Deep Learning

CISC 867

Project 1

By:

Name: Bilal Mohamed Fetouh Mohamed Morsy

email: 22kytb@queensu.ca

ID:20398551

Supervised by:

Dr. Hazem Abbas

Eng. **Asif Mahfuz**

Part 1:

Description and cleaning:

The data has 990 rows and 194 columns, the label column is species and it has 99 unique values

The data is clear, there are no missing values, no duplication values, there are no outliers, the data is normalized because the range of the values between zero and 1, AS shown in the virtualization figure (fig.1).

We will apply one hot encoding on the label since it has 99 unique values to convert them from categorical into numerical in which the model can deal.

We will divide the data into a training and test set using approximately 80% for training.

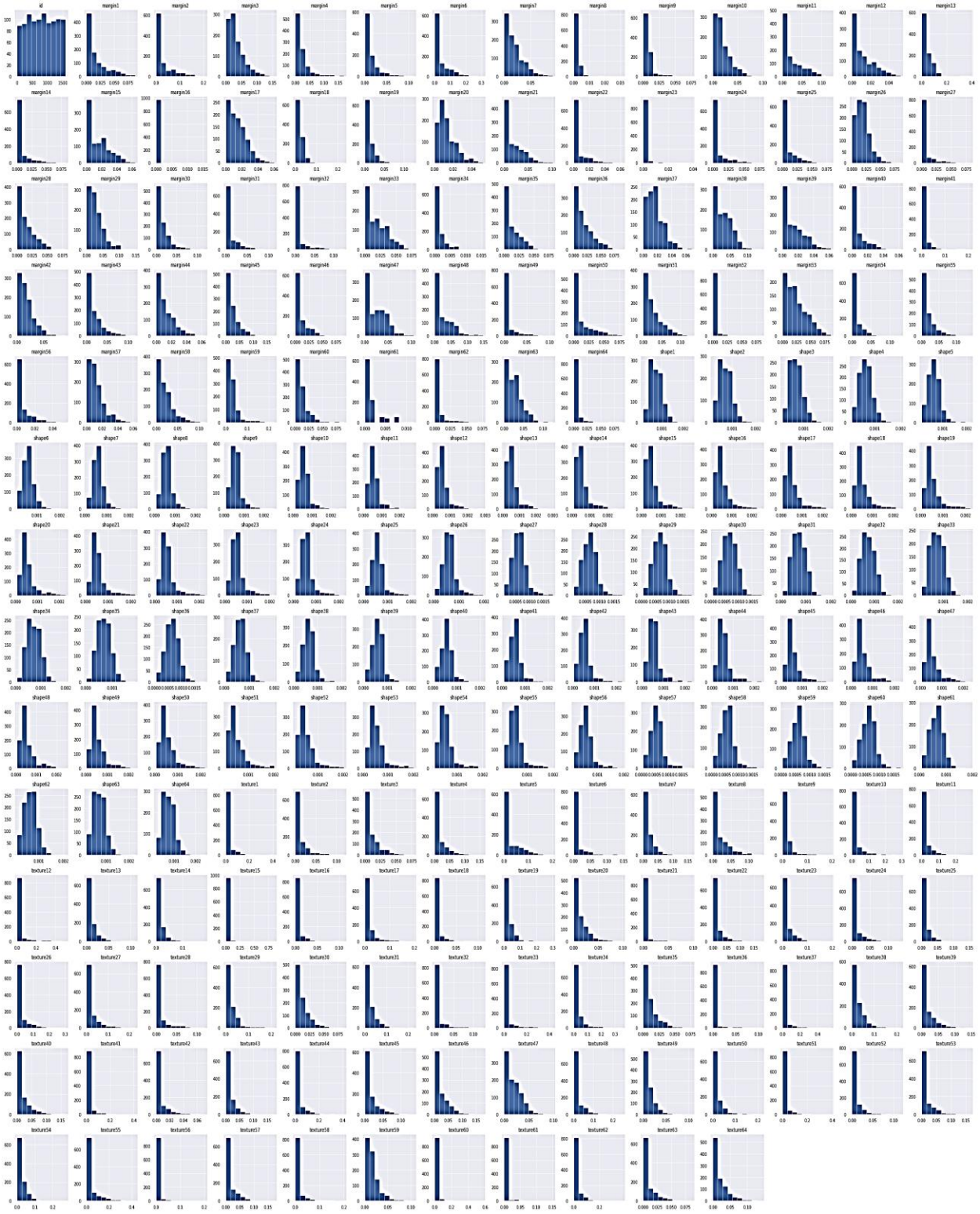
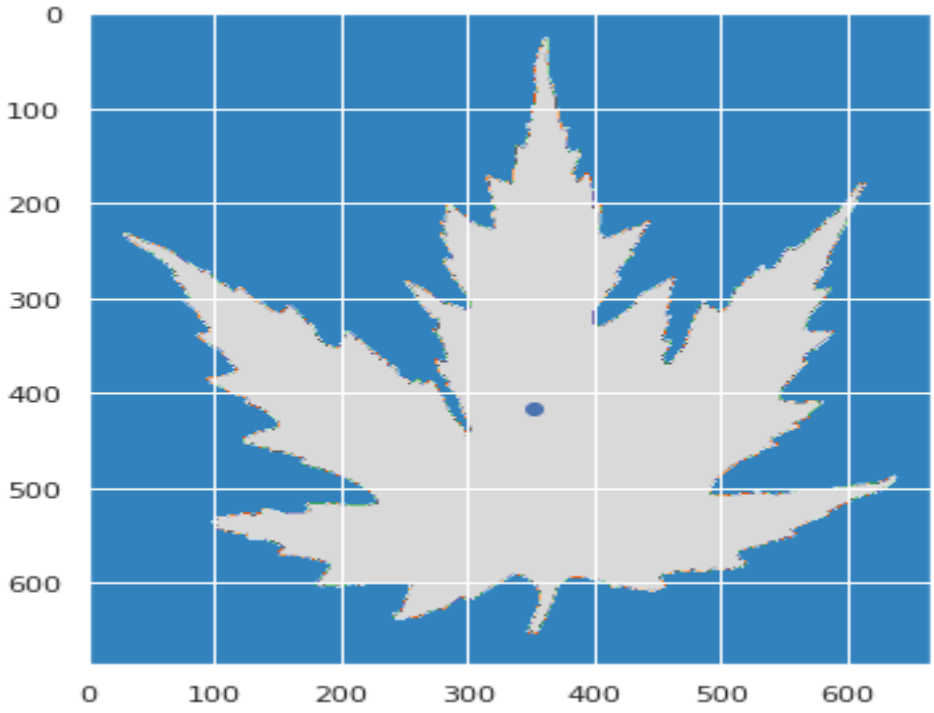
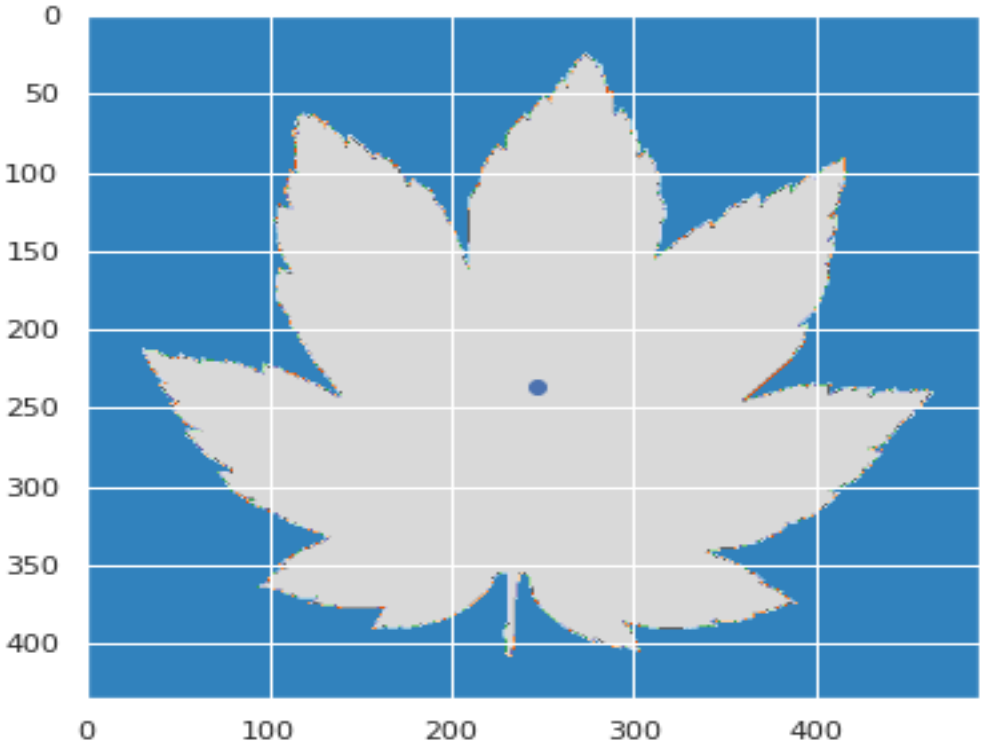


