focus on situation of specific clustering number



HC based Quantization Method

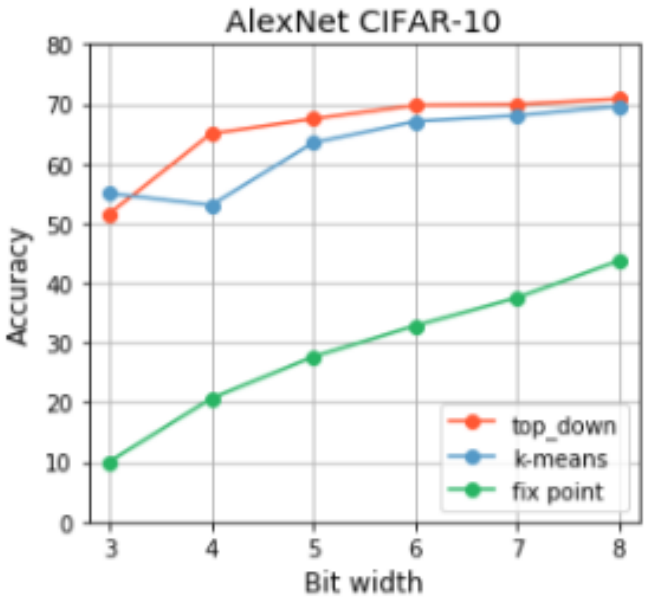
- Improvement: cluster multiple items in each layer, shorten clustering time.



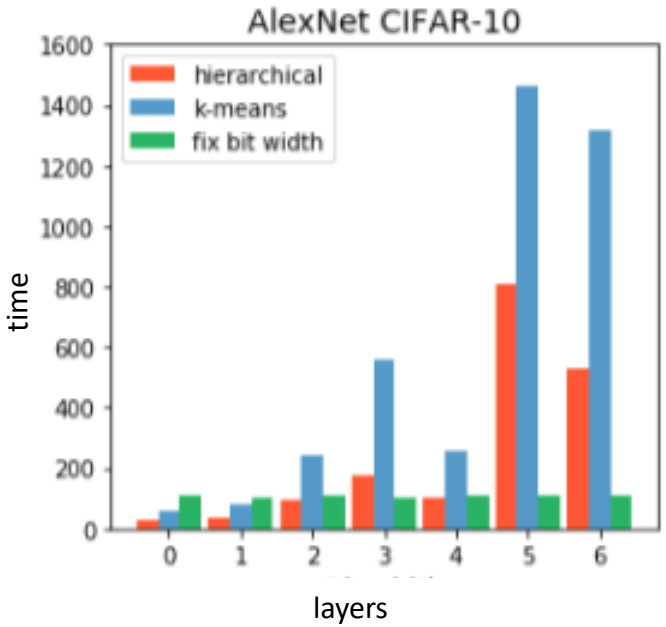
- Algorithm:
  1. Reshape matrix W into 1D array W_{sort}.
  2. Sort W_{sort} ascendingly.
  3. Traverse W_{sort}. For $w_i \in W_{sort}$ if $Dist(w_i, w_{i+1}) < Dist(w_i, w_{i-1})$, log index of w_{i+1} as w_i's nearest index. Vice versa.
  4. Traverse W_{sort}'s nearest index list, if nearest indexes of w_i, w_{i+1} are opposite, cluster them as one item.
  5. Repeat Step 3, 4 until item number is the clustering number we want.

HC VS K-Means Quantization Method

Accuracy



Training Time



Linkage Function Selection

Linkage options：Ward, Complete, Average, single

Table 1: Top-1 Error (%) of Hierarchical Clustering Quantization with different linkage function compared with pretrained model on AlexNet, CIFAR-10. W:ward, C:complete, A:average, S:single

Layer	6bit				5bit				4bit				3bit			
	W	C	A	S	W	C	A	S	W	C	A	S	W	C	A	S
Conv1	-1.35	-1.13	-1.39	-0.99	-0.24	-1.40	-0.46	0.54	-0.94	-0.15	12.10	5.92	1.41	28.25	27.87	17.84
Conv2	-0.97	-0.99	-0.37	-0.30	-0.87	0.37	1.35	0.71	-0.24	0.66	4.02	3.38	0.37	1.99	10.15	7.25
Conv3	-0.34	33.59	-0.81	-1.55	-0.61	39.52	-1.95	-1.12	-1.02	38.96	6.47	0.85	-1.12	38.6	9.81	0.87
Conv4	0.07	51.93	-0.77	-0.88	-0.78	0.43	0.18	0.68	-0.78	53.78	19.22	0.52	0.48	55.22	49.25	5.56
Conv5	0.40	50.32	1.75	3.40	0.52	54.73	11.38	6.61	1.33	57.69	10.30	13.61	2.06	57.56	48.34	54.99
fc1	0.62	48.13	3.65	2.93	0.66	46.91	6.89	3.19	1.75	45.87	14.36	6.50	3.27	51.18	29.33	16.84
fc2	0.37	50.86	4.69	7.55	0.70	50.45	5.76	13.67	1.44	53.48	17.58	21.32	2.59	57.39	23.17	38.11

Loss – Compression Rate Evaluation

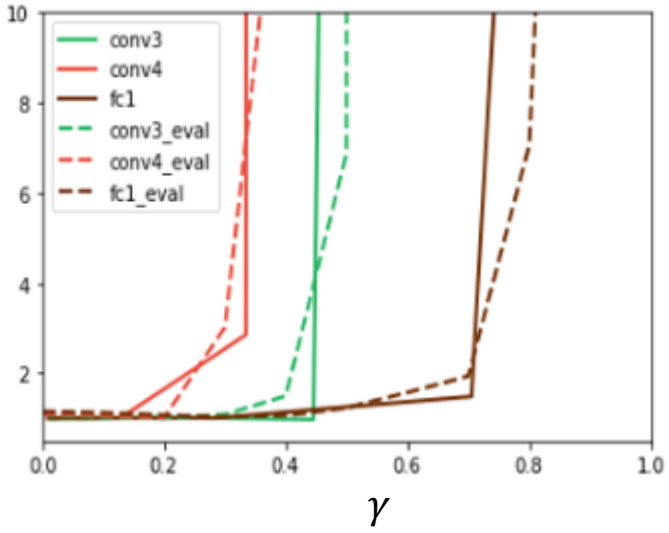
- Target: make different quantization bit width strategies for different kernels.
- Layer Graph: represent relation between change of weights and change of loss.
- Formula:

$$\delta w = w \times (1 - \gamma)$$

$$r_{cl} = \frac{L(X, \delta w)}{L(X, w)} \times log_{100}\left(\sum_{w_i \in w} ||w_i - E(w)||_2\right)$$

Sample of Data Set

1. The number of parameters
2. The distribution of weights



Outline

- Motivation
- Related Work
- Our Approach
- Conclusion

Conclusion

- HC based Quantization method has been proved to perform well in compression rate, accuracy preservation and time saving.
- In progress: experiments of Loss – Compression Rate Evaluation
- Experiments setting:
  - Models: U-Net, AlexNet, VGG16, ResNet50
  - Data Sets: ImageNet, CIFAR-10, MNIST, Cardiac CT images

Thanks