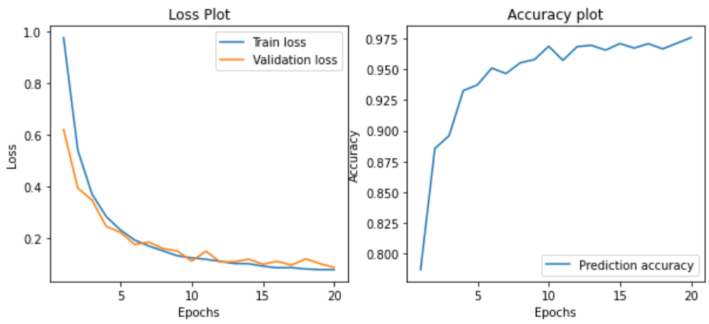
EE5179: Deep Learning for Imaging

Programming Assignment 3: RNN

1. MNIST classification using RNN

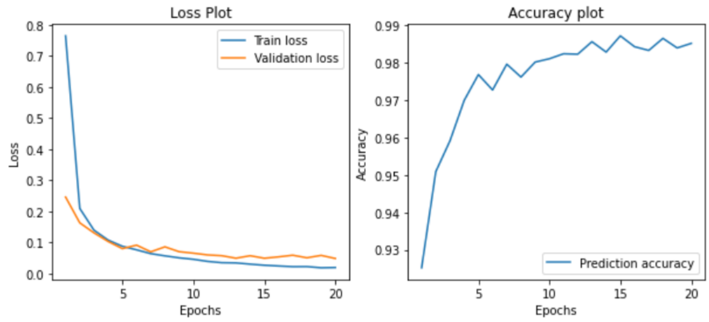
Training on different model:

