project-fake-news-detection

April 2, 2024

[1]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id
=947318989803-6bn6qk8qdgf4n4g3pfee6491hc0brc4i.apps.googleusercontent.com&redire
ct_uri=urn%3aietf%3awg%3aoauth%3a2.0%3aoob&response_type=code&scope=email%20http
s%3a%2f%2fwww.googleapis.com%2fauth%2fdocs.test%20https%3a%2f%2fwww.googleapis.c
om%2fauth%2fdrive%20https%3a%2f%2fwww.googleapis.com%2fauth%2fdrive.photos.reado
nly%20https%3a%2f%2fwww.googleapis.com%2fauth%2fpeopleapi.readonly

Enter your authorization code:
··········
Mounted at /content/drive
```

[0]:
```
#root_path = 'gdrive/My Drive/your_project_folder/'  #change dir to your␣
 ↪project folder
root_path = '/content/drive/My Drive/Hackathon'
```

[0]:
```
!wget http://nlp.stanford.edu/data/glove.6B.zip
```

```
--2019-12-24 04:28:05--  http://nlp.stanford.edu/data/glove.6B.zip
Resolving nlp.stanford.edu (nlp.stanford.edu)… 171.64.67.140
Connecting to nlp.stanford.edu (nlp.stanford.edu)|171.64.67.140|:80…
connected.
HTTP request sent, awaiting response… 302 Found
Location: https://nlp.stanford.edu/data/glove.6B.zip [following]
--2019-12-24 04:28:06--  https://nlp.stanford.edu/data/glove.6B.zip
Connecting to nlp.stanford.edu (nlp.stanford.edu)|171.64.67.140|:443…
connected.
HTTP request sent, awaiting response… 301 Moved Permanently
Location: http://downloads.cs.stanford.edu/nlp/data/glove.6B.zip [following]
--2019-12-24 04:28:06--  http://downloads.cs.stanford.edu/nlp/data/glove.6B.zip
Resolving downloads.cs.stanford.edu (downloads.cs.stanford.edu)… 171.64.64.22
Connecting to downloads.cs.stanford.edu
(downloads.cs.stanford.edu)|171.64.64.22|:80… connected.
HTTP request sent, awaiting response… 200 OK
Length: 862182613 (822M) [application/zip]
Saving to: ‘glove.6B.zip’
```

```
glove.6B.zip        100%[===================>] 822.24M  1.98MB/s    in 6m 29s

2019-12-24 04:34:35 (2.11 MB/s) - ‘glove.6B.zip’ saved [862182613/862182613]
```

[0]:
```
!unzip glove*.zip
```

```
Archive:  glove.6B.zip
  inflating: glove.6B.50d.txt
  inflating: glove.6B.100d.txt
  inflating: glove.6B.200d.txt
  inflating: glove.6B.300d.txt
```

[0]:
```
# Now just move the glove.6B.100d.txt file directly from local folder to your␣
 ↪drive folder from table of content file.
```

[2]:
```
import keras
from tensorflow.python.client import device_lib

print(device_lib.list_local_devices())
import numpy as np
import pandas

import pandas as pd
from collections import defaultdict
import re


import sys
import os

from keras.preprocessing.text import Tokenizer
from keras.preprocessing.sequence import pad_sequences
from keras.utils.np_utils import to_categorical

from keras.layers import Embedding
from keras.layers import Dense, Input, Flatten
from keras.layers import Conv1D, MaxPooling1D, Embedding, Dropout
from keras.models import Model



#NlTK

from wordcloud import WordCloud,STOPWORDS
import re
```

```
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches


#For Model

from keras.layers import Input, Dense, Embedding, Conv2D, MaxPool2D
from keras.layers import Reshape, Flatten, Dropout, Concatenate
from keras.callbacks import ModelCheckpoint
from keras.optimizers import Adam
from keras.models import Model
from sklearn.model_selection import train_test_split

# Train Test Split
from sklearn.model_selection import train_test_split

MAX_SEQUENCE_LENGTH = 1000
MAX_NB_WORDS = 200000
EMBEDDING_DIM = 100
VALIDATION_SPLIT = 0.2
```

```
Using TensorFlow backend.
```

```
<IPython.core.display.HTML object>
```

```
[name: "/device:CPU:0"
device_type: "CPU"
memory_limit: 268435456
locality {
}
incarnation: 10186032227981011991
, name: "/device:XLA_CPU:0"
device_type: "XLA_CPU"
memory_limit: 17179869184
locality {
}
incarnation: 4301372376794876670
physical_device_desc: "device: XLA_CPU device"
, name: "/device:XLA_GPU:0"
device_type: "XLA_GPU"
memory_limit: 17179869184
locality {
}
incarnation: 1230959728684556493
physical_device_desc: "device: XLA_GPU device"
```

```
, name: "/device:GPU:0"
device_type: "GPU"
memory_limit: 15956161332
locality {
  bus_id: 1
  links {
  }
}
incarnation: 2871042171773516752
physical_device_desc: "device: 0, name: Tesla P100-PCIE-16GB, pci bus id:
0000:00:04.0, compute capability: 6.0"
]
[nltk_data] Downloading package stopwords to /root/nltk_data…
[nltk_data]   Unzipping corpora/stopwords.zip.
```

[3]:
```
df = pd.read_csv("/content/drive/My Drive/Hackathon/train.tsv",sep='\t') # Here␣
↪train.tsv is the same dataset just upload it to your respective google drive␣
↪and copy the path here
df.head()
```

[3]:

	Unnamed: 0	…	image_url
0	0	…	NaN
1	1	…	NaN
2	2	…	NaN
3	3	…	NaN
4	4	…	https://preview.redd.it/zkz1pt6tojd11.jpg?widt…

[5 rows x 17 columns]

[4]:
```
df1=df.iloc[:
↪,[7,8,10,12,13]]#['num_comments','score','upvote_ratio','clean_title','2_way_label'])
df1=df1.iloc[:,:]
df1.head()
```

[4]:

	num_comments	…	2_way_label
0	1	…	1
1	8	…	0
2	3	…	1
3	224	…	0
4	0	…	0

[5 rows x 5 columns]

[5]:
```
df1.iloc[:,:-1]=df1.iloc[:,:-1].replace(to_replace = np.nan, value ='')
```