fig.1

Draw some of the images:





Part 2: MLP Model

We will tune the following hyperparameters:

- Batch Size
- Hidden Nodes Size
- Drop Rate
- Optimizer

Each one with three different values.

We will build our deep learning model using 3 function:

- 1- Deep learning architecture (3-layer MLP model (one input layer, one hidden layer with tanh activation and one output layer)
- 2- One function for virtualization of accuracy and val_accuracy
- 3- One function for virtualization of loss and val_loss

Tune the model with different optimizers using different hyperparameters.

- Batch size has [16, 32, 64]
- Hidden size has [64,128, 256]
- Dropout rate has [0.2, 0.4, 0.6]
- optimizer name has ['SGD', 'Adam', 'RMSprop']

We make 3*3*3*3 =81 Trials

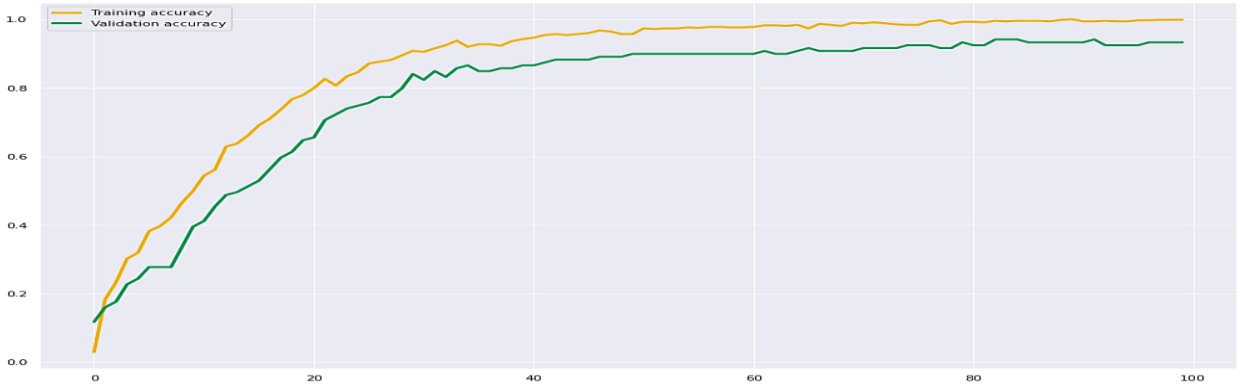
Trial: Adam optimizer

Every optimizer has 27 trials but I will choose the best 4.

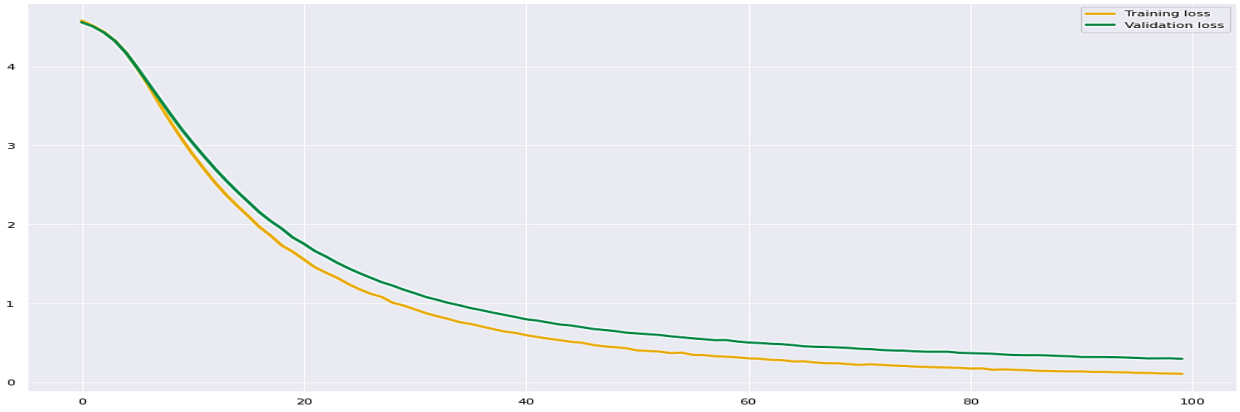
1) **The batch_size is: 16, The hidden_size is: 128, The dropout_rate is: 0.2, The optimizer_name is : Adam**

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 128)               24704

 dropout_14 (Dropout)        (None, 128)               0

 Output (Dense)              (None, 99)                12771

=================================================================
Total params: 37,475
Trainable params: 37,475
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 5ms/step - loss: 0.2913 - accuracy: 0.9444
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

