NAM: Normalization-based Attention Module

Yichao Liu
Helmholtz-Zentrum Dresden-Rossendorf
Dresden, Germany
y.liu@hzdr.de

Zongru Shao
Helmholtz-Zentrum Dresden-Rossendorf
Dresden, Germany
Center for Advanced Systems Understanding
Görlitz, Germany
z.shao@hzdr.de

Yueyang Teng
College of Medicine and Biological Information Engineering
Northeastern University
110004 Shenyang, China
tengyy@bmie.neu.edu.cn

Nico Hoffmann
Helmholtz-Zentrum Dresden-Rossendorf
Dresden, Germany
n.hoffmann@hzdr.de

Abstract

Recognizing less salient features is the key for model compression. However, it has not been investigated in the revolutionary attention mechanisms. In this work, we propose a novel normalization-based attention module (NAM), which suppresses less salient weights. It applies a weight sparsity penalty to the attention modules, thus, making them more computational efficient while retaining similar performance. A comparison with three other attention mechanisms on both Resnet and Mobilenet indicates that our method results in higher accuracy. Code for this paper can be publicly accessed at `https://github.com/Christian-lyc/NAM`.

1 Introduction

Attention mechanisms have been one of the heated research interests in recent years (Wang et al. [2017], Hu et al. [2018], Park et al. [2018], Woo et al. [2018], Gao et al. [2019]). It assists deep neural networks to suppress less salient pixels or channels. Many of the prior studies focus on capturing salient features with attention operations (Zhang et al. [2020], Misra et al. [2021]). Those methods successfully exploit the mutual information from different dimensions of features. However, they lack consideration on the contributing factors of weights, which is capable of further suppressing the insignificant channels or pixels. Inspired by Liu et al. [2017], we aim to utilize the contributing factors of weights for the improvement of attention mechanisms. We use a scaling factor of batch normalization which uses the standard deviation to represent the importance of weights. This can avoid adding fully-connected and convolutional layers, which is used in the SE, BAM and CBAM. Thus, we propose an efficient attention mechanism – Normalization-based Attention Module (NAM).

35th Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

2 Related work

Many prior works attempt to improve the performance of neural networks by suppressing insignificant weights. Squeeze-and-Excitation Networks (SENet) (Hu et al. [2018]) integrate the spatial information into channel-wise feature responses and compute the corresponding attention with two multi-layer-perceptron (MLP) layers. Later, Bottleneck Attention Module (BAM) (Park et al. [2018]) builds separated spatial and channel submodules in parallel and they can be embedded into each bottleneck block. Convolutional Block Attention Module (CBAM) (Woo et al. [2018]) provides a solution that embeds the channel and spatial attention submodules sequentially. To avoid the ignorance of cross-dimension interactions, Triplet Attention Module (TAM) (Misra et al. [2021]) takes account of dimension correlations by rotating the feature maps. However, these works neglect information from the tuned weights from training. Therefore, we aim to highlight salient features by utilizing the variance measurement of the trained model weights.

3 Methodology

We propose NAM as an efficient and lightweight attention mechanism. We **adopt the module integration from CBAM** (Woo et al. [2018]) and redesign the channel and spatial attention submodules. Then, a NAM module is embedded at the end of each network block. For residual networks, it is embedded at the end of the residual structures. For the **channel attention** submodule, we use a scaling factor from batch normalization (BN) (Ioffe and Szegedy [2015]), as shown in Equation (1). The scaling factor measures the variance of channels and indicates their importance.

$$B_{out} = BN(B_{in}) = \gamma \frac{B_{in} - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \beta \tag{1}$$

where $\mu_{\mathcal{B}}$ and $\sigma_{\mathcal{B}}$ are the mean and standard deviation of mini batch $\mathcal{B}$, respectively; γ and β are trainable affine transformation parameters (scale and shift) (Ioffe and Szegedy [2015]). The channel attention submodule is shown in Figure 1 and Equation (2), where $\mathbf{M}_c$ represents the output features. γ is the scaling factor for each channel, and the weights are obtained as $W_\gamma = \gamma_i / \sum_{j=0} \gamma_j$. We also apply a scaling factor of BN to the spatial dimension to measure the importance of pixels. We name it pixel normalization. The corresponding **spatial attention** submodule is shown in Figure 2 and Equation (3), where the output is denoted as $\mathbf{M}_s$. λ is the scaling factor, and the weights are $W_\lambda = \lambda_i / \sum_{j=0} \lambda_j$.