```
/usr/local/lib/python3.6/dist-packages/pandas/core/indexing.py:494:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  self.obj[item] = s
```

[6]:
```
df1.iloc[:,-1]=df1.iloc[:,-1].replace(to_replace = np.nan, value =0)
```

```
/usr/local/lib/python3.6/dist-packages/pandas/core/indexing.py:494:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  self.obj[item] = s
```

[0]:
```
texts = []
labels = []

for i in range(len(list(df1.clean_title))):
    text = str(df1['clean_title'][i])
    texts.append(text)
    labels.append(df1['2_way_label'][i])

tokenizer = Tokenizer(num_words=MAX_NB_WORDS)
tokenizer.fit_on_texts(texts)
sequences = tokenizer.texts_to_sequences(texts)
word_index = tokenizer.word_index

#print('Found %s unique tokens.' % len(word_index))
```

[8]:
```
# Pad input sequences
data = pad_sequences(sequences, maxlen=MAX_SEQUENCE_LENGTH)
labels = to_categorical(np.asarray(labels,dtype='int32'),num_classes = 2)
print('Shape of data tensor:', data.shape)
print('Shape of label tensor:', labels.shape)
```

```
Shape of data tensor: (669564, 1000)
Shape of label tensor: (669564, 2)
```

[0]:
```
from keras.models import Sequential
from keras.layers.convolutional import Conv3D
from keras.layers.convolutional_recurrent import ConvLSTM2D
from keras.layers.normalization import BatchNormalization
import numpy as np
from matplotlib import pyplot as plt
from keras.layers import Dense, Embedding, LSTM, GRU
```

[0]:
```
GLOVE_DIR = ""
embeddings_index = {}
f = open(os.path.join(GLOVE_DIR, '/content/drive/My Drive/glove.6B.100d.txt'),␣
 ↪encoding="utf8") #The file that you had downloaded and moved from local␣
 ↪session to drive.
```

[11]:
```
for line in f:
    values = line.split()
    #print(values[1:])
    word = values[0]
    coefs = np.asarray(values[1:], dtype='float32')
    embeddings_index[word] = coefs
f.close()

print('Total %s word vectors in Glove.' % len(embeddings_index))

embedding_matrix = np.random.random((len(word_index) + 1, EMBEDDING_DIM))
for word, i in word_index.items():
    embedding_vector = embeddings_index.get(word)
    if embedding_vector is not None:
        # words not found in embedding index will be all-zeros.
        embedding_matrix[i] = embedding_vector

embedding_layer = Embedding(len(word_index) + 1,
                            EMBEDDING_DIM,
                            weights=[embedding_matrix],
                            input_length=MAX_SEQUENCE_LENGTH)
```

```
Total 400000 word vectors in Glove.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:66: The name tf.get_default_graph
is deprecated. Please use tf.compat.v1.get_default_graph instead.
```

[0]:
```
from keras.callbacks import ModelCheckpoint
from keras import layers

embedding_vecor_length = 32
modell = Sequential()
modell.add(embedding_layer)
modell.add(Dropout(0.2))
modell.add(Conv1D(filters=32, kernel_size=3, padding='same', activation='relu'))
modell.add(MaxPooling1D(pool_size=2))
modell.add(Conv1D(filters=64, kernel_size=4, padding='same',␣
 ↪activation='sigmoid'))
modell.add(MaxPooling1D(pool_size=2))
```

```
modell.add(Conv1D(filters=128, kernel_size=5, padding='same',␣
 ↪activation='relu'))
modell.add(MaxPooling1D(pool_size=2))
modell.add(LSTM(100,activation='relu',recurrent_activation='sigmoid',dropout=0.
 ↪2, recurrent_dropout=0.2))
modell.add(BatchNormalization())
modell.add(Dense(1024, activation='relu'))
modell.add(layers.Dropout(0.2))
modell.add(Dense(512,  activation='relu'))
modell.add(layers.Dropout(0.2))
modell.add(Dense(256, activation='relu'))
modell.add(layers.Dropout(0.2))
modell.add(Dense(128, activation='relu'))
modell.add(layers.Dropout(0.2))
modell.add(Dense(64, activation='relu'))
modell.add(layers.Dropout(0.2))
modell.add(Dense(2, activation='softmax'))

modell.compile(loss='binary_crossentropy', optimizer='adam',␣
 ↪metrics=['accuracy'])
print(modell.summary())

filepath = "/content/hackerthon1/model.h5" # Location to save yor model
checkpoint = ModelCheckpoint(filepath, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')
callbacks_list = [checkpoint]
modell.fit(data, labels,validation_split=0.05, epochs=9, batch_size=1024 ,␣
 ↪callbacks=callbacks_list)
```

[0]:
```
import os

base_dir = 'hackerthon1'
os.mkdir(base_dir)
```

[0]:
```
base_dir = 'hackerthon'
os.mkdir(base_dir)
```

[0]:
```
from keras.models import load_model
filepath = "/content/hackerthon1/model.h5"  # Location to get yor model
filepath1 = "/content/hackerthon/hackerthon1.h5" # Location to save yor model
checkpoint = ModelCheckpoint(filepath1, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')

new_model = load_model(filepath)
#checkpoint = ModelCheckpoint(filepath, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')
```

