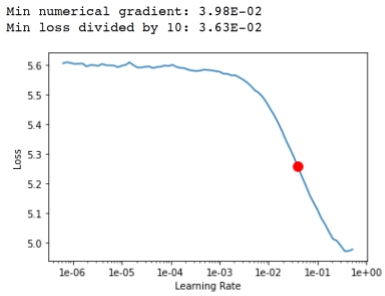
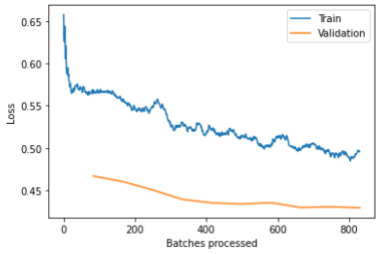
Data preparation, exploration, visualization

From lines 8-13 I did an initial EDA to test the type of language in the lists of the data. I first in line 7, imported the GLoVE pathway to set up for future modeling. In line 14-19, I began seeing the type of text and the number of NA columns associated with the text data. From there, I split the training data from the test data by text values and showed the values of the list through a batch. From there, I did an initial gradient to see the learning rate vs the loss in line 29 and found that the initial accuracy was around 47% without fine tuning. From there I began the actual modeling process to increase accuracy as well as decrease loss.





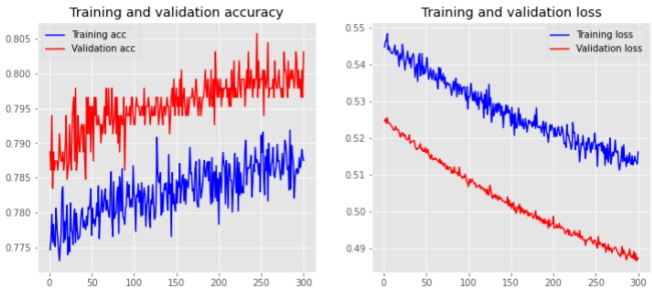
Review research design and modeling methods

For this Kaggle competition, I explored two different types of RNN varieties. The first one was a LSTM text classifier. I specifically used an AWD (weight dropped LSTM) because I wanted to be able to control the flow of inputs for the trained weights. The second type of RNN variety I used was pretrained GloVE with BERT. I wanted to use these two different models because I wanted to compare their accuracy vs their runtime. While LSTM is supposed to be more accurate, the runtime was extremely wrong. The reverse is true for GLoVE with BERT as well.

Review results, evaluate models

For the LSTM model, in lines 32-33, I classified, the data and in lines 34-36, I trained and sliced the data with different parameters (number of epochs). Lastly I plotted the training vs validation graph in line 36. Overall, the model for LSTM was accurate (71%) however took an extreme long time to process. To iterate through every epoch took around 3 minutes which totaled a time of 24 minutes for the model. In

line 88, I used the GLoVE model with BERT. In lines 88-91, I prepped the data by creating an embedded matrix and then in lines 93-95, I implemented the model and found that was 80% with a run time of 13 seconds total. This was not only more accurate than LSTM but also was extremely quick (13 seconds versus 23 minutes).



Implementation and programming as evidenced by Kaggle.com submission

My Kaggle score was 0.81612 with my username Ankita Avadhani.

Exposition, problem description, and management recommendations

If I were to recommend a solution to the problem of using a language model to classify customer reviews and complaints and assign a support personnel to a critical customer, I would highly recommend management to use GLoVE with BERT for their NLP model. Because customers also are impacted by response time in addition to accuracy of sentiment based on reviews, it's imperative have a model that is quick in addition to one that is accurate. By using pretrained GLoVe, management will be able to process a vast amount of reviews in around 13 seconds versus 23 minutes. In order to create this automated support system, I would recommend using pretrained sets of data as well. Data Scientists can use pretrained sets of sentiment analysis similar to GLoVe in order to create language models that can be quick yet effective in identifying key customers.

