One-Step Forward and Backtrack: Overcoming Zig-Zagging in Loss-Aware Quantization Training

Lianbo Ma[1], Yuee Zhou[1], Jianlun Ma[1], Guo Yu[2]*, Qing Li[3]

[1]Software College, Northeastern University, Shenyang, China
[2]Institute of Intelligent Manufacturing, NanJing Tech University, Nanjing, China
[3]Peng Cheng Laboratory, Shenzhen, China
malb@swc.neu.edu.cn, {zhouyuee, jianlunma}@stumail.neu.edu.cn, guo.yu@njtech.edu.cn, liq@pcl.ac.cn

1. Review of formulas

Firstly, let us revisit the formulas in the main manuscript, which will be used in the following proofs.

$$\begin{aligned} \min\ & \ell\left(\hat{\omega}^{t-1}\right) + \hat{g}^{T^{t-1}}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right) \\ & + \frac{1}{2}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right)^{T}\hat{H}^{t-1}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right), \\ s.t.\ & \hat{\omega}^{t} = \alpha^{t}\beta^{t}, \alpha^{t} = \frac{\|\omega^{t}\|}{n}, \beta^{t} = sign\left(\omega^{t}\right), \end{aligned} \tag{1}$$

$$\arg\min_{\hat{\omega}^{t}} \frac{1}{2}\|\omega^{t} - \hat{\omega}^{t}\|^{2}_{\hat{D}^{t-1}}, \tag{2}$$

where

$$\omega^{t} = \hat{\omega}^{t-1} - \hat{g}^{t-1} \oslash \hat{D}^{t-1}. \tag{3}$$

$$\arg\min_{\hat{\omega}^{*(t+1)}} \frac{1}{2}\|\omega^{*(t+1)} - \hat{\omega}^{*(t+1)}\|^{2}_{\hat{D}^{t}}, \tag{4}$$

where

$$\omega^{*(t+1)} = \omega^{t} - \hat{g}^{t} \oslash \hat{D}^{t}. \tag{5}$$

$$\arg\min_{\hat{\omega}^{t+1}} \frac{1}{2}\|\omega^{t+1} - \hat{\omega}^{t+1}\|^{2}_{\hat{D}^{t+1}}, \tag{6}$$

where

$$\omega^{t+1} = \omega^{t} - \hat{g}^{t+1} \oslash \hat{D}^{t+1}. \tag{7}$$

$$\hat{g}^{t+1} = a\hat{g}^{t} + (1-a)\hat{g}^{*(t+1)}, \tag{8}$$

$$\hat{D}^{t+1} = a\hat{D}^{t} + (1-a)\hat{D}^{*(t+1)}, \tag{9}$$

2. Eq. 1 can be written as Eq. 2 and Eq. 3

Due to Eq. 3, Eq. 2 can be expressed as

$$\frac{1}{2}\|\hat{\omega}^{t-1} - \hat{\omega}^{t} - \hat{g}^{t-1} \oslash \hat{D}^{t-1}\|^{2}_{\hat{D}^{t-1}}. \tag{10}$$

Since $\|x\|^{2}_{Q} = x^{T}Qx$, Eq. 2 can be further expressed as

$$\begin{aligned} & \frac{1}{2}(\hat{\omega}^{t} - \hat{\omega}^{t-1} + \frac{\hat{g}^{t-1}}{\hat{D}^{t-1}})^{T}\hat{D}^{t-1}(\hat{\omega}^{t} - \hat{\omega}^{t-1} + \frac{\hat{g}^{t-1}}{\hat{D}^{t-1}}) \\ = & \ \frac{1}{2}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right)^{T}\hat{H}^{t-1}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right) \\ & + (\hat{g}^{t-1} \oslash \hat{D}^{t-1})^{T}\hat{D}^{t-1}(\hat{\omega}^{t} - \hat{\omega}^{t-1}) \\ & + \frac{1}{2}(\hat{g}^{t-1} \oslash \hat{D}^{t-1})^{2}\hat{D}^{t-1} \\ = & \ \frac{1}{2}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right)^{T}\hat{H}^{t-1}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right) \\ & + \hat{g}^{T^{t-1}}(\hat{\omega}^{t} - \hat{\omega}^{t-1}) + c_{1}, \end{aligned} \tag{11}$$

where c_1 is independent of $\hat{\omega}^t$ and can be considered as a constant.

