Large Scale Training of Neural Networks

Zhewei Yao, Amir Gholami
& Kurt Keutzer, Michael Mahoney

University of California, Berkeley

November 2018
DNN’s Impact on a wide range of problems

A. Gholami, S. Subramanian, V. Shenoy, N. Himthani, X. Yue, S. Zhao, P. Jin, K. Keutzer, G. Biros, A novel domain adaptation framework for medical image segmentation, BRATS, MICCAI 2018
A Semantic Segmentation using Detectron, Facebook Research
DNN’s Impact on a wide range of problems

Slide from Bo Li
Susceptibility to Adversarial Example

Z Yao, A Gholami, M Mahoney, K Keutzer Trust Region Based Adversarial Attack on Neural Networks
Susceptibility to Adversarial Example

° Despite noticeable impact, NN can be easily fooled

Original Frames

Adversarial perturbation injected into every other 10 frames

Song et al.: Delving into adversarial attacks on deep policies. ICLR Workshop 2017

[Chaowei Xiao, Bo Li, Jun-yun Zhu, Warren He, Mingyan Liu, Dawn Song, 2017]
Our Research

**Rapid Training of DNNs**

- **Collect/annotate adequate training data**
- **Find the right DNN architectures using design space exploration**
- **Create efficient implementation for embedded hardware**

**Integrated Parallelism** [2]

- 2\(^{nd}\) Order Method [3,7,8,9]

- **SqueezeNext** [4]
- **ShiftNet** [6]
- \(112\times\) smaller models

**Hardware and model Co-design** [5]

**Domain Adaptation** [1]

---


[7] Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and **Second-Order Information**, (under review)


[9] Z. Yao, N. Mu, K. Keutzer, MW. Mahoney. **Weight** Re-Initialization through Cyclical Batch Scheduling
SGD is very sensitive to hyper-parameters and in particular **batch size**

Batch size inter-dependent with:

- Degradation in accuracy
- Poor generalizability
- Robustness of model
- Training time
- Parallel Scalability

\[
W^{t+1} = W^t - \alpha \sum_{i=0}^{B} \nabla W f_i(W^t, x)
\]
High Level Outline

° DNN design requires training on large datasets
  • Time consuming
  • Need fast training -> parallelization -> large batch

° Large batch training does not work:
  • Degrades accuracy
  • Poor robustness to adversarial inputs
  • Existing solutions either do not work or require extensive hyper-parameter tuning

Summary of Contributions

° Extensive analysis of mini-batch SGD behavior for deep neural networks
  • Saddle points, adversarial robustness, sharp/flat minima
° A new **Hessian based** large batch size training
  • Degrades accuracy
    - **Equal or better accuracy**
  • Poor robustness to adversarial inputs
    - **More robust model**
  • Existing solutions either do not work or require extensive hyper-parameter tuning
    - **No hyper-parameter tuning**
° Extensive testing of the proposed method on multiple datasets and multiple neural networks
  • Cifar-10/100, **ImageNet**, SVHN, Tiny ImageNet
Stochastic Gradient Descent (SGD)

Assume \( f(W^t, x) = \frac{1}{n} \sum_{i=1}^{n} f_i(W^t, x) \)

\[ W^{t+1} \leftarrow W^t - \alpha \cdot \nabla W f_i(W^t, x) \]

Pure SGD: compute gradient using 1 sample

\[ W^{t+1} \leftarrow W^t - \alpha \cdot \frac{1}{b} \sum_{i=k+1}^{k+b} \nabla W f_i(W^t, x) \]

Mini-batch: compute gradient using b samples

• Actually the name is a misnomer, *this is not a “descent” method*

Image from https://www.cs.umd.edu/~tomg/projects/landscapes/
Degradation in Accuracy

° Larger Batch often leads to degradation in accuracy

Why large batch suffers from poor generalization performance?

- A common belief is that large batch training gets attracted to "sharp minimas"
- Another theory is that large batch may get stuck in saddle points

Loss landscape from https://www.cs.umd.edu/~tomg/projects/landscapes/
Analysis through Hessian Lens!

- We back-propogate the Hessian operator (second derivative) and compute its spectrum during training along with total gradient.

- The Hessian spectrum is computed on all the training/testing examples using power method.