In [6]:
```
import numpy as np
import tensorflow as tf

# Python chakin package previously installed by
#    pip install chakin
import chakin
import json
import os
from collections import defaultdict

CHAKIN_INDEX = 11
NUMBER_OF_DIMENSIONS = 50
SUBFOLDER_NAME = "gloVe.6B"

DATA_FOLDER = "embeddings"
ZIP_FILE = os.path.join(DATA_FOLDER, "{}.zip".format(SUBFOLDER_NAME))
ZIP_FILE_ALT = "glove" + ZIP_FILE[5:]  # sometimes it's lowercase onl
y...
UNZIP_FOLDER = os.path.join(DATA_FOLDER, SUBFOLDER_NAME)
if SUBFOLDER_NAME[-1] == "d":
    GLOVE_FILENAME = os.path.join(
        UNZIP_FOLDER, "{}.txt".format(SUBFOLDER_NAME))
else:
    GLOVE_FILENAME = os.path.join(UNZIP_FOLDER, "{}.{}d.txt".format(
        SUBFOLDER_NAME, NUMBER_OF_DIMENSIONS))
```

In [7]:
```
if not os.path.exists(ZIP_FILE) and not os.path.exists(UNZIP_FOLDER):
    # GloVe by Stanford is licensed Apache 2.0:
    #     https://github.com/stanfordnlp/GloVe/blob/master/LICENSE
    #     http://nlp.stanford.edu/data/glove.twitter.27B.zip
    #     Copyright 2014 The Board of Trustees of The Leland Stanford Ju
nior University
    print("Downloading embeddings to '{}'".format(ZIP_FILE))
    chakin.download(number=CHAKIN_INDEX, save_dir='./{}'.format(DATA_FOL
DER))
else:
    print("Embeddings already downloaded.")

if not os.path.exists(UNZIP_FOLDER):
    import zipfile
    if not os.path.exists(ZIP_FILE) and os.path.exists(ZIP_FILE_ALT):
        ZIP_FILE = ZIP_FILE_ALT
    with zipfile.ZipFile(ZIP_FILE, "r") as zip_ref:
        print("Extracting embeddings to '{}'".format(UNZIP_FOLDER))
        zip_ref.extractall(UNZIP_FOLDER)
else:
    print("Embeddings already extracted.")

print('\nRun complete')
```

```
Downloading embeddings to 'embeddings/gloVe.6B.zip'

Test: 100% ||                                          | Time:  0:06:27
2.1 MiB/s

Extracting embeddings to 'embeddings/gloVe.6B'

Run complete
```

In [8]:
```
from __future__ import absolute_import
from __future__ import division
from __future__ import print_function

# Keras
from keras.preprocessing.text import Tokenizer
from keras.preprocessing.sequence import pad_sequences
from keras.models import Sequential
from keras.layers import Dense, Flatten, LSTM, Conv1D, MaxPooling1D, Dro
pout, Activation
from keras.layers.embeddings import Embedding

## Plot
from matplotlib import pyplot

# NLTK
import nltk
from nltk.corpus import stopwords
from nltk.stem import SnowballStemmer
from nltk.tokenize import TreebankWordTokenizer

# Other
import re
import string
import numpy as np
import pandas as pd
from sklearn.manifold import TSNE
```

In [13]:
```
from fastai import *
from fastai.text import *
import pandas as pd
```

In [14]:
```
train_df= pd.read_csv('train.csv')
test_df= pd.read_csv('test.csv')
train_df.head()
```

Out[14]:

	id	keyword	location	text	target
0	1	NaN	NaN	Our Deeds are the Reason of this #earthquake M...	1
1	4	NaN	NaN	Forest fire near La Ronge Sask. Canada	1
2	5	NaN	NaN	All residents asked to 'shelter in place' are ...	1
3	6	NaN	NaN	13,000 people receive #wildfires evacuation or...	1
4	7	NaN	NaN	Just got sent this photo from Ruby #Alaska as ...	1

In [15]:
```
train_df.iloc[0]["text"]
```

Out[15]:
```
'Our Deeds are the Reason of this #earthquake May ALLAH Forgive us all'
```

