In [1]:
```
# importing libraries

import warnings
warnings.filterwarnings("ignore")
import pandas as pd
import numpy as np
from keras.layers import Input, Embedding, LSTM, Dropout, BatchNormaliz
ation, Dense, concatenate, Flatten, Conv1D, MaxPool1D, LeakyReLU, ELU,
SpatialDropout1D, MaxPooling1D, GlobalAveragePooling1D, GlobalMaxPoolin
g1D
from keras.preprocessing.text import Tokenizer, one_hot
from keras.preprocessing.sequence import pad_sequences
from keras.models import Model, load_model
from keras import regularizers
from keras.optimizers import *
from keras.callbacks import ModelCheckpoint, EarlyStopping, TensorBoard
, ReduceLROnPlateau
from sklearn.feature_extraction.text import TfidfVectorizer, CountVecto
rizer
from sklearn.metrics import roc_auc_score
import tensorflow as tf
from tensorboardcolab import *
import matplotlib.pyplot as plt
%matplotlib inline
import re
from tqdm import tqdm
from sklearn.preprocessing import LabelEncoder
import seaborn as sns
import pickle
```

```
Using TensorFlow backend.
```

In [2]:
```
# https://medium.com/@rushic24/mounting-google-drive-in-google-colab-5e
cd1d3b735a
# https://towardsdatascience.com/3-ways-to-load-csv-files-into-colab-7c
14fcbdcb92#targetText=To%20start%2C%20log%20into%20your,Colab%20has%20i
```

```
t%20installed%20already).

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.googleuser
content.com&redirect_uri=urn%3Aietf%3Awg%3Aoauth%3A2.0%3Aoob&scope=emai
l%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fdocs.test%20https%3A%2F%2
Fwww.googleapis.com%2Fauth%2Fdrive%20https%3A%2F%2Fwww.googleapis.com%2
Fauth%2Fdrive.photos.readonly%20https%3A%2F%2Fwww.googleapis.com%2Faut
h%2Fpeopleapi.readonly&response_type=code

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

In [3]:
```
# reading datasets

project_data = pd.read_csv("/content/drive/My Drive/Data/preprocessed_d
ata.csv")
#pd.read_csv("preprocessed_data.csv")
project_data.head()
```

Out[3]:

	school_state	teacher_prefix	project_grade_category	teacher_number_of_previously_p
0	ca	mrs	grades_prek_2	53
1	ut	ms	grades_3_5	4

	school_state	teacher_prefix	project_grade_category	teacher_number_of_previously_p
2	ca	mrs	grades_prek_2	10
3	ga	mrs	grades_prek_2	2
4	wa	mrs	grades_3_5	2

In [4]:
```
print("Number of data points in train data", project_data.shape)
print('-'*50)
print("The attributes of data :", project_data.columns.values)
```

```
Number of data points in train data (109248, 9)
--------------------------------------------------
The attributes of data : ['school_state' 'teacher_prefix' 'project_grad
e_category'
 'teacher_number_of_previously_posted_projects' 'project_is_approved'
 'clean_categories' 'clean_subcategories' 'essay' 'price']
```

In [5]:
```
approved_project = project_data['project_is_approved'].values
project_data.drop(['project_is_approved'], axis=1, inplace=True)
project_data.head(1)
```

Out[5]:

	school_state	teacher_prefix	project_grade_category	teacher_number_of_previously_p
0	ca	mrs	grades_prek_2	53

In [0]:
```
# Data splitting

from sklearn.model_selection import train_test_split

# Splitting in train and test
X_train, X_test, y_train, y_test = train_test_split(project_data, appro
ved_project, test_size=0.33, stratify=approved_project)
```

In [18]:
```
tfidf_essay_vectorizer = TfidfVectorizer()
tfidf_essay_vectorizer.fit(X_train["essay"])
#X_Train_essay_tfidf = tfidf_essay_vectorizer.transform(X_train["essa
y"])
```

Out[18]:
```
TfidfVectorizer(analyzer='word', binary=False, decode_error='strict',
                dtype=<class 'numpy.float64'>, encoding='utf-8',
                input='content', lowercase=True, max_df=1.0, max_featur
es=None,
                min_df=1, ngram_range=(1, 1), norm='l2', preprocessor=N
one,
                smooth_idf=True, stop_words=None, strip_accents=None,
                sublinear_tf=False, token_pattern='(?u)\\b\\w\\w+\\b',
                tokenizer=None, use_idf=True, vocabulary=None)
```

In [22]:
```
tfidf_essay_vectorizer.idf_
```

Out[22]:
```
array([ 7.17702919,  5.86940787, 11.50776253, ..., 11.10229743,
       11.50776253, 11.50776253])
```

In [0]:
```
# we are converting a dictionary with word as a key, and the idf as a v
alue