- We visualize the landscape of loss along dominant eigenvectors of the Hessian.

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS’18 (arXiv:1802.08241)
Loss Landscape at the end of training

Training/testing loss at the end of training along the dominant eigenvector of the Hessian (for Cifar-10 dataset)
Large Batch Size Training

Top 20 eigenvalues of the total Hessian w.r.t. model parameters for different batch size. Clearly larger batch size converges to points with higher Hessian spectrum.

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS'18 (arXiv:1802.08241)
Large Batch Size Training and Hessian Spectrum

Changes in the dominant eigenvalue of the Hessian w.r.t. weights and the total gradient is shown for different epochs during training. (Dotted = Robust Optimiz.) Large batch gets attracted to areas with larger Hessian spectrum (blue curve)

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS’18 (arXiv:1802.08241)
Hessian Based Adaptive Batch Size

Illustration of learning rate (a) and batch size (b) schedules of adaptive batch size as a function of training epochs based on C2 model on Cifar-10.

Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Results – Cifar 10

Cifar-10 has ten classes
- ~5000 examples per class
- Total 50,000 training images
- 10,000 testing images

Learning Multiple Layers of Features from Tiny Images, Alex Krizhevsky, 2009.
Our proposed method (ABSA) achieves better performance

### Results – Cifar10 - ResNet

ResNet-18 on Cifar-10

<table>
<thead>
<tr>
<th>batch size</th>
<th>BL</th>
<th>FB</th>
<th>GG</th>
<th>ABS</th>
<th>ABSA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc</td>
<td># updates</td>
<td>Acc</td>
<td># updates</td>
<td>Acc</td>
</tr>
<tr>
<td>128</td>
<td>83.05</td>
<td>35156</td>
<td>N.A</td>
<td>N.A</td>
<td>N.A</td>
</tr>
<tr>
<td>640</td>
<td>81.01</td>
<td>7031</td>
<td>84.59</td>
<td>7031</td>
<td>83.99</td>
</tr>
<tr>
<td>3200</td>
<td>74.54</td>
<td>1406</td>
<td>78.7</td>
<td>1406</td>
<td>84.27</td>
</tr>
<tr>
<td>5120</td>
<td>70.64</td>
<td>878</td>
<td>74.65</td>
<td>878</td>
<td>83.47</td>
</tr>
<tr>
<td>10240</td>
<td>68.75</td>
<td>439</td>
<td>30.99</td>
<td>439</td>
<td>83.68</td>
</tr>
<tr>
<td>16000</td>
<td>67.88</td>
<td>281</td>
<td>10.0</td>
<td>281</td>
<td>84.0</td>
</tr>
</tbody>
</table>

ABS/ABSA: Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Results – Cifar10 – Wide ResNet

° Our proposed method (ABSA) achieves better performance

WResNet16-4 on Cifar-10

<table>
<thead>
<tr>
<th>batch size</th>
<th>BL</th>
<th>FB</th>
<th>GG</th>
<th>ABS</th>
<th>ABSA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc.</td>
<td># updates</td>
<td>Acc.</td>
<td># updates</td>
<td>Acc.</td>
</tr>
<tr>
<td>128</td>
<td>87.64</td>
<td>35156</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
</tr>
<tr>
<td>640</td>
<td>86.2</td>
<td>7031</td>
<td>87.9</td>
<td>7031</td>
<td>87.84</td>
</tr>
<tr>
<td>3200</td>
<td>82.59</td>
<td>1406</td>
<td>73.2</td>
<td>1406</td>
<td>87.59</td>
</tr>
<tr>
<td>5120</td>
<td>81.4</td>
<td>878</td>
<td>63.27</td>
<td>878</td>
<td>87.85</td>
</tr>
<tr>
<td>10240</td>
<td>79.85</td>
<td>439</td>
<td>0.1</td>
<td>439</td>
<td>87.52</td>
</tr>
<tr>
<td>16000</td>
<td>81.06</td>
<td>281</td>
<td>0.1</td>
<td>281</td>
<td>88.28</td>
</tr>
</tbody>
</table>

ABS/ABSA: Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
### Results – Cifar10 - SqueezeNext