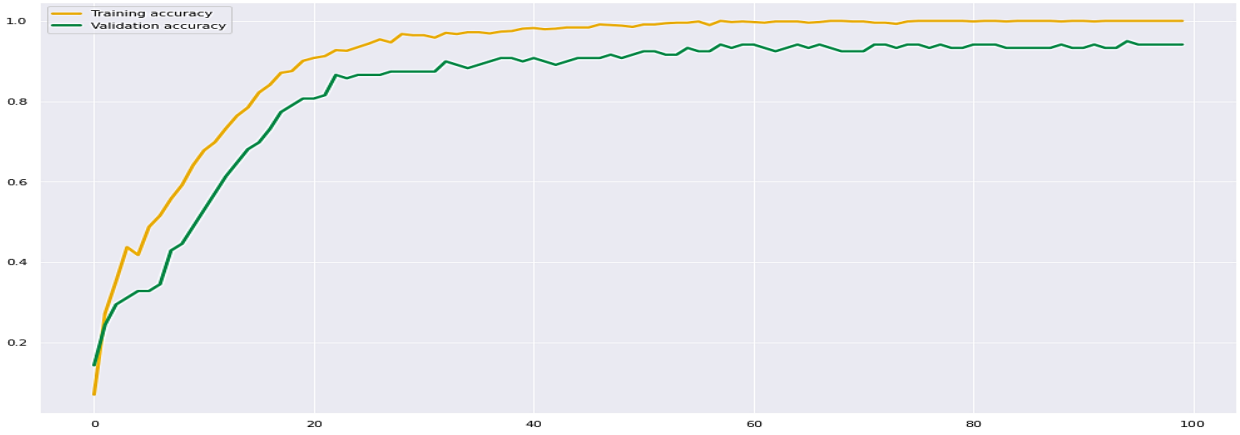


virtualization for the training loss and validation loss

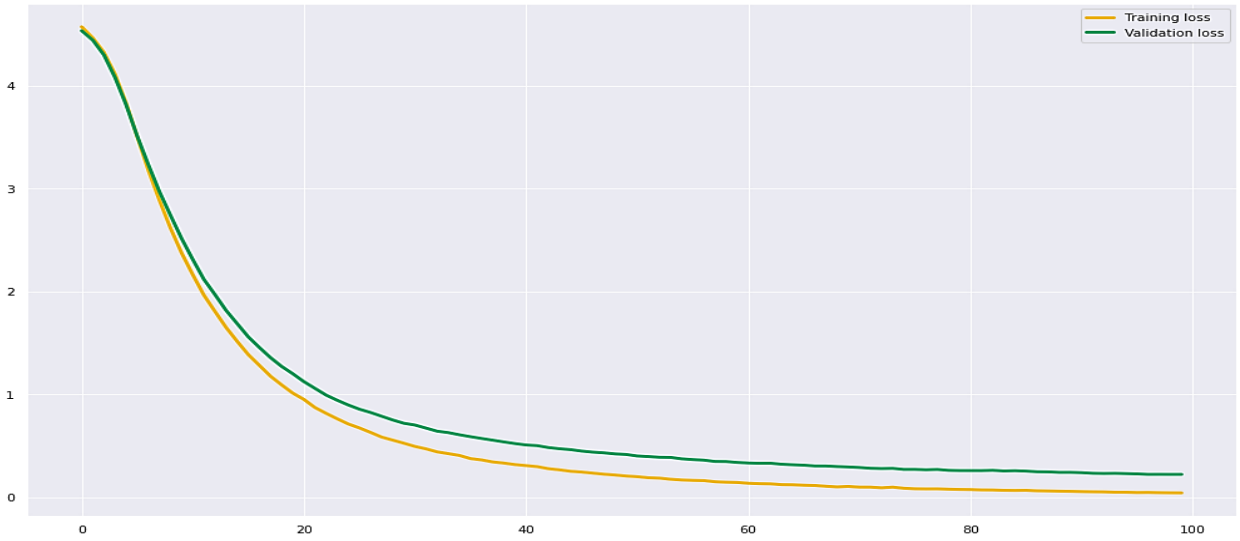
2) The batch_size is: 16, The hidden_size is: 128, The dropout_rate is: 0.2, The optimizer_name is : Adam

```
 Layer (type)                 Output Shape              Param #
=================================================================
 Layer_1 (Dense)              (None, 256)               49408

 dropout_23 (Dropout)         (None, 256)               0

 Output (Dense)               (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 6ms/step - loss: 0.2013 - accuracy: 0.9495
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

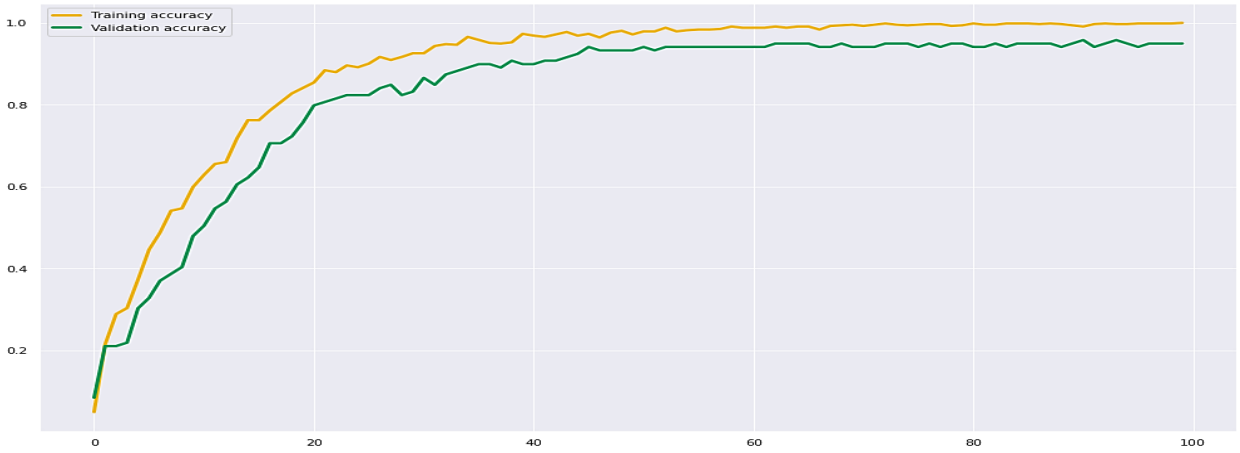


virtualization for the training loss and validation loss

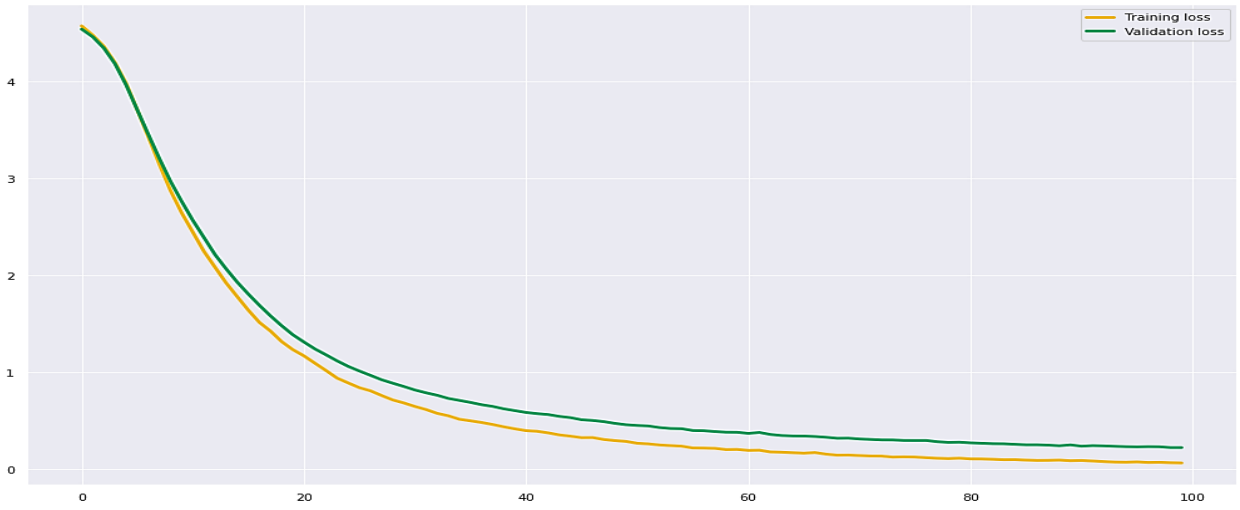
3) The batch_size is: 16, The hidden_size is: 256, The dropout_rate is: 0.4, The optimizer_name is : Adam

```
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_26 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 7ms/step - loss: 0.2108 - accuracy: 0.9495
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