1. Vanilla RNN



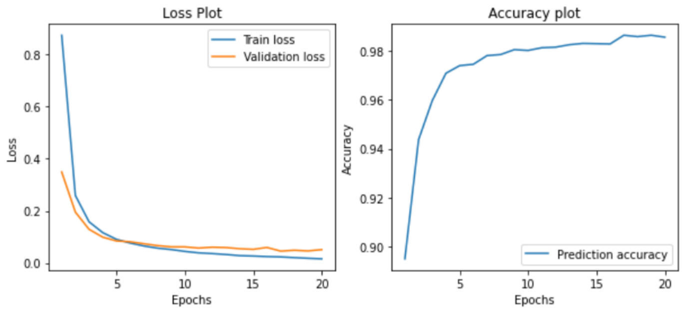
Test accuracy: 97.20%

2. Vanilla LSTM



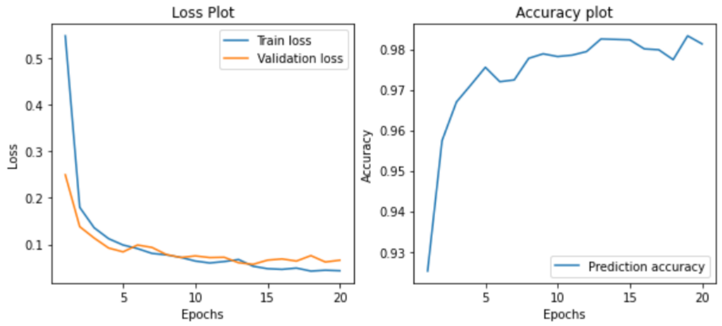
Test accuracy: 98.80%

3. Vanilla GRU



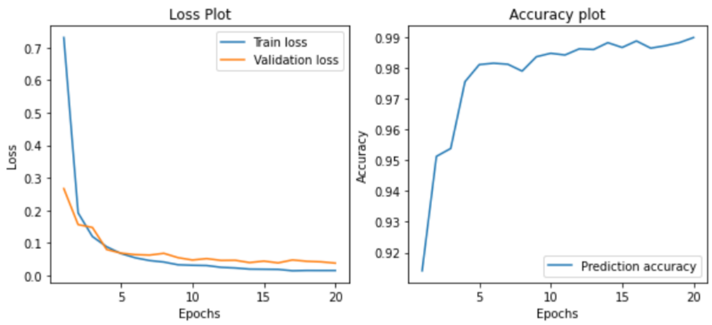
Test accuracy: 98.20%

4. Bidirectional RNN



Test accuracy: 97.70%

5. Bidirectional LSTM

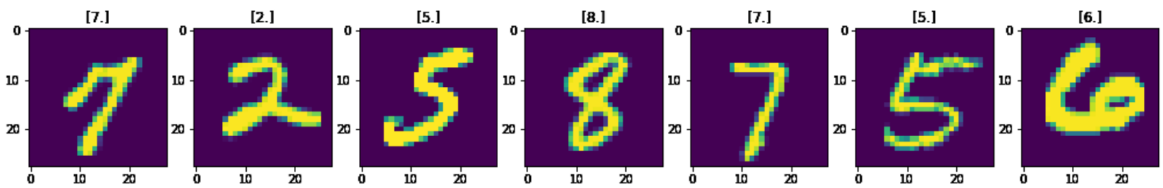


Test accuracy: 98.50%

Observations:

1. We see that more in not necessarily better. Increasing the number of layers does not improve the model accuracy by much.
2. We observe LSTMs to overall perform better

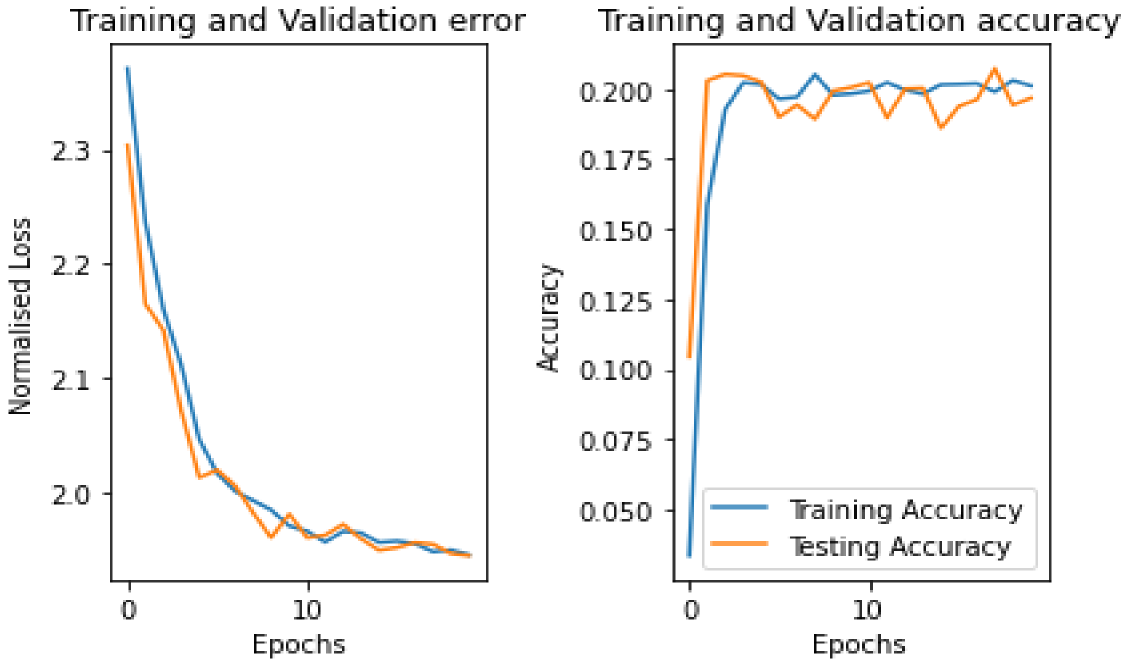
Predictions:



