DNQ: Dynamic Network Quantization

Yuhui Xu[1], Shuai Zhang[2], Yingyong Qi[2], Jiaxian Guo[1], Weiyao Lin[1] and Hongkai Xiong[1]

[1]Shanghai Jiao Tong University, Email: (yuhuixu, gjx925825293, wylin, xionghongkai)@sjtu.edu.cn
[2]Qualcomm AI Research, Email: (shuazhan, yingyong)@qti.qualcomm.com

In this paper, we propose a Dynamic Network Quantization (DNQ) framework. Unlike most existing quantization methods that use a universal quantization bit-width for the whole network, we utilize policy gradient [1] to train an agent to learn the bit-width of each layer by the bit-width controller.

Our bit-width controller in this work is based on reinforcement learning for training an agent to maximize the cumulative reward. This problem is solved by training a policy network M_θ, the input sequence is the embedding of the network and the output sequence $B_L = (b_1, \ldots, b_l, \ldots, b_L)$ is the bit-widths of the network, where b_l is the bit-width of the l^{th} layer. In time step l, the state s is the current produced bit-width sequence $(b_1, \ldots, b_{l-1},)$. The action a_l we choose in time step l indicates the bit-width used to quantize the layer, where $a_l \in (2, 3, \ldots, 8)$. Thus, the reward R is defined as $Acc + \lambda r$, where Acc is the accuracy of the quantized network without fine-tuning and r is the compression ratio. We should not only consider the fitness of previous layers' bit-widths but also the future outcome. Therefore, to evaluate the action a_t in time step t, we apply Monte Carlo search to sample the next $L - t$ bit-widths. We average the N times sampling results to reduce the variance:

$$R^{M_\theta}(s_t = B_{t-1}, a_t = b_t) = \frac{1}{N}\sum_{n=1}^{N} R_n(B_L), \quad B_L = MC(B_t; N), \tag{1}$$

where $MC(:)$ is the Monte Carlo sampling function.We train our policy networks by policy gradient [1].

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61425011, Grant 61720106001, Grant 61529101, and in part by Shanghai High Technology Project under Grant17511106603 andthe Program of Shanghai Academic Research Leader under Grant 17XD1401900.

References

[1] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in neural information processing systems*, 2000, pp. 1057–1063.

2375-0359/19/$31.00 ©2019 IEEE
DOI 10.1109/DCC.2019.00122