dictionary = dict(zip(tfidf_essay_vectorizer.get_feature_names(), list(
tfidf_essay_vectorizer.idf_)))
tfidf_words = set(tfidf_essay_vectorizer.get_feature_names())
```

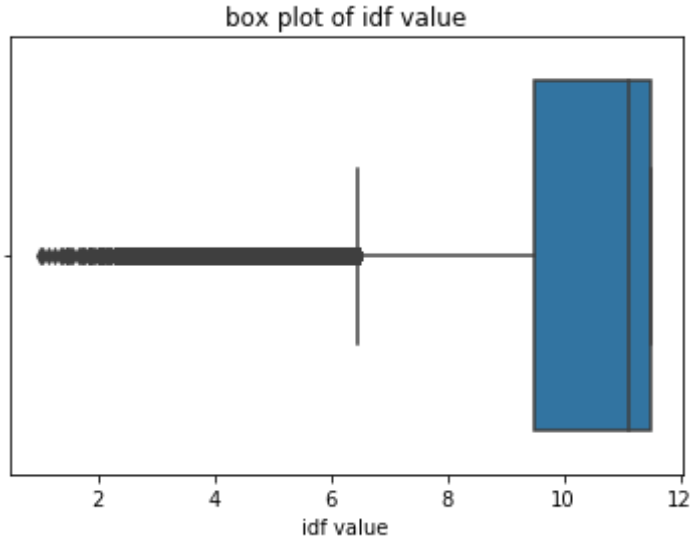
In [0]:
```
tfidf_dictinary_df = pd.DataFrame(list(dictionary.items()), columns=['W
ord', 'Value'])
```

In [0]:
```
tfidf_dictinary_df = tfidf_dictinary_df.sort_values(by ='Value' )
```

In [35]:
```
print(tfidf_dictinary_df["Value"].min())
print(tfidf_dictinary_df["Value"].max())
```

```
1.0080242390926728
11.50776253494305
```

In [38]:
```
sns.boxplot(x="Value", data=tfidf_dictinary_df )
plt.xlabel("idf value")
plt.title("box plot of idf value")
plt.show()
```



In [40]:
```
print("\nQuantiles:")
print(np.percentile(tfidf_dictinary_df['Value'],np.arange(0, 100, 10)))
```

```
Quantiles:
[ 1.00802424  7.48241084  8.94281318  9.89832462 10.5914718  11.1022974
3
 11.10229743 11.50776253 11.50776253 11.50776253]
```

In [41]:
```
final_tfidf = tfidf_dictinary_df[tfidf_dictinary_df["Value"] <=  np.per
centile(tfidf_dictinary_df['Value'],10)]
final_tfidf.shape
```

Out[41]:
```
(4833, 2)
```

In [42]:
```
#clearing the graph of tensorflow
tf.keras.backend.clear_session()

input_seq_total_text_data = Input(shape=(300,),name="input_seq_total_te
xt_data")
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/tensorflow_backend.py:541: The name tf.placeholder is deprecated.
```

```
Please use tf.compat.v1.placeholder instead.
```

In [0]:
```
final_word_list = final_tfidf["Word"].tolist()
```

In [47]:
```
# https://machinelearningmastery.com/use-word-embedding-layers-deep-learning-keras/