<table>
<thead>
<tr>
<th>BS</th>
<th>BL Acc.</th>
<th># Iters</th>
<th>GG Acc.</th>
<th># Iters</th>
<th>ABS Acc.</th>
<th># Iters</th>
<th>ABSA Acc.</th>
<th># Iters</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>92.02</td>
<td>78125</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
</tr>
<tr>
<td>256</td>
<td>91.88</td>
<td>39062</td>
<td>91.84</td>
<td>50700</td>
<td>91.7</td>
<td>40792</td>
<td>92.11</td>
<td>43352</td>
</tr>
<tr>
<td>512</td>
<td>91.68</td>
<td>19531</td>
<td>91.19</td>
<td>37050</td>
<td>92.15</td>
<td>32428</td>
<td>91.61</td>
<td>25388</td>
</tr>
<tr>
<td>1024</td>
<td>89.44</td>
<td>9766</td>
<td>91.12</td>
<td>31980</td>
<td>91.61</td>
<td>17046</td>
<td>91.66</td>
<td>23446</td>
</tr>
<tr>
<td>2048</td>
<td>83.17</td>
<td>4882</td>
<td>89.19</td>
<td>30030</td>
<td>91.57</td>
<td>21579</td>
<td>91.61</td>
<td>14027</td>
</tr>
<tr>
<td>4096</td>
<td>73.74</td>
<td>2441</td>
<td>91.83</td>
<td>29191</td>
<td>91.91</td>
<td>18293</td>
<td>92.07</td>
<td>21909</td>
</tr>
<tr>
<td>8192</td>
<td>63.71</td>
<td>1220</td>
<td>91.51</td>
<td>28947</td>
<td>91.77</td>
<td>22802</td>
<td>91.81</td>
<td>16778</td>
</tr>
<tr>
<td>16384</td>
<td>47.84</td>
<td>610</td>
<td>90.19</td>
<td>28828</td>
<td>92.12</td>
<td>17485</td>
<td>91.97</td>
<td>24361</td>
</tr>
</tbody>
</table>


ABS/ABSA: Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Results – SVHN

SVHN consists of ten classes

- Total 600,000 training images
- 26,000 testing images

Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A. Ng Reading Digits in Natural Images with Unsupervised Feature Learning NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011
### Results – SVHN - AlexNet

#### AlexNet on SVHN

<table>
<thead>
<tr>
<th>BS</th>
<th>BL Acc.</th>
<th># Iters</th>
<th>FB Acc.</th>
<th># Iters</th>
<th>GG Acc.</th>
<th># Iters</th>
<th>ABS Acc.</th>
<th># Iters</th>
<th>ABSA Acc.</th>
<th># Iters</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>94.90</td>
<td>81986</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
</tr>
<tr>
<td>512</td>
<td>94.76</td>
<td>20747</td>
<td>95.24</td>
<td>20747</td>
<td>95.49</td>
<td>51862</td>
<td>95.65</td>
<td>25353</td>
<td><strong>95.72</strong></td>
<td>24329</td>
</tr>
<tr>
<td>2048</td>
<td>95.17</td>
<td>5186</td>
<td>95.00</td>
<td>5186</td>
<td>95.59</td>
<td>45935</td>
<td>95.51</td>
<td>10562</td>
<td><strong>95.82</strong></td>
<td>16578</td>
</tr>
<tr>
<td>8192</td>
<td>93.73</td>
<td>1296</td>
<td>19.58</td>
<td>1296</td>
<td><strong>95.70</strong></td>
<td>44407</td>
<td>95.56</td>
<td>14400</td>
<td>95.61</td>
<td>7776</td>
</tr>
<tr>
<td>32768</td>
<td>91.03</td>
<td>324</td>
<td>10.0</td>
<td>324</td>
<td>95.60</td>
<td>42867</td>
<td>95.60</td>
<td>7996</td>
<td><strong>95.90</strong></td>
<td>12616</td>
</tr>
<tr>
<td>131072</td>
<td>84.75</td>
<td>81</td>
<td>10.0</td>
<td>81</td>
<td>95.58</td>
<td>42158</td>
<td>95.61</td>
<td>11927</td>
<td><strong>95.92</strong></td>
<td>11267</td>
</tr>
</tbody>
</table>

