Self Supervised Learning

Tego Chang, Cindy Chiu, Nansu Wang

Duke PRATT SCHOOL of ENGINEERING

Background & Overall Methodology

Over the past decade, deep convolutional neural networks have transformed the field of computer vision thanks to their ability to identify image features and classify images. However, in order to successfully learn image features, the networks usually require a vast amount of manually labeled data. Labeling huge sets of data is expensive and impractical to scale. Therefore, it is crucial to develop a technology that can learn with minimal labeling on the data, which is often referred as self-supervised learning method. Currently, there are two research in self-supervised learning field, SimCLR and Rotnet. They both leverage data augmentation on images to generate useful feature representations.

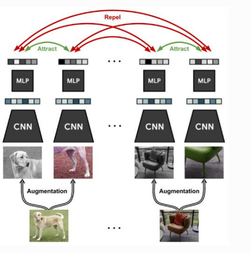
In this project:

- We implemented the data augmentation for data loader
- We implemented both SimCLR and RotNet structure
- We implemented the loss function and the training process for both models
- After implementing the network structure, we fine tuned model parameters and saved the image representation from the best performing model.
- We also implemented a supervised counterpart for both SimCLR and RotNet.
- Finally, we trained a simple linear classification on the feature representations and evaluated the classification accuracy on the testing data.

SimCLR

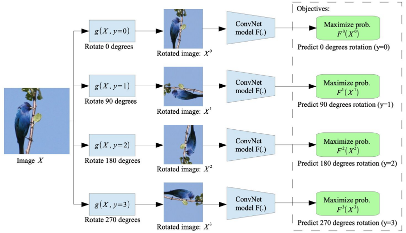
SimCLR introduces a simple framework for contrastive learning of visual representations. It learns representations by maximizing agreement between differently augmented views of the same image, including color distortion and cropping, via a contrastive loss in the latent space.

Specifically, we used a modified version of ResNet-18 as the backbone model, a MLP with 1 hidden layer to project into a 128 dimensional latent feature space. In each batch, for each image, 2 augmentations are implemented. Contrastive loss is calculated within the batch and is used to guide the back-propagation process. We trained 200 epochs with 0.0003 as the base learning rate with cosine scheduler using Adam optimizer.

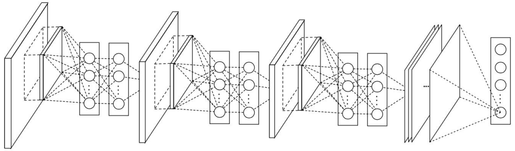


Rotnet

The idea of Rotnet (framework shown as below) is to train a deep convolutional neural nets to recognize the 2d rotations applied to its input images. The intuition behind it is that unless a ConvNet has already learnt the semantic parts in a image, it cannot identify the rotating transformation applied to it. Rotnet has been demonstrated to achieve state-of-the-art performance among many other unsupervised methods. It also approaches the performance of many supervised methods.



The implementation of Rotnet applies a network architecture called Network-In-Network (NIN, a 3-block network architecture shown as below) to learn the rotations applied to each image. NIN stacks 3 convolutional layers in one block and acts as a strong local modeling. The characteristic makes it possible to be connected by a global average pooling layer, instead of the traditional black-box fully connected layers. Thus, the entire network can not only correlate feature maps at the end of the convolutional layers with the final categorical level information but also avoid overfitting.



Implementation of the unsupervised framework:

- We applied 4 rotations to each sample image and built a Rotnet composed of 4 NIN blocks to learn the rotation (the first supervised learning task).
- We extracted the first 2 NIN blocks from the overall four blocks and treated its output as our feature representation.
- We connected the first 2NIN blocks with a linear classifier and trained it to perform the image classification task in CIFAR10 (the second supervised learning task).

Results / Conclusion

SimCLR is an effective self-learning framework. Our model achieves 54.16% accuracy on the testset of CIFAR-10.

However, the performance was not optimal due to some simplifications we adopted.

- Heavy hyper-parameter tuning was not conducted due to the computational limitation.
- Only 200 epochs were trained for which is not optimal based on the paper
- Adam optimizer was used instead of LARS

Rotnet is also a self-learning framework that extracts its feature representations through transforming its input samples into 4 rotations, 0, 90, 180, and 270 degrees and then making corresponding predictions. During its supervised process on learning the rotations, we trained 90 epochs and it converged and achieved best validation accuracy at the 30th epoch with an accuracy of 81.13%. After that, we connected the extracted feature representations from the first 2 blocks of NIN. We processed the second supervised process on predicting the correct labels in CIFAR10 and achieved an accuracy of 63.47%.

We do acknowledge that the accuracy from linear classifier does not approach the rotnet performance in the paper, 89.06%. However, we believe that is mostly because the feature representations was connected with a non-linear classifier in the paper, instead of the linear classifier we used.

Evaluation Results (Test Accuracy %)

	RotNet	SimCLR
Self-supervised	63.47%	54.16%
Supervised	82.83%	80.28%