```
import pandas as pd
import numpy as np
from tqdm import tqdm
from tqdm.notebook import tqdm_notebook
tqdm_notebook.pandas()
import warnings
warnings.filterwarnings('ignore')
import tensorflow as tf
from tqdm import tqdm
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.
```

```
! cp '/content/drive/My Drive/tweet-sentiment-extraction/preprocessed_train.csv' .
! cp '/content/drive/My Drive/tweet-sentiment-extraction/preprocessed_test.csv' .
#! cp '/content/drive/My Drive/tweet-sentiment-extraction/train.csv' .
#! cp '/content/drive/My Drive/tweet-sentiment-extraction/test.csv' .
```

```
train_df = pd.read_csv('preprocessed_train.csv')
test_df = pd.read_csv('preprocessed_test.csv')
```

```
train_df.shape,test_df.shape
```

```
((27469, 7), (3534, 3))
```

```
train_df.sample(5)
```

	textID	text	selected_text	sentiment	misspelled	start_indices	e
21780	2785cfe292	i have had the william shatner version of rock...	curse distracting	negative	No	18	
1949	b1626e8f40	wooo what a fight goooo	i am with you	positive	No	6	

```
test_df.sample(5)
```

	textID	text	sentiment
238	044f146e5c	had an amazing night with my favorite lady friend	positive
2696	e9c05ba1f9	I will be sure to	neutral

```
#train_df[train_df.end_indices<train_df.start_indices]
#train_df.loc[6393,'selected_text'] = 'amay the be with'
#train_df.loc[13668,'selected_text'] = 'utter curse these'
train_df.shape
```

```
(27469, 7)
```

```
train_df.head()
```

	textID	text	selected_text	sentiment	misspelled	start_indices	end_i
0	cb774db0d1	i would have responded if i were going	i would have responded if i were going	neutral	No	0	
1	549e992a42	sooo sad i will miss you here	sooo sad	negative	No	0	

```
!pip install transformers
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.6/dist-package
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: dataclasses; python_version < "3.7" in /usr/local/lib
Requirement already satisfied: numpy in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.6/dist-pa
Requirement already satisfied: filelock in /usr/local/lib/python3.6/dist-packages (f
Requirement already satisfied: tokenizers==0.8.1.rc2 in /usr/local/lib/python3.6/dis
Requirement already satisfied: sacremoses in /usr/local/lib/python3.6/dist-packages
Requirement already satisfied: packaging in /usr/local/lib/python3.6/dist-packages (
Requirement already satisfied: sentencepiece!=0.1.92 in /usr/local/lib/python3.6/dis
Requirement already satisfied: requests in /usr/local/lib/python3.6/dist-packages (f
Requirement already satisfied: joblib in /usr/local/lib/python3.6/dist-packages (fro
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from s
Requirement already satisfied: click in /usr/local/lib/python3.6/dist-packages (from
Requirement already satisfied: pyparsing>=2.0.2 in /usr/local/lib/python3.6/dist-pac
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.6/dist-pa
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.6/dist-p
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.6/dist-package
```

```
from transformers import RobertaTokenizer
tokenizer = RobertaTokenizer.from_pretrained('roberta-base',add_prefix_space=True)
tokenizer.encode(' hello world'),tokenizer.encode('hello world')
```

```
([0, 20760, 232, 2], [0, 20760, 232, 2])
```

```
tokenizer.encode('positive'),tokenizer.encode('negative'),tokenizer.encode('neutral')
```

```
tokenizer.encode('positive'),tokenizer.encode('negative'),tokenizer.encode('neutral')
```

```
([0, 1313, 2], [0, 2430, 2], [0, 7974, 2])
```

```
tokenizer.decode([0,1313,2])
```

```
'<s> positive</s>'
```

```
train_df = train_df[['text','selected_text','sentiment']]
train_df.head()
```

	text	selected_text	sentiment
0	i would have responded if i were going	i would have responded if i were going	neutral
1	sooo sad i will miss you here in san diego	sooo sad	negative
2	my boss is bullying me	bullying me	negative
3	what interview leave me alone	leave me alone	negative
	aone of curse why could not they put them on		

```
from sklearn.model_selection import train_test_split
x_train, x_val , y_train , y_val = train_test_split(train_df[['text','sentiment']],train_d
x_train.shape, x_val.shape , y_train.shape , y_val.shape
```

```
((21975, 2), (5494, 2), (21975,), (5494,))
```

input_ids - Indices of input sequence tokens in the vocabulary.