Moreover, Eq. 1 can be expressed as

$$\begin{aligned} & \frac{1}{2}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right)^{T}\hat{H}^{t-1}\left(\hat{\omega}^{t} - \hat{\omega}^{t-1}\right) \\ & + \hat{g}^{T^{t-1}}(\hat{\omega}^{t} - \hat{\omega}^{t-1}) + c_{2}, \end{aligned} \tag{12}$$

where where c_2 is independent of $\hat{\omega}^t$ and can be considered as a constant. Therefore, at iteration t, Eq. 1 can be written as Eq. 2 and Eq. 3.

3. Example to illustrate the zig-zagging-like issue

Given a simple loss function $f(\omega) = c\omega^{\frac{3}{2}}$, if Eq. 2 and Eq. 3 are used to calculate quantized weights, for any initial weights ω_0, the optimization of $f(\omega)$ cannot converge and end in oscillating.

Proof: Firstly, we compute the first derivative $g(\omega)$ and second derivative $H(\omega)$ of $f(\omega)$ as

$$g(\omega) = \frac{3}{2}c\omega^{\frac{1}{2}}, \tag{13}$$

$$H(\omega) = \tfrac{3}{4}c\omega^{-\frac{1}{2}}. \tag{14}$$

Then, calculate the quantized value $\hat{\omega}^0$ of initial weights ω^0 using Eq. 2. Following Eq. 3, we can obtain

$$\begin{aligned} \omega^{1} & = \hat{\omega}^{0} - g\left(\hat{\omega}^{0}\right) \oslash H\left(\hat{\omega}^{0}\right) \\ & = \hat{\omega}_{0} - \frac{3}{2}c\left(\hat{\omega}^{0}\right)^{\frac{1}{2}} \oslash \frac{3}{4}c\left(\hat{\omega}^{0}\right)^{-\frac{1}{2}} \\ & = -\hat{\omega}^{0}. \end{aligned} \tag{15}$$

[*] Corresponding author.

The result of Eq. 15 indicates that the optimization is unable to converge and will oscillate between the two points ($\hat{\omega}^0$ and $-\hat{\omega}^0$). The lemma is proven.

4. Example to illustrate the effectiveness of BLAQ

In this section, we provide the proofs of the theorems.

The proposed method does not oscillate when using Eq. 6 and Eq. 7to solve the optimization case $f(\omega) = c\omega^{\frac{3}{2}}$.

Proof: Similar to the proof process in section 2, we calculate the quantized value $\hat{\omega}^0$ of initial weight ω^0. Next, we can have ω^1 and then $\omega^{*1} = -\hat{\omega}^0$ by using Eq. 7 and Eq. 15. In the worst case, $\hat{\omega}^{*1} = -\hat{\omega}^0$. Hence, we can compute the first and second derivative of $f(\omega)$ as

$$\hat{g}^0 = \hat{g}^{*1} = g\left(\hat{\omega}^{*1}\right) = \frac{3}{2}c\left(\hat{\omega}^0\right)^{\frac{1}{2}}, \tag{16}$$

$$\hat{H}^0 = \hat{H}^{*1} = H\left(\hat{\omega}^{*1}\right) = \tfrac{3}{4}c\left(\hat{\omega}^0\right)^{-\frac{1}{2}}. \tag{17}$$

After that, we get Eq. 18 by adopting Eqs. 16-17 to calculate Eqs. 8-9, respectively.

$$\hat{g}^1 = a\hat{g}^0 + (1-a)\hat{g}^{*1} = \frac{3}{2}(2a-1)c\hat{\omega}^{\frac{1}{2}}, \tag{18}$$

$$\hat{H}^1 = a\hat{H}^0 + (1-a)\hat{H}^{*1} = \frac{3}{4}(2a-1)c\hat{\omega}^{-\frac{1}{2}}. \tag{19}$$

Then, we use Eq. 7 to calculate ω^1 as

$$\begin{aligned} \omega^1 &= \omega^0 - \hat{g}^1 \oslash \hat{H}^1 \\ &= \omega^0 - \frac{3}{2}c(\hat{\omega}^0)^{\frac{1}{2}} \oslash \frac{3}{4}c(\hat{\omega}^0)^{-\frac{1}{2}} \\ &= \omega^0 - 2\hat{\omega}^0. \end{aligned} \tag{20}$$

According to Eq. 20, it is clear that our method will not oscillate as long as ω^0 and $\hat{\omega}^0$ are not equal. Hence, the lemma is proven.