```
callbacks_list = [checkpoint]
#new_model.fit(x_train, y_train, epochs=5, batch_size=50,␣
 ↪callbacks=callbacks_list)


new_model.fit(data, labels,validation_split=0.05, epochs=9, batch_size=1024 ,␣
 ↪callbacks=callbacks_list)
```

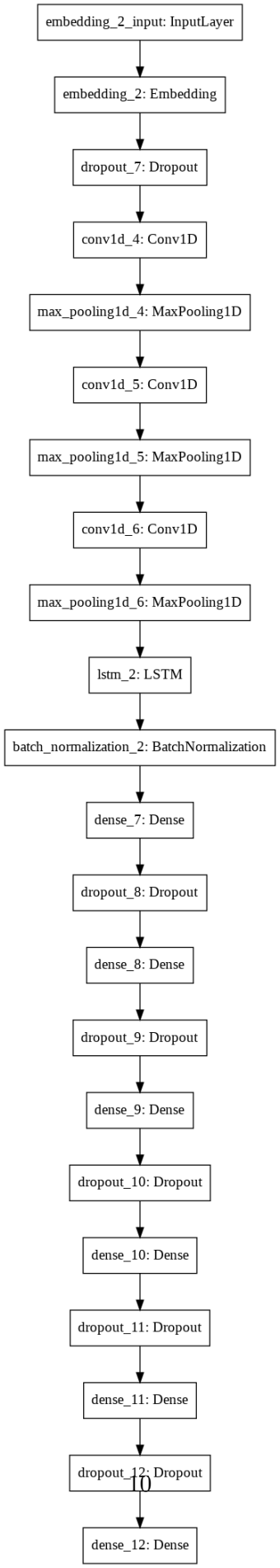
```
Train on 636085 samples, validate on 33479 samples
Epoch 1/9
636085/636085 [==============================] - 245s 386us/step - loss: 0.2404
- acc: 0.8991 - val_loss: 0.3768 - val_acc: 0.8351

Epoch 00001: loss improved from inf to 0.24035, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 2/9
636085/636085 [==============================] - 242s 380us/step - loss: 0.2301
- acc: 0.9033 - val_loss: 0.3854 - val_acc: 0.8373

Epoch 00002: loss improved from 0.24035 to 0.23010, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 3/9
636085/636085 [==============================] - 240s 377us/step - loss: 0.2214
- acc: 0.9078 - val_loss: 0.3849 - val_acc: 0.8342

Epoch 00003: loss improved from 0.23010 to 0.22140, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 4/9
636085/636085 [==============================] - 243s 382us/step - loss: 0.2131
- acc: 0.9111 - val_loss: 0.3867 - val_acc: 0.8360

Epoch 00004: loss improved from 0.22140 to 0.21307, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 5/9
636085/636085 [==============================] - 243s 382us/step - loss: 0.2048
- acc: 0.9148 - val_loss: 0.3918 - val_acc: 0.8410

Epoch 00005: loss improved from 0.21307 to 0.20480, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 6/9
636085/636085 [==============================] - 243s 382us/step - loss: 0.1975
- acc: 0.9179 - val_loss: 0.3992 - val_acc: 0.8380

Epoch 00006: loss improved from 0.20480 to 0.19745, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 7/9
636085/636085 [==============================] - 242s 380us/step - loss: 0.1907
```

```
 - acc: 0.9209 - val_loss: 0.4044 - val_acc: 0.8353

Epoch 00007: loss improved from 0.19745 to 0.19072, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 8/9
636085/636085 [==============================] - 252s 397us/step - loss: 0.1841
- acc: 0.9238 - val_loss: 0.4259 - val_acc: 0.8384

Epoch 00008: loss improved from 0.19072 to 0.18409, saving model to
/content/hackerthon/hackerthon1.h5
Epoch 9/9
636085/636085 [==============================] - 248s 389us/step - loss: 0.1781
- acc: 0.9263 - val_loss: 0.4223 - val_acc: 0.8384

Epoch 00009: loss improved from 0.18409 to 0.17806, saving model to
/content/hackerthon/hackerthon1.h5
```

[0]:
```
<keras.callbacks.History at 0x7f1888ed4400>
```

[0]:
```
from keras.utils import plot_model
plot_model(new_model, to_file='new_model.png')
```

[0]:


embedding_2_input: InputLayer
embedding_2: Embedding
dropout_7: Dropout
conv1d_4: Conv1D
max_pooling1d_4: MaxPooling1D
conv1d_5: Conv1D
max_pooling1d_5: MaxPooling1D
conv1d_6: Conv1D
max_pooling1d_6: MaxPooling1D
lstm_2: LSTM
batch_normalization_2: BatchNormalization
dense_7: Dense
dropout_8: Dropout
dense_8: Dense
dropout_9: Dropout
dense_9: Dense
dropout_10: Dropout
dense_10: Dense
dropout_11: Dropout
dense_11: Dense
dropout_12: Dropout
dense_12: Dense

[30]:
```
from keras.layers import Input, Dense, Embedding, Conv2D, MaxPool2D
from keras.layers import Reshape, Flatten, Dropout, Concatenate
from keras.callbacks import ModelCheckpoint
from keras.optimizers import Adam
from keras.models import Model
from sklearn.model_selection import train_test_split



sequence_length = 1000
vocabulary_size = 200000
embedding_dim = 100
filter_sizes = [3,4,5]
num_filters = 512
drop = 0.5

