In-Place Activated BatchNorm for Memory-Optimized Training of DNNs

Samuel Rota Bulò, Lorenzo Porzi, Peter Kontschieder
Mapillary Research
research@mapillary.com

Abstract

In this document we provide the following additional contributions to our CVPR 2018 main paper:

- *We provide side-by-side comparisons of ImageNet results for ResNeXt architectures when trained with* RELU *and* LEAKY RELU *activations and mutually replaced activations, respectively.*
- *We provide derivations of the gradients computed by* INPLACE-ABN *I and* INPLACE-ABN *II.*

1. Validation of LEAKY RELU vs. RELU

We compared the validation accuracy obtained when replacing RELU with LEAKY RELU in a `ResNeXt-101` trained with RELU. We also considered the opposite case, replacing LEAKY RELU with RELU in a LEAKY RELU-trained network (see Table 1). Our results are in line with [2], and never differ by more than a single point per training except for the 320^2 center crop evaluation top-1 results, probably also due to non-deterministic training behaviour.

Network	activation		224^2 center		224^2 10-crops		320^2 center	
	training	validation	top-1	top-5	top-1	top-5	top-1	top-5
`ResNeXt-101`	RELU	RELU	77.74	93.86	79.21	94.67	79.17	94.67
`ResNeXt-101`	RELU	LEAKY RELU	76.88	93.42	78.74	94.46	78.37	94.25
`ResNeXt-101`	LEAKY RELU	LEAKY RELU	77.04	93.50	78.72	94.47	77.92	94.28
`ResNeXt-101`	LEAKY RELU	RELU	76.81	93.53	78.46	94.38	77.84	94.20

Table 1. Imagenet validation set results using `ResNeXt-101` and RELU/LEAKY RELU exchanged activation functions during training and validation.

2. Derivation of Gradient $\frac{\partial L}{\partial x}$

We follow the gradient derivations as provided in the original batch normalization paper [1] and rewrite them as a function of $\hat{x}$, starting with generally required derivatives for INPLACE-ABN I & II and particular ones of INPLACE-ABN II.

$$\frac{\partial y_j}{\partial \gamma} = \hat{x}_j\,, \qquad \frac{\partial y_j}{\partial \beta} = 1\,, \qquad \frac{\partial y_j}{\partial \hat{x}_j} = \gamma\,,$$

$$\frac{\partial L}{\partial \gamma} = \sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\frac{\partial y_j}{\partial \gamma} = \sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\hat{x}_j\,, \qquad \frac{\partial L}{\partial \beta} = \sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\frac{\partial y_j}{\partial \beta} = \sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\,, \qquad \frac{\partial L}{\partial \hat{x}_j} = \frac{\partial L}{\partial y_j}\frac{\partial y_j}{\partial \hat{x}_j} = \frac{\partial L}{\partial y_j}\gamma\,,$$

$$\frac{\partial \hat{x}_j}{\partial \sigma^2_{\mathcal{B}}} = -\frac{1}{2(\sigma^2_{\mathcal{B}}+\epsilon)}\frac{x_j-\mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}} = -\frac{\hat{x}_j}{2(\sigma^2_{\mathcal{B}}+\epsilon)}\,, \qquad \frac{\partial \hat{x}_j}{\partial \mu_{\mathcal{B}}} = -\frac{1}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}}\,,$$

$$\frac{\partial L}{\partial \sigma^2_{\mathcal{B}}} = \sum_{j=1}^{m} \frac{\partial L}{\partial \hat{x}_j}\frac{\partial \hat{x}_j}{\partial \sigma^2_{\mathcal{B}}} = -\frac{\gamma}{2(\sigma^2_{\mathcal{B}}+\epsilon)}\sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\hat{x}_j = -\frac{\gamma}{2(\sigma^2_{\mathcal{B}}+\epsilon)}\frac{\partial L}{\partial \gamma}\,,$$

$$\frac{\partial L}{\partial \mu_{\mathcal{B}}} = \sum_{j=1}^{m} \frac{\partial L}{\partial \hat{x}_j}\frac{\partial \hat{x}_j}{\partial \mu_{\mathcal{B}}} = -\frac{\gamma}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}}\sum_{j=1}^{m} \frac{\partial L}{\partial y_j} = -\frac{\gamma}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}}\frac{\partial L}{\partial \beta}\,,$$

$$\frac{\partial \sigma^2_{\mathcal{B}}}{\partial x_i} = \frac{2(x_i-\mu_{\mathcal{B}})}{m}\,, \qquad \frac{\partial \mu_{\mathcal{B}}}{\partial x_i} = \frac{1}{m}\,, \qquad \frac{\partial \hat{x}_i}{\partial x_i} = \frac{1}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}}\,,$$

$$\frac{\partial L}{\partial x_i} = \frac{\partial L}{\partial \hat{x}_i}\frac{\partial \hat{x}_i}{\partial x_i} + \frac{\partial L}{\partial \sigma^2_{\mathcal{B}}}\frac{\partial \sigma^2_{\mathcal{B}}}{\partial x_i} + \frac{\partial L}{\partial \mu_{\mathcal{B}}}\frac{\partial \mu_{\mathcal{B}}}{\partial x_i} = \left(\frac{\partial L}{\partial y_i} - \frac{1}{m}\frac{\partial L}{\partial \gamma}\hat{x}_i - \frac{1}{m}\frac{\partial L}{\partial \beta}\right)\frac{\gamma}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}}\,.$$

For INPLACE-ABN II, we write gradients $\frac{\partial L}{\partial \gamma}$ and $\frac{\partial L}{\partial x}$ as functions of y instead of $\hat{x}$ in the following way:

$$\frac{\partial L}{\partial \gamma} = \sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\hat{x}_j = \sum_{j=1}^{m} \frac{\partial L}{\partial y_j}\frac{y_j-\beta}{\gamma} = \frac{1}{\gamma}\sum_{j=1}^{m} \frac{\partial L}{\partial y_j}y_j - \frac{\beta}{\gamma}\sum_{j=1}^{m} \frac{\partial L}{\partial y_j} = \frac{1}{\gamma}\left[\sum_{j=1}^{m} \frac{\partial L}{\partial y_j}y_j - \beta\frac{\partial L}{\partial \beta}\right]\,,$$

$$\begin{aligned}\frac{\partial L}{\partial x_i} &= \left(\frac{\partial L}{\partial y_i} - \frac{1}{m}\frac{\partial L}{\partial \gamma}\hat{x}_i - \frac{1}{m}\frac{\partial L}{\partial \beta}\right)\frac{\gamma}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}} \\ &= \left(\frac{\partial L}{\partial y_i} - \frac{1}{m}\frac{\partial L}{\partial \gamma}\frac{y_i-\beta}{\gamma} - \frac{1}{m}\frac{\partial L}{\partial \beta}\right)\frac{\gamma}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}} \\ &= \left[\frac{\partial L}{\partial y_i} - \frac{1}{\gamma m}\frac{\partial L}{\partial \gamma}y_i - \frac{1}{m}\left(\frac{\partial L}{\partial \beta} + \frac{\beta}{\gamma}\frac{\partial L}{\partial \gamma}\right)\right]\frac{\gamma}{\sqrt{\sigma^2_{\mathcal{B}}+\epsilon}}\,.\end{aligned}$$

References

[1] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. 2

[2] B. Xu, N. Wang, T. Chen, and M. Li. Empirical evaluation of rectified activations in convolutional network. *CoRR*, abs/1505.00853, 2015. 1