In [ ]:
```
#model for language
```

In [16]:
```
path = Path('../input/nlp-getting-started')
```

In [17]:
```
df_lm = pd.DataFrame({
    "text" : np.concatenate((train_df["text"].values, test_df["text"].va
lues), axis = 0)
})
```

In [18]:
```
data_lm = (TextList.from_df(df_lm).split_by_rand_pct(0.1).label_for_lm()
.databunch(bs=32))
```

In [19]:
```
data_lm.show_batch()
```

```
/Users/avadhani/anaconda3/lib/python3.7/site-packages/fastai/text/data.
py:339: UserWarning: This overload of nonzero is deprecated:
        nonzero()
Consider using one of the following signatures instead:
        nonzero(*, bool as_tuple) (Triggered internally at  ../torch/cs
rc/utils/python_arg_parser.cpp:766.)
  idx_min = (t != self.pad_idx).nonzero().min()
```

idx	text
0	orders in xxmaj california xxbos xxmaj just got sent this photo from xxmaj xxunk # xxmaj alaska as smoke from # wildfires xxunk into a school xxbos # rockyfire xxmaj update = > xxmaj california xxmaj hwy . 20 closed in both xxunk due to xxmaj lake xxmaj county fire - # xxunk # wildfires xxbos # flood # disaster xxmaj heavy rain causes flash flooding of streets in xxmaj
1	> https : / / t.co / xxunk > > @trubgme xxmaj xxunk xxup xxunk > > # xxup armageddon . xxbos xxmaj tomorrow is the day we start armageddon # xxunk xxrep 4 ? xxbos xxmaj xxunk does comedy : xxunk : xxmaj working class xxmaj xxunk prepare for your xxmaj armageddon . # xxunk xxbos 9 xxmaj charts xxmaj xxunk xxmaj financial xxmaj crisis xxmaj part 2 xxmaj
2	the back ? ? xxbos 3 xxmaj xxunk 1 xxmaj xxunk for sale in 29 xxmaj palms xxup ca . (http : / / t.co / xxunk) \n (youtube xxmaj video : ... http : / / t.co / xxunk xxbos ? ? xxmaj yes i do have 2 guns ? ? ? ? xxbos xxunk xxup xxunk xxup girl xxup xxunk xxup xxunk c :
3	xxbos xxmaj status : last seen buying body bags . xxbos xxunk xxunk i 'm xxunk by only xxunk cross - body bags . i really like xxmaj xxunk xxmaj xxunk bags : machine xxunk . http : / / t.co / xxunk xxbos xxmaj xxunk xxmaj vintage xxmaj leather xxmaj xxunk xxmaj messenger xxmaj satchel xxmaj tote xxmaj cross xxmaj body xxmaj handbags for xxmaj womens http : /
4	the xxmaj corners xxmaj of xxmaj the xxmaj planet [xxmaj on xxmaj all xxmaj levels] http : / / t.co / xxup xxunk xxbos xxunk : xxmaj warfighting xxmaj robots xxmaj could xxmaj reduce xxmaj civilian xxmaj casualties xxmaj so xxmaj calling for a xxmaj ban xxmaj now xxmaj is ... - ... http : / / t.co / xxunk # xxunk xxbos a xxunk reminder that in

In [20]:
```
learn = language_model_learner(data_lm, AWD_LSTM, drop_mult=0.3)
```

```
Downloading https://s3.amazonaws.com/fast-ai-modelzoo/wt103-fwd.tgz
```

In [21]:
```
learn.lr_find()
```

0.00% [0/1 00:00<00:00]

