```
import numpy as np

import keras.models

from scipy.misc import imread, imresize,imshow

import tensorflow as tf


from keras.models import Sequential

from keras.models import Sequential

from keras.layers import Dense, Dropout, Flatten

from keras.layers import Conv2D, MaxPooling2D


def init():

    num_classes = 10

    img_rows, img_cols = 28, 28

    input_shape = (img_rows, img_cols, 1)

    model = Sequential()

    model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=input_shape))

    model.add(Conv2D(64, (3, 3), activation='relu'))

    model.add(MaxPooling2D(pool_size=(2, 2)))

    model.add(Dropout(0.25))

    model.add(Flatten())

    model.add(Dense(128, activation='relu'))

    model.add(Dropout(0.5))

    model.add(Dense(num_classes, activation='softmax'))


    #load woeights into new model

    model.load_weights("weights.h5")

    print("Loaded Model from disk")


    #compile and evaluate loaded model

    model.compile(loss=keras.losses.categorical_crossentropy, optimizer=keras.optimizers.Adadelta(), metrics=['accuracy'])

    #loss,accuracy = model.evaluate(X_test,y_test)

    #print('loss:', loss)

    #print('accuracy:', accuracy)
```

```
graph = tf.get_default_graph()


return model,
```