To suppress the less salient weights, we add a regularization term into the loss function, as shown in Equation (4) (Liu et al. [2017]), where x denotes the input; y is the output; W represents network weights; $l(\cdot)$ is the loss function; $g(\cdot)$ is the l_1 norm penalty function; p is the penalty that balances $g(\gamma)$ and $g(\lambda)$.

$$\mathbf{M}_c = sigmoid(W_\gamma(BN(\mathbf{F}_1))) \tag{2}$$

$$\mathbf{M}_s = sigmoid(W_\lambda(BN_s(\mathbf{F}_2))) \tag{3}$$

$$Loss = \sum_{(x,y)} l(f(x, W), y) + p \sum g(\gamma) + p \sum g(\lambda) \tag{4}$$

4 Experiment

In this section, we compare the performance of NAM with SE, BAM, CBAM, and TAM for ResNet and MobileNet. We evaluate every method with four Nvidia Tesla V100 GPUs on a cluster. We first run ResNet50 on CIFAR-100 (Krizhevsky et al. [2009]) and use the same preprocess and training configurations as CBAM (Woo et al. [2018]), with p as 0.0001. The comparison in Table 1 indicates that NAM with channel or spatial attention alone outperforms the other four attention mechanisms. We then run MobileNet on ImageNet (Deng et al. [2009]) as it is one of the standard datasets for image classification benchmarks. We set p as 0.001 and the rest of the configurations the same as CBAM. The comparison in Table 2 shows that NAM with channel and spatial attention combined outperforms the other three with similar computation complexity.