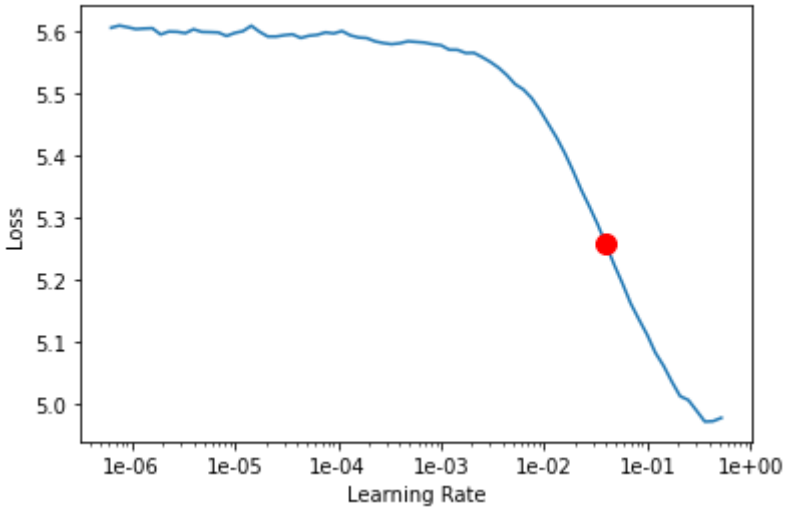
epoch	train_loss	valid_loss	accuracy	time

85.34% [99/116 02:40<00:27 9.3176]

```
LR Finder is complete, type {learner_name}.recorder.plot() to see the g
raph.
```

In [22]:
```
learn.recorder.plot(skip_end=15, suggestion = True)
```

```
Min numerical gradient: 3.98E-02
Min loss divided by 10: 3.63E-02
```



In [23]:
```
learn.fit_one_cycle(5, 1e-2, moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	4.173398	3.427854	0.410131	03:57
1	3.283376	3.061961	0.457349	03:45
2	2.835851	2.970045	0.471257	03:42
3	2.582863	2.941284	0.474279	03:42
4	2.419730	2.939342	0.474416	03:37

In [24]:
```
learn.save('fine_tuned')
```

In [25]:
```
learn.load('fine_tuned');
```

In [26]:
```
learn.save_encoder('fine_tuned_enc')
```

In [27]:
```
path
```

Out[27]:
```
PosixPath('../input/nlp-getting-started')
```

In [28]:
```
from sklearn.model_selection import train_test_split
X_train, X_val = train_test_split(train_df, test_size = 0.3)
```

In [29]:
```
data_classifier = (TextDataBunch.from_df('.', X_train, X_val, test_df, t
ext_cols = "text", label_cols = "target", vocab = data_lm.vocab))
```

In [30]:
```
data_classifier.show_batch()
```

text	target
xxbos _ \n xxrep 5 ? xxup retweet \n xxrep 7 ? \n xxrep 5 ? xxup follow xxup all xxup who xxup rt \n xxrep 7 ? \n xxrep 5 ? xxup xxunk \n xxrep 7 ? \n xxrep 5 ? xxup gain xxup with \n xxrep 7 ? \n xxrep 5 ? xxup follow ? xxunk # xxup xxunk	0
xxbos xxup info xxup u. xxup xxunk : xxup xxunk xxup xxunk . xxup exp xxup inst xxup apch . xxup rwy 05 . xxup curfew xxup in xxup oper xxup until 2030 xxup z. xxup taxiways xxup foxtrot 5 & & xxup foxtrot 6 xxup navbl . xxup tmp : 10 . xxup wnd : xxunk / 6 .	0
xxbos xxmaj no # news of # hostages in # xxmaj libya \n \n http : / / t.co / xxunk \n \n # xxmaj india # terrorism # xxmaj africa # xxup ap # xxup xxunk # xxup xxunk # xxmaj news # xxup xxunk # xxup xxunk # xxup bjp http : / / t.co / xxunk	1
xxbos xxmaj truth ... \n https : / / t.co / xxunk \n # xxmaj news \n # xxup bbc \n # xxup cnn \n # xxmaj islam \n # xxmaj truth \n # god \n # xxup isis \n # terrorism \n # xxmaj quran \n # xxmaj lies http : / / t.co / xxunk	1
xxbos xxmaj truth ... \n https : / / t.co / xxunk \n # xxmaj news \n # xxup bbc \n # xxup cnn \n # xxmaj islam \n # xxmaj truth \n # god \n # xxup isis \n # terrorism \n # xxmaj quran \n # xxmaj lies http : / / t.co / xxunk	0

In [31]:
```
data_classifier.save('data_clas.pkl')
```

In [ ]:
```
#RNN text classifier 1, LSTM
```

In [32]:
```
learn = text_classifier_learner(data_classifier, AWD_LSTM)
learn.load_encoder('fine_tuned_enc')
```

Out[32]:
```
RNNLearner(data=TextClasDataBunch;