The input ids are often the only required parameters to be passed to the model as input. They are token indices, numerical representations of tokens building the sequences that will be used as input by the model.

attention_mask – Mask to avoid performing attention on padding token indices. Mask values selected in [0, 1]:

1 for tokens that are not masked,

0 for tokens that are maked.

The attention mask is an optional argument used when batching sequences together. This argument indicates to the model which tokens should be attended to, and which should not.

```
MAX_LEN=92
count = x_train.shape[0]
input_ids = np.zeros((count,MAX_LEN),dtype='int32')
attention_mask = np.zeros((count,MAX_LEN),dtype='int32')
start_tokens = np.zeros((count,MAX_LEN),dtype='int32')
end_tokens = np.zeros((count,MAX_LEN),dtype='int32')
```

```
for i,each in tqdm(enumerate(x_train.values)):
  val = tokenizer.encode_plus(each[1],each[0],add_special_tokens=True,max_length=92,returr
  input_ids[i] = val['input_ids']
  attention_mask[i] = val['attention_mask']
  res = (tokenizer.encode(y_train.values[i]))
  res = res[1:-1] # to ignore <s> and </s>
  st = tf.where(val['input_ids']==res[0]).numpy()[0][1]
  start_tokens[i][st]=1
  ed = tf.where(val['input_ids']==res[-1]).numpy()[0][1]
  end_tokens[i][ed]=1
```

```
   21975it [00:32, 671.61it/s]
```

```
import random

for _ in range(35,40):
  i = random.randint(0,x_train.shape[0])
  print(x_train.iloc[i]['text'],'>>>>',y_train.iloc[i])
  print('Input ids',input_ids[i])
  print('attention mask',attention_mask[i])
  print('Start tokens',start_tokens[i])
  print('end   Tokens',end_tokens[i])
  print('*'*50)
```

```
thats what andy said a few hours ago i hope they actually are >>>> i hope they actua
Input ids [    0  1313     2     2 45365    99     8   219    26    10   367   722
   536   939  1034    51   888    32     2     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1]
attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Start tokens [0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
end   Tokens [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
**************************************************
i am new at this and i do not have any friends now please comment >>>> i am new at t
Input ids [   0 7974    2    2  939  524   92   23   42    8  939  109   45   33
  143  964  122 2540 1129    2    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1    1    1    1    1    1    1
    1    1    1    1    1    1    1    1]
attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Start tokens [0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
end   Tokens [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
**************************************************
nothing exciting from me tonight got some new toys to play with though happy mother
Input ids [    0  1313     2     2  1085  3571    31   162  3422   300   103    92
  9492     7   310    19   600  1372   985    16   183 10717     2     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1]
attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Start tokens [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
end   Tokens [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
**************************************************
you finally watched he is just not that into you arizzard justin long is sexy is not
Input ids [    0  7974     2     2    47  1747  3996    37    16    95    45    14
    88    47    10  1069  7399  1120    95   179   251    16 13350    16
    45    37   939   174 13531     2     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
```

```
            1     1     1     1     1     1     1     1     1     1     1     1
            1     1     1     1     1     1     1     1     1     1     1     1
            1     1     1     1     1     1     1     1     1     1     1     1
            1     1     1     1     1     1     1     1]
      attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
