Knowledge Diversification in Ensembles of Identical Neural Networks

Indian Institute of Technology Delhi

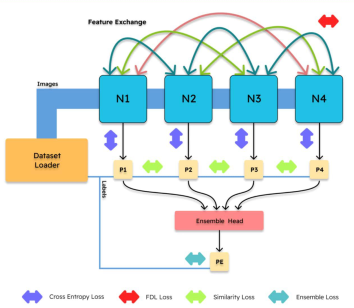
Bishshoy Das[1], Sumantra Dutta Roy[1]
[1]Indian Institute of Technology Delhi, India

BMVC 2022

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 to **NOT** learn what others have already learnt.
  - Networks train adversarially to reach different local optimas 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.
    - Separable stages for training flexibility.
    - Ability to train more than two networks simultaneously.
    - Stability inducing losses (Similarity loss).
    - **No additional hyperparamters!** (other than the default ones - learning rate, momentum, weight decay, etc.)

2 FDL Ensemble architecture



The architecture of four identical neural networks trained with FDL. All networks share a common minibatch. The double sided arrows represent the different loss functions and their position indicates the location where they are invoked.

- 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 (**PE**).
- 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_1 and N_2.
- Feature difference loss at the i'th layer over networks N_1 and N_2.
- Feature tensors are pixel-like, and therefore we take mean-squared difference as a loss.

$$L_i^{N_1,N_2} = \frac{1}{BCHW} \sum_{b=0}^{B-1} \sum_{c=0}^{C-1} \sum_{h=0}^{H-1} \sum_{w=0}^{W-1} \left(F_i^{N_1}(b,c,h,w) - F_i^{N_2}(b,c,h,w)\right)^2$$

$$L_{FDL}^{N_1,N_2} = \frac{1}{D} \sum_{d=0}^{D-1} L_d^{N_1,N_2}$$

- Similarity loss between networks N_1 and N_2.

$$S^{N_1,N_2} = (L_1 - L_2)^2$$

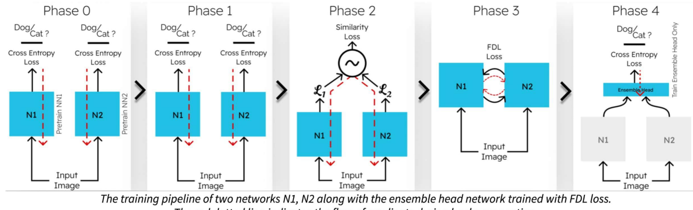
- The overall optimization criteria:

$$N_E^*(W) = \underset{W}{\operatorname{argmin}}\left(-L_{FDL}^{N_1,N_2} + k\, S^{N_1,N_2} + k_1\, L_X(\hat{y}_{N_1}, y) + k_2\, L_X(\hat{y}_{N_2}, y)\right)$$

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

4 Phased training

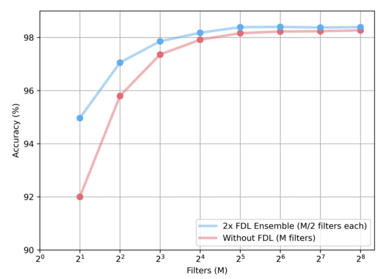
- Splitting the training routine into multiple phases helps.



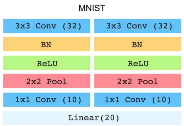
The training pipeline of two networks N1, N2 along with the ensemble head network trained with FDL loss. The red dotted line indicates the flow of gradients during backpropagation.

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

5 Experiments



Number of Filters (M) vs Accuracy plot of a one layer ConvNet. Red line indicates single network of M filters. Blue line indicates FDL ensemble of 2x networks (M/2 filters each).



Ensemble architecture for MNIST 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**.

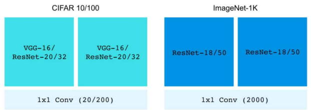
Method	Accuracy (%)	
	CIFAR-10	CIFAR-100
VGG-16 (1x) baseline	93.66	74.61
VGG-16 RIE (2x)	93.7	76.95
VGG-16 SSE [18]	94.05	75.31
VGG-16 FGE [11]	94.34	76.46
VGG-16 AE Full [44]	93.93	72.16
VGG-16 FDL (2x) [ours]	**94.93**	**77.02**

Comparisons of ensemble methods in image classification task, performed on CIFAR-10 and CIFAR-100, with VGG-16.

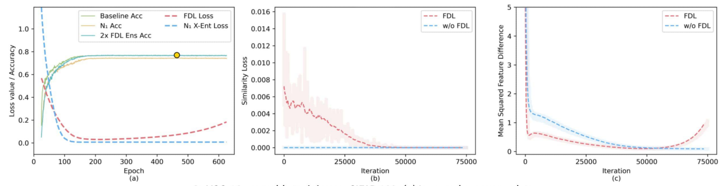
Method	Accuracy (%)
	ImageNet-1K
ResNet-50 (1x) baseline	76.38
ResNet-50 RIE (2x)	76.96
ResNet-50 SSE [18]	76.67
ResNet-50 FGE [11]	76.69
ResNet-50 FDL (2x) [ours]	**77.06**

Comparisons of ensemble methods in image classification task, performed on ImageNet-1K, with ResNet-50.

- FDL ensembles outperform other competing methods.
- FDL forces the base networks to find diverse feature representations.
  - We shall explain this in detail in the next slide.



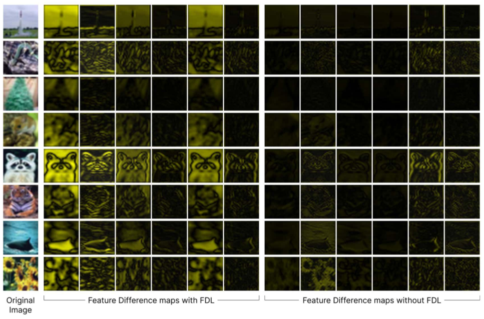
Ensemble architecture for CIFAR and ImageNet experiments.



2x VGG-16 ensemble training on CIFAR-100. (a) Loss and accuracy plots. (b) Similarity loss plots. (c) Plot of Mean Squared Feature Differences during the training.

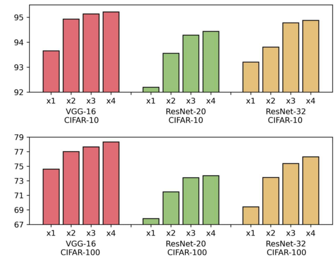
Key highlights:

- **Plot (a)**: The FDL loss decreases initially (red line). After 200 epochs it starts to increase, even though the cross-entropy loss of N_1 keeps on decreasing (blue line).
- **Plot (b)**: With FDL, the networks' states initially moves away from each other jumping across different local optimas as it explores the entire loss landscape, hence the fluctuations in the similarity loss. In case of without-FDL, the similarity loss is very close to zero right from the beginning and till the end of training.
- **Plot (c)**: Mean square feature differences steadily decreases to zero if the networks are not trained with it (blue line).



- Feature differences with and without FDL loss.
- With FDL - Feature differences are more prominent.
- Without FDL - The chances of the base networks arriving at similar feature sets increase drastically.

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.