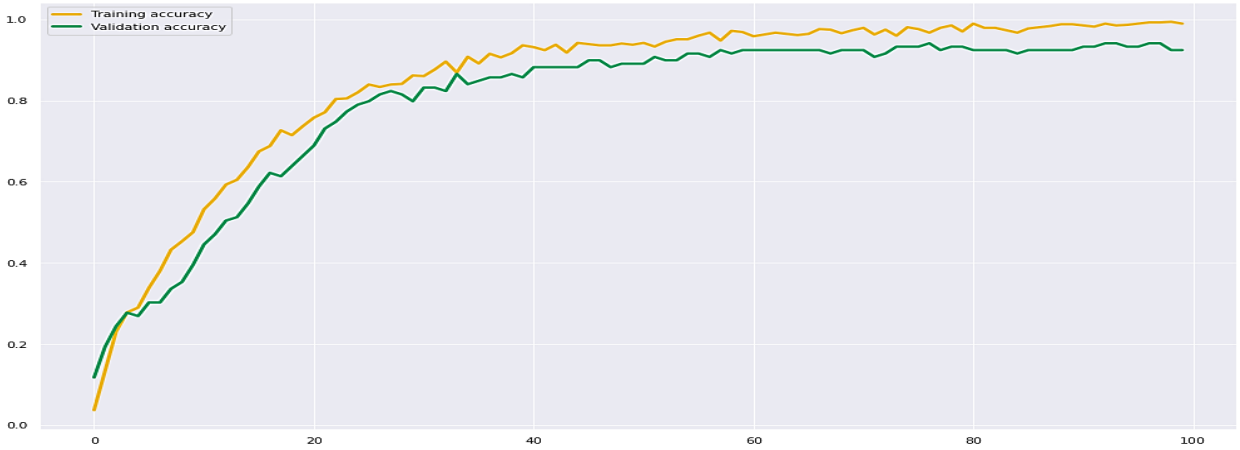


virtualization for the training loss and validation loss

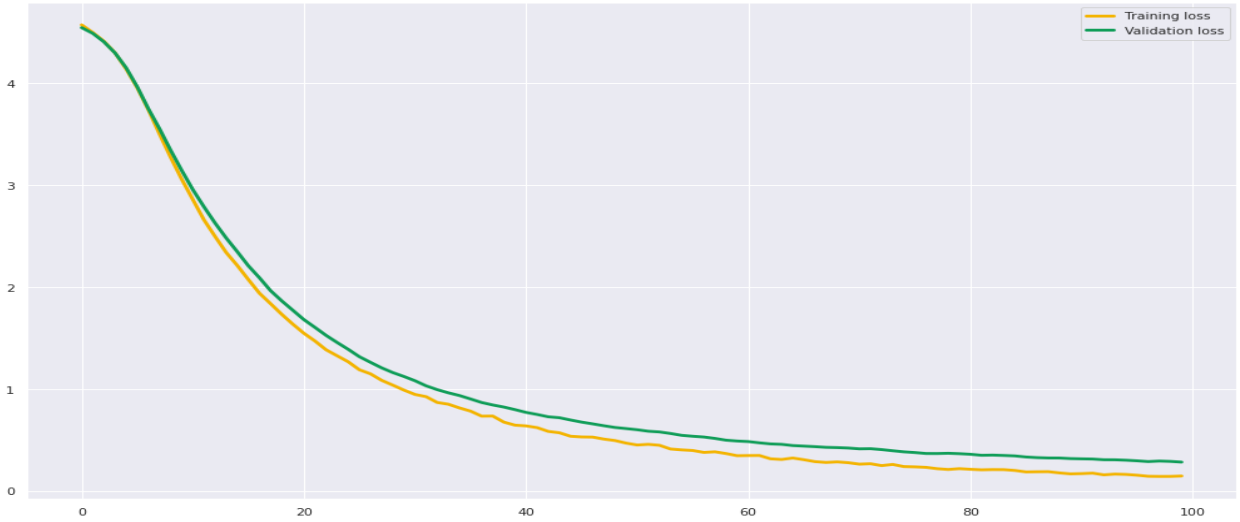
4) The batch_size is: 16, The hidden_size is: 256, The dropout_rate is: 0.6, The optimizer_name is : Adam

```
 Layer (type)                 Output Shape              Param #
=================================================================
 Layer_1 (Dense)              (None, 256)               49408

 dropout_29 (Dropout)         (None, 256)               0

 Output (Dense)               (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 4ms/step - loss: 0.2706 - accuracy: 0.9545
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy



virtualization for the training loss and validation loss

Compared the four Adam optimizers, we find that the best one with highest accuracy and lowest loss is the one which:

The batch_size is: 16
The hidden_size is: 256
The dropout_rate is: 0.6
The optimizer_name is : Adam

We will do the same for the SGD optimizer and RMSprop optimizer. Choose the best four for each one and then choose the best one from the four according to the loss and accuracy.

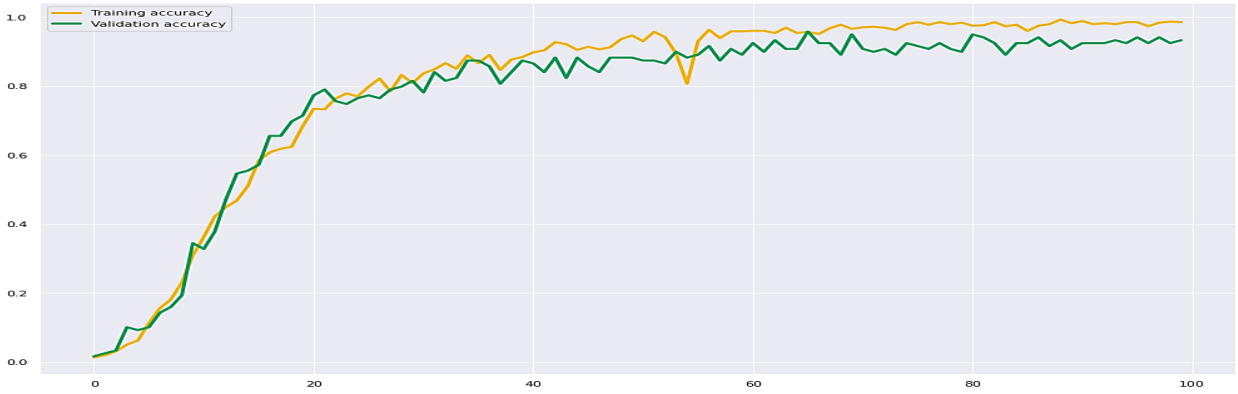
Trial: SGD optimizer

SGD optimizer has 27 trials but I will choose the best 4.

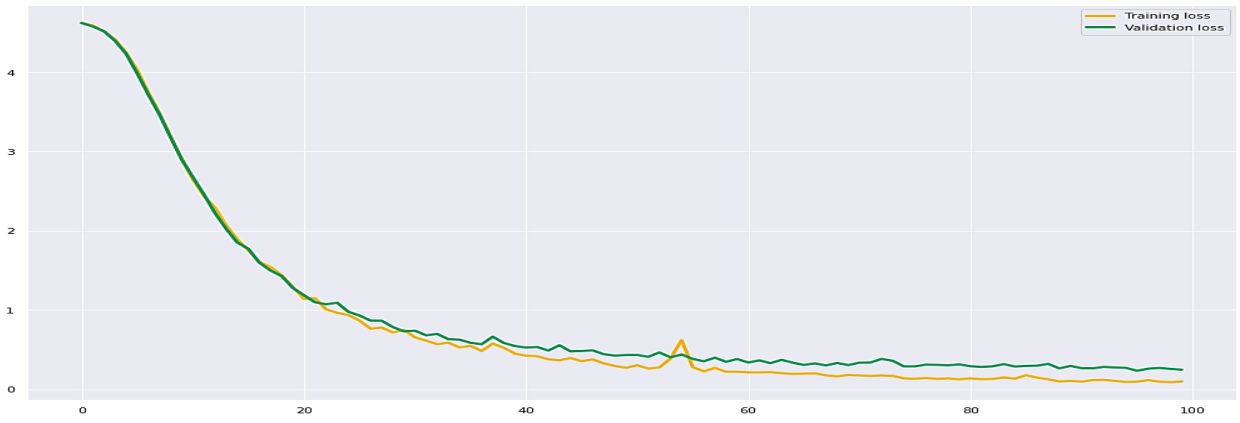