5. The proof of Theorem 1 and Theorem 2

Theorem 1. For the loss function $\ell(\omega)$ with the learning rate η^t, the convergence of our method is as

$$\ell\left(\omega^{t+1}\right) - \ell\left(\omega^*\right) \le \frac{L_1 + L_1^3\left(\eta^{t+1}\right)^2 - 2\mu^2\eta^{t+1}}{2}\Delta^2. \tag{21}$$

Proof: Since $\ell(\omega)$ is convex and $L_1 - smooth$ with respect to ω, according to the corresponding function properties, we have

$$\begin{aligned} \ell\left(\omega^{t+1}\right) \le \ell\left(\omega^t\right) &+ < \nabla\ell\left(\omega^t\right), \omega^{t+1} - \omega^t > \\ &+ \frac{L_1}{2}\|\omega^{t+1} - \omega^t\|^2, \end{aligned} \tag{22}$$

$$\begin{aligned} \ell\left(\omega^{t+1}\right) - \ell\left(\omega^*\right) &\le \ell\left(\omega^t\right) - \ell\left(\omega^*\right) \\ &+ < \nabla\ell\left(\omega^t\right), \omega^{t+1} - \omega^t > \\ &+ \frac{L_1}{2}\left\|\omega^{t+1} - \omega^t\right\|^2. \end{aligned} \tag{23}$$

Algorithm 1: BLAQ for Training Quantized Network

```
Input: Minibatch $({x_0}^t, y^t)$, current full-precision weights $\omega^t$, diagonal Hessian matrix $D^t$, first moment $m^t$, second moment $v^t$, learning rate $\eta^t$
Output: ${\omega_l}^{t+1}$
1: /*Forward Propagation*/
2: Save the current model $\omega^t$;
3: for $l = 1$ to $L$ do
4:     Use first moment ${m_l}^{t-1}$ and diagonal Hessian matrix ${D_l}^{t-1}$ to compute ${\alpha_l}^t$ and ${\beta_l}^t$ in Algorithm 2;
5:     Utilize batch-normalization and activation function to obtain outputs ${z_l}^t$;
6: end for
7: Obtain the loss value $\ell$;
8: /*Backward Propagation*/
9: Initialize gradient ${g_L}^t$ of output layer's activation;
10: for $l = L - 1$ to 1 do
11:     Obtain gradients $\hat{g}_{l-1}^t$ from $\hat{g}_l^t$, ${\alpha_l}^t$ and ${\beta_l}^t$;
12: end for
13: /*Update Parameters using Adam*/
14: Update first moment $\hat{m}_l^t$ and second moment $\hat{v}_l^t$;
15: Compute Hessian matrix $\hat{D}_l^t = \left(\varepsilon + \sqrt{\hat{v}_l^t}\right)/\eta^t$;
16: Update parameter $\hat{\omega}^{*(t+1)}$ using Eq. 4;
17: Obtain $\hat{g}_L^{*(t+1)}$ by repeating step 1-step 12 based on $\hat{\omega}^{*(t+1)}$;
18: for $l = L - 1$ to 1 do
19:     Obtain gradients $\hat{g}_{l-1}^{*(t+1)}$ from $\hat{g}_l^{*(t+1)}$, ${\alpha_l}^t$ and ${\beta_l}^{*(t+1)}$;
20: end for
21: Compute gradients $\hat{g}^{t+1}$ using Eq. 8;
22: Update $\hat{m}_l^{*(t+1)}$, $\hat{v}_l^{*(t+1)}$ and $\hat{D}_l^{*(t+1)} = (\varepsilon + \sqrt{\hat{v}_l^{*(t+1)}})/\eta^t$;
23: Compute Hessian matrix $\hat{D}_l^{t+1}$ using Eq. 9;
24: Update full-precision weights ${\omega_l}^{t+1}$ and quantization weights using Eq. 7 and Eq. 6;
25: return ${\omega_l}^{t+1}$;
```