epochs = 4
batch_size = 30

# this returns a tensor
#print("Creating Model...")
inputs = Input(shape=(sequence_length,), dtype='int32')
#embedding = Embedding(input_dim=vocabulary_size, output_dim=embedding_dim,␣
 ↪input_length=sequence_length)(inputs)
reshape = Reshape((sequence_length,embedding_dim,1))(embedding_layer(inputs))
conv_0 = Conv2D(num_filters, kernel_size=(filter_sizes[0], embedding_dim),␣
 ↪padding='valid', kernel_initializer='normal', activation='relu')(reshape)
conv_1 = Conv2D(num_filters, kernel_size=(filter_sizes[1], embedding_dim),␣
 ↪padding='valid', kernel_initializer='normal', activation='relu')(reshape)
conv_2 = Conv2D(num_filters, kernel_size=(filter_sizes[2], embedding_dim),␣
 ↪padding='valid', kernel_initializer='normal', activation='relu')(reshape)

maxpool_0 = MaxPool2D(pool_size=(sequence_length - filter_sizes[0] + 1, 1),␣
 ↪strides=(1,1), padding='valid')(conv_0)
maxpool_1 = MaxPool2D(pool_size=(sequence_length - filter_sizes[1] + 1, 1),␣
 ↪strides=(1,1), padding='valid')(conv_1)
maxpool_2 = MaxPool2D(pool_size=(sequence_length - filter_sizes[2] + 1, 1),␣
 ↪strides=(1,1), padding='valid')(conv_2)

concatenated_tensor = Concatenate(axis=1)([maxpool_0, maxpool_1, maxpool_2])
flatten = Flatten()(concatenated_tensor)
dropout = Dropout(drop)(flatten)
output = Dense(units=2, activation='softmax')(dropout)
```

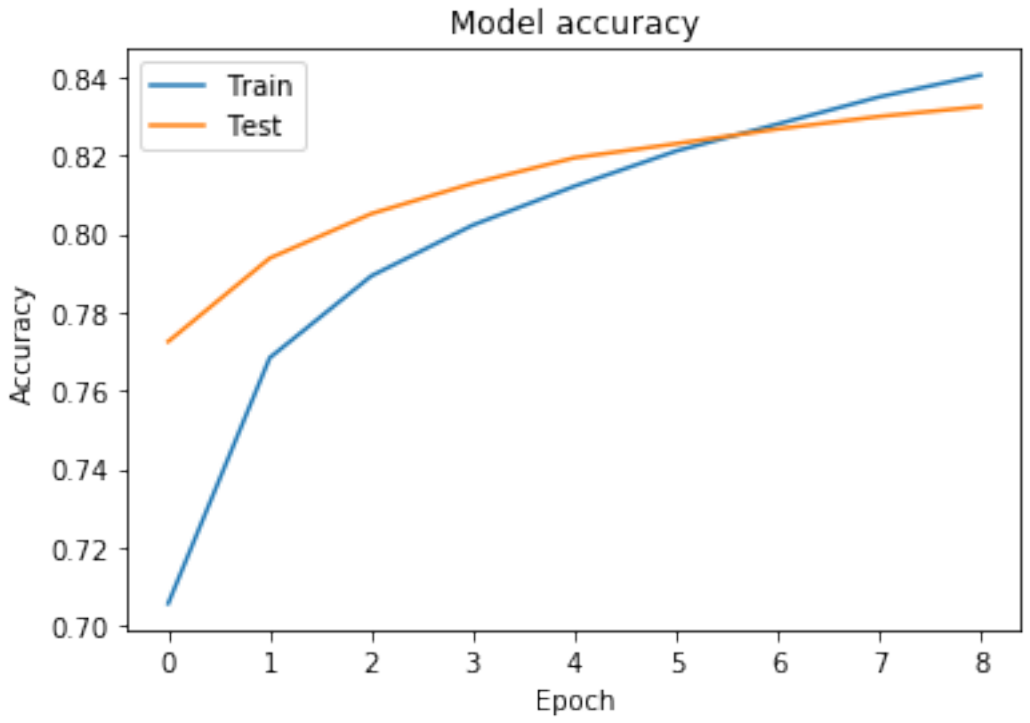
```
# this creates a model that includes
model = Model(inputs=inputs, outputs=output)



filepath = "/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5" # Location␣
 ↪to get yor model
checkpoint = ModelCheckpoint(filepath, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')
callbacks_list = [checkpoint]

#checkpoint = ModelCheckpoint('weights.{epoch:03d}-{val_acc:.4f}.hdf5',␣
 ↪monitor='val_acc', verbose=1, save_best_only=True, mode='auto')
adam = Adam(lr=1e-4, beta_1=0.9, beta_2=0.999, epsilon=1e-08, decay=0.0)

model.compile(optimizer=adam, loss='binary_crossentropy', metrics=['accuracy'])
print("Traning Model...")

hist=model.fit(data, labels,validation_split=0.05, epochs=9, batch_size=1024 ,␣
 ↪callbacks=callbacks_list)