1) **The batch_size is: 16, The hidden_size is: 128, The dropout_rate is: 0.4, The optimizer_name is : SGD**

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 128)               24704

 dropout_16 (Dropout)        (None, 128)               0

 Output (Dense)              (None, 99)                12771

=================================================================
Total params: 37,475
Trainable params: 37,475
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 3ms/step - loss: 0.2858 - accuracy: 0.9444
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

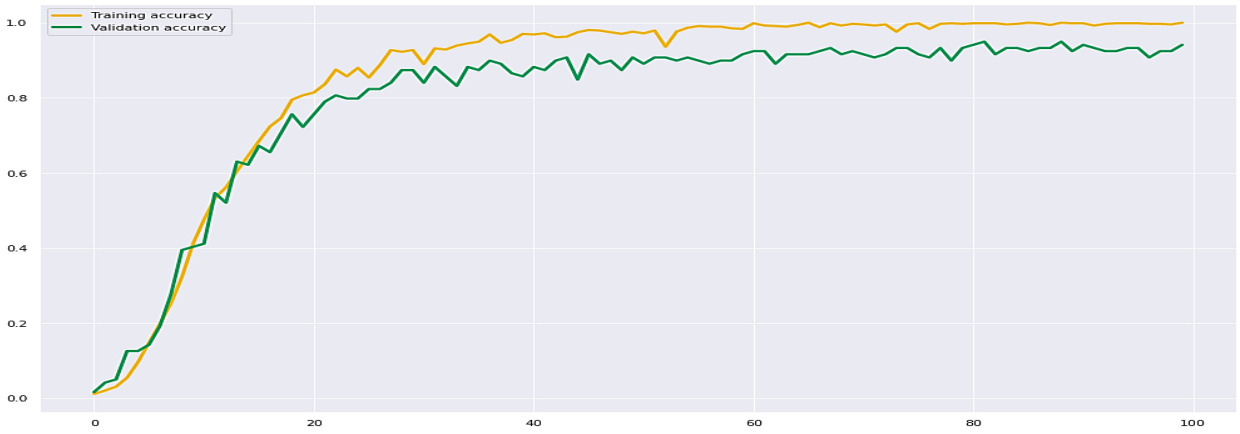


virtualization for the training loss and validation loss

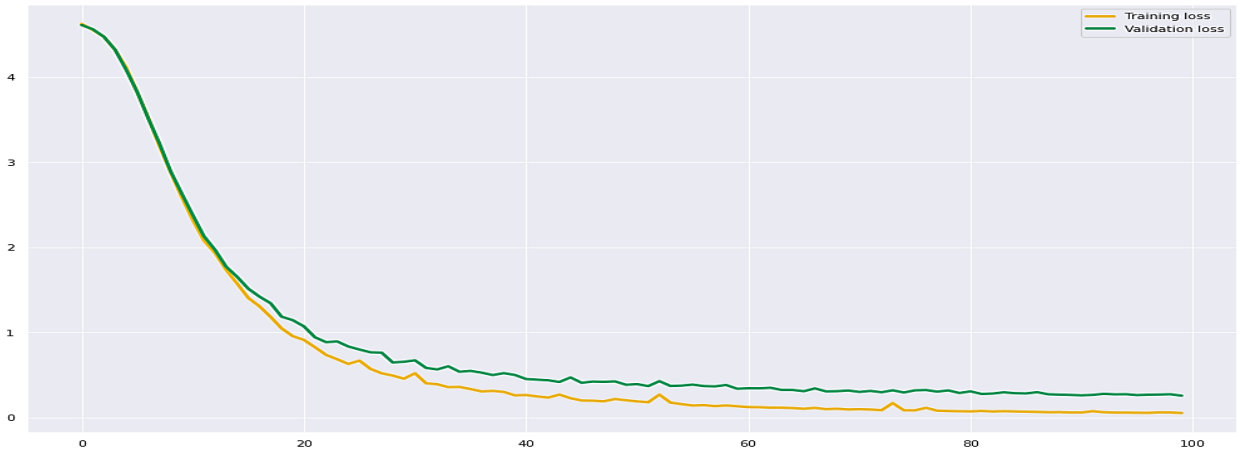
2）The batch_size is: 16, The hidden_size is: 256, The dropout_rate is: 0.2, The optimizer_name is : SGD

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_22 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 5ms/step - loss: 0.2698 - accuracy: 0.9495
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

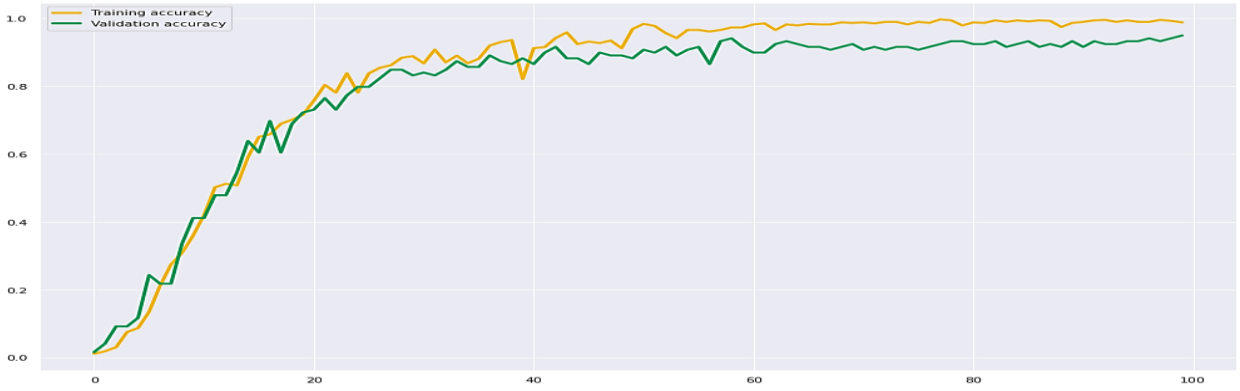


virtualization for the training loss and validation loss

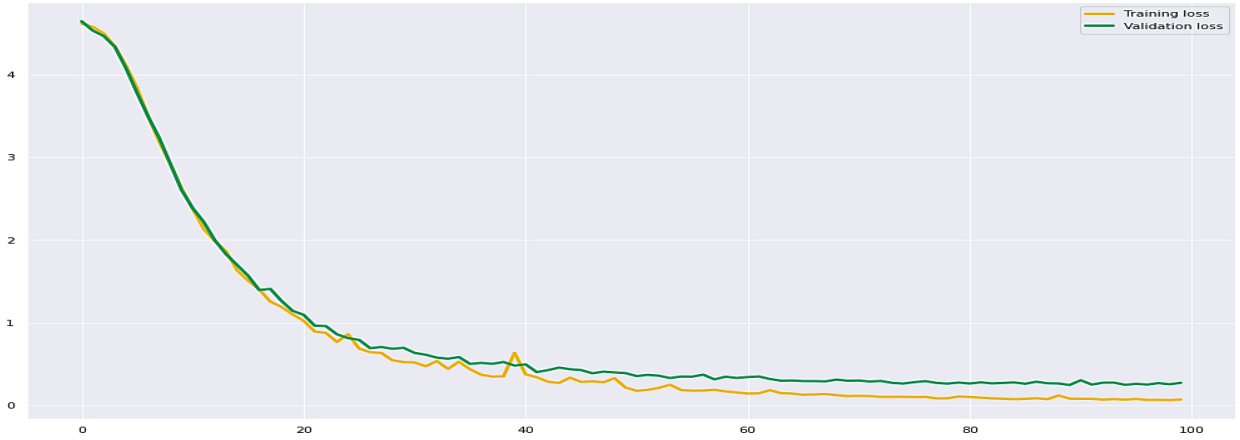
3) The batch_size is: 16, The hidden_size is: 256, The dropout_rate is: 0.4, The optimizer_name is : SGD

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_25 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 5ms/step - loss: 0.2714 - accuracy: 0.9394