ABS/ABSA: Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Results – ImageNet

ImageNet consists of 1000 classes

- Total 1.2 million training images
- 50,000 testing images

### Results – ImageNet - AlexNet

**ResNet50 on Tiny ImageNet**

<table>
<thead>
<tr>
<th>BS</th>
<th>Acc.</th>
<th># Iters</th>
<th>Acc.</th>
<th># Iters</th>
<th>Acc.</th>
<th># Iters</th>
<th>Acc.</th>
<th># Iters</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>60.41</td>
<td>93750</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
<td>N.A.</td>
</tr>
<tr>
<td>256</td>
<td>58.24</td>
<td>46875</td>
<td>59.82</td>
<td>46875</td>
<td>60.31</td>
<td>70290</td>
<td>61.28</td>
<td>60684</td>
</tr>
<tr>
<td>512</td>
<td>57.48</td>
<td>23437</td>
<td>59.28</td>
<td>23437</td>
<td>59.94</td>
<td>58575</td>
<td>60.55</td>
<td>51078</td>
</tr>
<tr>
<td>1024</td>
<td>54.14</td>
<td>11718</td>
<td>59.62</td>
<td>11718</td>
<td>59.72</td>
<td>52717</td>
<td>60.72</td>
<td>19011</td>
</tr>
<tr>
<td>2048</td>
<td>50.89</td>
<td>5859</td>
<td>59.18</td>
<td>5859</td>
<td>59.82</td>
<td>50667</td>
<td>60.43</td>
<td>17313</td>
</tr>
<tr>
<td>4096</td>
<td>40.97</td>
<td>2929</td>
<td>58.26</td>
<td>2929</td>
<td>60.09</td>
<td>49935</td>
<td>61.14</td>
<td>22704</td>
</tr>
<tr>
<td>8192</td>
<td>25.01</td>
<td>1464</td>
<td>16.48</td>
<td>1464</td>
<td>60.00</td>
<td>49569</td>
<td>60.71</td>
<td>22334</td>
</tr>
<tr>
<td>16384</td>
<td>10.21</td>
<td>732</td>
<td>0.05</td>
<td>732</td>
<td>60.37</td>
<td>48995</td>
<td>60.71</td>
<td>20348</td>
</tr>
</tbody>
</table>


**ABSA/AWSA:** Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Results – ImageNet – ResNet18

° Baseline:
  
  - **450k** SGD iterations, **70.4%** validation accuracy

° ABSA:
  
  - **66k** SGD iterations, **70.2%** validation accuracy

° GG would have required **166k** SGD iterations
Hessian Based Adaptive Batch Size

Illustration of learning rate (a) and batch size (b) schedules of adaptive batch size as a function of training epochs based on C2 model on Cifar-10.

Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
What is Robust Optimization?

° Instead of minimizing for the average case, consider the worst case under “some metric”

Robust Optimization

° Consider a simple linear programming problem

$$\min_{x} \{c^T x : Ax \leq b\}$$

° Now assume input data (A,b,c) is uncertain

$$\min_{x} \sup_{(A,b,c) \in \mathcal{U}} \{c^T x : Ax \leq b\}$$

° Where \( \mathcal{U} \) is the uncertainty set

Robust Optimization and Regularization

There is an interesting connection between the solution to robust optimization and a properly regularized problem.

Robust solution to least squares under bounded uncertainty in $A$ is equivalent to lasso regularized one:

$$\min_x \max_{\|\Delta A\|_\infty, 2 \leq \rho} \| (A + \Delta A)x - b \|$$

$$\min_x \|Ax - b\| + \lambda \|x\|_1$$


Robust Optimization in NN

- Find the NN’s parameters using a min-max optimization, instead of just the min
  - Solving the max problem is computationally infeasible
  - A practical solution is to perform gradient ascent to iteratively solve the max problem and then gradient descent for the min part

\[
\min_{\theta} \tilde{J}(\theta, x, y) = \min_{\theta} \sum_{i=1}^{m} \max_{\tilde{x}_i \in U_i} J(\theta, \tilde{x}_i, y_i)
\]
Problem with SGD

° For convex problem, under proper conditions, we have

\[ R(W) = \mathbb{E}[f(W,X)] \leq \frac{1}{n} \sum_{i=0}^{n} f(W, x_i) + C1 \sqrt{Var(f(W,X))/n} + C2/n \]

° Empirical Risk Minimization only minimizes the first bias term \( \frac{1}{n} \sum_{i=0}^{n} f(W, x_i) \)

° Robust Optimization considers bias term \( \frac{1}{n} \sum_{i=0}^{n} f(W, x_i) \) and variance term \( \sqrt{Var(f(W,X))/n} \) together, back to the famous tradeoff, bias-variance tradeoff, in Machine Learning community. That can improve test performance.

