Ridge-i Assignment

Design a network that combines supervised and unsupervised architectures in one model to achieve a classification task

January 17 – February 01 , 2022

Report
by
Atul Sharma

Ridge-i assignment

Supervised and unsupervised training

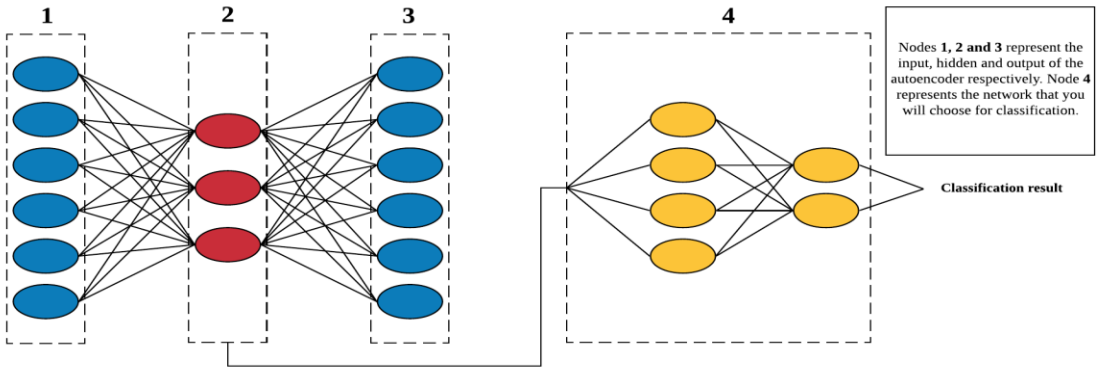
Welcome to Ridge-I assignment. This task aims to evaluate your technical skills in managing a neural network design and dataflow for a classification task with a strictly imbalanced training-data situation. Code must be readable, extendable, follows object oriented standards, fair performance, mobile, and converging. Aspects are descendingly ordered with respect to their importance.

Introduction:

We want you to design a network that combines supervised and unsupervised architectures in one model to achieve a classification task.

Architecture:

The model must start with autoencoder(s) (stacking autoencoder is ok) that is connected through its hidden layer to another network of your choice, as shown in the figure below:



This autoencoder takes an input (image) at node 1 and reconstructs it at its output at node 3. It creates valuable features at its hidden layers (node 2) during this process. it is hypothesized that if node 2 is used as input for the CNN (node 4) then the classification can be improved.

Data:

We want to use this model to classify Cifar-10 dataset by using at max 2500 training images for each of the bird, deer, and truck classes while using 5000 for the other classes. Delivering a working model with less than 2500 training images for the three referenced classes shows your skills. The model should be evaluated by the test set of 10000 images (1000 for each class).

Training:

You are free to propose the best number of epochs, mini batch size, optimizer and regularizing technique and data augmentation that you think is suitable for this scenario. It is recommended to try a variety of training schemes (e.g. end to end and separated) for sake comparison. Be creative.

Deliverables:

We wish to receive your code in any convenient sharing from (GitHub, Collab, Jupiter or docker) plus a detailed report analyzing the model, its contribution and a discussion of your findings.

Background of Image Classification

Image classification is the task of categorizing and assigning labels to groups of pixels or vectors within an image dependent on rules. Image classification techniques are mainly divided into two categories: Supervised and unsupervised image classification techniques. Unsupervised classification technique is a fully automated method that does not leverage training data. This means machine learning algorithms are used to analyze and cluster unlabeled datasets by discovering hidden patterns or data groups without the need for human intervention. Two popular algorithms used for unsupervised image classification are 'K-mean' and 'ISODATA' (Iterative Self-Organizing Data Analysis Technique). Supervised image classification methods use previously classified reference samples (the ground truth) in order to train the classifier and subsequently classify new, unknown data.

Convolutional Neural Network (CNN)

A CNN is a framework developed using machine learning concepts. CNNs are able to learn and train from data on their own without the need for human intervention.

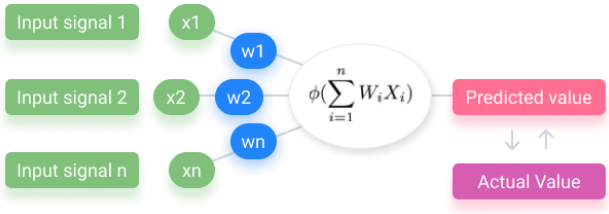


Figure 2. Concept of neural Network (credits: viso.ai)

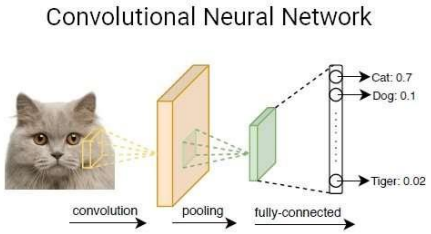


Figure 1. Concept of Convolutional neural Network (CNN) (credits: viso.ai)

The CIFAR-10 dataset

The CIFAR-10 dataset consists of 60000 32x32 color images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain exactly 5000 images from each class. Classes in the dataset is shown in figure 3.