#model.fit(X_train, y_train, batch_size=batch_size, epochs=epochs, verbose=1,␣
 ↪callbacks=[checkpoint], validation_data=(X_test, y_test))  # starts training
```

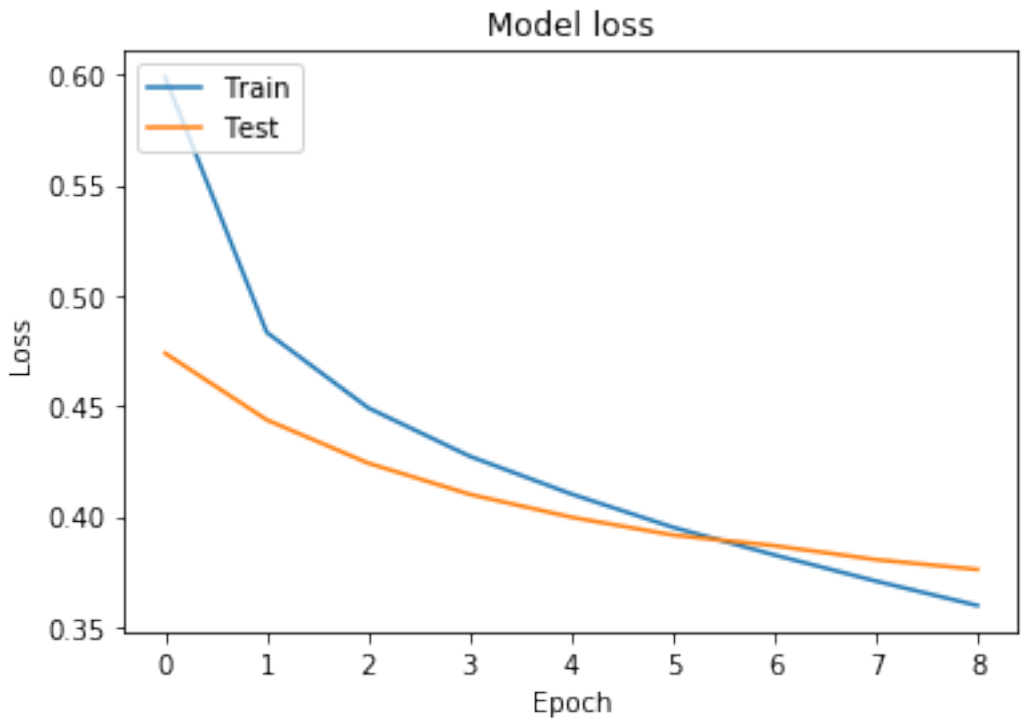
```
Traning Model…
Train on 636085 samples, validate on 33479 samples
Epoch 1/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.5987 -
acc: 0.7056 - val_loss: 0.4737 - val_acc: 0.7725

Epoch 00001: loss improved from inf to 0.59875, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 2/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.4832 -
acc: 0.7683 - val_loss: 0.4436 - val_acc: 0.7938

Epoch 00002: loss improved from 0.59875 to 0.48321, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 3/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.4490 -
acc: 0.7892 - val_loss: 0.4240 - val_acc: 0.8051

Epoch 00003: loss improved from 0.48321 to 0.44903, saving model to
```

```
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 4/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.4271 -
acc: 0.8022 - val_loss: 0.4099 - val_acc: 0.8129

Epoch 00004: loss improved from 0.44903 to 0.42712, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 5/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.4101 -
acc: 0.8121 - val_loss: 0.3995 - val_acc: 0.8194

Epoch 00005: loss improved from 0.42712 to 0.41005, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 6/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.3950 -
acc: 0.8211 - val_loss: 0.3915 - val_acc: 0.8230

Epoch 00006: loss improved from 0.41005 to 0.39497, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 7/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.3824 -
acc: 0.8280 - val_loss: 0.3866 - val_acc: 0.8268

Epoch 00007: loss improved from 0.39497 to 0.38245, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 8/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.3706 -
acc: 0.8349 - val_loss: 0.3803 - val_acc: 0.8300

Epoch 00008: loss improved from 0.38245 to 0.37061, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
Epoch 9/9
636085/636085 [==============================] - 992s 2ms/step - loss: 0.3596 -
acc: 0.8405 - val_loss: 0.3759 - val_acc: 0.8325

Epoch 00009: loss improved from 0.37061 to 0.35957, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNN.h5
```

[32]:
```python
import matplotlib.pyplot as plt

#history = model.fit(x, y, validation_split=0.25, epochs=50, batch_size=16,␣
 ↪verbose=1)

# Plot training & validation accuracy values
plt.plot(hist.history['acc'])
plt.plot(hist.history['val_acc'])
plt.title('Model accuracy')
```

```
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()

# Plot training & validation loss values
plt.plot(hist.history['loss'])
plt.plot(hist.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```





[14]:
```
from keras.models import load_model
filepath = "/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated.h5"  #␣
 ↪Location to get yor model
filepath1 = "/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated.h5" #␣
 ↪Location to save yor model
checkpoint = ModelCheckpoint(filepath1, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')

new_model = load_model(filepath)
#checkpoint = ModelCheckpoint(filepath, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')
callbacks_list = [checkpoint]
#new_model.fit(x_train, y_train, epochs=5, batch_size=50,␣
 ↪callbacks=callbacks_list)


hist=new_model.fit(data, labels,validation_split=0.05, epochs=3,␣
 ↪batch_size=1024 , callbacks=callbacks_list)
```

