
Project 1.1: Software 1.0 versus Software 2.0

Lakshmi Chaitanya Chapala

UB NO: 50290974

lakshmic@buffalo.edu

Abstract

The project will compare two different problem-solving approaches to software development which are logic based (software 1.0) and machine learning approach (software 2.0). It uses Keras framework. It will let you familiarize with model used in the program.

Introduction:

Machine Learning is a science which makes computers learn like humans by feeding machine with set of data and some real-world interactions. Here we used two different software's for producing an output. In the Software 1.0 we will directly implement logic using python which need to give output "fizz", "buzz" or "fizzbuzz" based on the inputs which are divisible by "3", "5", both "3 and 5" in a set of numbers from 1 to 100.

In the Software 2.0 also same logic is implemented but using machine learning algorithm. According to this we will first feed the algorithm with training set on which the algorithm works and the neural network is trained such that corresponding outputs are given for respective inputs. For this we will use different machine learning models and it depends on the kind of data that we are processing in the real world. The Machine learning model that is used in this project for this problem is "Sequential" model. After training the test set will be fed to the algorithm for the outputs. The accuracy of the outputs depends on different hyper parameters that are in the program.

Model in Machine Learning:

Model in machine is set of guides which leads machine learning algorithm to process data manner depending on the type of data that we take. There are different kind of models that are available. Linear Regression, Linear Discriminant Analysis-nearest neighbors etc.. The model that we used here is Sequential model as the data that we used here is classified and in tabular form

There are different hyper parameters such as no of epochs, batch size, model batch size, drop out, early patience and no of inputs, number of data split which are going to effect the accuracy of the data. By varying these all parameters we need to calculate which is more accurate and giving highest accuracy.

Drop-out: It refers to dropping out of some of neurons from the network which will regularize the network and avoids interdependency of the neurons. If the network where neurons are co-dependent it will lead to

overfitting of training data so we will maintain drop-out in between 0.1 to 1 a minimum value. The figure 1 below shows the variation of accuracy based on drop-out.

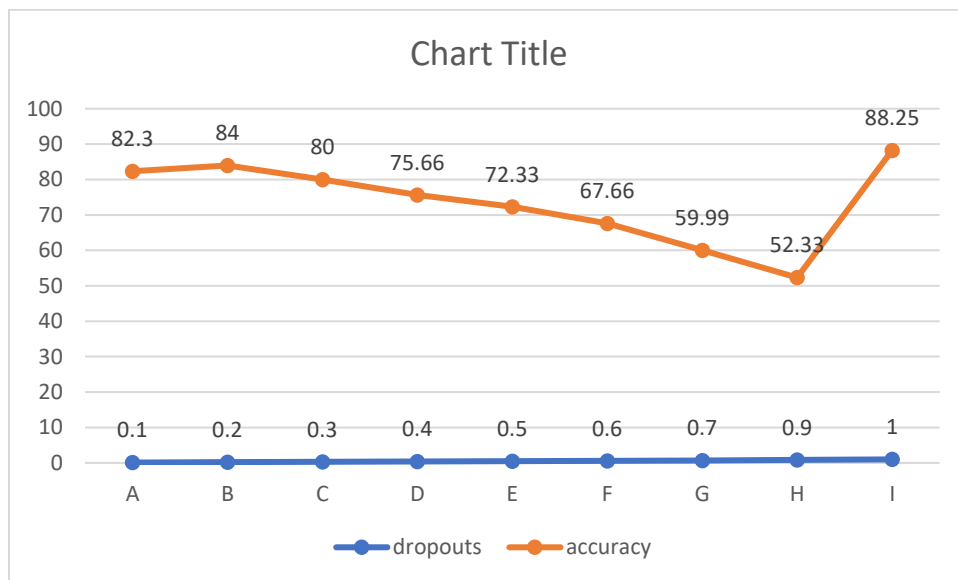


Figure1: Graph between drop-out and Accuracy

Number of Epochs: One epoch is when an entire dataset is passed forward and backward in the neural network only once. As one epoch is very big for a machine we feed a lot of epochs at one. The variation of accuracy with epochs is given in below figure 2