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

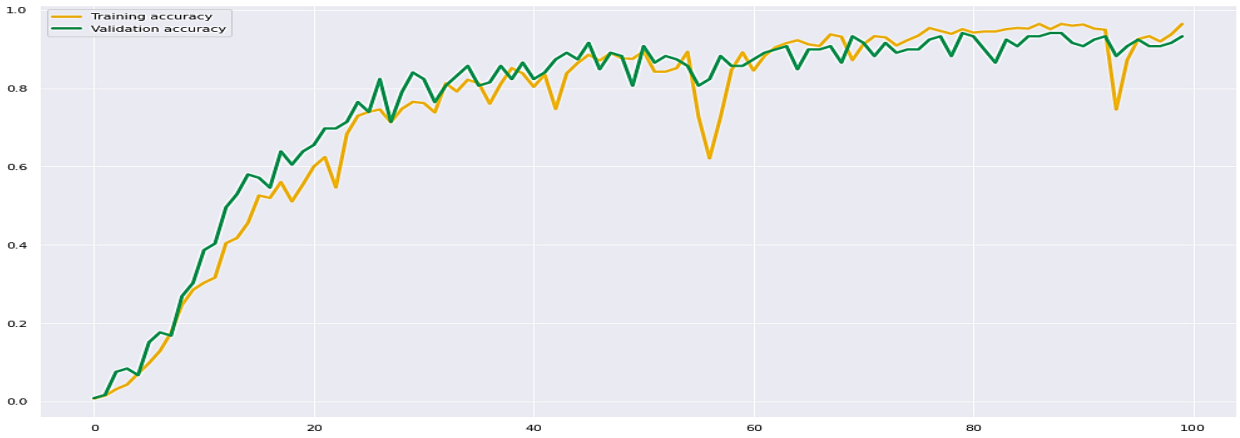


virtualization for the training loss and validation loss

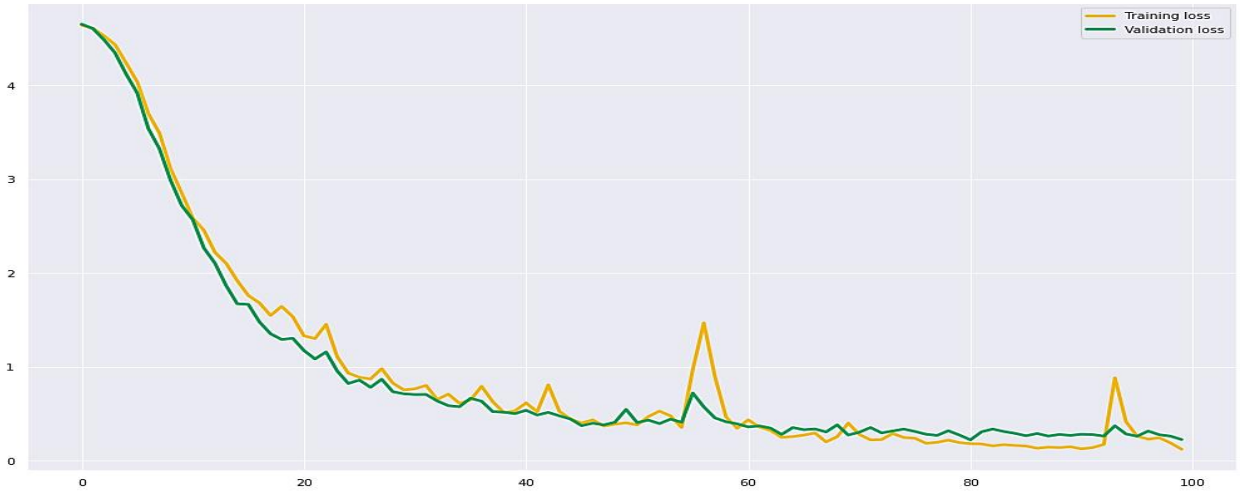
4) The batch_size is: 16, The hidden_size is: 256, The dropout_rate is: 0.6, The optimizer_name is : SGD

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_28 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 5ms/step - loss: 0.2296 - accuracy: 0.9444
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy



virtualization for the training loss and validation loss

Compared the four SGD optimizers, we find that the best one with highest accuracy and lowest loss is the one which:

The batch_size is: 16
The hidden_size is: 256
The dropout_rate is: 0.2
The optimizer_name is : SGD

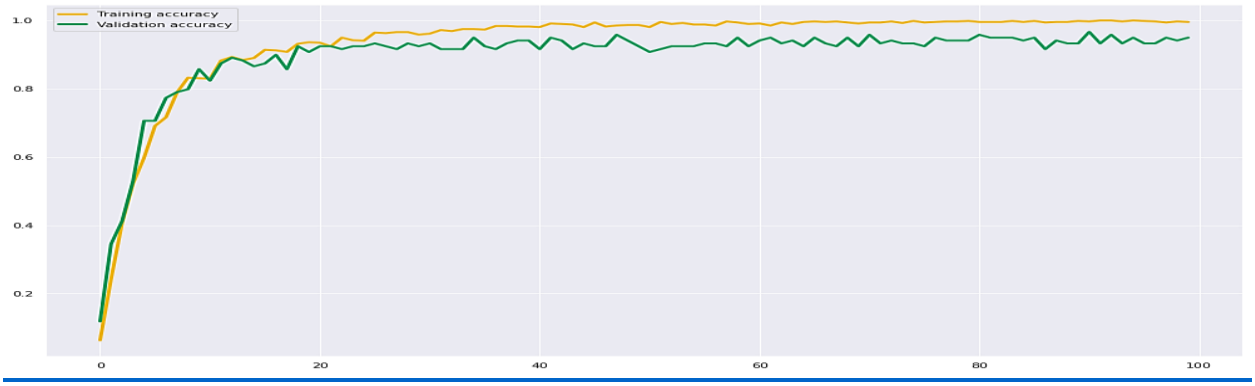
Trial: RMSprop optimizer

RMSprop optimizer has 27 trials, I will choose the best 4.

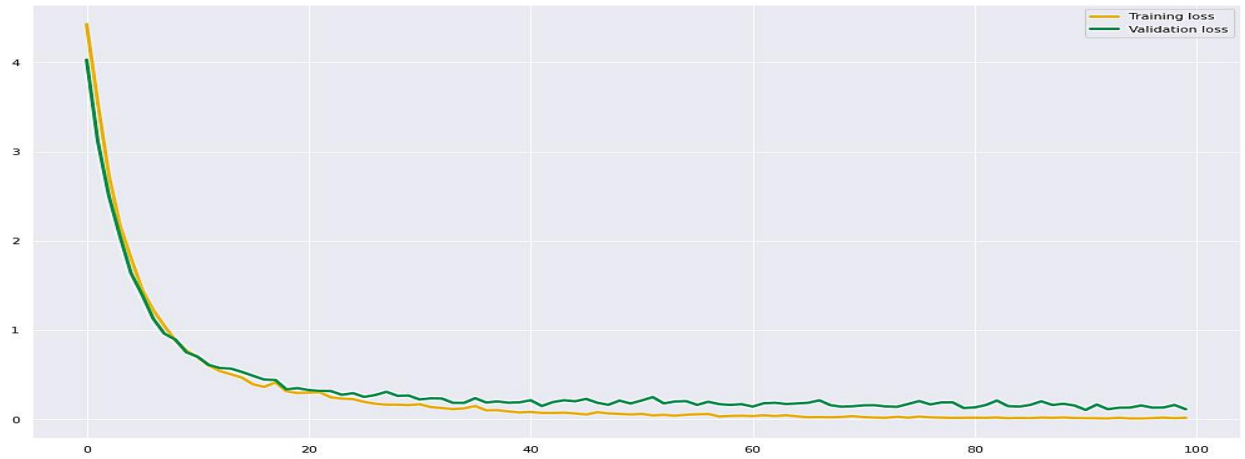
1) **The batch_size is: 16, The hidden_size is: 256, The dropout_rate is: 0.6, The optimizer_name is : RMSprop**

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_30 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 4ms/step - loss: 0.1631 - accuracy: 0.9646
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

