Speeding up Convolutional Neural Networks

November 2016

1 Team

- Egor Krivov
- Arsen Sagoyan
- Pavel Tolmachev
- ShahRukh Athar

2 Background and Problem Formulation

Convolutional Neural Networks (CNNs) have now become ubiquitous in computer vision due their amazing ability to generate extremely rich feature maps and be close to human accuracy on a number of supervised learning tasks. However, this performance comes at the expense of computational and storage cost due to the many stacked layers of convolutions most CNNs use. Our goal is to use a number of linear algebra methods to try to speed up the computation and reduce their memory usage. Our starting point would be [1] which discusses a number of methods generate low rank approximations to filters and convolutions within a CNN which leads to a reasonable amount of speedup. We plan to add onto these methods my also simultaneously reducing the floating point precision of the weights and biases and observe the extent to which it affects the accuracies of neural networks.

3 Data

The dataset we would be using is the MNIST dataset, to evaluate the performance of the neural networks we train.

4 Related Work

Jaderberg *et al.* (2014) worked on developing methods to reduce computation by generating approximations to filters and convolutions. Courbariaux *et al.*

(2015) and Vanhoucke *et al.* (2011) explored reducing floating point precisions during training and testing of neural networks and observed how their accuracies changed.

5 Scope

By the end of the project we hope to have explored a number of ways in which CNNs could be sped up and made more memory efficient.

6 Evaluation

Our final evaluation would be based on how well the modified CNNs perform on the test MNIST images.

References

[1] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. "Speeding up Convolutional Neural Networks with Low Rank Expansions". In: *CoRR* abs/1405.3866 (2014). URL: `http://arxiv.org/abs/1405.3866`.