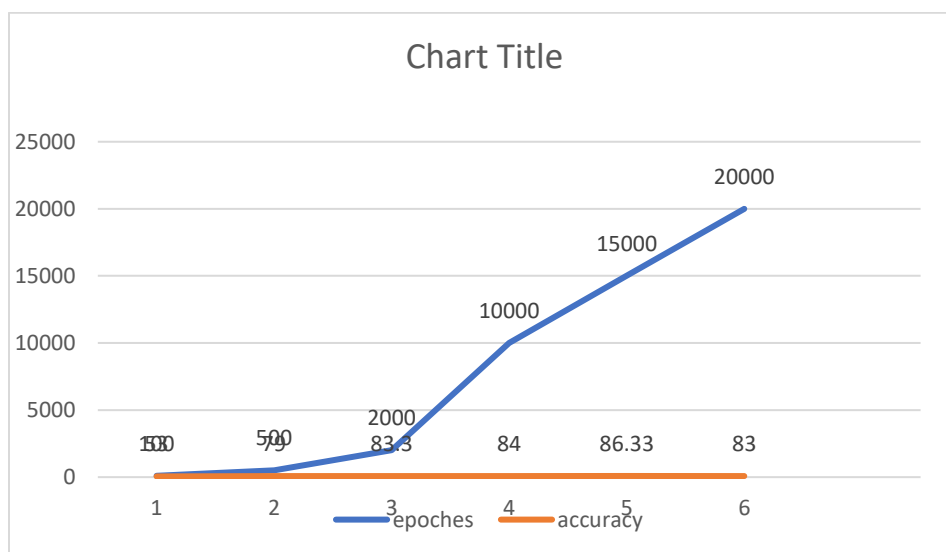


Figure2: Graph between epochs and Accuracy

Batch size: Batch size is total number of training examples present in a single batch. The variation of accuracy based on batch size is given in the below figure 3

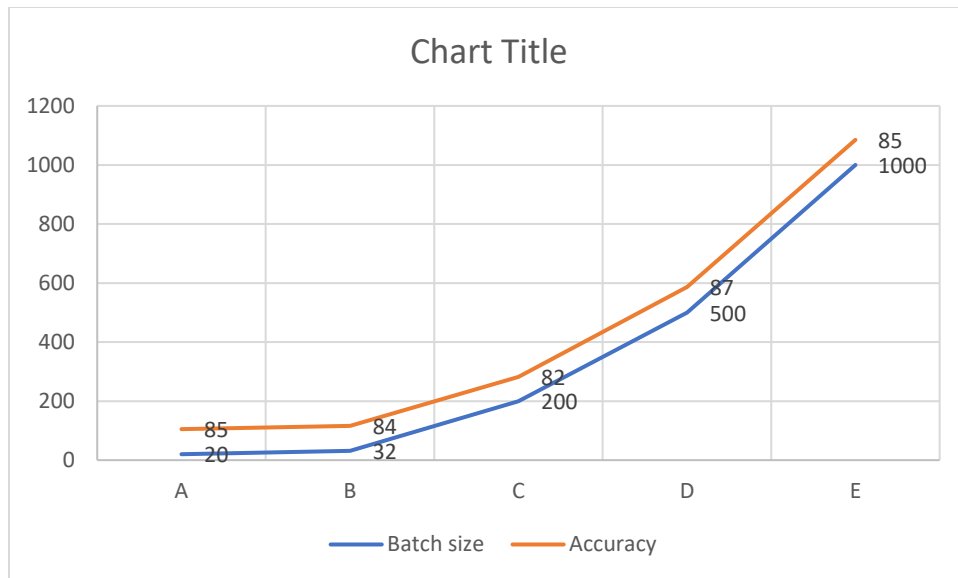


Figure3: Graph between batch size and Accuracy

Model batch size: The variation of accuracy based on model batch size is given in figure 4

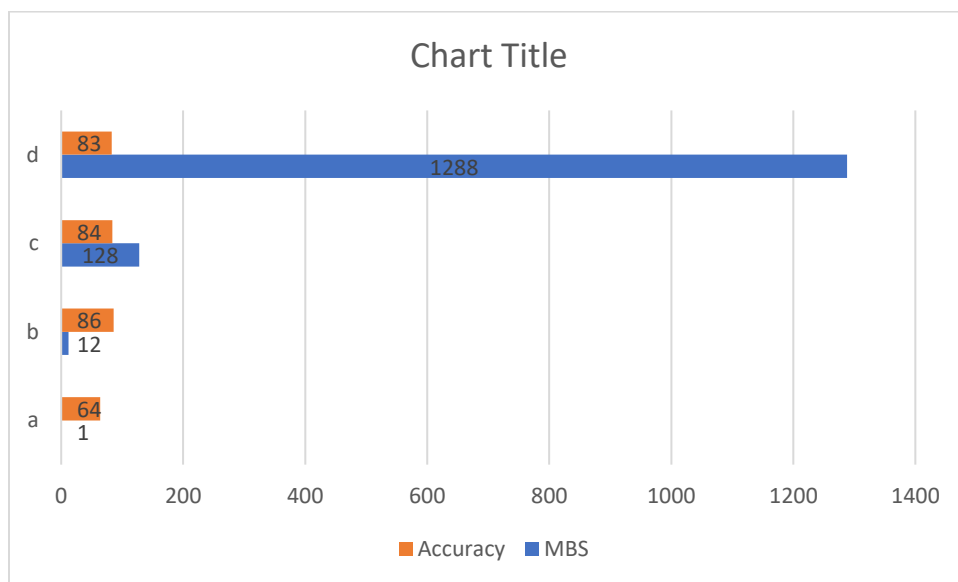


Figure4: Graph between model batch size and Accuracy

These are all some of the important parameters that we need to consider while we are training a model in machine learning algorithm. Not only these there are also other parameters such as number of iterations ,number of layers and the type of model that we use re also going to effect the neural network output.

Conclusion:

