Knowledge Diversification in Ensembles of Identical Neural Networks

Bisheshy Das¹, Sumantra Dutta Roy¹
¹Indian Institute of Technology Delhi, India

1 Overview

- In many scenarios, multiple instances of identical neural networks are used to perform a certain task.
- If these networks have the same weights, then the total knowledge of the pool is no more than the knowledge of one single network.

Our goal in this paper:
- Enhance the joint knowledge of the networks residing in the pool.
- Make networks train jointly.
  - Pool aware training: Make each network know what other networks are learning.
  - Each network trains itself: NOT learn what others have already learned.
- Networks train adversarially to reach different local optima on the loss landscape.
- Network work together (as an ensemble) during inference.
  - Optionally allow predictions of each network to be assembled at single point that combines everything to form a better prediction (similar to a command and control center).

We propose:
- FDL - Feature Difference Loss functions.
  - Adversarial training routine.
    - Information sharing during training to enhance pool knowledge.
    - Separate stage for training flexibility.
    - Ability to train more than two networks simultaneously.
    - Stability reducing losses (Similarity loss).
  - No additional hyperparameters (other than the default ones - learning rate, momentum, weight decay, etc.)

2 FDL Ensemble architecture

- N identical base networks (without loss of generality, N=4 in the diagram).
- At any point of time, they share a common minibatch.

Training:
- Feature tensors are shared across all networks
- Prediction vectors of each network (P1, P2, P3, P4) is accumulated in an Ensemble Head network.
- A combined prediction vector is produced (P).

Losses invoked during training:
- FDL losses (to invoke adversarial behavior).
- Similarity loss (to stabilize training).
- Cross-entropy loss (default for classification).  
- Ensemble loss (same as cross-entropy loss, applied to the ensemble head network).

3 Feature Difference Losses and Similarity Loss

Let’s say we have network N, and N_r
- Feature difference loss at the r-th layer over networks N and N_r
- Feature tensors are pixel-like, and therefore we take mean-squared difference as a loss.

\[ \frac{f_{N_r}}{f_{N}} = \frac{1}{2} \sum_{w} \sum_{b} \left( f_{N}(b,c,h,w) - f_{N}(b,c,h,w) \right)^2 \]

- Similarity loss between networks N and N_r

\[ S_{N,N_r} = (L_1 - L_2)^2 \]

The overall optimization criteria:

\[ N_r(W) = \text{argmin}_{W} (-L_{N_r}(N) + k \cdot S_{N,N_r} + L_1(S_{N_1,y}) + L_2(S_{N_2,y})) \]

Problems with this approach:
- Unstable training (just like GAN).
- Introduces new (and sensitive) hyperparameters - \( k, k_1, k_2, \ldots \)
- Extending the criteria to many network scenario is complicated.

4 Phased training

Splitting the training routine into multiple phases helps.

Phase 0
Phase 1
Phase 2
Phase 3
Phase 4

The training pipeline of four networks ALL along with the ensemble head network trained with FDL loss. The red dotted line indicates the flow of gradients during backpropagation.

- Easily adaptable to many networks.
- Easy to check which phase is problematic.
- Hyperparameters pertaining to a particular stage can be modified accordingly.

5 Experiments

- Experiments on MNIST -
  - One layer network with M filters,
  - versus 2x FDL ensemble with M/2 filters.
- The FDL ensemble performs better for any value of M.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy (%)</th>
<th>CIFAR-10</th>
<th>CIFAR-100</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG-16 (1x) baseline</td>
<td>93.66</td>
<td>74.81</td>
<td></td>
</tr>
<tr>
<td>VGG-16 BIE (2x)</td>
<td>91.7</td>
<td>76.95</td>
<td></td>
</tr>
<tr>
<td>VGG-16 SBE (1x)</td>
<td>94.05</td>
<td>75.31</td>
<td></td>
</tr>
<tr>
<td>VGG-16 FOG (11x)</td>
<td>94.34</td>
<td>76.64</td>
<td></td>
</tr>
<tr>
<td>VGG-16 All (11x)</td>
<td>93.91</td>
<td>72.16</td>
<td></td>
</tr>
<tr>
<td>VGG-16 FDL (2x) (ours)</td>
<td>94.93</td>
<td>77.82</td>
<td></td>
</tr>
</tbody>
</table>

Comparisons of ensemble methods in image classification task, performed on CIFAR-10 and CIFAR-100, with VGG-10.

6 Many network FDL ensembles

- 2x, 3x and 4x FDL ensembles show strong response to the FDL loss function.
- Phased training routine ensure stability across all many network experiments.
- All hyperparameters and training details are listed in supplementary section.

7 Conclusion

- FDL - A strong method of optimizing ensemble performance.
- Adversarial training to achieve diversity in feature representation among base networks of an ensemble.
- Custom training routine that ensures stability and ease of training ensembles.