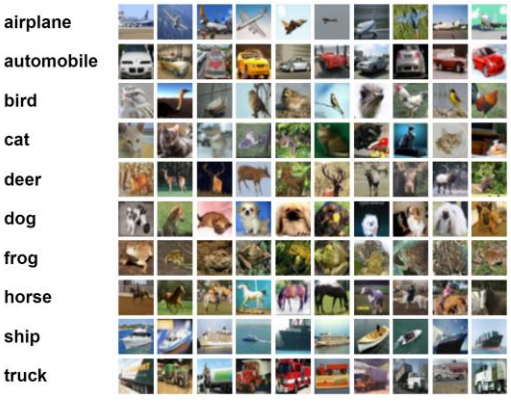


Figure 3. Classes in the dataset of CIFAR (credit:cs.toronto.edu)

Concept

The CIFAR-10 dataset consists of 60000 32x32 color images in 10 classes, with 6000 images per class (5000 for training and 1000 for testing purpose). I implemented autoencoders as feature extractor and using those features as inputs to a classifier model to predict the different classes of images in the CIFAR-10 dataset. For training 3 classes: bird, deer and truck only 50% of the images (2500 images per class) were used to make the training data imbalance.

System and software environment used:

System : Intel i9-10900X CPU , 32 Gb RAM, Ubuntu-18.04, GPU:RTX 3090.

Software: Pytorch1.10.2, Jupyterlab3.2.1, matplotlib3.5.0, scikit-learn1.0.2, seaborn0.11.2, pandas1.3.5.

The code is shared on github : https://github.com/atul-sh/supervised-unsupervised-learning-using-pytorch

Result:

1) Epoch =150 batch_size= 512, learningrate= 0.001

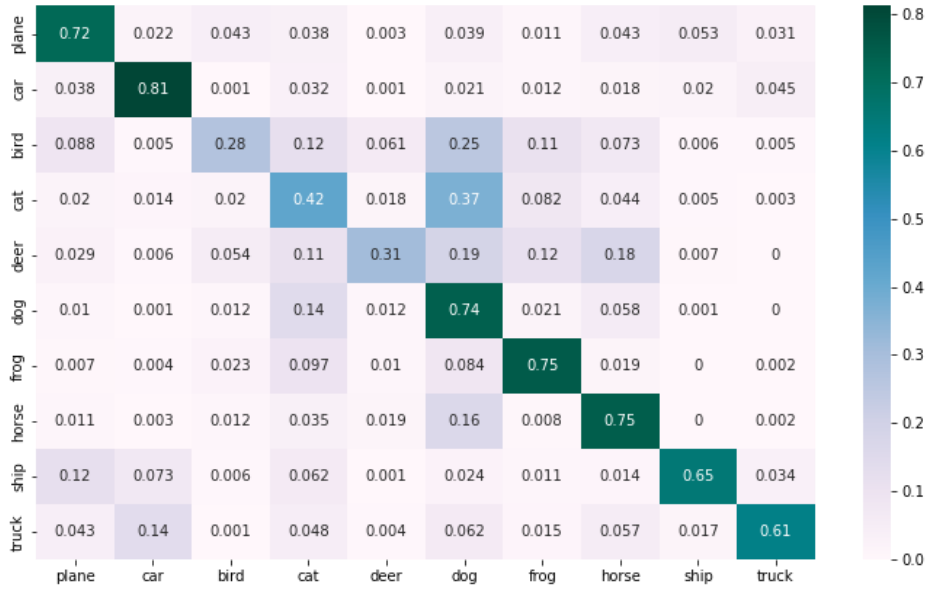


Figure 4 Confusion Matrix

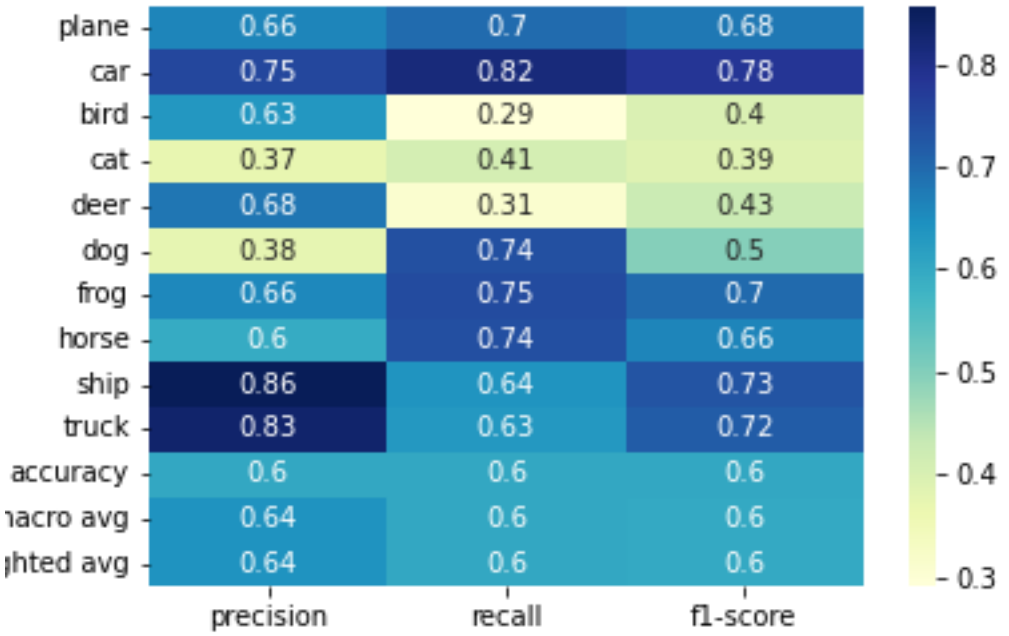


Figure 5 Classification report

2) Epoch =250 batch_size= 512, learningrate= 0.001

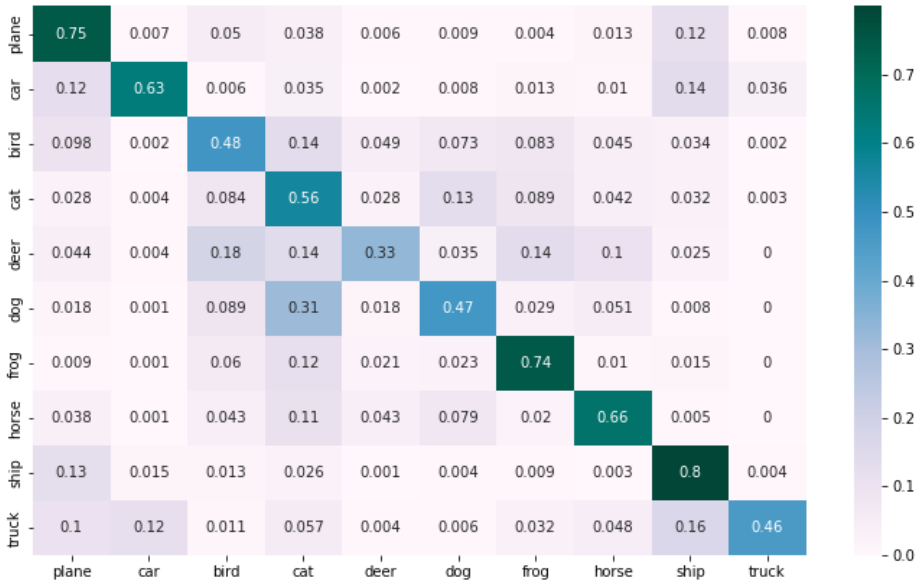


Figure 6 Confusion Matrix

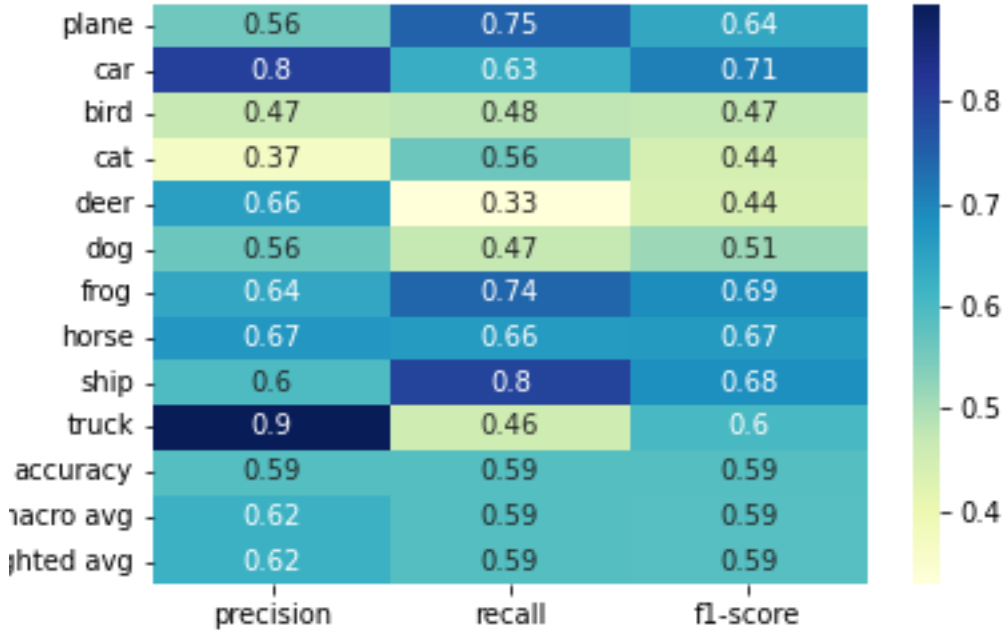


Figure 7 Classification report

3) Epoch =100 batch_size= 512, learningrate= 0.01