2. Remembering the number at a particular index in a given sequence

Training the RNN model with different hidden size

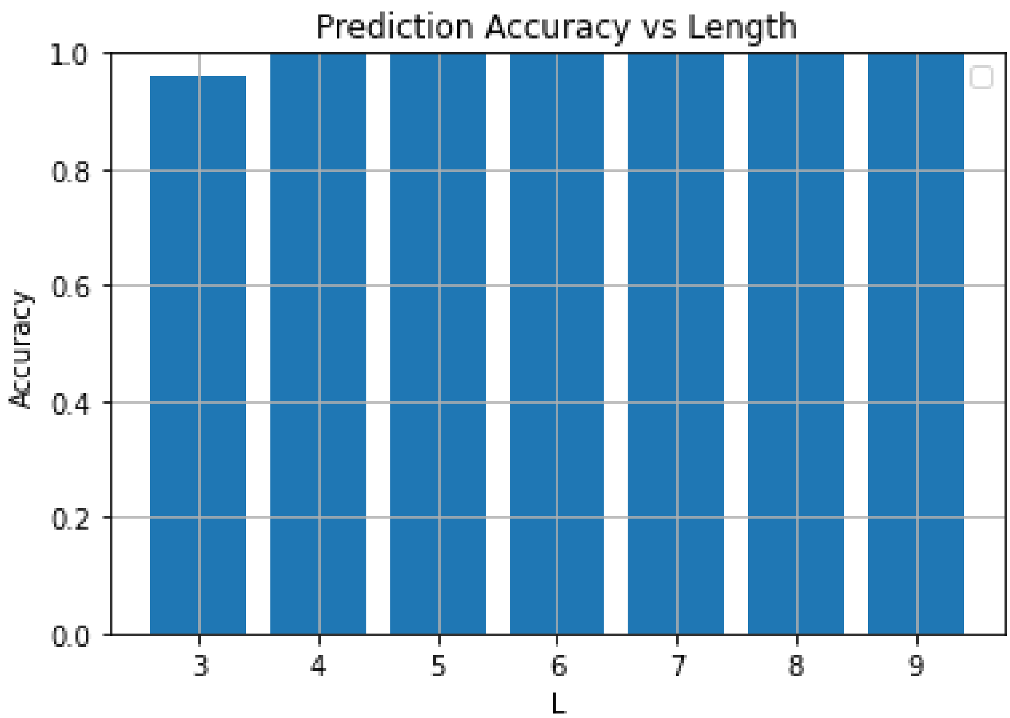
1. Hidden size = 2



2. Hidden size = 5

3. Hidden size = 10

Prediction accuracy vs length:

