How to handle the BatchNorm layer when training fully convolutional networks by finetuning?

Training fully convolutional nerworks (FCNs) for pixelwise semantic segmentation is very memory intensive. So we often use batchsize=1 for traing FCNs. However, when we finetune the pretrained networks with BatchNorm (BN) layers, batchsize=1 doesn't make sense for the BN layers. So, how to handle the BN layers?

Some options:

1. delete the BN layers (merge the BN layers with the preceding layers for the pretrained model)
2. Freeze the parameters and statistics of the BN layers
3. ....

which is better and any demo for implementation in pytorch/tf/caffe?

tensorflow deep-learning caffe pytorch

edited Jun 19 '17 at 3:17

asked Jun 19 '17 at 3:10

Liang Xiao
459 5 12

5 Answers

Having only one element will make the batch normalization zero if epsilon is non-zero
(variance is zero, mean will be same as input).
Its better to delete the BN layers from the network and try the activation function SELU (scaled exponential linear units). This is from the paper 'Self normalizing neural networks' (SNNs).

Quote from the paper:

> While batch normalization requires explicit normalization, neuron activations of SNNs automatically converge towards zero mean and unit variance. The activation function of SNNs are “scaled exponential linear units” (SELUs), which induce self-normalizing properties.

The SELU is defined as:

```
def selu(x, name="selu"):
  alpha = 1.6732632423543772848170429916717
  scale = 1.0507009873554804934193349852946
  return scale * tf.where(x >= 0.0, x, alpha * tf.nn.elu(x))
```

answered Jun 19 '17 at 23:10

vijay m
4,546 1 4 24

According to my experiments in PyTorch, if convolutional layer before the BN outputs more than one value (i.e. 1 x feat_nb x height x width, where height > 1 or width > 1), then the BN still works fine even when the batch size is equal to one. However, I suspect that in this case the variance estimate might be very biased since all samples that are used for variance calculation come from the same image. Therefore in my case I still decided to use small batch.

edited Jul 1 '17 at 21:49

answered Jul 1 '17 at 21:34

Stephan