In [22]:

```
##########Load the kaggle api token##########
!pip install --upgrade --force-reinstall --no-deps kaggle
!mkdir -p ~/.kaggle
!cp kaggle.json ~/.kaggle/
!chmod 600 /root/.kaggle/kaggle.json
!kaggle competitions download -c digit-recognizer
!kaggle datasets download -d keras/resnet50
!unzip digit-recognizer
!unzip resnet50
```

```
Processing /root/.cache/pip/wheels/a1/6a/26/d30b7499ff85a4a4593377a87ecf55f7d08af42f0de9b
60303/kaggle-1.5.12-cp37-none-any.whl
Installing collected packages: kaggle
  Found existing installation: kaggle 1.5.12
    Uninstalling kaggle-1.5.12:
      Successfully uninstalled kaggle-1.5.12
Successfully installed kaggle-1.5.12
digit-recognizer.zip: Skipping, found more recently modified local copy (use --force to f
orce download)
Downloading resnet50.zip to /content
 97% 169M/174M [00:07<00:00, 20.1MB/s]
100% 174M/174M [00:07<00:00, 23.1MB/s]
Archive:  digit-recognizer.zip
replace sample_submission.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: n
replace test.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: n
replace train.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: n
Archive:  resnet50.zip
  inflating: imagenet_class_index.json
  inflating: resnet50_weights_tf_dim_ordering_tf_kernels.h5
  inflating: resnet50_weights_tf_dim_ordering_tf_kernels_notop.h5
```

Libraries and Constants

In [47]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import tensorflow.keras as keras
from sklearn.model_selection import train_test_split
from tensorflow.python.keras.preprocessing.image import ImageDataGenerator
import time

from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten
from keras.applications.resnet50 import ResNet50


resnet_path = 'resnet50_weights_tf_dim_ordering_tf_kernels_notop.h5'
```

In [104]:

```
#CONSTANTS
NUM_CLASSES=10
VALID_SIZE=0.2
LEARNING_RATE=1e-5
BATCH_SIZE=64
EPOCHS=5
MOMENTUM=0.9
```

Data Loading and Preprocessing

In [65]:

```
training_data = pd.read_csv('train.csv')

def prepare_data_for_resnet50(data_to_transform):
    data = data_to_transform.copy().values
    data = data.reshape(-1, 28, 28) / 255
    data = X_rgb = np.stack([data, data, data], axis=-1)
    return data

y = training_data.pop('label').values
X = training_data

y = keras.utils.to_categorical(y, NUM_CLASSES)
X_rgb = prepare_data_for_resnet50(X)
```

Models

In [110]:

```
from tensorflow.keras.optimizers import SGD, RMSprop, Adam, Adadelta, Adagrad, Adamax, N
adam

def build_model(opt):
  model = Sequential()
  model.add(ResNet50(include_top=False, pooling='avg', weights='imagenet'))
  model.add(Dropout(0.50))
  model.add(Dense(NUM_CLASSES, activation='softmax'))
  model.compile(optimizer=opt, loss='categorical_crossentropy', metrics=['accuracy'])
  return model

optimizer_list = [SGD(learning_rate=LEARNING_RATE, momentum=MOMENTUM),
                  RMSprop(learning_rate=LEARNING_RATE, momentum=MOMENTUM),
                  Adam(learning_rate=LEARNING_RATE),
                  Adadelta(learning_rate=LEARNING_RATE),
                  Adagrad(learning_rate=LEARNING_RATE)
                  ]

optimizers = ['Momentum', 'RMSprop', 'Adam', 'Adadelta', 'Adagrad']
```

In [111]:

```
Results = pd.DataFrame()
for opt, str_opt in zip(optimizer_list, optimizers):
  print(str_opt)
  start = time.time()
  model = build_model(Adam(learning_rate=LEARNING_RATE))
  model.fit(X_rgb, y,
            epochs=EPOCHS,
            validation_split=VALID_SIZE,
            batch_size=BATCH_SIZE)
  end = time.time()
  acc = model.evaluate(X_rgb, y)

  loss = round(acc[0],4)
  acc = round(acc[1],4)
  conv_time = end-start
  Results = Results.append([str_opt,conv_time,loss,acc])
  print(str_opt,"Finished")
```

```
Momentum
Epoch 1/5
525/525 [==============================] - 117s 131ms/step - loss: 3.0365 - accuracy: 0.2
718 - val_loss: 24.0772 - val_accuracy: 0.0658
Epoch 2/5
525/525 [==============================] - 67s 128ms/step - loss: 0.9243 - accuracy: 0.71
52 - val_loss: 1.1606 - val_accuracy: 0.6256
Epoch 3/5
525/525 [==============================] - 66s 126ms/step - loss: 0.4609 - accuracy: 0.86
24 - val_loss: 0.3136 - val_accuracy: 0.9102
Epoch 4/5
```