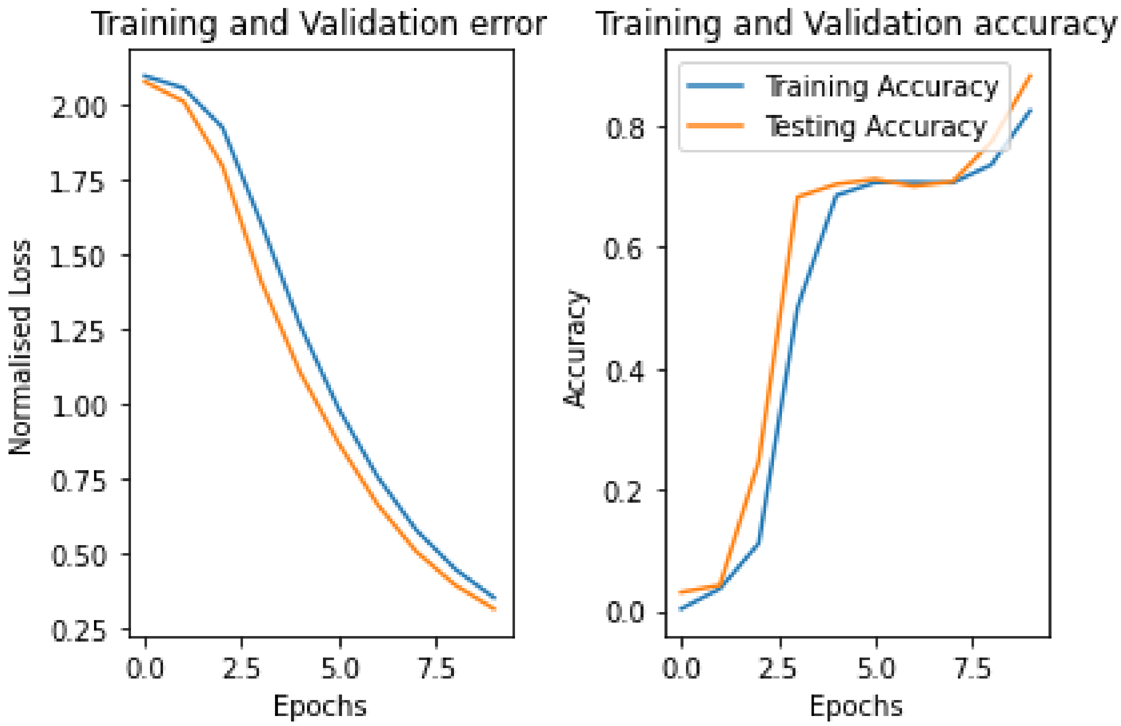


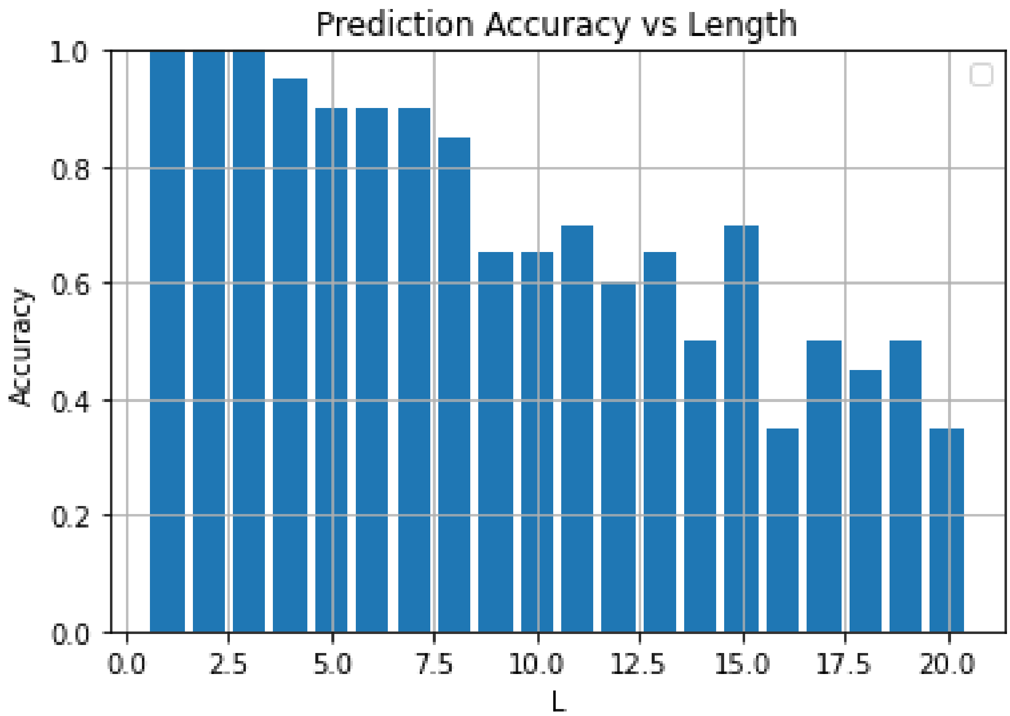
We observe that by training the model on varying L's, the prediction accuracy is high. Increasing the hidden