      Start tokens [0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
       0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
```

```
MAX_LEN=92
count = y_val.shape[0]
input_ids_val = np.zeros((count,MAX_LEN),dtype='int32')
attention_mask_val = np.zeros((count,MAX_LEN),dtype='int32')
start_tokens_val = np.zeros((count,MAX_LEN),dtype='int32')
end_tokens_val = np.zeros((count,MAX_LEN),dtype='int32')
```

```
          1     1     1     1     1     1     1     1     1     1     1     1
```

```
for i,each in tqdm(enumerate(x_val.values)):
  val = tokenizer.encode_plus(each[1],each[0],add_special_tokens=True,max_length=92,returr
  input_ids_val[i] = val['input_ids']
  attention_mask_val[i] = val['attention_mask']
  res = (tokenizer.encode(y_val.values[i]))
  res = res[1:-1] # to ignore <s> and </s>
  st = tf.where(val['input_ids']==res[0]).numpy()[0][1]
  start_tokens_val[i][st]=1
  ed = tf.where(val['input_ids']==res[-1]).numpy()[0][1]
  end_tokens_val[i][ed]=1
```

```
    5494it [00:07, 691.65it/s]
```

```
import random

for _ in range(35,40):
  i = random.randint(0,x_val.shape[0])
  print(x_val.iloc[i]['text'],'>>>>',y_val.iloc[i])
  print('Input ids',input_ids_val[i])
  print('attention mask',attention_mask_val[i])
  print('Start tokens',start_tokens_val[i])
  print('end   Tokens',end_tokens_val[i])
  print('*'*50)
```

```
thanks so much >>>> thanks so much
Input ids [    0  1313     2     2  2446    98   203     2     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1]
attention mask [1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Start tokens [0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
end   Tokens [0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
**************************************************
wow i just install twitter fox i am tired to keep refresh my browser >>>> wow i just
Input ids [    0  7974     2     2 26388   939    95  8486  7409 23602   939   524
  7428     7   489 14240   127 11407     2     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1]
attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Start tokens [0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
end   Tokens [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
**************************************************
i saw you started following me welcome what do you do in akl and for whom >>>> welco
Input ids [    0  1313     2     2   939   794    47   554   511   162  2814    99
   109    47   109    11    10 16291     8    13  2661     2     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1]
attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
Start tokens [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
end   Tokens [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
**************************************************
let me know how it goes i am praying ummmph i still ca not believe it >>>> i still c
Input ids [    0  1313     2     2   905   162   216   141    24  1411   939   524
 17587  1717  5471 17055   939   202  6056    45   679    24     2     1
     1     1     1     1     1     1     1     1     1     1     1     1
     1     1     1     1     1     1     1     1     1     1     1     1
```

```
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1]
    attention mask [1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
    Start tokens [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
    end   Tokens [0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
     0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
    ************************************************
    curse it i want to see that movie sooo bad do not tell me anything hehe >>>> curse i
    Input ids [    0  7974     2     2 26020    24   939   236     7   192    14  1569
        98  3036  1099   109    45  1137   162   932    37   700     2     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1     1     1     1     1
        1     1     1     1     1     1     1     1]
```

