Building a Convolutional Neural Network Model to Classify MNIST Handwritten Dataset with Comparing the Accuracy of Different Optimizers (Adam, SGD, RMSprop)

Dewan Fahim Uddin

18-38049-2

Abstract

Convolutional neural network is very popular in processing and recognising image for its amazing image processing algorithm. This algorithm is used in this project to classify MNIST handwritten dataset 3 different optimisers. All of them managed to get a sufficient accuracy of over 99% in both train and test images. With the same model, Adam, SGD and RMSprop comes up with 99.06%, 99.12% and 99.15% accuracy respectively. It seems that the model is quite good and CNN proves that it is the best algorithm to classify MNIST handwritten dataset. Though with different dataset this model may not produce results as good as this experiment.

Introduction

Machine learning, neural network, deep learning are invented to implement artificial intelligence in machines. In this article we are going to implement a CNN model to classify MNIST handwritten dataset.

Neural Network

Neural networks are a machine learning model which were initially introduced in 1940s. As it reflects the behaviour of human brain, it is used to recognise patterns and solving the problems in AI, machine learning and deep learning models. Neural networks are composed of 3 layers - input layer, hidden layer and output layer. A neural network model take one or more inputs and return one or more outputs.

Convolutional Neural Network (CNN)

A convolutional neural network is a deep learning algorithm which is also a type of artificial neural network. It is mainly used in image processing and image recognition. It is a powerful image processing algorithm as it is specially designed to process pixel data. It can detect lines, colours, edges and other visual elements of an image by using filters. With more filters, it can detect more elements. A filter is a square shaped object which scans the image.

Optimizers

Optimizers are methods which are used to change weights, learning rates and some other attributes neural network to reduce the loss. As they minimise the losses or errors, accuracy goes up automatically. To get results faster, optimisers help significantly. In this article, only three TensorFlow keras optimisers will be discussed. So, first we should know what is TensorFlow. According to TensorFlow itself, it is an end-to-end open source platform for machine learning [1]. In other words, it is an artificial intelligence library to build AI models. Inside TensorFlow, there is a high level API named keras. Keras has some different types of optimisers. Adam is one of them. It stands for adaptive moment estimation. It calculates new gradients using the past gradients. Another optimiser is SGD or stochastic gradient descent optimisation. RMSprop is another optimiser developed by Geoffrey Hinton which is an exclusive version of Adagard optimiser. It is also a gradient based optimisation technique which maintains a lots of the square of gradients and then divides the gradient by the root of this average.

MNIST Handwritten Digit Classification Dataset

MNIST stands for the Modified National Institute of Standards and Technology. It is a dataset of 60,000 pictures of handwritten numeric values. Each is a grayscale image of 28 by 28 pixels and consists of a single digit between 0 and 9. It also has a dataset of 10,000 test images. It is a widely used dataset. CNN algorithm can achieve a classification accuracy of over 99% on this dataset.

Results

In this experiment 3 different optimisers show 3 different outputs. The outputs are given below:

