Hyomin Seo

Title : Probes, Batch Norms and ReNorm

Describe the main method and key takeaways from "probes." What was the main hang-up that prevented the authors from completing more significant experimentation?

Definition of ‘Probes’

The linear classifier hat takes the features of each layer separately to fit a linear classifier to predict the original classes; fit linear classifier probes to prodiet the certain classes.
Given a linear classifier fk evaluated over the training set, where fk is itself optimized that at any given time, it reflects the current optimal thing that can be done with the features present. The linear classifier fk is considered ‘probe’.

Usages

Probes do not affect the model training. They only measure the level of linear separability of the features at a given layer. Probes are largely used to characterize different layers, debug bad models, get a sense of how the training progresses in well-behaved models.

Facts regarding the probes regarding numerous studies

The deeper the analysis goes, the more abstract original convolutional neural networks are.
The level of linear separability increases monotonically as we go to deeper layers. This is purely an indirect consequence of enforcing this constraint on the last layer.
The probes can be used to identify certain problematic behaviors in models that might not be apparent when we traditionally have access to only the prediction loss and error.

Concerned Failure on Probes

If the dimension is not properly scaled, probes can simply overfit on the features because there are too many features.
Failure example :
Considerably large model fitting random labels on ImageNet (Zhang *et al.*, 2016), in this incident, the probe measurements would be entirely meaningless in that situation.
With Skip connection, using probes we show that this solution was not working as intended, because half of the model stays unused. The weights are not zero, but there is no useful signal passing through that segment. The skip connection left a dead segment and skipped over it.

Describe the main method and key takeaways from "confidence penalty."
Is there a difference at inference time between batch-norm and batch-renorm?

Definition batch-norm and batch-renorm

Confidence penalty is a regulation (output regularizers) during an approach to supervised neural network training in which the loss function is augmented by a term that penalized over-confident output distribution. It is closely related to Label smoothing regularization**,** which estimated the marginalized effect of label-dropout during training, reducing overfitting by preventing a network from assigning full probability to each training example and maintaining a reasonable ratio between the logits of the incorrect classes.

BatchNorm/ ReNorm

Batch Norm is a technique adopted to soothe the change of input to layers in the network distribution, called ‘internal covariate shift’.

‘Internal covariate shift’ is an issue that arises in transfer learning is that the distribution of the inputs to layers deep in the network may change after each mini-batch when the weights are updated, which can cause the learning algorithm to forever chase a moving target.

Here, Batch normalization standardizes the input to a layer for each smaller dimension (mini) batch. Doing so stabilizes the learning process and reduces the number of training epochs required to train deep networks Batch renormalization adopts changing average of both mean and variance for training and inference steps, renormalization is used when changing average of the small batches is expected to have false approximation of the population mean and variance