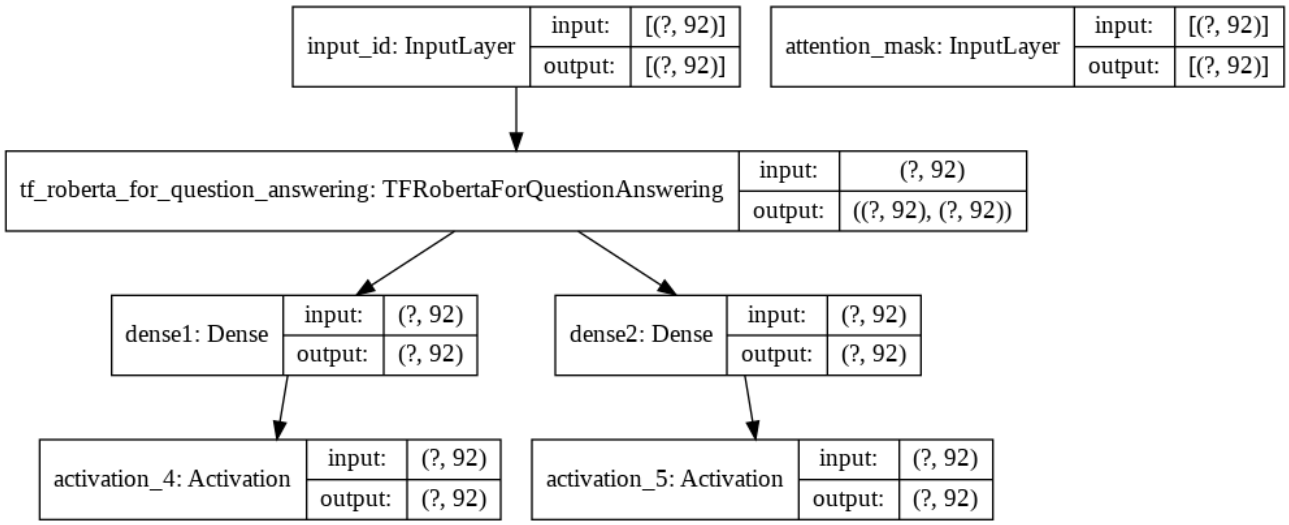
```
from transformers import TFRobertaForQuestionAnswering
roberta = TFRobertaForQuestionAnswering.from_pretrained('roberta-base')
```

```
    Some weights of the model checkpoint at roberta-base were not used when initializing
    - This IS expected if you are initializing TFRobertaForQuestionAnswering from the ch
    - This IS NOT expected if you are initializing TFRobertaForQuestionAnswering from th
    Some weights of TFRobertaForQuestionAnswering were not initialized from the model ch
    You should probably TRAIN this model on a down-stream task to be able to use it for
```

```
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input,Softmax,Dense,Activation,Dropout


input1 = Input(shape=(MAX_LEN,),name='input_id',dtype=tf.int32)
input2 = Input(shape=(MAX_LEN,),name='attention_mask',dtype=tf.int32)
start_scores,end_scores = roberta(input1,attention_mask = input2)
dense1 = Dense(units=MAX_LEN,activation='relu',name='dense1',kernel_regularizer = tf.keras
softmax1 = Activation('softmax')(dense1)
dense2 = Dense(units=MAX_LEN,activation='relu',name='dense2',kernel_regularizer = tf.keras
softmax2 = Activation('softmax')(dense2)


model = Model(inputs=[input1,input2],outputs=[softmax1,softmax2])


model.summary()
```

```
Model: "functional_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_id (InputLayer)           [(None, 92)]         0
__________________________________________________________________________________________________
attention_mask (InputLayer)     [(None, 92)]         0
__________________________________________________________________________________________________
tf_roberta_for_question_answeri ((None, 92), (None,  124647170   input_id[0][0]
                                                                 attention_mask[0][0
__________________________________________________________________________________________________
dense1 (Dense)                  (None, 92)           8556        tf_roberta_for_ques
__________________________________________________________________________________________________
dense2 (Dense)                  (None, 92)           8556        tf_roberta_for_ques
__________________________________________________________________________________________________
activation_4 (Activation)       (None, 92)           0           dense1[0][0]
__________________________________________________________________________________________________
```

```
import tensorflow as tf
tf.keras.utils.plot_model(model, 'Model.png',show_shapes=True)
```