# prepare tokenizer
text_tokenizer = Tokenizer()
text_tokenizer.fit_on_texts(final_word_list)
vocab_size = len(text_tokenizer.word_index) + 1
vocab_size
```

Out[47]:
```
4834
```

In [0]:
```
# integer encode the data

encoded_essay_train = text_tokenizer.texts_to_sequences(X_train["essay"])
encoded_essay_test = text_tokenizer.texts_to_sequences(X_test["essay"])
```

In [49]:
```
# Padding data

padded_text_train = pad_sequences(encoded_essay_train, maxlen=300, padding='post', truncating='post')
padded_text_test = pad_sequences(encoded_essay_test, maxlen=300, padding='post', truncating='post')

print(padded_text_train.shape)
print(padded_text_test.shape)
```

```
(73196, 300)
(36052, 300)
```

In [0]:
```
f = open("/content/drive/My Drive/Data/glove_vectors", "rb")
glove_words = pickle.load(f)
```

In [51]:
```
# create a weight matrix for words in training docs

embedding_matrix = np.zeros((vocab_size, 300))
for word, i in text_tokenizer.word_index.items():

    embedding_vector = glove_words.get(word)
    if embedding_vector is not None:

        embedding_matrix[i] = embedding_vector

print(embedding_matrix.shape)
```

```
(4834, 300)
```

In [52]:
```
Emb_Txt_Data = Embedding(vocab_size, 300, weights = [embedding_matrix],
 input_length = 300, trainable=False)
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/tensorflow_backend.py:66: The name tf.get_default_graph is deprec
ated. Please use tf.compat.v1.get_default_graph instead.
```

In [53]:
```
Emb_Text_Data = (Emb_Txt_Data)(input_seq_total_text_data)
```

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

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/tensorflow_backend.py:190: The name tf.get_default_session is dep
recated. Please use tf.compat.v1.get_default_session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/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/ba
ckend/tensorflow_backend.py:203: The name tf.Session is deprecated. Ple
```

```
ase use tf.compat.v1.Session instead.

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

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/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/ba
ckend/tensorflow_backend.py:223: The name tf.variables_initializer is d
eprecated. Please use tf.compat.v1.variables_initializer instead.
```

In [54]:
```
Emb_Text_Data
```

Out[54]:
```
<tf.Tensor 'embedding_1/embedding_lookup/Identity:0' shape=(?, 300, 30
0) dtype=float32>
```

In [55]:
```
lstm = LSTM(64, recurrent_dropout=0.5, kernel_regularizer = regularizer
s.l2(0.001), return_sequences = True)(Emb_Text_Data)
```

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

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/tensorflow_backend.py:3733: calling dropout (from tensorflow.pyth
on.ops.nn_ops) with keep_prob is deprecated and will be removed in a fu
ture version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate =
1 - keep_prob`.
```

In [56]:
```
lstm
```

Out[56]:
```
<tf.Tensor 'lstm_1/transpose_1:0' shape=(?, ?, 64) dtype=float32>
```

In [0]:
```
flatten = Flatten()(lstm)
```

In [58]:
```
flatten
```

Out[58]:
```
<tf.Tensor 'flatten_1/Reshape:0' shape=(?, ?) dtype=float32>
```

In [0]:
```
# processing for school state

input_school_state = Input(shape=(50,),name="input_school_state")
```

In [60]:
```
unique_school_state  = X_train["school_state"].nunique()
print(unique_school_state)

Emb_State_Data = Embedding(unique_school_state, 300, input_length = 50)
(input_school_state)
```

```
51
```

In [0]:
```
flatten_1 = Flatten()(Emb_State_Data)
```

In [0]:
```
encoded_school_state_train = [one_hot(d, unique_school_state) for d in
X_train['school_state']]
encoded_school_state_test = [one_hot(d, unique_school_state) for d in X
_test['school_state']]
```

In [63]:
```
padded_school_state_train = pad_sequences(encoded_school_state_train, m
axlen=50, padding='post')
padded_school_state_test = pad_sequences(encoded_school_state_test, max
len=50, padding='post')

print(padded_school_state_train.shape)
print(padded_school_state_test.shape)
```

```
(73196, 50)
(36052, 50)
```

In [64]:
```
# processing for project grade category
```

```
input_project_grade_category = Input(shape=(50,),name="input_project_gr
ade_category")

unique_project_grade  = X_train["project_grade_category"].nunique()
print(unique_project_grade)

Emb_PGC_Data = Embedding(unique_project_grade, 300, input_length = 50)(
input_project_grade_category)
flatten_2 = Flatten()(Emb_PGC_Data)
```