Algorithm 2: Parameter Learning Algorithm

```
Input: Full-precision weights $\omega^{t+1}$, diagonal approximate Hessian $\hat{D}^{t+1}$, gradient $\hat{g}^{t+1}$, number of bits $k$
Output: $\hat{\omega}^{t+1} = \alpha\beta$
$\alpha = 1.0$, $\alpha_{old} = 0$, $\beta^t + 1 = \beta^t$, $\varepsilon = 10^{-6}$
while $|\alpha - \alpha_{old}| > \varepsilon$ do
    $\alpha_{old} = \alpha$
    $\alpha = \frac{\|\beta \odot D^{t+1} \odot \omega^{t+1}\|_1}{\|\beta \odot D^{t+1} \odot \beta\|_1}$
    for $\beta_i \in Q$ do
        $\beta_i = \arg\min_{\beta_i} d_i(\beta_i - \frac{{\omega_i}^{t+1}}{\alpha})$
    end for
    $\beta = [\beta]_Q$
end while
return $\alpha, \beta$
```

We can obtain Eq. 24 by calculating Eq. 23 with Eq. 7:

$$\begin{aligned}\ell\left(\omega^{t+1}\right)-\ell\left(\omega^{*}\right) \leq\, & \ell\left(\omega^{t}\right)-\ell\left(\omega^{*}\right) \\ & -\eta^{t+1}\left(\hat{g}^{t}, \hat{g}^{t+1}\right) \\ & +\left(\eta^{t+1}\right)^{2} \frac{L_{1}}{2}<\hat{g}^{t+1}, \hat{g}^{t+1}>.\end{aligned} \tag{24}$$

According to the property of the $\mu-strongly$ convex function, we can achieve

$$\begin{aligned}\ell\left(\omega^{t}\right) \geq\, & \ell\left(\omega^{*}\right)+<\nabla \ell\left(\omega^{*}\right), \omega^{t}-\omega^{*}> \\ & +\frac{\mu}{2}\|\omega^{t}-\omega^{*}\|^{2}.\end{aligned} \tag{25}$$

Due to the fact that ω^{*} is the optimal point, i.e., $\nabla \ell\left(\omega^{*}\right)=0$, Eq. 25 can be rewritten as

$$\ell\left(\omega^{t}\right) \geq \ell\left(\omega^{*}\right)+\frac{\mu}{2}\left\|\omega^{t}-\omega^{*}\right\|^{2}. \tag{26}$$

Then, we take the first derivative of both sides of Eq. 26 with respect to ω^{t}, so that Eq. 27 can be deduced.

$$\hat{g}^{t} \geq \mu\left\|\omega^{t}-\omega^{*}\right\|. \tag{27}$$

Since $\ell\left(\omega^{t}\right)$ is $L_{1}-smooth$, we can obtain

$$\begin{aligned}\ell\left(\omega^{t}\right) \leq\, & \ell\left(\omega^{*}\right)+<\nabla \ell\left(\omega^{*}\right), \omega^{t}-\omega^{*}> \\ & +\frac{L_{1}}{2}\|\omega^{t}-\omega^{*}\|^{2}.\end{aligned} \tag{28}$$

Since ω^{*} is the optimal point, i.e., $\nabla \ell\left(\omega^{*}\right)=0$, Eq. 28 can be written as

$$\ell(\omega^{t})-\ell(\omega^{*}) \leq \frac{L_{1}}{2}\|\omega^{t}-\omega^{*}\|^{2}. \tag{29}$$

Taking the first derivative of both sides of Eq.29 with respect to ω^{t}, we obtain

$$\hat{g}^{t} \leq L_{1}\|\omega^{t}-\omega^{*}\|. \tag{30}$$

By calculating Eq. 24 with Eqs. 27 and 29-30, we can infer that

$$\begin{aligned}\ell(\omega^{t+1})-\ell(\omega^{*}) \leq\, & \frac{L_{1}}{2}\|\omega^{t}-\omega^{*}\|^{2} \\ & -\eta^{t+1}\mu^{2}\|\omega^{t}-\omega^{*}\|\|\omega^{t+1} \\ & -\omega^{*}\|+\frac{{L_{1}}^{3}}{2}(\eta^{t+1})^{2}\|\omega^{t+1}-\omega^{*}\|^{2}.\end{aligned} \tag{31}$$

For simplicity, we use "Δ" to denote $max\{\|\omega^{t+1}-\omega^{*}\|, \|\omega^{t}-\omega^{*}\|\}$. From Eq. 31, we can achieve the final convergence of our method via induction:

$$\ell(\omega^{t+1})-\ell(\omega^{*}) \leq \frac{L_{1}{L_{1}}^{3}(\eta^{t+1})^{2}-2\mu^{2}\eta^{t+1}}{2}\Delta^{2}. \tag{32}$$

Therefore, the theorem is proven.