size also improves the model accuracy over epochs.

3. Adding two binary strings

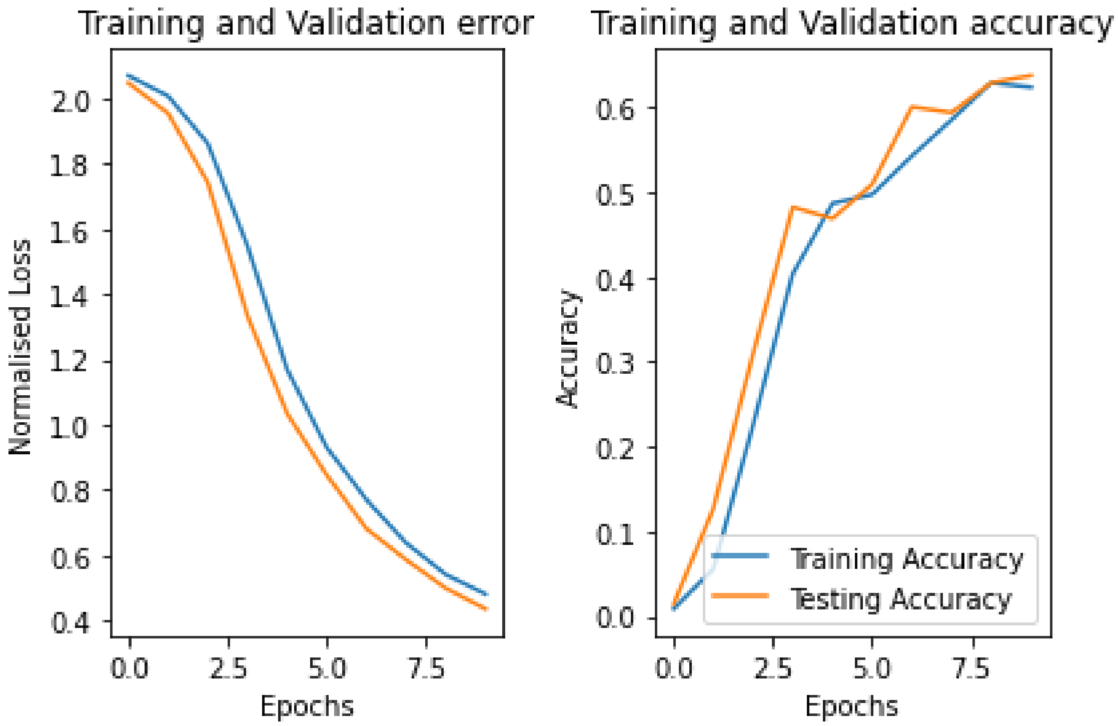
With L=5, we observe the effect of using cross-entropy and MSE loss