Train: LabelList (5329 items)
x: TextList
xxbos xxunk xxmaj is my pick for http : / / t.co / thoyhrhkfj xxmaj fan
xxmaj army # xxmaj beyhive http : / / t.co / wvj39a3bgm,xxbos xxmaj dr.
xxmaj jim & & the tsunami : xxmaj the latest xxmaj new xxmaj yorker war
ned us in no xxunk terms . xxmaj have n't you heard ? xxmaj the tsunami
's ... http : / / t.co / xxunk,xxbos xxmaj man xxmaj found xxmaj dead i
n xxmaj demi xxmaj moore 's xxmaj swimming xxmaj pool ! http : / / t.co
/ xxunk,xxbos xxmaj some poor xxunk arriving in xxmaj xxunk during yest
erday 's dust storm were xxunk to xxmaj ben xxmaj xxunk airport : http
: / / t.co / xxunk,xxbos xxunk xxunk xxunk xxunk xxmaj appears to alrea
dy be arriving in xxmaj xxunk in body bags .
y: CategoryList
0,0,1,1,0
Path: .;

Valid: LabelList (2284 items)
x: TextList
xxbos xxmaj welcome xxunk xxunk xxunk xxunk xxunk # xxmaj family # xxma
j cleveland # xxunk xxup xxunk http : / / t.co / xxunk,xxbos xxup u.s x
xmaj national xxmaj park xxmaj services xxmaj tonto xxmaj national xxma
j forest : xxmaj stop the xxmaj annihilation of the xxmaj salt xxmaj ri
ver xxmaj wild xxmaj horse ... https : / / t.co / xxunk via @change,xxb
os xxunk xxup gm ! i pray any attack of the enemy 2 derail ur destiny i
s blocked by the xxmaj lord & & that xxmaj he floods ur life w / heaven
ly xxmaj blessings,xxbos # xxmaj australia # xxmaj news ; xxup rt xxunk
: ' xxmaj high xxunk ' aircraft wreckage is from # xxup mh370 according
to xxmaj deputy xxmaj prime xxup  û _ http : / / t.co / xxunk,xxbos xx
maj ron & & xxmaj xxunk - xxmaj xxunk 's xxmaj high xxmaj school xxmaj
crush https : / / t.co / xxunk via @youtube
y: CategoryList
0,0,0,1,1
Path: .;

Test: LabelList (3263 items)
x: TextList
xxbos xxmaj just happened a terrible car crash,xxbos xxmaj heard about
# earthquake is different cities , stay safe everyone .,xxbos there is
a forest fire at spot pond , xxunk are fleeing across the street , i ca
n not save them all,xxbos xxmaj apocalypse lighting . # xxmaj xxunk # w
ildfires,xxbos xxmaj typhoon xxmaj soudelor kills 28 in xxmaj china and
xxmaj taiwan
y: EmptyLabelList
,,,,
Path: ., model=SequentialRNN(
  (0): MultiBatchEncoder(
    (module): AWD_LSTM(
      (encoder): Embedding(5432, 400, padding_idx=1)
      (encoder_dp): EmbeddingDropout(
        (emb): Embedding(5432, 400, padding_idx=1)
      )
      (rnns): ModuleList(
        (0): WeightDropout(
          (module): LSTM(400, 1152, batch_first=True)
        )
```