```
4
```

In [65]:
```
encoded_project_grade_train = [one_hot(d, unique_project_grade) for d i
n X_train['project_grade_category']]
encoded_project_grade_test = [one_hot(d, unique_project_grade) for d in
 X_test['project_grade_category']]

padded_project_grade_train = pad_sequences(encoded_project_grade_train,
 maxlen=50, padding='post')
padded_project_grade_test = pad_sequences(encoded_project_grade_test, m
axlen=50, padding='post')

print(padded_project_grade_train.shape)
print(padded_project_grade_test.shape)
```

```
(73196, 50)
(36052, 50)
```

In [66]:
```
# processing for clean categories

input_clean_categories = Input(shape=(50,),name="input_clean_categorie
s")

unique_clean_categories  = X_train["clean_categories"].nunique()
print(unique_clean_categories)

Emb_clean_categories_Data = Embedding(unique_clean_categories, 300, inp
```

```
ut_length = 50)(input_clean_categories)
flatten_3 = Flatten()(Emb_clean_categories_Data)
```

```
51
```

In [67]:
```
encoded_clean_categories_train = [one_hot(d, unique_clean_categories) f
or d in X_train['clean_categories']]
encoded_clean_categories_test = [one_hot(d, unique_clean_categories) fo
r d in X_test['clean_categories']]

padded_clean_categories_train = pad_sequences(encoded_clean_categories_
train, maxlen=50, padding='post')
padded_clean_categories_test = pad_sequences(encoded_clean_categories_t
est, maxlen=50, padding='post')


print(padded_clean_categories_train.shape)
print(padded_clean_categories_test.shape)
```

```
(73196, 50)
(36052, 50)
```

In [68]:
```
# processing for clean subcategories

input_clean_subcategories = Input(shape=(50,),name="input_clean_subcate
gories")

unique_clean_subcategories  = X_train["clean_subcategories"].nunique()
print(unique_clean_subcategories)

Emb_clean_subcategories_Data = Embedding(unique_clean_subcategories, 30
0, input_length = 50)(input_clean_subcategories)
flatten_4 = Flatten()(Emb_clean_subcategories_Data)
```

```
395
```

In [69]:
```
encoded_clean_subcategories_train = [one_hot(d, unique_clean_subcategor
ies) for d in X_train['clean_subcategories']]
encoded_clean_subcategories_test = [one_hot(d, unique_clean_subcategori
```

```
es) for d in X_test['clean_subcategories']]

padded_clean_subcategories_train = pad_sequences(encoded_clean_subcateg
ories_train, maxlen=50, padding='post')
padded_clean_subcategories_test = pad_sequences(encoded_clean_subcatego
ries_test, maxlen=50, padding='post')

print(padded_clean_subcategories_train.shape)
print(padded_clean_subcategories_test.shape)
```

```
(73196, 50)
(36052, 50)
```

In [70]:
```
# processing for teacher prefix

input_teacher_prefix = Input(shape=(50,),name="input_teacher_prefix")

unique_teacher_prefix  = X_train["teacher_prefix"].nunique()
print(unique_teacher_prefix)

Emb_teacher_prefix_Data = Embedding(unique_teacher_prefix, 300, input_l
ength = 50)(input_teacher_prefix)
flatten_5 = Flatten()(Emb_teacher_prefix_Data)
```

```
5
```

In [71]:
```
encoded_teacher_prefix_train = [one_hot(d, unique_teacher_prefix) for d
 in X_train['teacher_prefix']]
encoded_teacher_prefix_test = [one_hot(d, unique_teacher_prefix) for d
in X_test['teacher_prefix']]

padded_teacher_prefix_train = pad_sequences(encoded_teacher_prefix_trai
n, maxlen=50, padding='post')
padded_teacher_prefix_test = pad_sequences(encoded_teacher_prefix_test,
 maxlen=50, padding='post')

print(padded_teacher_prefix_train.shape)
print(padded_teacher_prefix_test.shape)
```

```
(73196, 50)
(36052, 50)
```

In [72]:
```
# Processing numerical features

input_numerical_data = Input(shape=(2,),name="input_numerical_data")