Theorem 2. When $\frac{2}{L_{1}\eta}-1<a<1$ holds, the convergence of our method is better than that of LAQ (Hou et al. 2017).

Proof. Since $\ell(\omega)$ is convex and $L_{1}-smooth$ with respect to ω, according to the corresponding function properties, we have

$$\begin{aligned}\ell\left(\omega^{t}\right) \leq\, & \ell\left(\omega^{*}\right)+<\nabla \ell\left(\omega^{*}\right), \omega^{t}-\omega^{*}> \\ & +\frac{L_{1}}{2}\left\|\omega^{t}-\omega^{*}\right\|^{2},\end{aligned} \tag{33}$$

$$\begin{aligned}\ell\left(\omega^{t+1}\right) \leq\, & \ell\left(\omega^{t}\right)+<\nabla \ell\left(\omega^{t}\right), \omega^{t+1}-\omega^{t}> \\ & +\frac{L_{1}}{2}\left\|\omega^{t+1}-\omega^{t}\right\|^{2}.\end{aligned} \tag{34}$$

Since ω^{*} is the optimal point, namely, $\nabla \ell\left(\omega^{*}\right)=0$, Eq. 33 can be written as

$$\ell\left(\omega^{t}\right)-\ell\left(\omega^{*}\right) \leq \frac{L_{1}}{2}\left\|\omega^{t}-\omega^{*}\right\|^{2}. \tag{35}$$

By calculating Eq. 7 through Eqs. 8-9, we can have

$$\omega^{t+1}-\omega^{t}=-\eta^{t+1}\left[a\hat{g}^{t}+(1-a)\hat{g}^{*(t+1)}\right]. \tag{36}$$

Then, through taking inner product operation on Eq. 36, we can further obtain

$$\begin{aligned}&\frac{L_{1}}{2}\left\|\omega^{t+1}-\omega^{t}\right\|^{2}=\frac{L_{1}}{2}\left(\eta^{t+1}\right)^{2}\cdot \\ &<a\hat{g}^{t}+(1-a)\hat{g}^{*(t+1)}, a\hat{g}^{t}+(1-a)\hat{g}^{*(t+1)}>,\end{aligned} \tag{37}$$

$$\begin{aligned}&<\nabla \ell\left(\omega^{t}\right), \omega^{t+1}-\omega^{t}>= \\ &-\eta^{t+1}<\hat{g}^{t}, a\hat{g}^{t}+(1-a)\hat{g}^{*(t+1)}>.\end{aligned} \tag{38}$$

By calculating Eq. 34 with Eq. 35 and Eqs. 37-38, we have

$$\begin{aligned}&\ell(\omega^{t+1})-\ell(\omega^{*}) \leq \frac{L_{1}}{2}\|\omega^{t}-\omega^{*}\|^{2} \\ &+[\frac{L_{1}a^{2}}{2}(\eta^{t+1})^{2}-a\eta^{t+1}]<\hat{g}^{t}, \hat{g}^{t}> \\ &+\left[\left(\eta^{t+1}\right)^{2}L_{1}a(a-1)-(1-a)\eta^{t+1}\right]<g^{t}, \hat{g}^{*(t+1)}> \\ &+\frac{\left(\eta^{t+1}\right)^{2}L_{1}}{2}(1-a)^{2}<\hat{g}^{*(t+1)}, \hat{g}^{*(t+1)}>.\end{aligned} \tag{39}$$

Similar to the above deduction, we can infer the convergence of LAQ (Hou et al. 2017) as

$$\begin{aligned}\ell\left(\omega^{t+1}\right)-\ell\left(\omega^{*}\right) \leq\, & \frac{L_{1}}{2}\left\|\omega^{t}-\omega^{*}\right\|^{2} \\ & +\left[\frac{L_{1}\left(\eta^{t+1}\right)^{2}}{2}-\eta^{t+1}\right]\cdot<\hat{g}^{t}, \hat{g}^{t}>.\end{aligned} \tag{40}$$

Hence, we can achieve the necessary condition (Eq. 41) of the fact that the loss of BLAQ in Eq. 39 is closer to the loss of the global optimum than the loss of LAQ in Eq. 40:

$$\frac{2}{L_{1}\eta}-1<a<1. \tag{41}$$

Therefore, the theorem is proven.