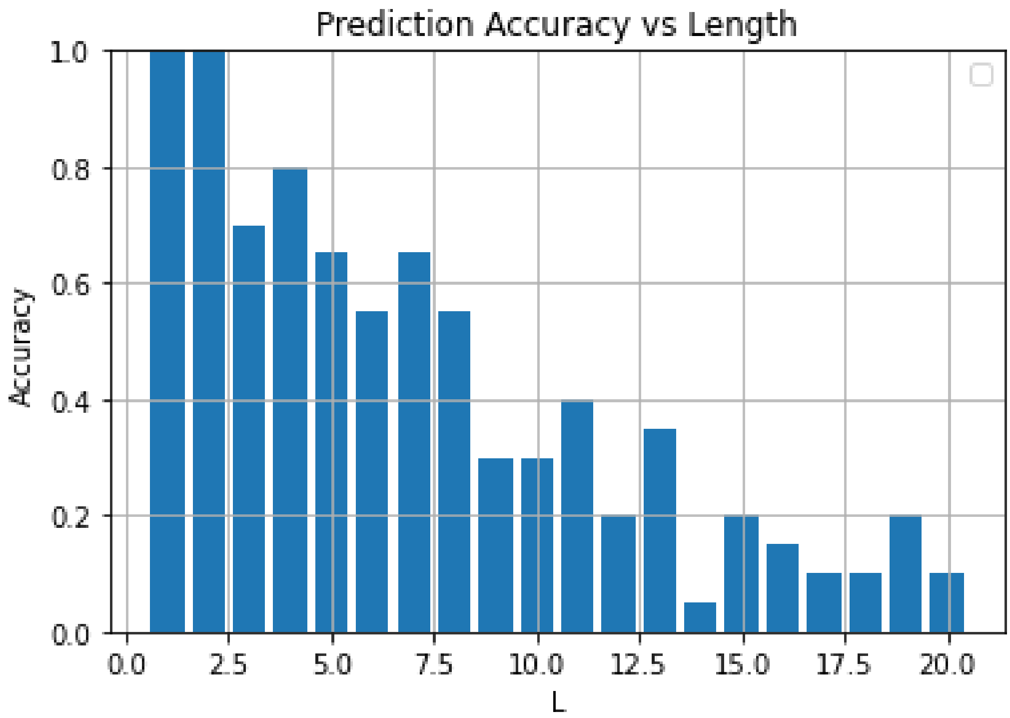
1. Cross entropy loss





2. MSE loss



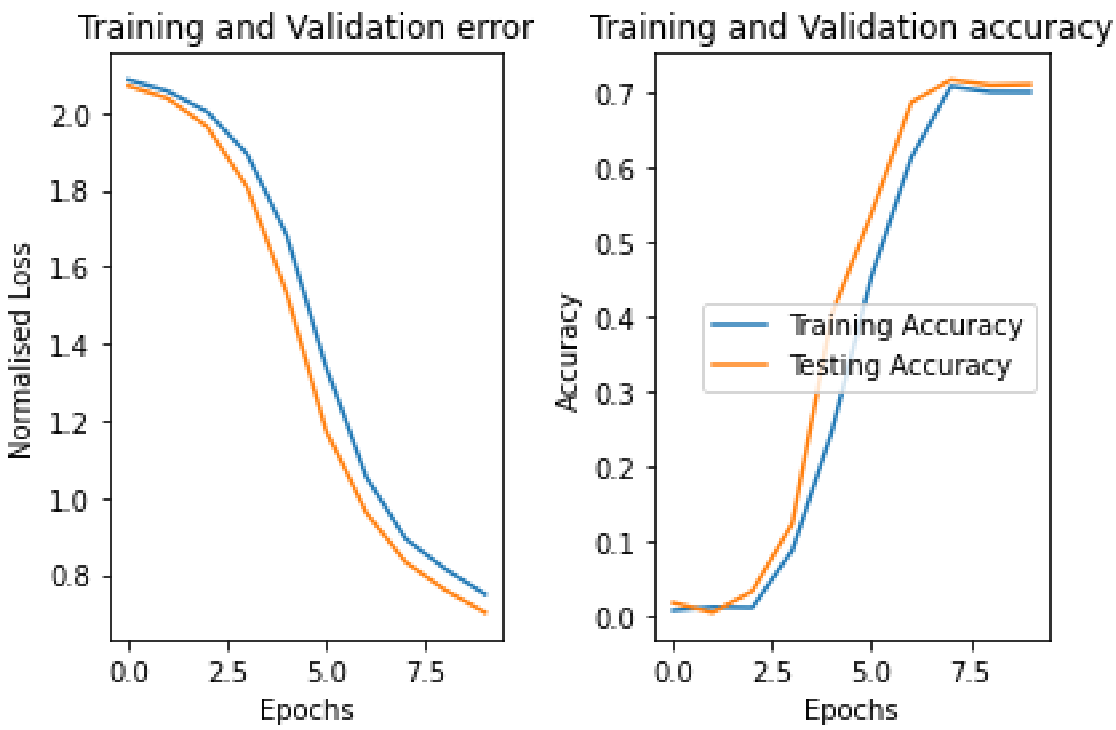


We observe cross entropy loss performs better