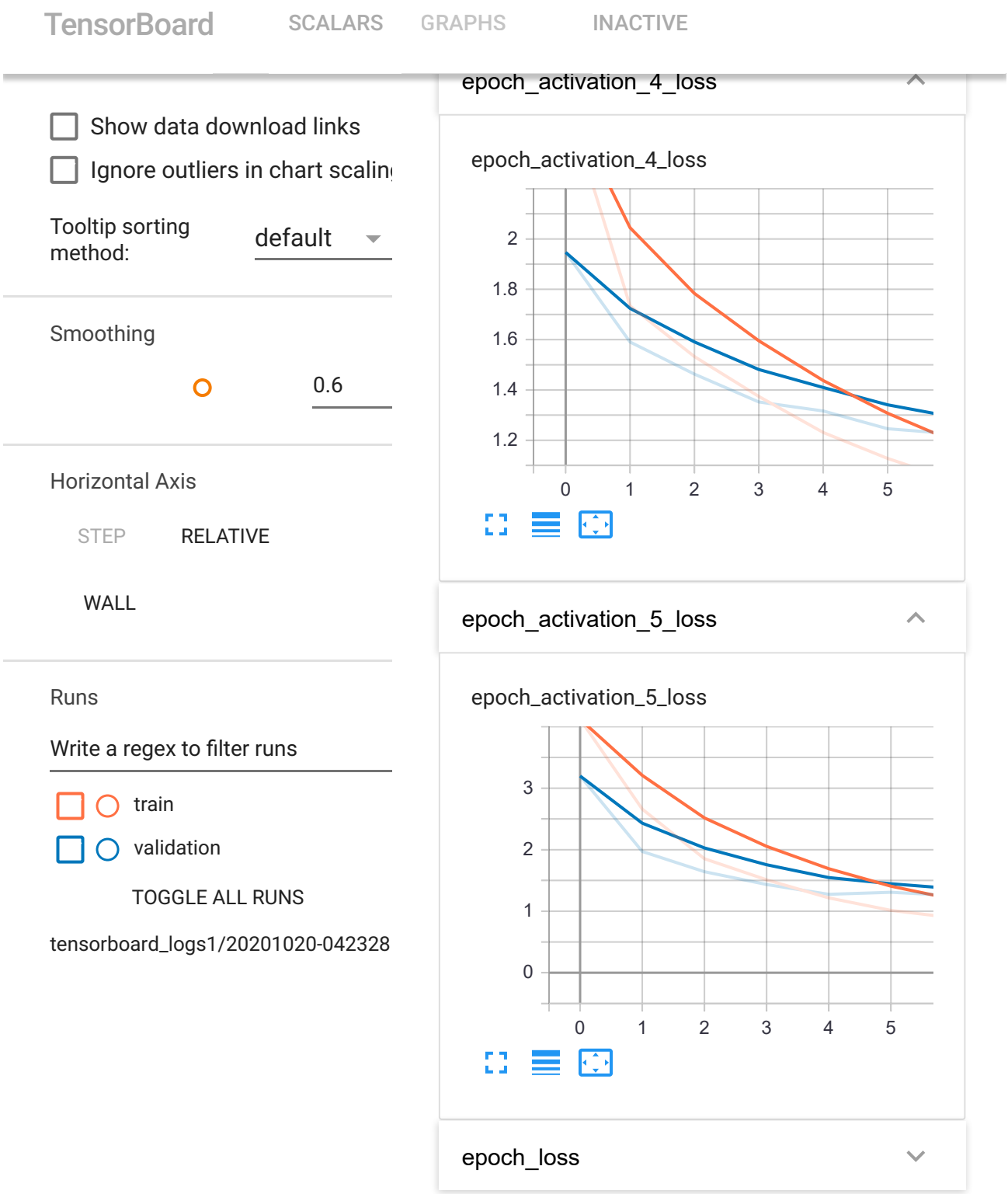



```
! rm -r '/content/checkpt'
! rm -r '/content/tensorboard_logs1'


input_data = (input_ids,attention_mask)
output_data = (start_tokens,end_tokens)

val = (input_ids_val,attention_mask_val)
output_val = (start_tokens_val,end_tokens_val)
val_data = (val,output_val)


%load_ext tensorboard
import datetime
```

```
import datetime
import os
log_dir= os.path.join("tensorboard_logs1" , datetime.datetime.now().strftime("%Y%m%d-%H%M%
tensorboard_callback = tf.keras.callbacks.TensorBoard(log_dir=log_dir,histogram_freq=1, wr
! mkdir  'checkpt'
file_path = os.path.join('checkpt/model.hdf5')
checkpt_save = tf.keras.callbacks.ModelCheckpoint(filepath=file_path,save_weights_only=Tru
callbacks=[tensorboard_callback,checkpt_save]


opt = tf.keras.optimizers.Adam(learning_rate=3e-5, epsilon=1e-08, clipnorm=1.0)
model.compile(optimizer=opt,loss='categorical_crossentropy')


train_dataset = tf.data.Dataset.from_tensor_slices((input_data, output_data)).shuffle(buff
val_dataset = tf.data.Dataset.from_tensor_slices(val_data).batch(32)


#model.fit(input_data,output_data,epochs=10,batch_size=128,validation_data=val_data,callba
model.fit(train_dataset,epochs=7,validation_data=val_dataset,callbacks=callbacks)
```

```
WARNING:tensorflow:Model failed to serialize as JSON. Ignoring...

WARNING:tensorflow:Gradients do not exist for variables ['tf_roberta_for_question_an
```

```
tf.keras.backend.clear_session()
%tensorboard --logdir $log_dir --port 0
```




```
model.load_weights('checkpt/model.hdf5')
```

```
start_pred_train , end_pred_train = model.predict((input_data))
```

```
type(output_data),output_data[0].shape,output_data[1].shape
```

```
(tuple, (21975, 92), (21975, 92))
```

```
start_pred_train.shape,end_pred_train.shape
```

```
((21975, 92), (21975, 92))
```

```
strt =[]
end=[]
for i in tqdm(range(start_pred_train.shape[0])):
  s = tf.math.argmax(start_pred_train[i],axis=0).numpy()
  e = tf.math.argmax(end_pred_train[i],axis=0).numpy()
  strt.append(s)
  end.append(e)
```

```
100%|██████████| 21975/21975 [00:10<00:00, 2133.72it/s]
```

```
