Name:Md.Nazmul Hasan

ID: 22-92444-1

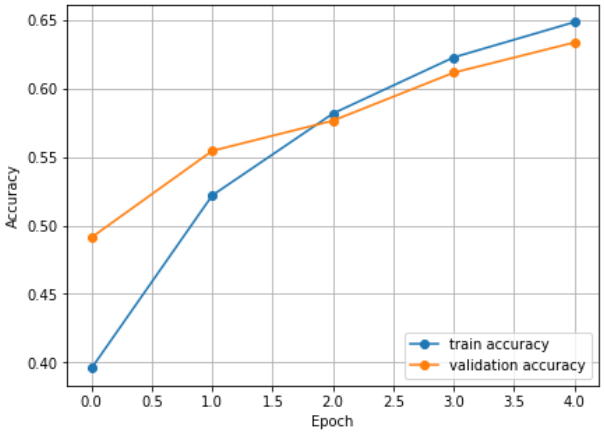
Adam Optimization:

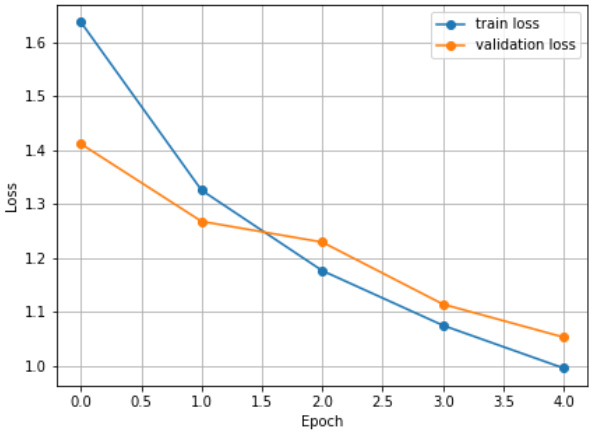
Here are the results, I have found in the adam optimization function

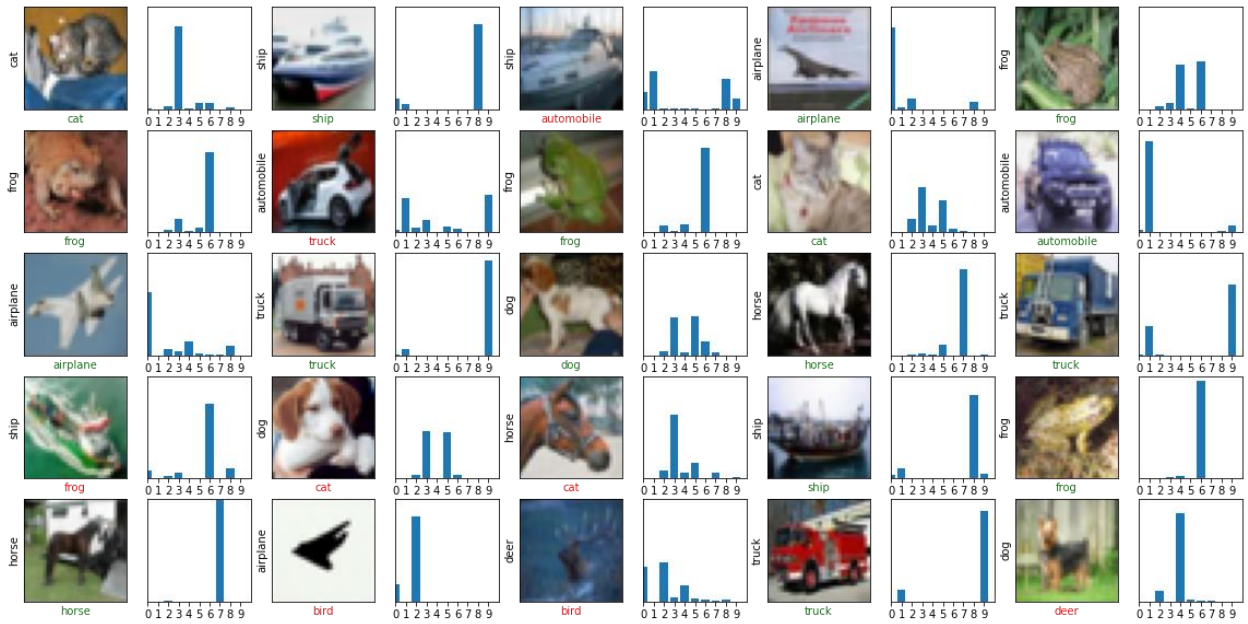
```
[9]  model.compile(
         optimizer='adam',
         loss='sparse_categorical_crossentropy',
         metrics=['accuracy']
     )

     h = model.fit(x=X_train, y=Y_train, epochs=5, validation_split=0.2, batch_size=32)

     Epoch 1/5
     1250/1250 [==============================] - 62s 49ms/step - loss: 1.6369 - accuracy: 0.3959 - val_loss: 1.4111 - val_accuracy: 0.4913
     Epoch 2/5
     1250/1250 [==============================] - 59s 47ms/step - loss: 1.3250 - accuracy: 0.5219 - val_loss: 1.2672 - val_accuracy: 0.5544
     Epoch 3/5
     1250/1250 [==============================] - 59s 47ms/step - loss: 1.1758 - accuracy: 0.5816 - val_loss: 1.2288 - val_accuracy: 0.5764
     Epoch 4/5
     1250/1250 [==============================] - 60s 48ms/step - loss: 1.0743 - accuracy: 0.6226 - val_loss: 1.1137 - val_accuracy: 0.6114
     Epoch 5/5
     1250/1250 [==============================] - 61s 48ms/step - loss: 0.9950 - accuracy: 0.6482 - val_loss: 1.0524 - val_accuracy: 0.6334
```







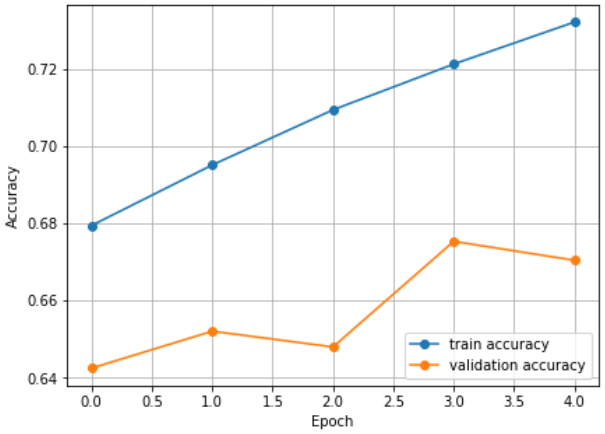
RMSProp

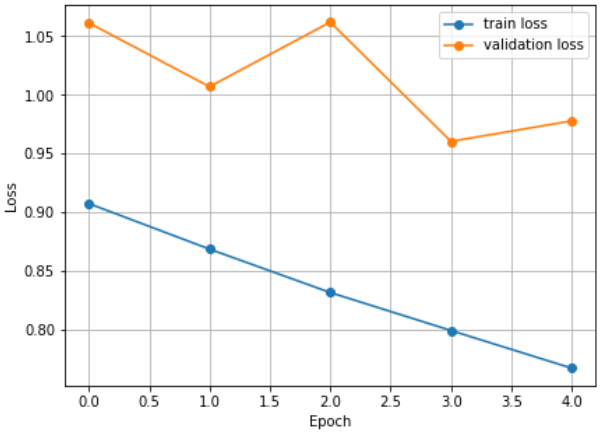
Here are the results, I have found in the RMSProp optimization function