training than mean square error.

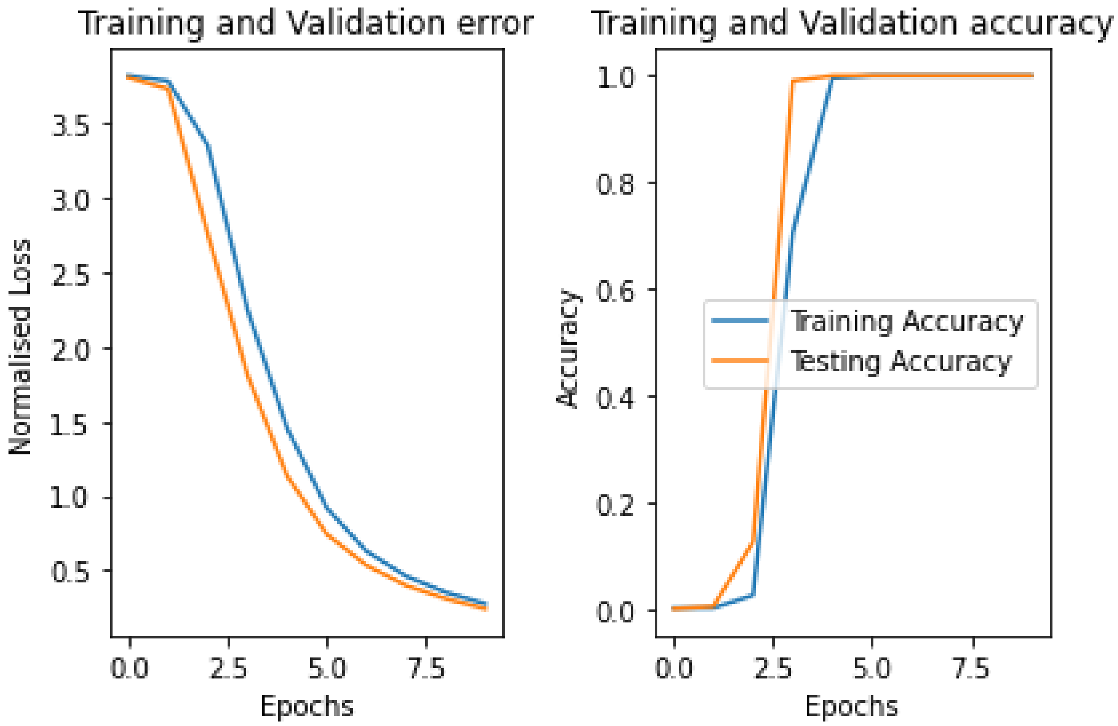
We now try to vary L=3,5,10 to observe the training loss and the accuracy of the prediction

