Attention Module

1 Introduction

Attention Module seems to help improve the accuracy of CNN without introducing much more parameters and computation. In [1], they designed the attention module as following:

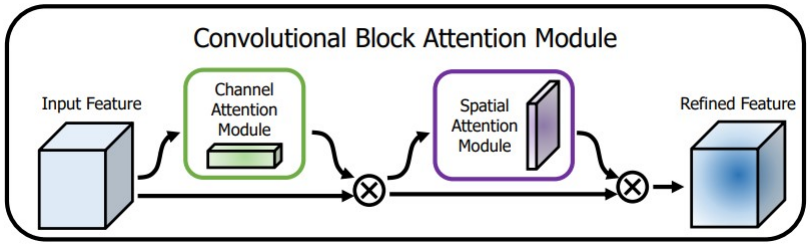

Figure 1: Convolutional Block Attention Module

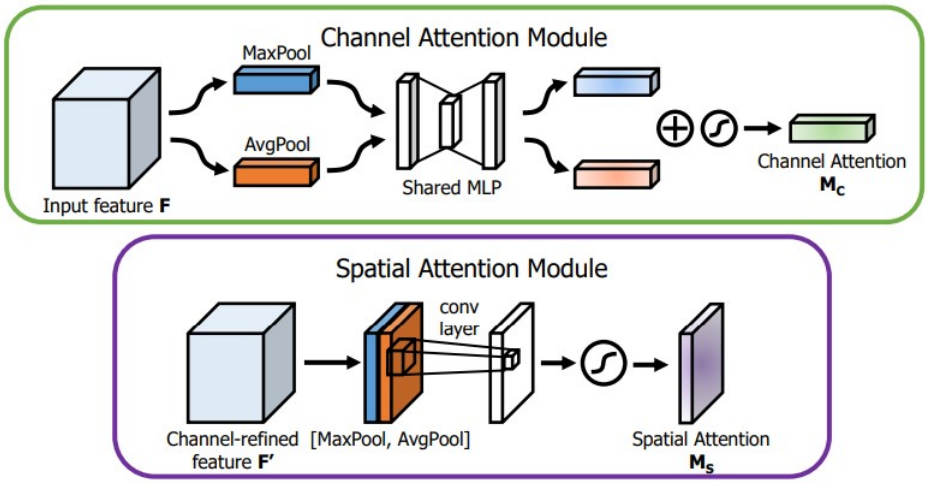

Figure 2: Channel Attention and Spatial Attention

In their paper, they added the attention modules to all blocks in ResNet. But maybe not all the blocks need the attention modules. Also when merging the MaxPool and AvgPool, they simply added them together. Perhaps assign them with different weights may improve the accuracy. So an interesting topic is that where to insert the attention modules and how to merge MaxPool and AvgPool.

2 Method

Let $CBAM()$ be the function of Attention Module. Then original codes work in this way in each ResNet Block:

$$\begin{aligned} Given\ input: &\quad x \\ residual &= x \\ out &= ResBlock(x) \\ out &= CBAM(out) \\ out &= out + residual \end{aligned}$$

where $ResBlock$ is made up of $Conv2d,\ BatchNorm,\ ReLU$.
We consider searching the structure and position of the attention module. In CBAM, the attention module is made up of a channel attention(CA) and followed by a spatial attention(SA). We design a structure that the network can search a proper structure of attention module, which can be CA follow SA, SA follow CA or CA parallel SA.
Now we let $\oplus$ denotes sequential with, and $(,)$ be parallel to. Given hyperparameters α, the attention module becomes:

$$\begin{aligned} Given\ input: &\quad x, \alpha \\ define: p_i &= softmax(\alpha_1; \alpha) \\ residual &= x \\ out &= ResBlock(x) \\ out &= out * p_0 + CA \oplus SA(out) * p_1 + SA \oplus CA(out) * p_2 \\ &\quad + CA(out) * p_3 + SA(out) * p_4 \\ out &= out + residual \end{aligned}$$

And we choose the highest p_i operation as the final operation after certain epochs. We also consider the case that p_3 and p_4 are close and their sum is the highest($p_3 + p_4 > max(p_0, p_1, p_2)$ and $|p_3 - p_4| < 0.1$). In this case, we will choose operation (CA, SA), which we assign p_3, p_4 to be $1/2$ and then fuse them together. As a consequence, the potential structure of attention module contains: Identity, CA, SA, CA$\oplus$SA, SA$\oplus$CA, (CA,SA).