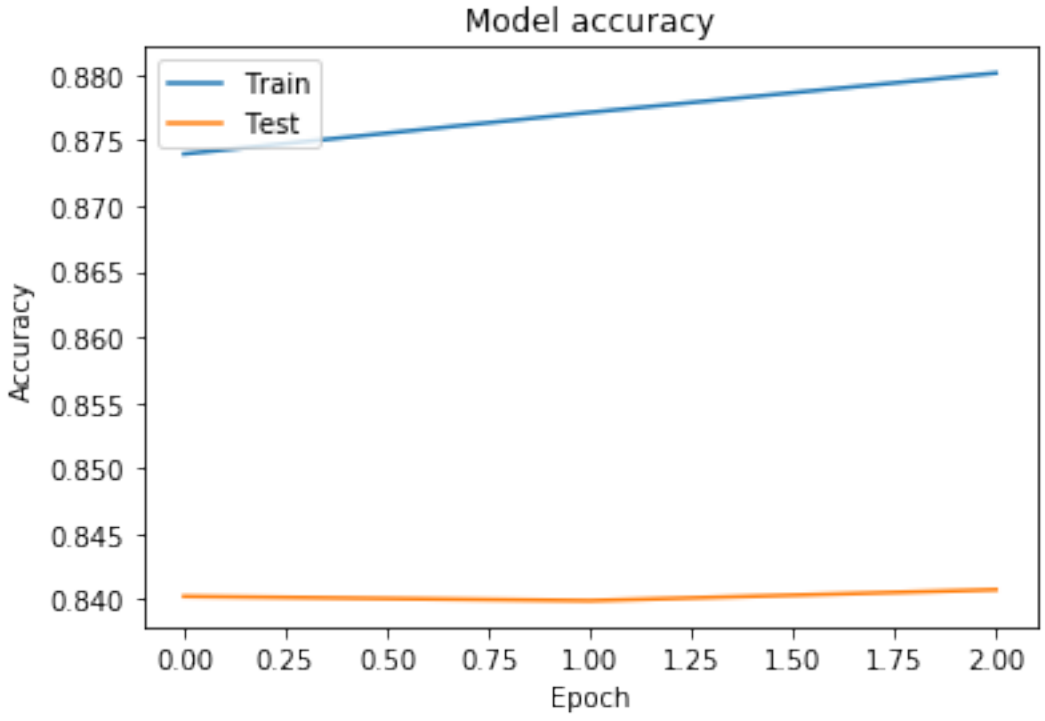
```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:541: The name tf.placeholder is
deprecated. Please use tf.compat.v1.placeholder instead.
```

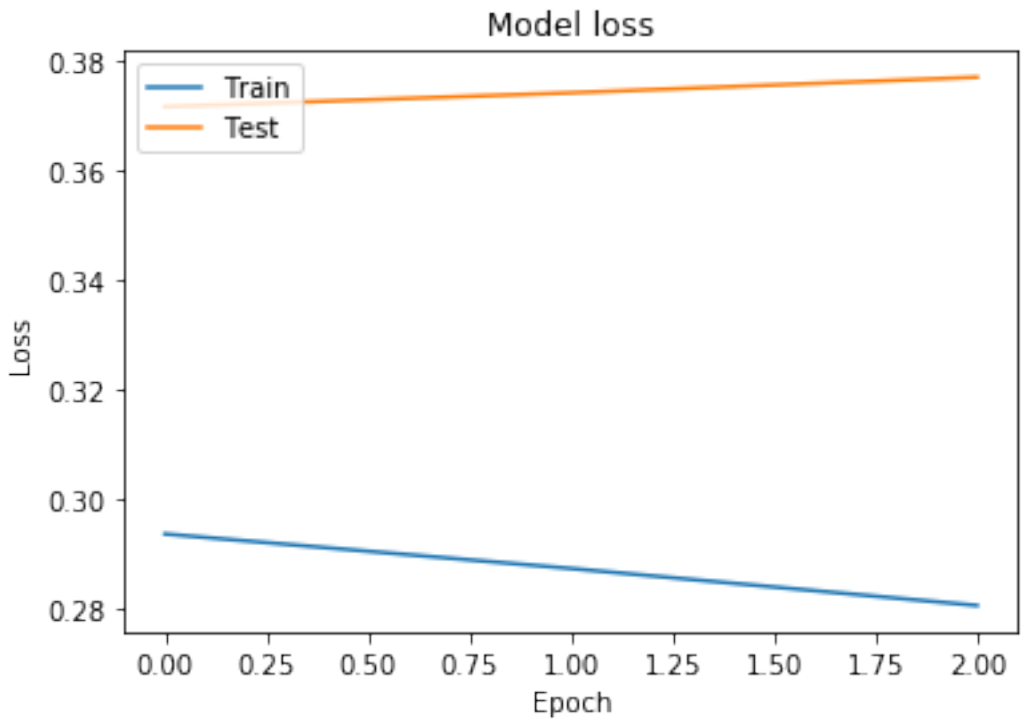
```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:4432: The name tf.random_uniform is
deprecated. Please use tf.random.uniform instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:4409: The name tf.random_normal is
deprecated. Please use tf.random.normal instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:4267: The name tf.nn.max_pool is
deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:148: The name
tf.placeholder_with_default is deprecated. Please use
tf.compat.v1.placeholder_with_default instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:3733: calling dropout (from
tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed
in a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 -
keep_prob`.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:190: The name
tf.get_default_session is deprecated. Please use
tf.compat.v1.get_default_session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:197: The name tf.ConfigProto is
deprecated. Please use tf.compat.v1.ConfigProto instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:203: The name tf.Session is
deprecated. Please use tf.compat.v1.Session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:207: The name tf.global_variables
is deprecated. Please use tf.compat.v1.global_variables instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:216: The name
tf.is_variable_initialized is deprecated. Please use
tf.compat.v1.is_variable_initialized instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:223: The name
```

```
tf.variables_initializer is deprecated. Please use
tf.compat.v1.variables_initializer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/optimizers.py:793: The name tf.train.Optimizer is deprecated.
Please use tf.compat.v1.train.Optimizer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:3657: The name tf.log is
deprecated. Please use tf.math.log instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/tensorflow_core/python/ops/nn_impl.py:183: where (from
tensorflow.python.ops.array_ops) is deprecated and will be removed in a future
version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:1033: The name tf.assign_add is
deprecated. Please use tf.compat.v1.assign_add instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:1020: The name tf.assign is
deprecated. Please use tf.compat.v1.assign instead.

Train on 636085 samples, validate on 33479 samples
Epoch 1/3
636085/636085 [==============================] - 1003s 2ms/step - loss: 0.2936 -
acc: 0.8739 - val_loss: 0.3716 - val_acc: 0.8402

Epoch 00001: loss improved from inf to 0.29361, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated.h5
Epoch 2/3
636085/636085 [==============================] - 993s 2ms/step - loss: 0.2873 -
acc: 0.8771 - val_loss: 0.3742 - val_acc: 0.8399

Epoch 00002: loss improved from 0.29361 to 0.28731, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated.h5
Epoch 3/3
636085/636085 [==============================] - 993s 2ms/step - loss: 0.2806 -
acc: 0.8801 - val_loss: 0.3770 - val_acc: 0.8407

Epoch 00003: loss improved from 0.28731 to 0.28056, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated.h5
```

