Background

- Training with larger datasets generally leads to better performance for trained models.
- However, training with a large amount of data requires an enormous amount of energy, time, and resources.
- Existing methods provide heuristics for pruning examples during training pipeline [1, 2].

Problem Statement

1. Do existing pruning methods select the same set of images?
2. Can we combine different pruning methods?
3. What is the best pruning method to train a model on CIFAR-100 quickly?

Dataset

- CIFAR-100 (50,000 training & 10,000 validation images).

Methods

- **EL2N**: prune at initialization with EL2N scores from ensemble of early models
- **Selective-Backprop (SB)**: sample data per batch based on loss
- **Diet-SB**: combine EL2N and SB
- **Random-Backprop**: sample data per batch randomly

Experiments & Analysis

- **(a) Accuracy of various backprop methods.** We see Random-Backprop performs on par with Selective Backprop when dropping 50% data.
- **(b) Accuracy of EL2N, SB and Diet SB.** Diet SB seems to be limited by EL2N’s performance. However, Diet SB is able to decrease computation cost on top of EL2N.
- **(c) Overlap of examples pruned with EL2N at different fractions of the dataset on two different CNN architectures: GoogleNet and ResNet18.**
- **(d) Images that are least used in SB are often have low score in EL2N.** This graph shows that for the top 1% of images that are least used in SB, how many are also in the 1% lowest rated set in EL2N.
- **(e) Examples with the 20 lowest EL2N scores on CIFAR-100.**
- **(f) Examples with the lowest combined rank for EL2N and SB on CIFAR-100.**

- **EL2N** has stronger impact on accuracy since it removes data permanently. This reduces cost in both compute and memory.
- Backprop-based methods have little impact on accuracy while decreasing computational costs (note that our SB implementation has some extra overhead).
- Random-Backprop can drop up to 75% of CIFAR-100 without impacting accuracy, but at higher prune rates, SB performs better.
- Similar trends are observed in CIFAR-10 with ResNet20.

Runtime

<table>
<thead>
<tr>
<th>Method</th>
<th>Total Pruned (Pre-Training)</th>
<th>Runtime / Epoch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>7% (0%)</td>
<td>43.77s</td>
</tr>
<tr>
<td>SB</td>
<td>50% (0%)</td>
<td>43.37s</td>
</tr>
<tr>
<td>Random-Backprop</td>
<td>50% (0%)</td>
<td>27.19s</td>
</tr>
<tr>
<td>Random-Backprop</td>
<td>75% (0%)</td>
<td>18.83s</td>
</tr>
<tr>
<td>EL2N</td>
<td>25% (25%)</td>
<td>35.97s</td>
</tr>
<tr>
<td>Diet SB</td>
<td>25% (25%)</td>
<td>32.66s</td>
</tr>
<tr>
<td>Diet SB</td>
<td>62.5% (25%)</td>
<td>37.86s</td>
</tr>
</tbody>
</table>

Table 1. Typical runtime per epoch when training with different methods. All experiments are ran on AWS g4dn.xlarge on a Tesla T4 GPU. All percentages are with respect to the total training dataset.

Conclusions

- EL2N scoring can be transferable across different CNN architectures.
- SB and EL2N tend to pick similar sets of images from certain classes that seem to lack diversity.
- Diet Selective-Backprop reduces computation cost without sacrificing performance.
- Randomly choosing data to backprop per batch can speed up training without affecting model performance on CIFAR-100.

Future Work

- Explore combination of EL2N and Random-Backprop.
- Experiment with other datasets (ImageNet and SVHN) and other models (ResNet20, WideResNet, GoogleNet).
- Do multiple runs on backprop methods to account for randomness.

References


Advisor: Jonathan Frankle (Chief Scientist at MosaicML, PhD at MIT, new faculty at Harvard).