6. Ablation Study

In this section, we conduct a set of ablation experiments to determine the optimal setting of the main hyperparameters of BLAQ, including m (i.e., the period to activate two-stage weight updating operation) and a (the coefficient to update gradient in Eq.8 and Eq.9.

To be specific, the values of m are selected from 5 to 10, and the values of a are selected as 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 0.95. The experimental results are shown in Table 1, Table 2, Table 3, Table 4, Table 5, and Table 6.

Table 1: Results of BLAQ with different m and a for VGG on CIFAR10.

VGG on CIFAR10		
m	a	Top-1 accuracy
5	0.95	90.05%
	0.9	90.51%
	0.8	90.80%
	0.7	91.02%
	0.6	**91.50%**
	0.5	90.60%
	0.4	90.87%
	0.3	90.39%
10	0.95	89.85%
	0.9	89.93%
	0.8	90.14%
	0.7	90.46%
	0.6	90.83%
	0.5	91.07%
	0.4	90.64%
	0.3	89.60%

Effect of m on BLAQ. From these tables, it can be observed that BLAQ performs well, getting good Top-1 accuracy results when $m = 5$ on VGG, 4-layer Model, SVHNNet, and LeNet5, and when $m = 10$ on ResNet18 network.

Effect of a on BLAQ. BLAQ performs one-step-forward search to find the trial gradient at the next-step, and then backtracks to update the new weights through the current-step gradient and next-step trial gradient. In this way, the estimated new gradient can be more accurate, as shown in Eq.8 and Eq.9. Hence, it is important to investigate the effect of the coefficient a in Eq.8 and Eq.9. From the tables, we can observe that the methed obtains the best results when $a = 0.6$ on VGG, 4-layer Model, SVHNNet, and LeNet5, and when $a = 0.9$ on ResNet18 network.

7. Implementation Details

In this section, the illustration of experimental settings are detailed as follows.

1) The CIFAR10 dataset contains 60000 32×32 color images from 10 object classes. We sample 45000 images from the dataset for training, another 5000 for validation, and the rest 10000 for testing. The mini-batch with size is set to 100. Following the setting in (Hou et al. 2017), we use the modified VGG:

$$(2 \times 128C3) - MP2 - (2 \times 256C3) - MP2 - (2 \times 512C3) - MP2 - (2 \times 1024FC) - 10SVM.$$

where $C3$ is a 3×3 ReLU convolution layer, $MP2$ is a 2×2 max-pooling layer, FC is a fully-connected layer, and SVM is a $L2$-SVM output layer using the square hinge loss. The maximum number of epochs is 100. The learning rate for the weight-binarized network starts at 0.02, and decays by a factor of 0.1 after every 5 epochs. As suggested in the ablation study, the update operation for the model training is conducted in a one-step forward and backtrack way

Table 2: Results of BLAQ with different m and a for 4-layer Model on MNIST.

4-layer model on MNIST		
m	a	Top-1 accuracy
5	0.95	99.062%
	0.9	99.076%
	0.8	99.078%
	0.7	99.081%
	0.6	**99.110%**
	0.5	99.070%
	0.4	99.064%
	0.3	99.057%
10	0.95	99.029%
	0.9	99.044%
	0.8	99.093%
	0.7	99.078%
	0.6	99.089%
	0.5	99.107%
	0.4	99.084%
	0.3	99.043%

Table 3: Results of BLAQ with different m and a for LeNet5 on MNIST.

LeNet5 on MNIST		
m	a	Top-1 accuracy
5	0.95	99.304%
	0.9	99.339%
	0.8	99.348%
	0.7	99.332%
	0.6	**99.380%**
	0.5	99.354%
	0.4	99.318%
	0.3	99.314%
10	0.95	99.125%
	0.9	99.150%
	0.8	99.331%
	0.7	99.334%
	0.6	99.354%
	0.5	99.351%
	0.4	99.352%
	0.3	99.324%

Table 4: Results of BLAQ with different m and a for SVHNNet on SVHN.

SVHNNet on SVHN		
m	a	Top-1 accuracy
5	0.95	97.81%
	0.9	97.85%
	0.8	97.83%
	0.7	97.90%
	0.6	**98.13%**
	0.5	97.95%
	0.4	97.90%
	0.3	97.80%
10	0.95	97.75%
	0.9	97.78%
	0.8	97.79%
	0.7	97.81%
	0.6	97.86%
	0.5	97.80%
	0.4	97.74%
	0.3	97.60%

Table 5: Results of BLAQ with different m and a for ResNet18 on ILSVRC12 (1-bit).