H. Namkoong and J. C. Duchi. Variance regularization with convex objectives, NIPS’17
Robust Optimization as a Regularizer

Robust optimization regularizes the model away from “sharp minima”

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS’18 (arXiv:1802.08241)
we present the breakdown of one SGD update training time in terms of forward/backwards computation ($T_{\text{comp}}$), one step communication time ($T_{\text{comm}}$), one total Hessian spectrum computation (if any $T_{\text{Hess}}$), and the total training time. The results correspond to ResNet18 model on ImageNet.

<table>
<thead>
<tr>
<th>Method</th>
<th>$T_{\text{comp}}$</th>
<th>$T_{\text{comm}}$</th>
<th>$T_{\text{Hess}}$</th>
<th>Total Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>BL</td>
<td>2.2E-2</td>
<td>1.5E-2</td>
<td>0.</td>
<td>16666</td>
</tr>
<tr>
<td>GG</td>
<td>2.2E-2</td>
<td>1.5E-2</td>
<td>0.</td>
<td>6150 (2.71× faster)</td>
</tr>
<tr>
<td>ABS</td>
<td>2.2E-2</td>
<td>1.5E-2</td>
<td>1.15</td>
<td>2666 (6.25× faster)</td>
</tr>
<tr>
<td>ABSA</td>
<td>3.6E-2</td>
<td>1.5E-2</td>
<td>1.15</td>
<td>3467 (4.80× faster)</td>
</tr>
</tbody>
</table>
ABS Convergence Proof

° For a convex problem we have the following:

**Theorem 2.** Under Assumption 1, let assume at step $t$, the batch size used for parameter update is $b_t$, the step size is $b_t \eta_0$, where $\eta_0$ is fixed and satisfies,

$$0 < \eta_0 \leq \frac{1}{L_g(M_v + B_{\text{max}})},$$

(10)

where $B_{\text{max}}$ is the maximum batch size during training. Then, the expected optimality gap satisfies the following inequality,

$$\mathbb{E}[L(\theta_{t+1})] - L_* \leq \prod_{k=1}^{t} (1 - b_k \eta_0 c_s) (L(\theta_0) - L_* - \frac{\eta_0 L_g M}{2c_s}) + \frac{\eta_0 L_g M}{2c_s},$$

(11)

where $\theta_0$ is the initialization.

Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Approximate Hessian Computation

- Using block approximation to Hessian and analyzing last layer of a deep neural network seems to contain enough signal for ABSA

Z. Yao, A. Gholami, K. Keutzer, M. Mahoney, Large Batch Size Training of Neural Networks with Adversarial Training and Second-Order Information, (under review)
Problem with SGD

° SGD only reduces the bias term

$B_H = 50000$

Top 20 eigenvalues of the total Hessian w.r.t. model parameters is shown after the training is finished.

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS’18 (arXiv:1802.08241)
Problem with SGD

SGD only reduces the bias term

Top 20 eigenvalues of one sample Hessian w.r.t. model parameters is shown after the training is finished.

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS’18 (arXiv:1802.08241)
Robust Optimization

- Bring back the plots for the hessian and explain that now the results look better

Top 20 eigenvalues of the total Hessian w.r.t model parameters is shown after the training is finished.

Z. Yao, A. Gholami, Q. Lei, K. Keutzer, M. Mahoney. Hessian-based Analysis of Large Batch Training and Robustness to Adversaries, NIPS’18 (arXiv:1802.08241)
Summary

° Stochastic optimization using SGD often times does not lead to robust and it is very sensitive to hyper-parameters that are not optimal

° Robust optimization helps increase stability of the model to adversarial inputs

° Incorporating robust optimization can often time lead to solutions that have superior generalization performance than the baseline network trained with SGD and is more robust to adversarial perturbation

° Future work would include combining robust optimization with stochastic second order methods such as inexact sub-sampled Hessian
Thank You