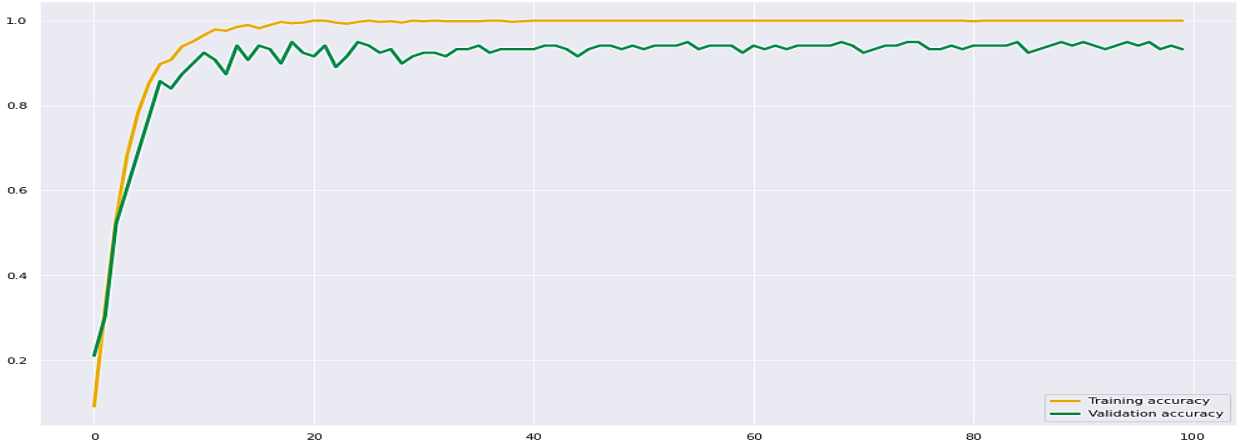


virtualization for the training loss and validation loss

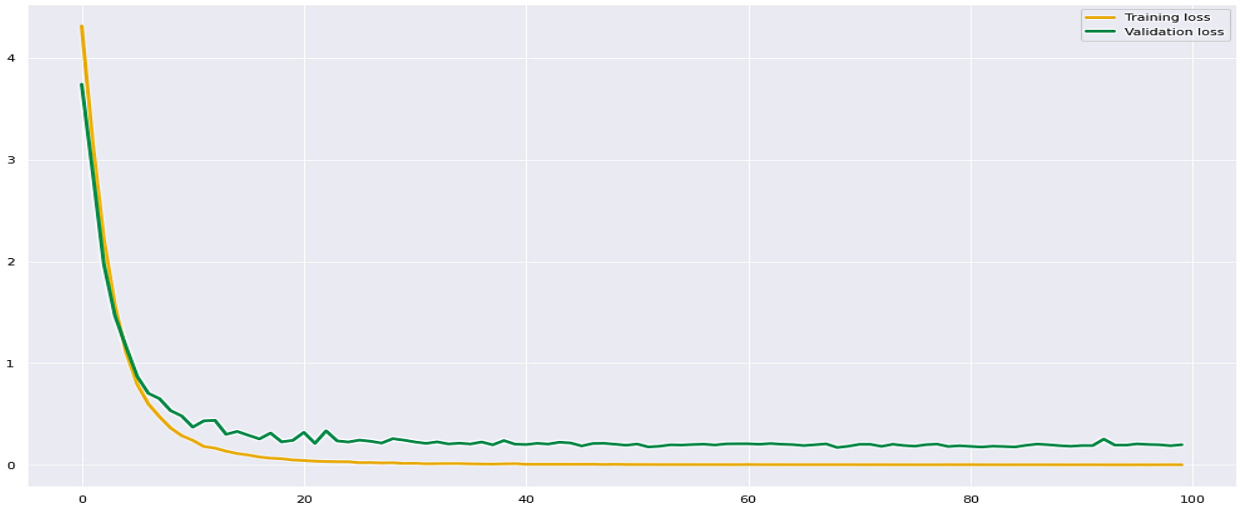
2) The batch_size is: 32, The hidden_size is: 256, The dropout_rate is: 0.2, The optimizer_name is : RMSprop

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_51 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 8ms/step - loss: 0.1608 - accuracy: 0.9697
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

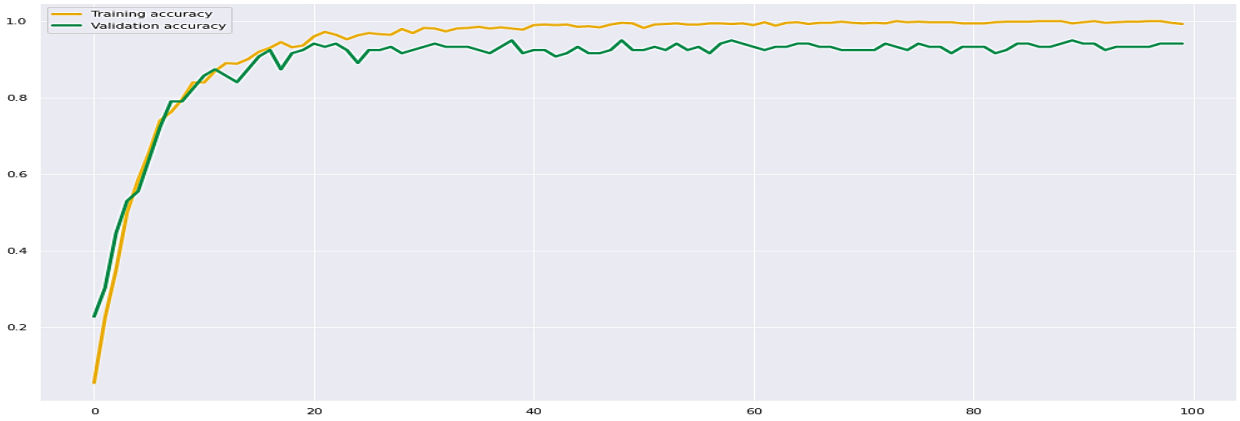


virtualization for the training loss and validation loss

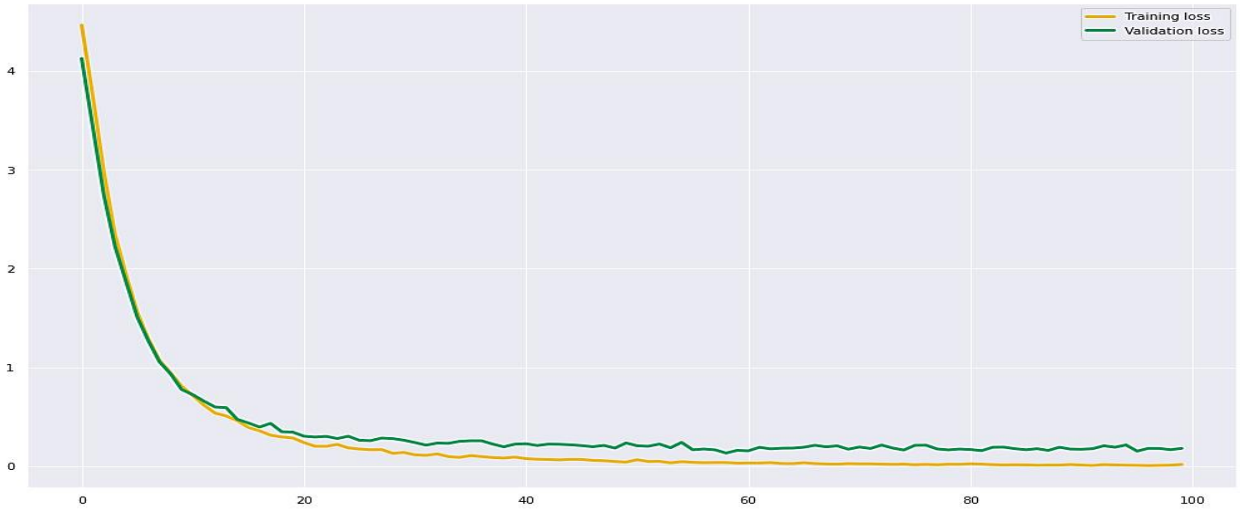
3) The batch_size is: 32, The hidden_size is: 256, The dropout_rate is: 0.6, The optimizer_name is : RMSprop

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_57 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 4ms/step - loss: 0.1603 - accuracy: 0.9646
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy

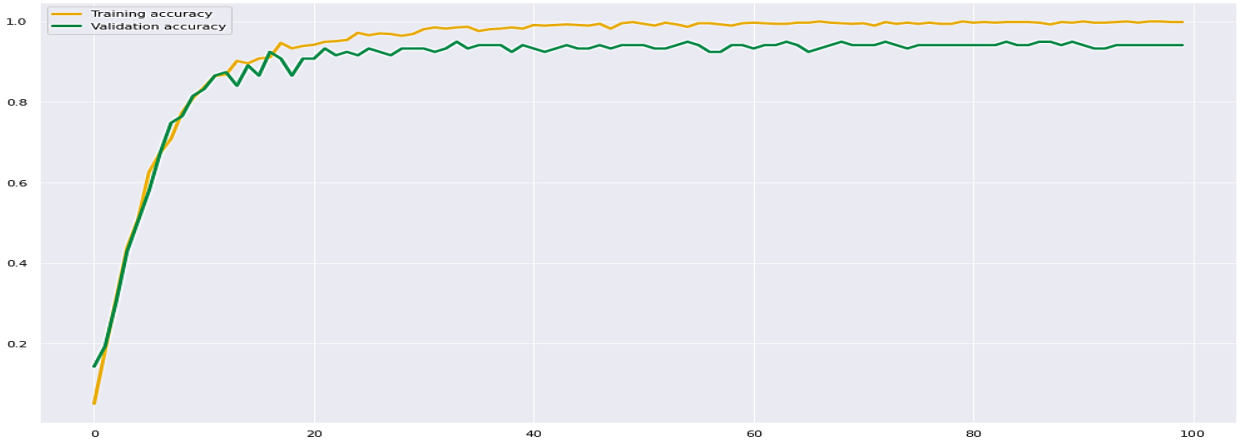


virtualization for the training loss and validation loss

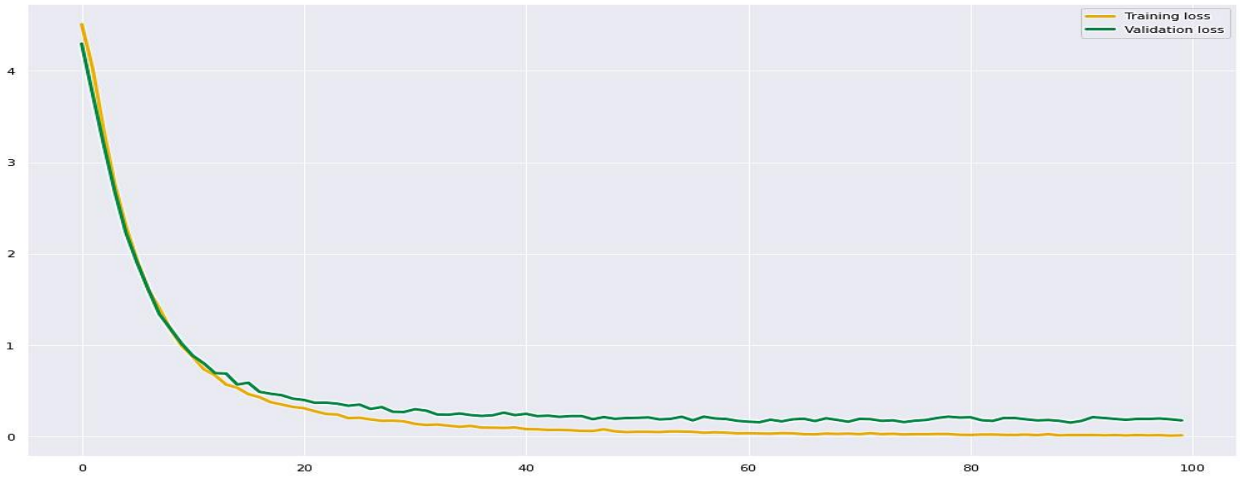
4) The batch_size is: 64, The hidden_size is: 256, The dropout_rate is: 0.6, The optimizer_name is : RMSprop

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 Layer_1 (Dense)             (None, 256)               49408