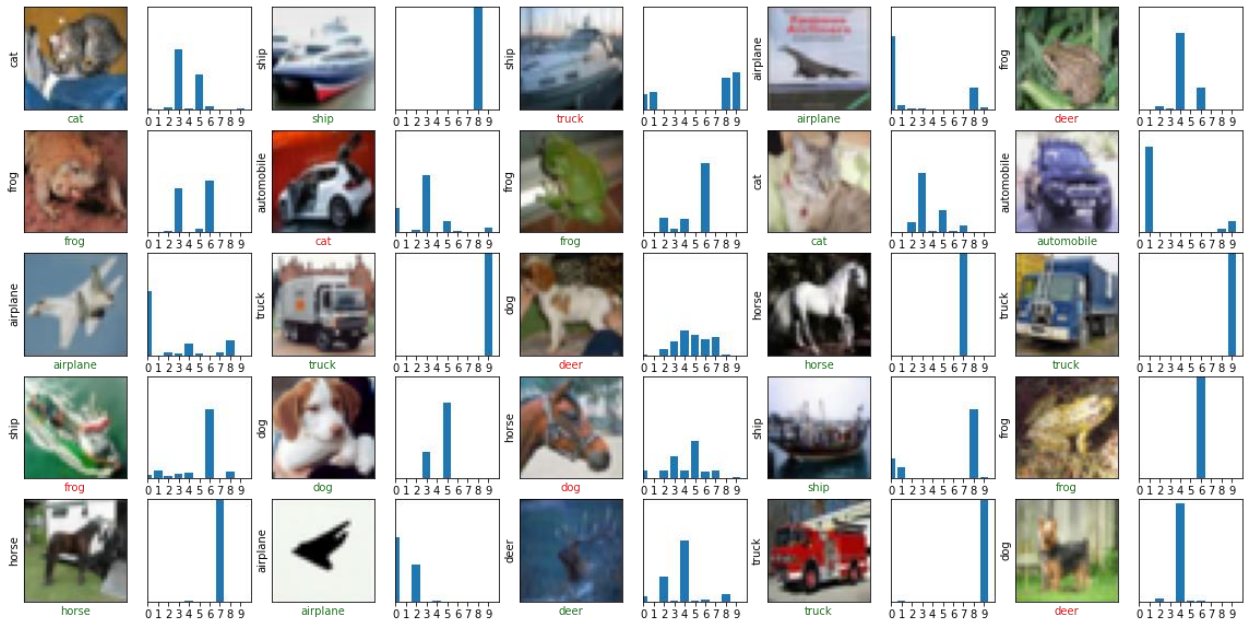
```
model.compile(
    optimizer='rmsprop',
    loss='sparse_categorical_crossentropy',
    metrics=['accuracy']
)

h = model.fit(x=X_train, y=Y_train, epochs=5, validation_split=0.2, batch_size=32)

Epoch 1/5
1250/1250 [==============================] - 61s 48ms/step - loss: 0.9074 - accuracy: 0.6795 - val_loss: 1.0614 - val_accuracy: 0.6425
Epoch 2/5
1250/1250 [==============================] - 59s 47ms/step - loss: 0.8685 - accuracy: 0.6952 - val_loss: 1.0070 - val_accuracy: 0.6521
Epoch 3/5
1250/1250 [==============================] - 60s 48ms/step - loss: 0.8315 - accuracy: 0.7095 - val_loss: 1.0621 - val_accuracy: 0.6480
Epoch 4/5
1250/1250 [==============================] - 60s 48ms/step - loss: 0.7991 - accuracy: 0.7214 - val_loss: 0.9603 - val_accuracy: 0.6754
Epoch 5/5
1250/1250 [==============================] - 59s 47ms/step - loss: 0.7670 - accuracy: 0.7323 - val_loss: 0.9779 - val_accuracy: 0.6705
```







SGD

Here are the results, I have found in the SGD optimization function

```
[ ]     loss='sparse_categorical_crossentropy',
        metrics=['accuracy']
    )

h = model.fit(x=X_train, y=Y_train, epochs=5, validation_split=0.2, batch_size=32)

Epoch 1/5
1250/1250 [==============================] - 59s 47ms/step - loss: 0.9181 - accuracy: 0.6768 - val_loss: 1.0145 - val_accuracy: 0.6498
Epoch 2/5
1250/1250 [==============================] - 59s 47ms/step - loss: 0.8764 - accuracy: 0.6925 - val_loss: 1.0184 - val_accuracy: 0.6477
Epoch 3/5
1250/1250 [==============================] - 59s 47ms/step - loss: 0.8557 - accuracy: 0.7007 - val_loss: 1.0249 - val_accuracy: 0.6485
Epoch 4/5
1250/1250 [==============================] - 60s 48ms/step - loss: 0.8409 - accuracy: 0.7023 - val_loss: 1.0051 - val_accuracy: 0.6500
Epoch 5/5
1250/1250 [==============================] - 60s 48ms/step - loss: 0.8249 - accuracy: 0.7101 - val_loss: 0.9920 - val_accuracy: 0.6537
```