```
525/525 [==============================] - 66s 126ms/step - loss: 0.2783 - accuracy: 0.91
72 - val_loss: 0.1501 - val_accuracy: 0.9557
Epoch 5/5
525/525 [==============================] - 67s 127ms/step - loss: 0.2112 - accuracy: 0.93
85 - val_loss: 0.1148 - val_accuracy: 0.9669
1313/1313 [==============================] - 33s 25ms/step - loss: 0.0738 - accuracy: 0.9
791
Momentum Finished
RMSprop
Epoch 1/5
525/525 [==============================] - 77s 133ms/step - loss: 3.2137 - accuracy: 0.24
95 - val_loss: 25.5099 - val_accuracy: 0.0962
Epoch 2/5
525/525 [==============================] - 66s 126ms/step - loss: 0.9796 - accuracy: 0.69
49 - val_loss: 1.4812 - val_accuracy: 0.5536
Epoch 3/5
525/525 [==============================] - 67s 127ms/step - loss: 0.4820 - accuracy: 0.85
54 - val_loss: 0.3459 - val_accuracy: 0.9005
Epoch 4/5
525/525 [==============================] - 66s 127ms/step - loss: 0.2922 - accuracy: 0.91
37 - val_loss: 0.1561 - val_accuracy: 0.9533
Epoch 5/5
525/525 [==============================] - 66s 126ms/step - loss: 0.2012 - accuracy: 0.94
22 - val_loss: 0.1210 - val_accuracy: 0.9645
1313/1313 [==============================] - 32s 25ms/step - loss: 0.0792 - accuracy: 0.9
767
RMSprop Finished
Adam
Epoch 1/5
525/525 [==============================] - 76s 131ms/step - loss: 3.1918 - accuracy: 0.25
47 - val_loss: 17.6264 - val_accuracy: 0.0874
Epoch 2/5
525/525 [==============================] - 66s 127ms/step - loss: 0.9497 - accuracy: 0.70
08 - val_loss: 1.3356 - val_accuracy: 0.5864
Epoch 3/5
525/525 [==============================] - 66s 127ms/step - loss: 0.4612 - accuracy: 0.86
20 - val_loss: 0.3150 - val_accuracy: 0.9094
Epoch 4/5
525/525 [==============================] - 66s 126ms/step - loss: 0.2853 - accuracy: 0.91
61 - val_loss: 0.1409 - val_accuracy: 0.9601
Epoch 5/5
525/525 [==============================] - 67s 128ms/step - loss: 0.1991 - accuracy: 0.94
17 - val_loss: 0.1107 - val_accuracy: 0.9681
1313/1313 [==============================] - 32s 24ms/step - loss: 0.0743 - accuracy: 0.9
787
Adam Finished
Adadelta
Epoch 1/5
525/525 [==============================] - 76s 130ms/step - loss: 3.3311 - accuracy: 0.24
41 - val_loss: 24.4572 - val_accuracy: 0.1077
Epoch 2/5
525/525 [==============================] - 66s 126ms/step - loss: 1.0437 - accuracy: 0.67
41 - val_loss: 1.3588 - val_accuracy: 0.5506
Epoch 3/5
525/525 [==============================] - 66s 126ms/step - loss: 0.5154 - accuracy: 0.84
35 - val_loss: 0.3982 - val_accuracy: 0.8835
Epoch 4/5
525/525 [==============================] - 66s 126ms/step - loss: 0.3086 - accuracy: 0.90
77 - val_loss: 0.1561 - val_accuracy: 0.9555
Epoch 5/5
525/525 [==============================] - 68s 129ms/step - loss: 0.2236 - accuracy: 0.93
44 - val_loss: 0.1178 - val_accuracy: 0.9675
1313/1313 [==============================] - 32s 24ms/step - loss: 0.0827 - accuracy: 0.9
767
Adadelta Finished
Adagrad
Epoch 1/5
525/525 [==============================] - 76s 131ms/step - loss: 3.3389 - accuracy: 0.23
38 - val_loss: 20.9112 - val_accuracy: 0.1112
Epoch 2/5
525/525 [==============================] - 66s 126ms/step - loss: 1.0520 - accuracy: 0.66
99 - val_loss: 1.2983 - val_accuracy: 0.5726
```

```
Epoch 3/5
525/525 [==============================] - 66s 126ms/step - loss: 0.4968 - accuracy: 0.85
22 - val_loss: 0.3205 - val_accuracy: 0.9082
Epoch 4/5
525/525 [==============================] - 66s 126ms/step - loss: 0.3018 - accuracy: 0.90
93 - val_loss: 0.1474 - val_accuracy: 0.9575
Epoch 5/5
525/525 [==============================] - 66s 127ms/step - loss: 0.2104 - accuracy: 0.93
79 - val_loss: 0.1165 - val_accuracy: 0.9665
1313/1313 [==============================] - 32s 24ms/step - loss: 0.0817 - accuracy: 0.9
766
Adagrad Finished
```

In [ ]:

```
```