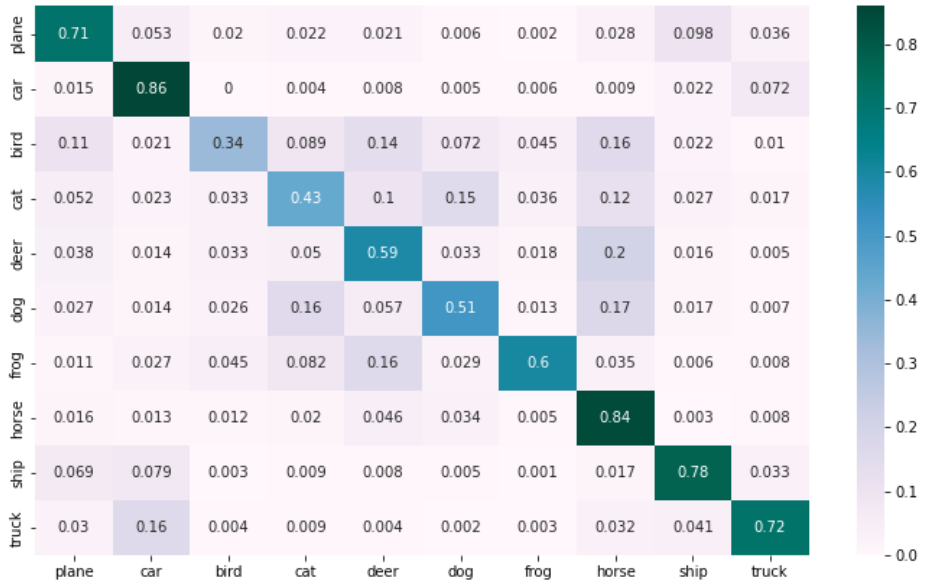


Figure 8 Confusion Matrix

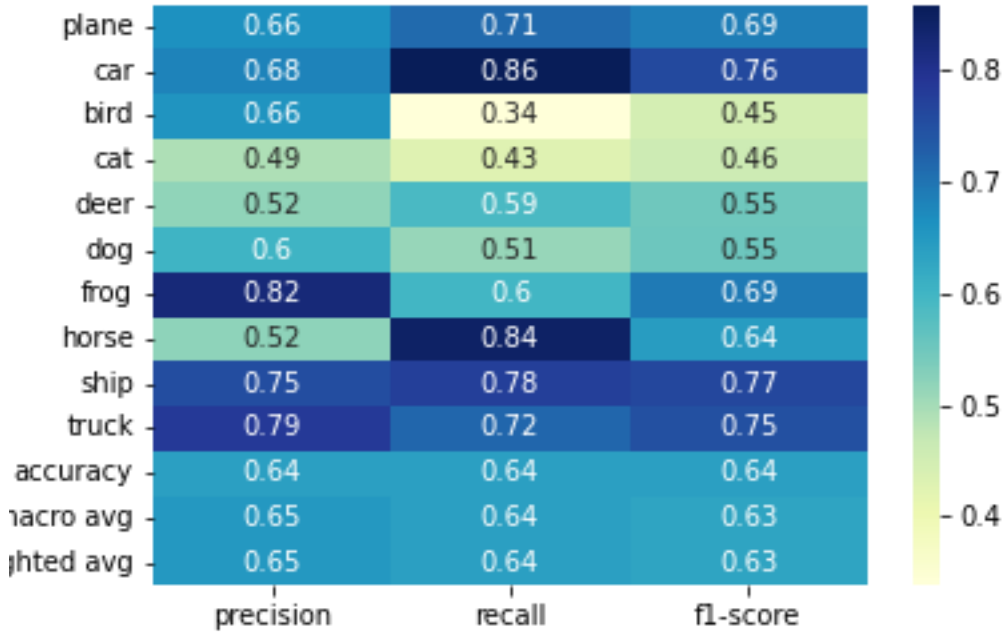


Figure 9 Classification Report

Conclusion:

The training accuracy is below 30% when trained for epoch <50.

Maximum accuracy obtained on the above model is about 63% for epoch=100.

Batch size is kept 512 throughout training and learning rate is varied but it also does not have a deep impact on the model at values 0.01 and 0.001.