1. L = 3

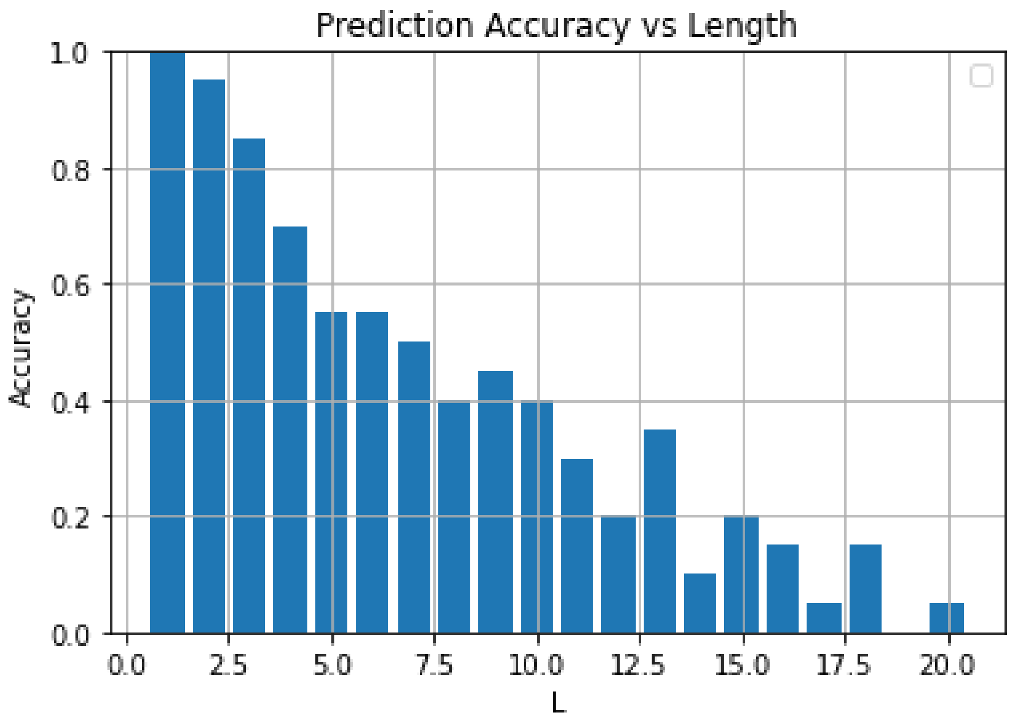
2. L = 5



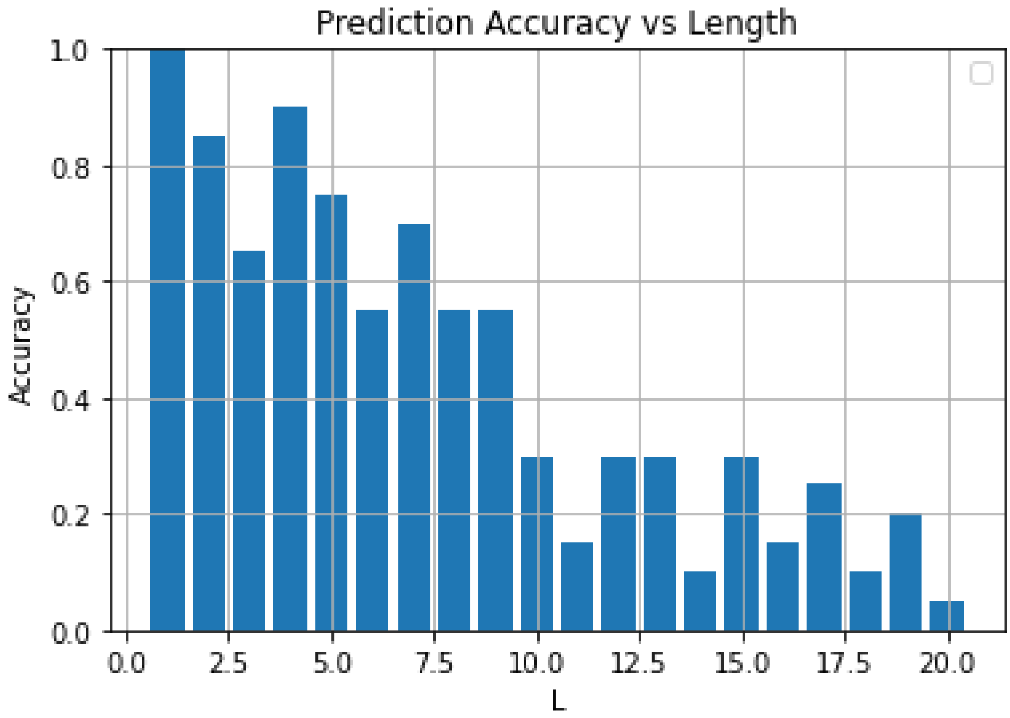
3. L = 10


Training and Validation error
Normalised Loss
Epochs
Training and Validation accuracy
Accuracy
Epochs
Training Accuracy
Testing Accuracy

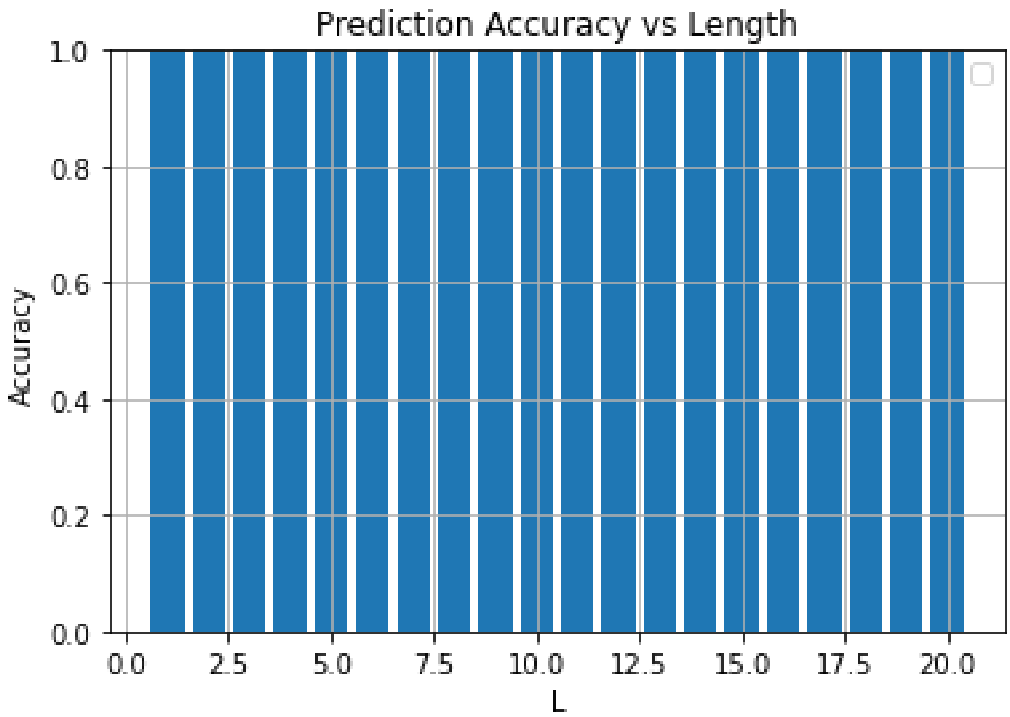
1. L = 3


Prediction Accuracy vs Length
Accuracy
L

2. L = 5



3. L = 10



Increasing L, increases the training and validation accuracy.

While observing the prediction accuracy vs length L, the accuracy depreciates