Dense_for_rem_input = Dense(units=32,activation='relu',kernel_initializ
er='he_normal',name="Dense_for_rem_input")(input_numerical_data)
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/tensorflow_backend.py:4479: The name tf.truncated_normal is depre
cated. Please use tf.random.truncated_normal instead.
```

In [0]:
```
teacher_number_of_previously_posted_projects_train = X_train['teacher_n
umber_of_previously_posted_projects'].values.reshape(-1, 1)
price_train = X_train['price'].values.reshape(-1, 1)

teacher_number_of_previously_posted_projects_test = X_test['teacher_num
ber_of_previously_posted_projects'].values.reshape(-1, 1)
price_test = X_test['price'].values.reshape(-1, 1)

concat_numerical_train = np.concatenate((teacher_number_of_previously_p
osted_projects_train,price_train),axis=1)
concat_numerical_test = np.concatenate((teacher_number_of_previously_po
sted_projects_test,price_test),axis=1)
```

In [0]:
```
concatenate = concatenate(inputs=[flatten, flatten_1, flatten_2, flatte
n_3, flatten_4, flatten_5, input_numerical_data],name="concatenate")
```

In [75]:
```
Dense_layer1_after_concat = Dense(256,activation="relu", kernel_initial
izer="he_normal", kernel_regularizer=regularizers.l2(0.001))(concatenat
e)

dropout = Dropout(0.5)(Dense_layer1_after_concat)

Dense_layer2_after_concat = Dense(128,activation="relu", kernel_initial
```

```
izer="he_normal", kernel_regularizer=regularizers.l2(0.001))(dropout)

dropuout_1 = Dropout(0.3)(Dense_layer2_after_concat)

Dense_layer3_after_concat = Dense(64,activation='relu',kernel_initializ
er='he_normal', kernel_regularizer=regularizers.l2(0.001))(dropuout_1)

dropuout_2 = Dropout(0.7)(Dense_layer3_after_concat)

Dense_layer4_after_concat = Dense(32,activation='relu',kernel_initializ
er='he_normal', kernel_regularizer=regularizers.l2(0.001))(dropuout_2)

batchnormalization_1 = BatchNormalization()(Dense_layer4_after_concat)

output_layer_to_classify_with_softmax = Dense(2,activation='softmax',ke
rnel_initializer="he_normal",name="output")(batchnormalization_1)

model_1 = Model(inputs=[input_seq_total_text_data,input_school_state,in
put_project_grade_category,input_clean_categories,input_clean_subcatego
ries,input_teacher_prefix,input_numerical_data],outputs=[output_layer_t
o_classify_with_softmax])

model_1.summary()
```

```
WARNING:tensorflow:Large dropout rate: 0.7 (>0.5). In TensorFlow 2.x, d
ropout() uses dropout rate instead of keep_prob. Please ensure that thi
s is intended.
Model: "model_1"
__________________________________________________________________________
_____
Layer (type)                    Output Shape         Param #     Connec
ted to
==========================================================================
=============================
input_seq_total_text_data (Inpu (None, 300)          0


__________________________________________________________________________
_____
embedding_1 (Embedding)         (None, 300, 300)     1450200     input_
seq_total_text_data[0][0]
```

```
__________________________________________________________________________
_____________________________
input_school_state (InputLayer) (None, 50)           0

__________________________________________________________________________
_____________________________
input_project_grade_category (I (None, 50)           0

__________________________________________________________________________
_____________________________
input_clean_categories (InputLa (None, 50)           0

__________________________________________________________________________
_____________________________
input_clean_subcategories (Inpu (None, 50)           0

__________________________________________________________________________
_____________________________
input_teacher_prefix (InputLaye (None, 50)           0

__________________________________________________________________________
_____________________________
lstm_1 (LSTM)                   (None, 300, 64)      93440       embedd
ing_1[0][0]
__________________________________________________________________________
_____________________________
embedding_2 (Embedding)         (None, 50, 300)      15300       input_
school_state[0][0]
__________________________________________________________________________
_____________________________
embedding_3 (Embedding)         (None, 50, 300)      1200        input_
project_grade_category[0][0
__________________________________________________________________________
_____________________________
embedding_4 (Embedding)         (None, 50, 300)      15300       input_
clean_categories[0][0]
__________________________________________________________________________
_____________________________
embedding_5 (Embedding)         (None, 50, 300)      118500      input_
clean_subcategories[0][0]
```

