Shape Classification System

Prepared by Marvin Kangangi, KNGMAR027 and Lindani Khuzwayo, KHZLIN012

Introduction

The purpose of this project is to design a system that can classify different types of images. For this particular project, the images will be of shapes and specifically, a triangle, square, circle and star. After classifying the shapes, this project will also investigate the effects of different parameters on both the accuracy and execution time of the system. The parameters that will be investigated include the type of data, epoch. This will be investigated by testing them. The results obtained will be comprehensively discussed and conclusions will be drawn.

Literature Review

Machine learning has shown the fastest growth in the area of computer science. Machine learning entails the automated detection of not only useful but meaningful trends and patterns in data, its tools also entails endowing programs that have the capability to learn and adapt. In this specific project, deep learning, specifically deep neural networks will be implemented and investigated.

Deep learning is one of the subsets of ML. It's a subset that learns and improves itself by examining different algorithms. Deep learning is mainly based on artificial neural networks such as deep neural networks. Artificial neural networks were actually inspired by biological neural networks specifically how they process and distribute information. Human brains comprise a network of neurons that are interconnected. As information is brought to the brain, a layer of neurons process this information, providing insight then afterwards forwards it to the next layer. This is the mechanism that artificial neural networks are based on. Artificial neural networks are made up of many different layers that process inputs and produce outputs based on their activation functions.

ANN artificial neural network depends on a few thing the fundamental aspects of ANN are namely the input and activation functions, network architecture and the weights of each input connection while the ANN's behavior is defined or described by the current values of the weights.Below is a basic representation of an ANN

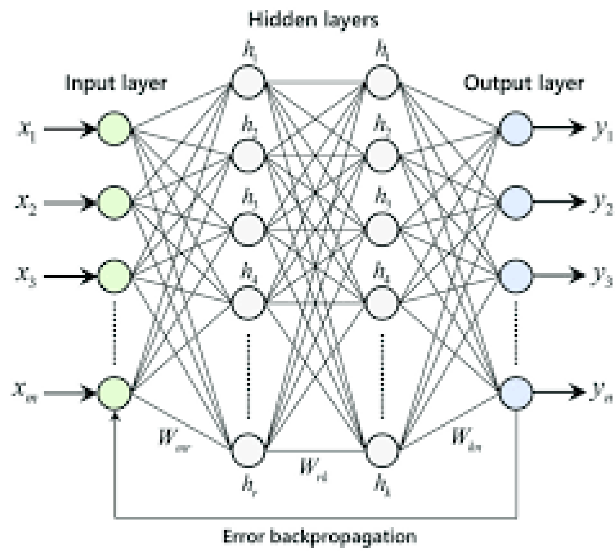

Figure 1: ANN structure

There are different types of layers used in ANN. Some commonly used layers are Dense layers aka Fully connected layers, convolution layers, pooling layers, normalization layers just to mention a few. Different types of layers perform different kinds of transformations on their inputs. These layers are added and connected depending on the task at hand

This project borrows from the theory and concepts discussed above to solve the problem described below.

Problem

The project will look at the classification problem in three parts.

1. Is the model accurate i.e does the model identify circles, squares, triangles and stars?
2. What is the effect of a different data split on both the validation and test accuracy
3. What is the effect of different epochs on the validation and test accuracy.

Technology used

In order to implement this deep learning system , Tensorflow and Keras were used. Tensorflow is an end-to-end platform that allows one to build machine learning programs. It allows one to make dataflow graphs which defines how data is processed and transferred over a series of nodes. This is done using the Python language.

Keras is a deep learning API that was developed to assist in implementing neural networks. Tensorflow supports Keras. Keras has various layers that it supports in the implementation of the neural network.

Keras has different layers that were used in the project. This include:

1. The convolution layer - this makes a convolution kernel where a single input is convolved to produce a tensor of outputs.
2. The dropout layer which drops out the hyperparameters defined by users
3. A pooling layer that helps with pooling tasks
4. A dense layer - generally, a dense layer in a neural network is a layer that is deeply connected whereby it receives inputs from all of the previous neurons.

Implementation

The system accepts a dataset, processes the dataset, splits the dataset into training, validation and testing subsets. The training subset it used to train the model implemented. This mainly involves setting of parameters. The validation subset evaluates the model implemented. After the evaluation has produced desirable results, the model is then tested with unseed data and evaluated. The main metric that will be used to evaluate the model is its accuracy. This is essentially how close or far off the predicted results are from the ground truth. After this, the different parameters will be tested by checking the accuracy of the model with different parameters. Analysis will be done for the results obtained.

Results

The program was run with the following specifications:

1. The dataset was split into two equal halves. One for training and evaluation. The other for testing.
2. The training and evaluation data was again split. 80% of it is for training and 20% for evaluation.
3. The epochs were set to 10. Epochs are the number of times the training dataset is passed through the algorithm.
4. A batch size of 256

With these specifications, the evaluation accuracy obtained was 0.5124 and testing accuracy was 0.5002.

The following tests were run:

1. Changing the training/evaluation split to 50% for training and 50% for evaluation

With this, the following results were obtained:

Evaluation accuracy	Testing accuracy
0.2466	0.2101

2. Changing the training/evaluation split to 25% for training and 75% for evaluation

With this, the following results were obtained:

Evaluation accuracy	Testing accuracy
0.2556	0.2467

3. Splitting the data to 33% for training/validation and 66% for testing. The validation split was 20% of the training/validation data.

With this, the following results were obtained.

Validation accuracy	Testing accuracy

0.3312	0.315

4. Splitting the data to 66% for training/validation and 33% for testing. The validation split was 20% of the training/validation data.
With that, the following results were

Validation accuracy	Testing accuracy
0.5364	0.51795

5. Changing the epochs to 5
With that, the following results were obtained

Validation accuracy	Testing accuracy
0.2615	0.28

6. Changing the epochs to 20

Validation accuracy	**Testing accuracy**
0.5560	0.5163

Analysis and Discussion
In this section, the data obtained from the tests above will be analyzed.

The first problem is the accuracy of the model. The validation accuracy was found to be 0.5124 with a testing accuracy 0.500. There proved a difficulty that required a trade-off. The more layers we used, the higher the accuracy. However, this also increased the run-time to a level where we could not perform the majority of the tests. Therefore we opted for 10 layers. This would give us a decent accuracy with a run-time that can allow us to run multiple tests. Therefore we can say that the system did indeed perform the classification.

The second problem to be investigated was the effect of the data split. With tests (1) and (2), the training data was significantly reduced. This led to a decrease of the overall accuracy. We determined this to be because of the less data that the model had to learn.

With tests (3) , the amount of data used for training and validation was significantly less than that used in the testing. This gave a significantly lower validation accuracy than that of the initial conditions. We can say that the model did not perform well since it did not have a lot of data to learn from. The tes estimates are pesimistically biased.

With test(4), the amount of data used for training and validation was significantly more than that of the testing. The accuracy was slightly higher than the original one. This could be because of more training data. However, it could be noted that using less testing data could increase the variance.

The third problem that was investigated was the effect of different epochs. From the results, it was noted that the higher the epochs the greater the validation and test accuracy. This is because the model is given more time to learn the data provided. With an epoch of 5, the training data only passes through the model 5 times whereas with an epoch of 20, the model has 4x the time to learn the data.

Conclusion

From the experiment, all the problems outlined were investigated and conclusions were drawn. For the epoch, it was concluded that a higher epoch gives a higher accuracy. For the data split, we concluded that a higher training set gives a higher accuracy due to more data for learning. And finally, we concluded that the algorithm implemented did indeed classify the shapes.