len(strt),len(end)
```

```
(21975, 21975)
```

```
pred_values = []
for i in tqdm(range(len(strt))):
  index1 = strt[i]
  index2 = end[i] +1
  pred = input_ids[i][index1:index2]
  mystring = tokenizer.decode(pred)
  pred_values.append(mystring)
```

```
100%|██████████| 21975/21975 [00:00<00:00, 68814.68it/s]
```

```
actual_values = (y_train.values)
len(actual_values),len(pred_values)
```

```
(21975, 21975)
```

```
def jaccard(str1, str2):
    a = set(str1.lower().split())
    b = set(str2.lower().split())
    c = a.intersection(b)
    return float(len(c)) / (len(a) + len(b) - len(c))
```

```
scores=[]
for i in tqdm(range(len(actual_values))):
  scores.append(jaccard(actual_values[i],pred_values[i]))
```

```
100%|██████████| 21975/21975 [00:00<00:00, 223020.30it/s]
```

```
print('jaccard score for training data:',np.mean(scores))
```

```
jaccard score for training data: 0.7891184959838908
```

```
start_pred_val , end_pred_val = model.predict((val))
print(start_pred_val.shape,end_pred_val.shape)
strt_val =[]
end_val=[]
for i in tqdm(range(start_pred_val.shape[0])):
  s = tf.math.argmax(start_pred_val[i],axis=0).numpy()
  e = tf.math.argmax(end_pred_val[i],axis=0).numpy()
  strt_val.append(s)
  end_val.append(e)
print(len(strt_val),len(end_val))

pred_values_val = []
for i in tqdm(range(len(strt_val))):
  index1 = strt_val[i]
  index2 = end_val[i] +1
  pred = input_ids_val[i][index1:index2]
  mystring = tokenizer.decode(pred)
  pred_values_val.append(mystring)


actual_values_val = (y_val.values)
print(len(actual_values_val),len(pred_values_val))

scores_val=[]
for i in tqdm(range(len(actual_values_val))):
  scores_val.append(jaccard(actual_values_val[i],pred_values_val[i]))
```

```
  4%|▍          | 207/5494 [00:00<00:02, 2065.28it/s](5494, 92) (5494, 92)
100%|██████████| 5494/5494 [00:02<00:00, 2149.73it/s]
100%|██████████| 5494/5494 [00:00<00:00, 53996.53it/s]
100%|██████████| 5494/5494 [00:00<00:00, 216423.79it/s]5494 5494
5494 5494
Score 0.6862293506443297
```

```
print('Jaccard Score for val data:',np.mean(scores_val))
```

```
Jaccard Score for val data: 0.6862293506443297
```