[15]:
```
import matplotlib.pyplot as plt
```

```
#history = model.fit(x, y, validation_split=0.25, epochs=50, batch_size=16,␣
 ↪verbose=1)

# Plot training & validation accuracy values
plt.plot(hist.history['acc'])
plt.plot(hist.history['val_acc'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()

# Plot training & validation loss values
plt.plot(hist.history['loss'])
plt.plot(hist.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```





[12]:
```
from keras.models import load_model
filepath = "/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated.h5"  #␣
 ↪Location to get yor model
filepath1 = "/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5" #␣
 ↪Location to save yor model
checkpoint = ModelCheckpoint(filepath1, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')

new_model = load_model(filepath)
#checkpoint = ModelCheckpoint(filepath, monitor='loss', verbose=1,␣
 ↪save_best_only=True, mode='min')
callbacks_list = [checkpoint]
#new_model.fit(x_train, y_train, epochs=5, batch_size=50,␣
 ↪callbacks=callbacks_list)


hist=new_model.fit(data, labels,validation_split=0.05, epochs=10,␣
 ↪batch_size=1024 , callbacks=callbacks_list)
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:541: The name tf.placeholder is
deprecated. Please use tf.compat.v1.placeholder instead.
```

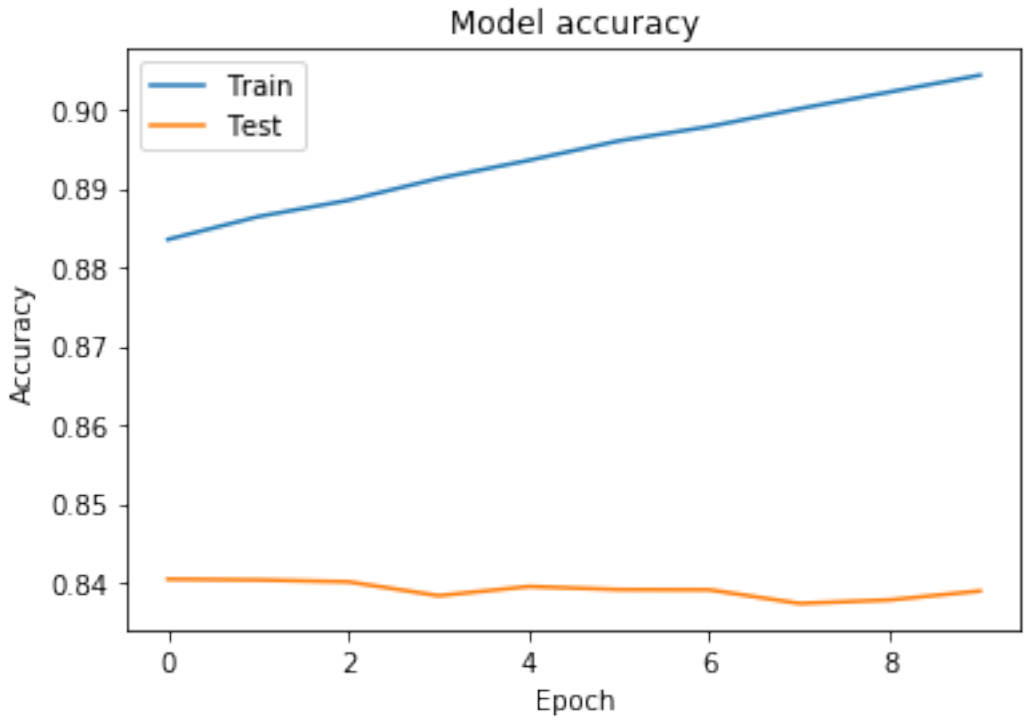
```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:4432: The name tf.random_uniform is
deprecated. Please use tf.random.uniform instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:4409: The name tf.random_normal is
deprecated. Please use tf.random.normal instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:4267: The name tf.nn.max_pool is
deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:148: The name
tf.placeholder_with_default is deprecated. Please use
tf.compat.v1.placeholder_with_default instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:3733: calling dropout (from
tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed
in a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 -
keep_prob`.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:190: The name
tf.get_default_session is deprecated. Please use
tf.compat.v1.get_default_session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:197: The name tf.ConfigProto is
deprecated. Please use tf.compat.v1.ConfigProto instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:203: The name tf.Session is
deprecated. Please use tf.compat.v1.Session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:207: The name tf.global_variables
is deprecated. Please use tf.compat.v1.global_variables instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:216: The name
tf.is_variable_initialized is deprecated. Please use
tf.compat.v1.is_variable_initialized instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:223: The name
```

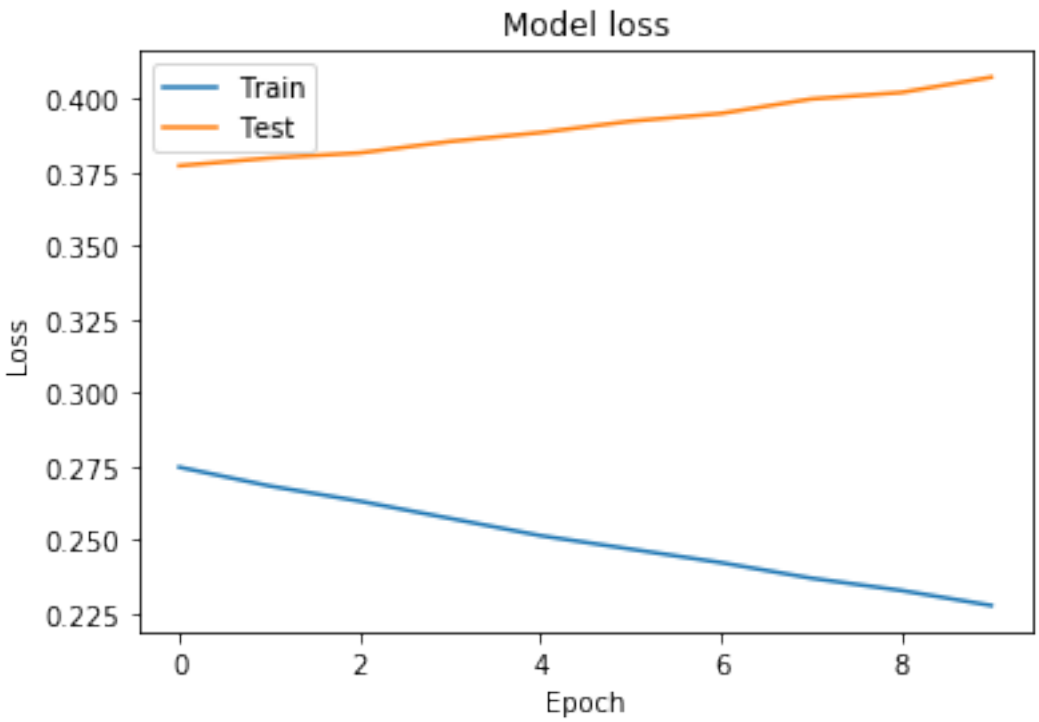
```
tf.variables_initializer is deprecated. Please use
tf.compat.v1.variables_initializer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/optimizers.py:793: The name tf.train.Optimizer is deprecated.
Please use tf.compat.v1.train.Optimizer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:3657: The name tf.log is
deprecated. Please use tf.math.log instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/tensorflow_core/python/ops/nn_impl.py:183: where (from
tensorflow.python.ops.array_ops) is deprecated and will be removed in a future
version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:1033: The name tf.assign_add is
deprecated. Please use tf.compat.v1.assign_add instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/keras/backend/tensorflow_backend.py:1020: The name tf.assign is
deprecated. Please use tf.compat.v1.assign instead.