```
__________________________________________________________________________________________________
__________________________________
embedding_6 (Embedding)         (None, 50, 300)      1500        input_
teacher_prefix[0][0]
__________________________________________________________________________________________________
__________________________________
flatten_1 (Flatten)             (None, 19200)        0           lstm_1
[0][0]
__________________________________________________________________________________________________
__________________________________
flatten_2 (Flatten)             (None, 15000)        0           embedd
ing_2[0][0]
__________________________________________________________________________________________________
__________________________________
flatten_3 (Flatten)             (None, 15000)        0           embedd
ing_3[0][0]
__________________________________________________________________________________________________
__________________________________
flatten_4 (Flatten)             (None, 15000)        0           embedd
ing_4[0][0]
__________________________________________________________________________________________________
__________________________________
flatten_5 (Flatten)             (None, 15000)        0           embedd
ing_5[0][0]
__________________________________________________________________________________________________
__________________________________
flatten_6 (Flatten)             (None, 15000)        0           embedd
ing_6[0][0]
__________________________________________________________________________________________________
__________________________________
input_numerical_data (InputLaye (None, 2)            0

__________________________________________________________________________________________________
__________________________________
concatenate (Concatenate)       (None, 94202)        0           flatte
n_1[0][0]
                                                                 flatte
n_2[0][0]
                                                                 flatte
```

```
n_3[0][0]

                                                                flatte
n_4[0][0]
                                                                flatte
n_5[0][0]
                                                                flatte
n_6[0][0]
                                                                input_
numerical_data[0][0]
__________________________________________________________________________
_____________________________
dense_1 (Dense)                 (None, 256)           24115968    concat
enate[0][0]
__________________________________________________________________________
_____________________________
dropout_1 (Dropout)             (None, 256)           0           dense_
1[0][0]
__________________________________________________________________________
_____________________________
dense_2 (Dense)                 (None, 128)           32896       dropou
t_1[0][0]
__________________________________________________________________________
_____________________________
dropout_2 (Dropout)             (None, 128)           0           dense_
2[0][0]
__________________________________________________________________________
_____________________________
dense_3 (Dense)                 (None, 64)            8256        dropou
t_2[0][0]
__________________________________________________________________________
_____________________________
dropout_3 (Dropout)             (None, 64)            0           dense_
3[0][0]
__________________________________________________________________________
_____________________________
dense_4 (Dense)                 (None, 32)            2080        dropou
t_3[0][0]
__________________________________________________________________________
```

```
_____________________________
batch_normalization_1 (BatchNor (None, 32)           128         dense_

4[0][0]
_____________________________
_____________________________
output (Dense)                  (None, 2)            66          batch_
normalization_1[0][0]
==================================================================
==========================
Total params: 25,854,834
Trainable params: 24,404,570
Non-trainable params: 1,450,264
_____________________________
_____________________________
```

In [76]:
```
# https://github.com/taomanwai/tensorboardcolab/blob/master/README.md

tbc=TensorBoardColab()
```

```
Wait for 8 seconds...
TensorBoard link:
https://6faa743d.ngrok.io
```

In [0]:
```
# https://machinelearningmastery.com/check-point-deep-learning-models-keras/
# https://machinelearningmastery.com/how-to-stop-training-deep-neural-networks-at-the-right-time-using-early-stopping/
# https://medium.com/singlestone/keras-callbacks-monitor-and-improve-your-deep-learning-205a8a27e91c
# https://www.tensorflow.org/tensorboard/get_started
# https://keras.rstudio.com/reference/callback_tensorboard.html
# https://colab.research.google.com/drive/1afN2SALDooZIHbBGmWZMT6cZ8ccVElWk#scrollTo=4pxUfiLhbS4Y&forceEdit=true&sandboxMode=true

#tensorboard_model_1 = TensorBoard(log_dir='./log', histogram_freq=1, write_graph=True, write_grads=True, batch_size=512, write_images=True)

#callbacks_1 = [tensorboard_model_1]
```

In [0]:
```
model_1_train_data = [padded_text_train,padded_school_state_train,padde
d_project_grade_train,padded_clean_categories_train,padded_clean_subcat
egories_train,padded_teacher_prefix_train,concat_numerical_train]

model_1_test_data = [padded_text_test,padded_school_state_test,padded_p
roject_grade_test,padded_clean_categories_test,padded_clean_subcategori
es_test,padded_teacher_prefix_test,concat_numerical_test]
```

In [0]:
```
from keras.utils import np_utils

Y_train = np_utils.to_categorical(y_train, 2)
Y_test = np_utils.to_categorical(y_test, 2)
```

In [0]:
```
# https://stackoverflow.com/questions/41032551/how-to-compute-receiving
-operating-characteristic-roc-and-auc-in-keras

def auroc(y_true, y_pred):
    return tf.py_func(roc_auc_score, (y_true, y_pred), tf.double)
```

In [81]:
```
model_1.compile(optimizer='adam', loss='categorical_crossentropy', metr
ics=[auroc])
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/op
timizers.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/ba
ckend/tensorflow_backend.py:3576: The name tf.log is deprecated. Please
use tf.math.log instead.