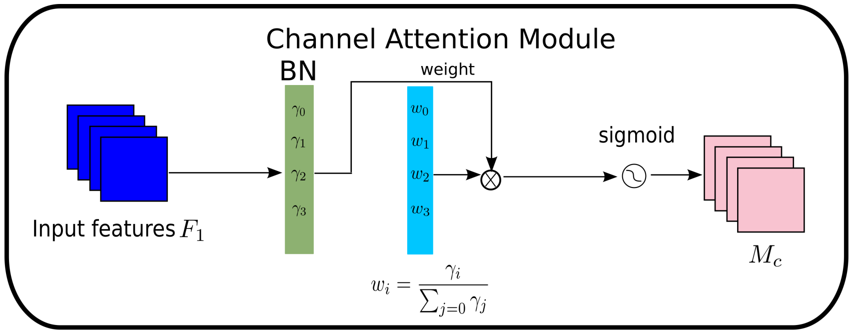

Figure 1: Channel attention mechanism

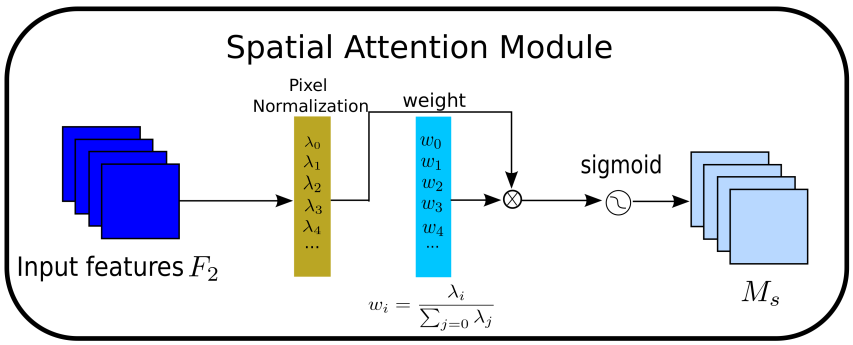

Figure 2: Spatial attention mechanism

Table 1: Classification results on CIFAR-100

Architecture	Parameters	FLOPs	Top-1 Error (%)	Top-5 Error (%)
ResNet 50	23.71M	1.30G	22.74	6.37
ResNet 50 + SE	26.22M	1.31G	20.29	5.18
ResNet 50 + BAM	24.06M	1.33G	19.97	5.03
ResNet 50 + CBAM	26.24M	1.31G	19.44	4.66
ResNet 50 + TAM	23.71M	1.33G	20.15	5.13
ResNet 50 + NAM(ch*)	23.74M	1.31G	19.09	4.5
ResNet 50 + NAM(sp*)	23.71M	1.31G	19.38	4.72

$\star$ ch stands for channel attention only; sp indicates spatial attention only.

Table 2: Classification results on ImageNet

Architecture	Parameters	FLOPs	Top-1 Error (%)	Top-5 Error (%)
MobileNet V2	3.51M	0.31G	30.52	11.20
MobileNet V2 + SE	3.53M	0.32G	29.77	10.65
MobileNet V2 + BAM	3.54M	0.32G	29.91	10.80
MobileNet V2 + CBAM	3.54M	0.32G	29.74	10.66
MobileNet V2 + NAM	3.51M	0.32G	29.34	10.18

5 Conclusion

We proposed a NAM module that is more efficient by suppressing the less salient features. Our experiments indicate that NAM provides efficiency gain on both ResNet and MobileNet. We are conducting a detailed analysis of the performance of NAM regarding its integration variations and hyper-parameter tuning. We also plan to optimize NAM with different model compression techniques to promote its efficiency. In the future, we will investigate its effects on other deep learning architectures and applications.

References

Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2017.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

Jongchan Park, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Bam: Bottleneck attention module. *arXiv preprint arXiv:1807.06514*, 2018.

Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

Zilin Gao, Jiangtao Xie, Qilong Wang, and Peihua Li. Global second-order pooling convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033, 2019.

Zhizheng Zhang, Cuiling Lan, Wenjun Zeng, Xin Jin, and Zhibo Chen. Relation-aware global attention for person re-identification. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 3186–3195, 2020.

Diganta Misra, Trikay Nalamada, Ajay Uppili Arasanipalai, and Qibin Hou. Rotate to attend: Convolutional triplet attention module. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3139–3148, 2021.

Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *Proceedings of the IEEE international conference on computer vision*, pages 2736–2744, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

A Appendix

A.1 Comparison of CBAM and NAM regarding the number of parameters

Table 3: The comparison of channel attention for CBAM and NAM on ResNet50

Parameters	CBAM	NAM
Block1	512*4*512*4/16*2 ($C * R * C * R/r * 2$)	512*4 ($C * R$)
Block2	256*4*256*4/16*2	256*4
Block3	128*4*128*4/16*2	128*4
Block4	64*4*64*4/16*2	64*4
Overhead	696320	3840

We show a comparison of the number of parameters in CBAM and NAM in Table 4 and 3. They empirically verify the parameter reduction of NAM. In the channel attention module, C represents the number of the input channels of each block. R represents the expanding ratio of each block. r represents the reduction ratio utilized in the MLP to compute the channel attention, which is set to 16 in CBAM. The kernel size is denoted as k, which is 7. In NAM, H and W represent the height and width of the input images respectively. From Table 4 and 3, we observe a significant parameter reduction in the channel attention module and an insignificant increase of parameters in the spatial attention module of NAM against CBAM. As a result, NAM has fewer parameters than CBAM.

Table 4: The comparison of spatial attention for CBAM and NAM on ResNet50

Parameters	CBAM	NAM
Block1	2*1*7*7 ($2 * 1 * k^2$)	32*32 ($H * W$)
Block2	2*1*7*7	16*16
Block3	2*1*7*7	8*8
Block4	2*1*7*7	4*4
Overhead	392	1360