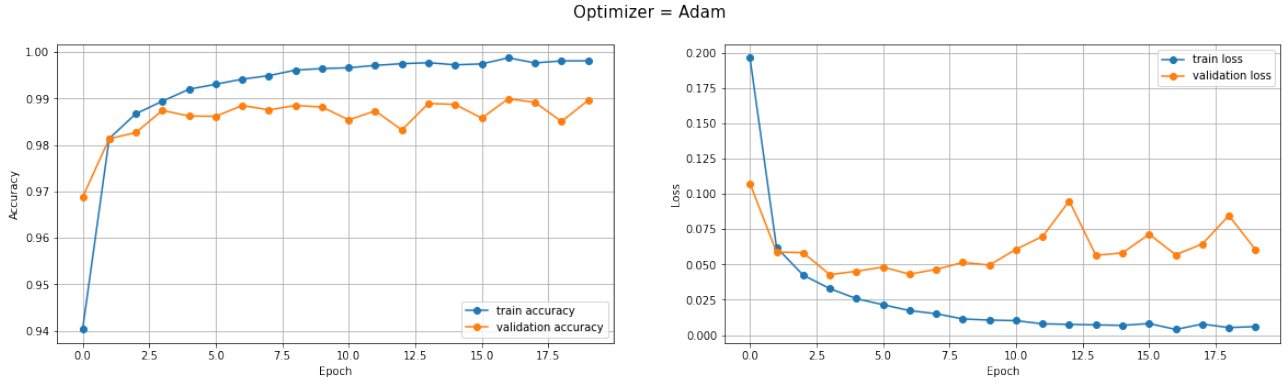

Fig-1: train and test accuracy obtained using Adam classifier

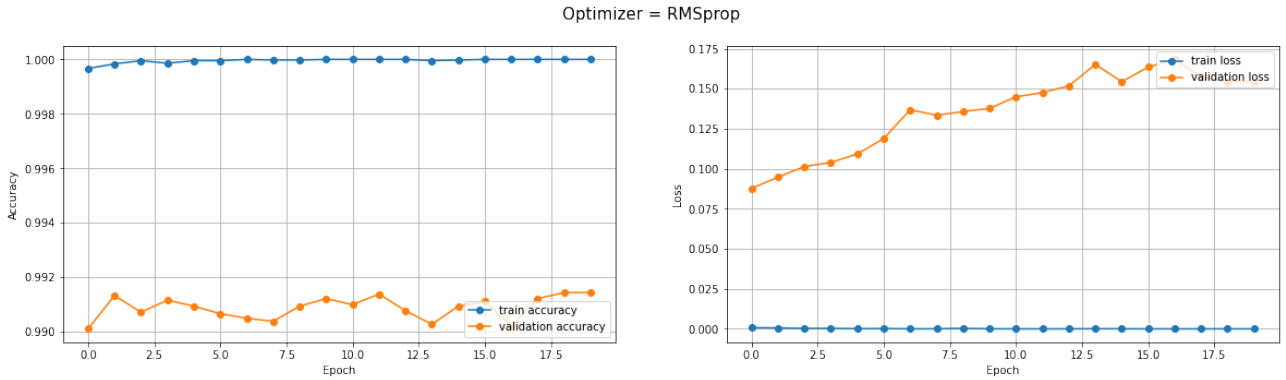

Fig-2: train and test accuracy obtained using SGD classifier

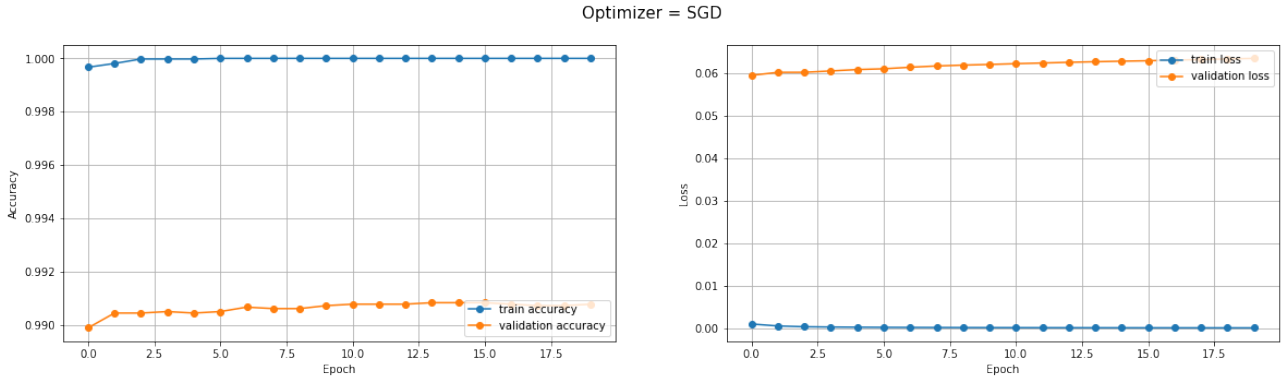

Fig-3: train and test accuracy obtained using RMSprop classifier

Also, we get an accuracy of Adam, SGD and RMSprop optimisers in test dataset of 99.06%, 99.12% and 99.15% respectively.

Discussion

The figures above shown that, both Adam, SGD and RMSprop individually manages a good accuracy of over 99% and there is hardly any difference. Though is terms of stability, SGD looks efficient. On the other hand, among them RMSprop manages to get the higher accuracy of 99.15% in the test dataset. As expected, the train accuracy is always higher than the validation accuracy. Both SGD and RMSprop got almost no loss in data when train. There are a lots of ups and downs in validation accuracy in both Adam and RMSprop optimiser. Though the model performs well as well as the optimisers used. Though this experiment was run multiple times with changing the model. And it seems that the system already memorise the datasets and giving a good result. Testing with different dataset may not produce the same results like this.

References

[1] https://www.tensorflow.org/