WARNING:tensorflow:From <ipython-input-80-2303c0155baf>:3: py_func (fro
m tensorflow.python.ops.script_ops) is deprecated and will be removed i
n a future version.
Instructions for updating:
tf.py_func is deprecated in TF V2. Instead, there are two
    options available in V2.
    - tf.py_function takes a python function which manipulates tf eager
    tensors instead of numpy arrays. It's easy to convert a tf eager te
nsor to
```

```
    an ndarray (just call tensor.numpy()) but having access to eager te
nsors
    means `tf.py_function`s can use accelerators such as GPUs as well a
s
    being differentiable using a gradient tape.
    - tf.numpy_function maintains the semantics of the deprecated tf.py
_func
    (it is not differentiable, and manipulates numpy arrays). It drops
the
    stateful argument making all functions stateful.
```

In [82]:
```
history = model_1.fit(model_1_train_data, Y_train, batch_size=512, epoc
hs=20, verbose=1, validation_data=(model_1_test_data, Y_test), callback
s=[TensorBoardColabCallback(tbc)])
```

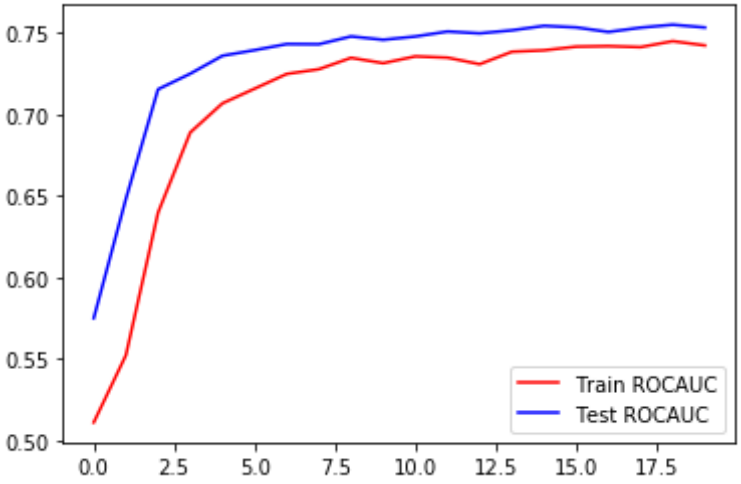
```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorfl
ow_core/python/ops/math_grad.py:1424: where (from tensorflow.python.op
s.array_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ba
ckend/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/ba
ckend/tensorflow_backend.py:1020: The name tf.assign is deprecated. Ple
ase use tf.compat.v1.assign instead.

Train on 73196 samples, validate on 36052 samples
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorbo
ardcolab/core.py:49: The name tf.summary.FileWriter is deprecated. Plea
se use tf.compat.v1.summary.FileWriter instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/ca
llbacks.py:1122: The name tf.summary.merge_all is deprecated. Please us
e tf.compat.v1.summary.merge_all instead.