Train on 636085 samples, validate on 33479 samples
Epoch 1/10
636085/636085 [==============================] - 1001s 2ms/step - loss: 0.2748 -
acc: 0.8835 - val_loss: 0.3772 - val_acc: 0.8404

Epoch 00001: loss improved from inf to 0.27476, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 2/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2684 -
acc: 0.8864 - val_loss: 0.3798 - val_acc: 0.8403

Epoch 00002: loss improved from 0.27476 to 0.26837, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 3/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2632 -
acc: 0.8885 - val_loss: 0.3816 - val_acc: 0.8401

Epoch 00003: loss improved from 0.26837 to 0.26319, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 4/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2573 -
acc: 0.8912 - val_loss: 0.3855 - val_acc: 0.8383
```

```
Epoch 00004: loss improved from 0.26319 to 0.25734, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 5/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2515 -
acc: 0.8935 - val_loss: 0.3885 - val_acc: 0.8395

Epoch 00005: loss improved from 0.25734 to 0.25149, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 6/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2469 -
acc: 0.8960 - val_loss: 0.3924 - val_acc: 0.8391

Epoch 00006: loss improved from 0.25149 to 0.24691, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 7/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2423 -
acc: 0.8978 - val_loss: 0.3950 - val_acc: 0.8391

Epoch 00007: loss improved from 0.24691 to 0.24231, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 8/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2370 -
acc: 0.9001 - val_loss: 0.3999 - val_acc: 0.8373

Epoch 00008: loss improved from 0.24231 to 0.23703, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 9/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2328 -
acc: 0.9022 - val_loss: 0.4021 - val_acc: 0.8378

Epoch 00009: loss improved from 0.23703 to 0.23283, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
Epoch 10/10
636085/636085 [==============================] - 992s 2ms/step - loss: 0.2277 -
acc: 0.9043 - val_loss: 0.4073 - val_acc: 0.8389

Epoch 00010: loss improved from 0.23283 to 0.22773, saving model to
/content/drive/My Drive/Hackathon/modelFakeNewsCNNUpdated1.h5
```

[13]:
```
import matplotlib.pyplot as plt

#history = model.fit(x, y, validation_split=0.25, epochs=50, batch_size=16,␣
 ↪verbose=1)

# Plot training & validation accuracy values
plt.plot(hist.history['acc'])
plt.plot(hist.history['val_acc'])
```

```
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()

# Plot training & validation loss values
plt.plot(hist.history['loss'])
plt.plot(hist.history['val_loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Test'], loc='upper left')
plt.show()
```




Model loss
Train
Test
0.400
0.375
0.350
0.325
0.300
0.275
0.250
0.225
Loss
0
2
4
6
8
Epoch

[0]: