Report

Table of content

Project Title :

Medical Image Analysis

Mentor -

Dr. Deepak Mishra

Team Members-

Riyanshu Jain (B20AI060)

Divyam Patel (B20EE082)

Dhruv Viradiya (B20CS079)

Abstract

Data Augmentation has proved to be an effective method for classification of images. It significantly increases the diversity of data available for training our models, without actually collecting new data samples. In this project, we have looked on recently proposed augmentation techniques popularly known as **MixUp, CutMix, AugMix** and **CutOut**. We found that models trained using augmentation techniques are calibrated more better. Ensemble models consisting of deep Convolutional Neural Networks (CNN) have shown significant improvements in model generalization but at the cost of large computation and memory requirements. We try to distillate all of the augmentation techniques in different teacher models and combine the knowledge of all the well calibrated teacher models into one shallow well calibrated student model.

Design Problem Formulation

Nowadays, ML algorithms are being used in several applications in real life and out of them Large DNN's have enabled breakthroughs in various fields. But one of the major challenges using such DNN is the issue of uncertainty in predictions done by them. The DNN's must not only be accurate but also their predictive scores should be indicative of the actual likelihood of correctness. They should quantify how likely they are to get the wrong answers. Although these DNNs show significant improvement in generalizing models, they take a larger cost of computation to fulfill requirements.

Proposed Solution

Calibration

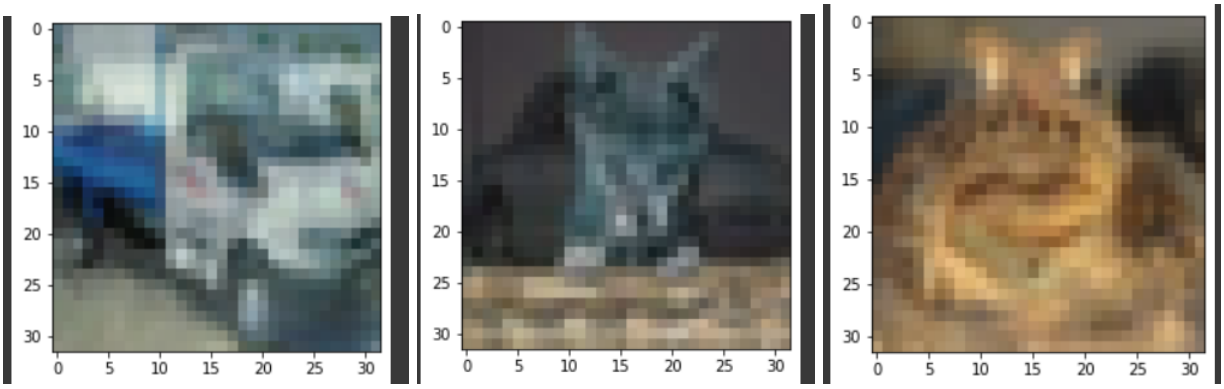
Calibration is comparison of the actual output and the expected output given by a system. In calibration we try to improve our model such that the distribution and behavior of the probability predicted is similar to the distribution and behavior of probability observed in training data. In our experiments we have seen 4 major data augmentation techniques to calibrate the model -

1. **MixUp -** In this augmentation technique, samples are generated during training by convexly combining random pairs of images and their associated labels. Hence, we can say that the classifier is trained not only on the training data, but also in the vicinity of each training sample. The mathematics involved behind this technique is as follows -

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j$$

where x_i and x_j are two randomly sampled input points, and y_i and y_j are their associated one-hot encoded labels.



2. **CutMix -** CutMix is an augmentation strategy incorporating region-level replacement. For a pair of images, patches from one image are randomly cut and pasted onto the other image along with the ground truth labels being mixed together proportionally to the area of patches. CutMix replaces the removed

regions with a patch from another image, which utilizes the fact that there is no uninformative pixel during training, making it more efficient and effective. The mathematics is as follows -

$$\tilde{x} = Mx_i + (1 - M)x_j$$

$$\tilde{y} = \lambda y_i + (1 - \lambda)y_j$$

where M is the binary mask which indicates the cutout and the fill-in regions from the two randomly drawn images and λ (in [0, 1]) is drawn from a Beta(α, α) distribution

The coordinates of bounding boxes are:

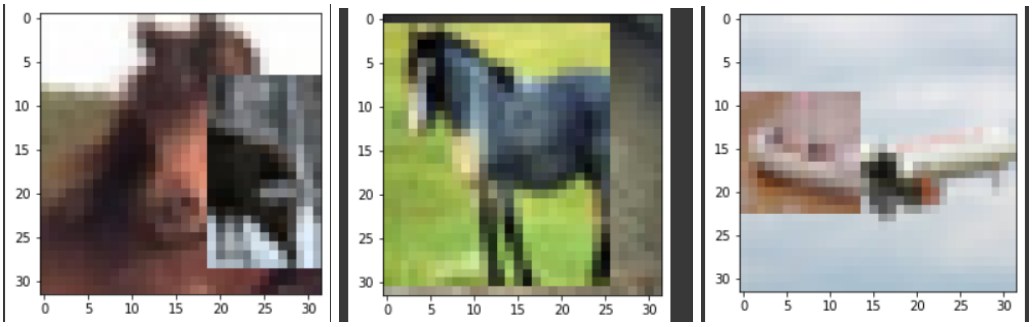
$$B = (r_x, r_y, r_w, r_h)$$

which indicates the cutout and fill-in regions in case of the images.

The bounding box sampling is represented by:

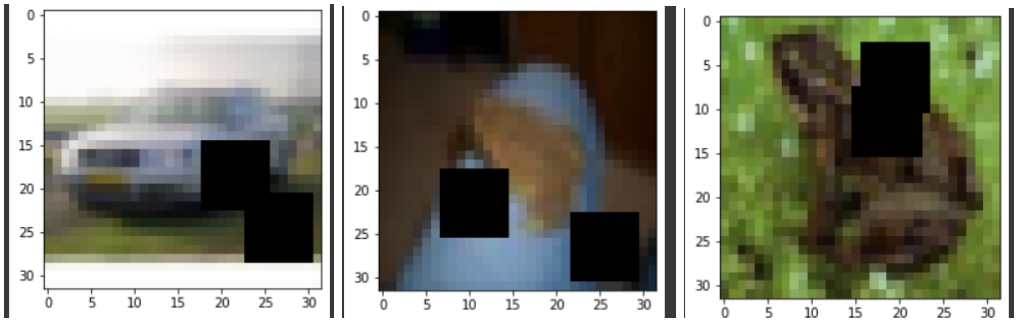
$$r_x \sim U(0, W), r_w = W(1-\lambda)^{(½)}$$

$$r_y \sim U(0, H), r_h = H(1-\lambda)^{(½)}$$



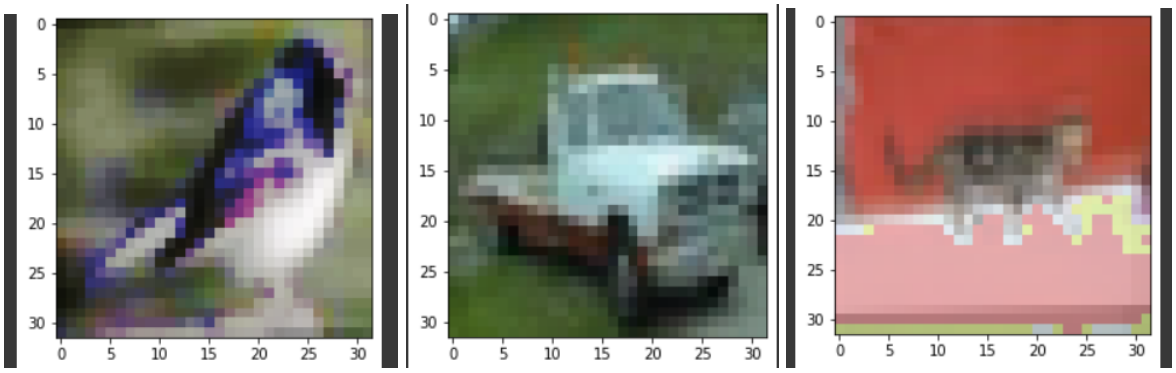
Cutmix with number of cuts = 1 and beta = 1

3. **CutOut -** Cutout augmentation is a kind of regional dropout strategy in which a random patch from an image is zeroed out (replaced with black pixels). Cutout samples suffer from the decrease in information.



Cutout with number of holes = 2 and size = 8 x 8

4. **AugMix -** AugMix is a data processing technique which mixes randomly generated augmentations, improves model robustness and slots easily existing training pipelines. Augmix performs data mixing using the input image itself. It transforms (translate, shear, rotate and etc) the input image and mixes it with the original image. AugMix prevents degradation of images while maintaining diversity as a result of mixing the results of augmentation techniques in a convex combination. At a high level, it is characterized by its utilization of simple augmentation operations in concert with a consistency loss.



Knowledge Distillation

Knowledge distillation is the process of transferring knowledge from a large model to a smaller one. While large models (such as very deep neural networks or ensembles of many models) have higher knowledge capacity than small models, this capacity might not be fully utilized. It can be computationally just as expensive to evaluate a model even if it utilizes little of its knowledge capacity. Knowledge distillation transfers knowledge from a large model to a smaller model without loss of validity. As smaller models are less expensive to evaluate, they can be deployed on less powerful hardware (such as a mobile device).

We present an Ensemble Knowledge Distillation (EKD) framework which improves classification performance and model generalization of small and compact networks by distilling knowledge from multiple teacher networks into a compact student network using an ensemble architecture.

Implementations/ Experiments we have done so far

CALIBRATION -

The following hyper parameters were set

- epochs - 500,
- batch_size = 128
- learning_rate = 0.05
- momentum = 0.9
- learning_rate_milestones = [150, 180, 210]
- learning_gamma = 0.1
- NUM_BINS = 100
- weight_decay = 5e-4

- **Model - Wide Resnet 40-2, Dataset - CIFAR 100 (SGD optimiser and multistep lr scheduler)**

Technique	Best_Acc	Epoch no.	ECE (corresponding to that epoch)	OE (corresponding to that epoch)
None	75.63	358	0.1104	0.0893
Cutmix	78.54	498	0.0315	0.0186
Cutout	77.31	370	0.0775	0.0591
Mixup	.76.98	287	0.0351	0.0027
Augmix	76.66	204	0.0437	0.0300
Calibration ensemble Ratio - 1:1:1:1:1	81.74	-	0.0709	0.0003
Ratio - 5:1:5:1:1	80.26	-	0.0306	0.0044

5 - None
1 - Cutmix
5 - Cutout
1 - Mixup
1 - Augmix
defines the ratio in which the techniques are combined in the calibration ensemble, to lower the ece by increasing the confidence of the models, as we can notice from the normal ensemble, the ece is higher because the model proves to be underconfident, which is good in terms of medical diagnosis.

- **Model - ShuffleNet V1, Dataset - CIFAR 100, Epochs - 500**

  Here, batch_size = 64 and learning_rate = 0.01, rest all parameters are same as above

Augmentation Technique	Best Accuracy at Epoch	Accuracy	ECE	OE
None	186	71.45	0.0890	0.0646

Small Scale Experiments :

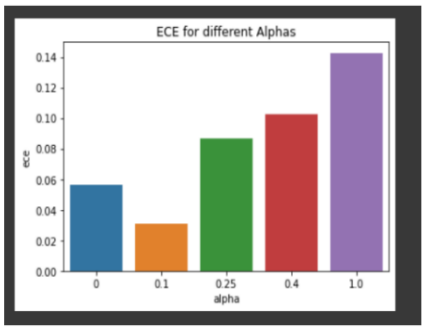
1. **Implementation of MixUp on CIFAR-100 Dataset using ResNet18 for various alphas**

Hyperparameters :

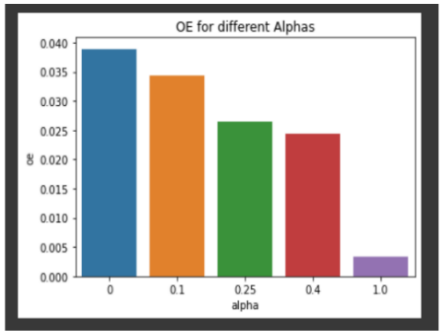
Number of epochs = 20, Initial learning rate = 0.1 , Batch size = 128

Value of alpha in mixup	Accuracy (on 20th epoch)
0 (no mixup)	0.6957
0.1	0.6954
0.25	0.6894
0.4	0.6904
1	0.698

Expected Calibration Error Plot :



Overconfidence Error plot -



KNOWLEDGE DISTILLATION -

The following hyper parameters were set

- epochs - 500
- batch_size = 64
- learning_rate = 0.01
- momentum = 0.9
- learning_rate_milestones = [150, 180, 210]
- learning_gamma = 0.1
- NUM_BINS = 100
- weight_decay = 5e-4

- **Teacher Model - Wide Resnet 40-2, Student Model - Shufflenet V1**
  Dataset - CIFAR 100
  (SGD optimiser and multistep lr scheduler)

Here, **alpha** is the weightage we have given to **KL div loss** (more dependent on learning with Teacher model) and, **gamma** is weightage we have given to **Cross Entropy loss** (less dependent on learning with itself)

T, the temperature which we set in KD, is applied to logits to affect the final probabilities from the softmax.

Technique	Alpha = 0.8, gamma = 0.2	Best_Acc	ECE	OE
Vanilla KD	**T = 4**	0.7471	0.1435	0.1198
	T = 50	0.7541	0.1200	0.0992
KD Ensemble (Add loss)	**T = 4**	0.7642	0.0933	0.0738
	T = 50	0.7728	0.06101	0.04604
KD Ensemble (Avg loss)	**T = 4**	0.7616	0.0881	0.0701
	T = 50	0.7571	0.0752	0.0563
KD Ensemble (Avg softmax)	**T = 50**	0.7555	0.0707	0.0522

Ensemble of 5:1:5:1:1 in KD models -
(Model on eval mode with the weights of individually trained KD models)
ECE - 0.0282
OE - 0.0146
Accuracy - 77.47

Results, Analysis and Conclusion

- From the calibration, we can see that we have got many calibrated models after applying the respective calibration techniques, when compared to the base model in terms of accuracy, ECE (Expected Calibration Error), OE (Overconfidence Error).
- From the KD results seen above, we can notice that the student model, i.e. Shufflenet V1 alone is able to reach the accuracy to 71.45% along with Expected Calibration Error = 0.0890.
- But when distilled the knowledge from calibrated Teacher model (wide resnets), the student model Shufflenet V1 reached good scores. The accuracy jumped up to 76 and 77 % with lower Expected Calibration Error.
- Also the ensemble of individually trained KD models is much better than the ensemble of KD applied on calibrated models in terms of ECE and OE.
- Hence, we can improve classification performance and model generalization of small and compact networks by distilling knowledge from multiple teacher networks into a compact student network using an ensemble architecture.

----------END of REPORT----------