ResNet18 on ILSVRC12 (1-bit)		
m	a	Top-1 accuracy
5	0.95	66.66%
	0.9	66.70%
	0.8	66.71%
	0.7	66.70%
	0.6	66.65%
	0.5	66.59%
	0.4	66.60%
	0.3	66.50%
10	0.95	66.69%
	0.9	**66.73%**
	0.8	66.70%
	0.7	66.71%
	0.6	66.65%
	0.5	66.50%
	0.4	66.59%
	0.3	66.57%

Table 6: Results of BLAQ with different m and a for ResNet18 on ILSVRC12 (2-bit).

ResNet18 on ILSVRC12 (2-bit)		
m	a	Top-1 accuracy
5	0.95	69.51%
	0.9	69.59%
	0.8	69.61%
	0.7	69.54%
	0.6	69.50%
	0.5	69.49%
	0.4	69.38%
	0.3	69.45%
10	0.95	69.50%
	0.9	**69.62%**
	0.8	69.58%
	0.7	69.55%
	0.6	69.31%
	0.5	69.49%
	0.4	69.45%
	0.3	69.45%

proposed in BLAQ after every 5 epochs, and in conventional way of LAQ (Hou et al. 2017) for other epochs.

2) The MNIST dataset contains 28×28 gray scale images from 10 digit classes, from which we use 50000 images for training, 10000 for validation, and the rest 10000 for testing. For fair comparison, we use two network architectures (Hou et al. 2017): The first one is a 4-layer model:

$$784FC - 2048FC - 2048FC - 2048FC - 10SVM.$$

The second is a modified LeNet5 model:

$$20C3 - MP2 - 50C3 - MP2 - 500FC - 10SVM.$$

The first model's maximum number of epochs is set to 50, and the mini-batch size 100. The second model's maximum number of epochs and the mini-batch size are set to 20 and 128, respectively. The learning rate of weight binarization networks starts at 0.005, and decays by a factor of 0.1 after every 5 epochs. As suggested in the ablation study, the update operation for the model training is conducted in a one-step forward and backtrack way proposed in BLAQ after every 5 epochs, and in conventional way of LAQ (Hou et al. 2017) for other epochs.

3) The SVHN dataset comprises of 32×32 color images from 10 digit classes. It contains the train file, test folder and extra folder, with 33402, 13068 and 202353 tag images, respectively. We use 598388 images for the training, another 6000 for the validation, and the rest 26032 for the testing. The mini-batch size is set to 50, and the used network architecture is as follows:

$$(2 \times 64C3) - MP2 - (2 \times 128C3) - MP2 - (2 \times 256C3) - MP2 - (2 \times 1024FC) - 10SVM.$$

The maximum number of epochs is set to 50. The learning rate for the weight-binarized network starts at 0.001, and decays by a factor of 0.1 at epochs 5. Again, as suggested in the ablation study, the update operation for the model training is performed in the way proposed in BLAQ after every 5 epochs, and in conventional way of LAQ (Hou et al. 2017) for other epochs.

4) ImageNet (ILSVRC12) dataset consists of 1.2 million high-resolution images from 1000 object classes. The validation set contains 50000 images for reporting accuracy levels. The experimental setup follows ALQ (Qu et al. 2020): the mini-batch size is set to 256; the network model uses ResNet18. In addition, the settings of data preprocessing also follows (Qu et al. 2020). Besides, the maximum number of epochs is set to 175. The learning rate for the weight-binarized network starts at 2e-4, and decays by a factor of 0.95 after every epoch. Again, as suggested in the ablation study, the update operation for the model training is performed in the way proposed in BLAQ after every 10 epochs, and in conventional way of LAQ (Hou et al. 2017) for other epochs.

References

Hou, L.; Yao, Q.; Kwok, J.; and Tin, Y. 2017. Loss-Aware Binarization of Deep Networks. In *Proceedings of the 5th International Conference on Learning Representations*.

Qu, Z.; Zhou, Z.; Cheng, Y.; and Thiele, L. 2020. Adaptive loss-aware quantization for multi-bit networks. In *Proceedings of the 33th IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7988–7997.