 dropout_84 (Dropout)        (None, 256)               0

 Output (Dense)              (None, 99)                25443

=================================================================
Total params: 74,851
Trainable params: 74,851
Non-trainable params: 0
_________________________________________________________________
7/7 [==============================] - 0s 5ms/step - loss: 0.1274 - accuracy: 0.9697
```

This is the loss and the accuarcy(100 epochs)



virtualization for the training accuracy and validation accuracy



virtualization for the training loss and validation loss

Compared the four RMSprop optimizers, we find that the best one with highest accuracy and lowest loss is the one which:

The batch_size is: 64
The hidden_size is: 256
The dropout_rate is: 0.6
The optimizer_name is : RMSprop

Compared the three optimizers (SGD, Adam, RMSProp) in descending order is:

1- RMSProp Optimizer
- The batch size = 64
- The hidden size = 256
- The dropout rate = 0.6
- The optimizer name = RMSProp

2- Adam Optimizer
- The batch size = 16
- The hidden size = 256
- The dropout rate = 0.6
- The optimizer name = RMSProp

3- SGD Optimizer
- The batch size = 16
- The hidden size = 256
- The dropout rate = 0.2
- The optimizer name = RMSProp

Finally, From the above order we notice that the best hidden size is 256, also the best drop rate is 0.6