Meanwhile, using Darts method will have 'collapse' issue where architecture parameters and weights are competing against each others. As a result, the model tends to select identity as the operation. To solve this, we also add an penalty function, forcing the model unwilling to choose identity.

$$\begin{aligned} P(\alpha, r, m) &= w \sum_{i=1}^{n} softmax(\alpha_i; \alpha)^m * r_i \\ &= w \sum_{i=1}^{n} {p_i}^m * r_i \end{aligned}$$

Here r is the ratio term, we use $r = [4, 1, 1, 2, 2]$. m is the power term, we use $m = 1.1$, $w = 0.1$ here.

3 Experiment

We test ResNet-18 on Cifar-100. And here are the experiment results:

Model	Acc1.	Acc2.	Acc3.	Acc4.	Acc5.	Best Acc.	Avg Acc.	Param.	FLOPs.
ResNet18	76.44	76.08	76.43	76.08	76.29	76.44	76.264		
w/ CBAM	76.69	76.34	76.61	76.66	76.33	76.69	76.526		
w/ SSCBAM	77.25	76.84	76.97	77.28	76.81	77.28	77.03		
w/ MVAM	76.67	76.79	77.13	76.83	76.49	77.13	76.782		
w/ SSMVAM	76.64	76.33	76.67	76.93	76.73	76.93	76.66		

Table 1: Experiment Results (Spatial kernel=7)

Model	Acc1.	Acc2.	Acc3.	Acc4.	Acc5.	Best Acc.	Avg Acc.	Param.	FLOPs.
ResNet18	76.44	76.08	76.43	76.08	76.29	76.44	76.264		
w/ CBAM	76.55	76.51	76.20	76.55	76.42	76.55	76.446		
w/ SSCBAM	77.27	77.12	76.54	76.87	76.93	77.27	76.946		
w/ MVAM	76.37	76.90	76.69	76.65	76.76	76.90	76.674		
w/ SSMVAM	76.78	76.90	76.73	76.47	76.51	76.90	76.678		

Table 2: Experiment Results (Spatial kernel=3)

Here CBAM is Convolutional Block Attention Module, and SSCBAM is Searched Structure Convolutional Block Attention Module. MVAM is Mean and Variance Attention Module. Instead of using Mean and Max for AM, while MVAM uses Mean and Variance. SSMVAM is Searched Structure Mean and Variance Attention Module.

And we can also look at the hyper-parameter . Here is one example in our search method. We first initialize α with all 0. And after 10 epochs' searching, viewing $softmax(\alpha)$ in all ResBlock in ResNet-18 as weights, the weights become:

ResBlock	Id	CA$\oplus$ SA	SA$\oplus$ CA	CA	SA	Op.
α_0	4.7375e-04	4.4295e-01	5.5134e-01	2.0592e-03	3.1687e-03	SA$\oplus$ CA
α_1	1.5267e-04	4.1338e-01	5.8425e-01	7.1946e-04	1.4979e-03	SA$\oplus$ CA
α_2	1.4487e-04	2.9350e-01	7.0241e-01	5.5964e-04	3.3896e-03	SA$\oplus$ CA
α_3	3.0173e-04	3.4739e-01	6.4989e-01	1.3028e-03	1.1119e-03	SA$\oplus$ CA
α_4	3.8662e-04	4.1580e-01	5.1593e-01	8.8905e-04	6.6994e-02	SA$\oplus$ CA
α_5	3.4632e-04	5.1557e-01	4.8145e-01	1.0539e-03	1.5781e-03	CA$\oplus$ SA
α_6	7.9483e-05	4.9560e-01	5.0244e-01	5.1867e-04	1.3604e-03	SA$\oplus$ CA
α_7	6.4367e-05	4.9740e-01	5.0105e-01	5.5208e-04	9.3461e-04	SA$\oplus$ CA

Table 3: Score and Operation for SSCBAM

ResBlock	Id	CA⊕ SA	SA⊕ CA	CA	SA	Op.
α_0	1.1785e-03	9.0141e-01	8.1482e-02	1.4495e-02	1.4338e-03	CA⊕ SA
α_1	5.8767e-04	3.1331e-01	6.7345e-01	9.8350e-03	2.8173e-03	CA⊕ SA
α_2	0.0071	0.4613	0.4444	0.0653	0.0218	SA⊕ CA
α_3	0.0028	0.3896	0.5789	0.0128	0.0160	SA⊕ CA
α_4	0.0208	0.2131	0.6099	0.0522	0.1041	SA⊕ CA
α_5	0.0054	0.4769	0.4700	0.0235	0.0243	CA⊕ SA
α_6	7.8822e-03	1.9313e-02	3.3015e-01	6.8491e-03	6.3580e-01	SA
α_7	8.3575e-03	3.1563e-04	2.3673e-03	4.5235e-04	9.8851e-01	SA

Table 4: Score and Operation for SSMVAM

References

[1] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module, 2018.

[2] https://github.com/developer0hye/ZAM