Epoch 1/20
```

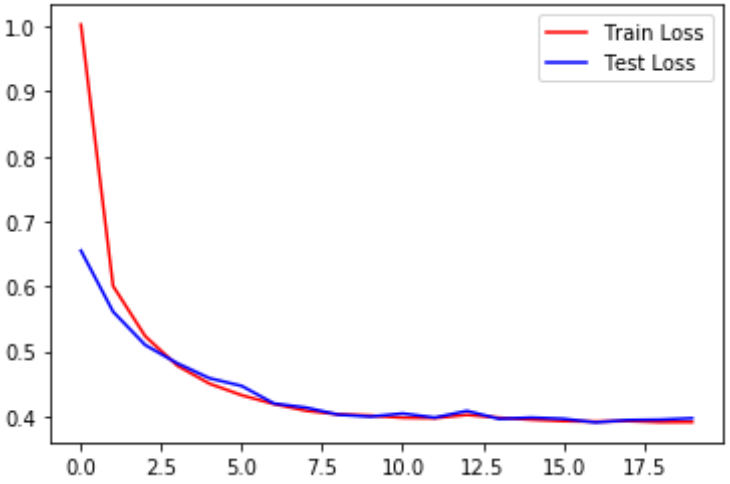
```
73196/73196 [==============================] - 112s 2ms/step - loss: 1.
0029 - auroc: 0.5110 - val_loss: 0.6552 - val_auroc: 0.5750
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorbo
ardcolab/callbacks.py:51: The name tf.Summary is deprecated. Please use
tf.compat.v1.Summary instead.

Epoch 2/20
73196/73196 [==============================] - 112s 2ms/step - loss: 0.
6007 - auroc: 0.5522 - val_loss: 0.5611 - val_auroc: 0.6482
Epoch 3/20
73196/73196 [==============================] - 112s 2ms/step - loss: 0.
5233 - auroc: 0.6398 - val_loss: 0.5098 - val_auroc: 0.7154
Epoch 4/20
73196/73196 [==============================] - 113s 2ms/step - loss: 0.
4779 - auroc: 0.6889 - val_loss: 0.4817 - val_auroc: 0.7250
Epoch 5/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4503 - auroc: 0.7068 - val_loss: 0.4587 - val_auroc: 0.7360
Epoch 6/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4326 - auroc: 0.7158 - val_loss: 0.4469 - val_auroc: 0.7395
Epoch 7/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4191 - auroc: 0.7249 - val_loss: 0.4198 - val_auroc: 0.7432
Epoch 8/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4086 - auroc: 0.7278 - val_loss: 0.4135 - val_auroc: 0.7431
Epoch 9/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4031 - auroc: 0.7348 - val_loss: 0.4026 - val_auroc: 0.7479
Epoch 10/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4013 - auroc: 0.7315 - val_loss: 0.3997 - val_auroc: 0.7458
Epoch 11/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
3980 - auroc: 0.7357 - val_loss: 0.4045 - val_auroc: 0.7479
Epoch 12/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
3973 - auroc: 0.7349 - val_loss: 0.3982 - val_auroc: 0.7509
```

```
Epoch 13/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
4024 - auroc: 0.7309 - val_loss: 0.4084 - val_auroc: 0.7498
Epoch 14/20
73196/73196 [==============================] - 111s 2ms/step - loss: 0.
3979 - auroc: 0.7385 - val_loss: 0.3961 - val_auroc: 0.7517
Epoch 15/20
73196/73196 [==============================] - 112s 2ms/step - loss: 0.
3950 - auroc: 0.7394 - val_loss: 0.3982 - val_auroc: 0.7544
Epoch 16/20
73196/73196 [==============================] - 110s 2ms/step - loss: 0.
3928 - auroc: 0.7416 - val_loss: 0.3963 - val_auroc: 0.7534
Epoch 17/20
73196/73196 [==============================] - 110s 1ms/step - loss: 0.
3920 - auroc: 0.7420 - val_loss: 0.3907 - val_auroc: 0.7506
Epoch 18/20
73196/73196 [==============================] - 110s 2ms/step - loss: 0.
3929 - auroc: 0.7413 - val_loss: 0.3944 - val_auroc: 0.7533
Epoch 19/20
73196/73196 [==============================] - 110s 2ms/step - loss: 0.
3911 - auroc: 0.7450 - val_loss: 0.3950 - val_auroc: 0.7551
Epoch 20/20
73196/73196 [==============================] - 110s 2ms/step - loss: 0.
3913 - auroc: 0.7424 - val_loss: 0.3971 - val_auroc: 0.7534
```

In [83]:

```
plt.plot(history.history['auroc'], 'r')
plt.plot(history.history['val_auroc'], 'b')
plt.legend({'Train ROCAUC': 'r', 'Test ROCAUC':'b'})
plt.show()


plt.plot(history.history['loss'], 'r')
plt.plot(history.history['val_loss'], 'b')
plt.legend({'Train Loss': 'r', 'Test Loss':'b'})
plt.show()
```





Model ROCAUC value is 0.7534