```
        (1): WeightDropout(
          (module): LSTM(1152, 1152, batch_first=True)
        )
        (2): WeightDropout(
          (module): LSTM(1152, 400, batch_first=True)
        )
      )
      (input_dp): RNNDropout()
      (hidden_dps): ModuleList(
        (0): RNNDropout()
        (1): RNNDropout()
        (2): RNNDropout()
      )
    )
  )
  (1): PoolingLinearClassifier(
    (layers): Sequential(
      (0): BatchNorm1d(1200, eps=1e-05, momentum=0.1, affine=True, trac
k_running_stats=True)
      (1): Dropout(p=0.4, inplace=False)
      (2): Linear(in_features=1200, out_features=50, bias=True)
      (3): ReLU(inplace=True)
      (4): BatchNorm1d(50, eps=1e-05, momentum=0.1, affine=True, track_
running_stats=True)
      (5): Dropout(p=0.1, inplace=False)
      (6): Linear(in_features=50, out_features=2, bias=True)
    )
  )
), opt_func=functools.partial(<class 'torch.optim.adam.Adam'>, betas=
(0.9, 0.99)), loss_func=FlattenedLoss of CrossEntropyLoss(), metrics=[<
function accuracy at 0x163b6b950>], true_wd=True, bn_wd=True, wd=0.01,
train_bn=True, path=PosixPath('.'), model_dir='models', callback_fns=[f
unctools.partial(<class 'fastai.basic_train.Recorder'>, add_time=True,
silent=False)], callbacks=[RNNTrainer
learn: ...
alpha: 2.0
beta: 1.0], layer_groups=[Sequential(
  (0): Embedding(5432, 400, padding_idx=1)
  (1): EmbeddingDropout(
    (emb): Embedding(5432, 400, padding_idx=1)
  )
), Sequential(
  (0): WeightDropout(
    (module): LSTM(400, 1152, batch_first=True)
  )
  (1): RNNDropout()
), Sequential(
  (0): WeightDropout(
    (module): LSTM(1152, 1152, batch_first=True)
  )
  (1): RNNDropout()
), Sequential(
  (0): WeightDropout(
    (module): LSTM(1152, 400, batch_first=True)
  )
  (1): RNNDropout()
), Sequential(
```

```
  (0): PoolingLinearClassifier(
    (layers): Sequential(
      (0): BatchNorm1d(1200, eps=1e-05, momentum=0.1, affine=True, trac
k_running_stats=True)
      (1): Dropout(p=0.4, inplace=False)
      (2): Linear(in_features=1200, out_features=50, bias=True)
      (3): ReLU(inplace=True)
      (4): BatchNorm1d(50, eps=1e-05, momentum=0.1, affine=True, track_
running_stats=True)
      (5): Dropout(p=0.1, inplace=False)
      (6): Linear(in_features=50, out_features=2, bias=True)
    )
  )
)], add_time=True, silent=False)
```

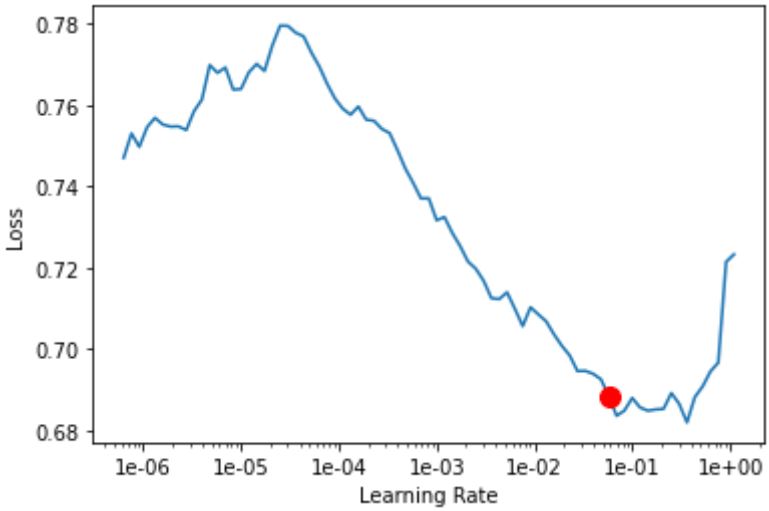
In [33]:
```
learn.lr_find()
learn.recorder.plot(suggestion= True)
```

50.00% [1/2 01:01<01:01]

epoch	train_loss	valid_loss	accuracy	time
0	0.681947	#na#	01:01	

12.05% [10/83 00:08<00:59 1.5818]

```
LR Finder is complete, type {learner_name}.recorder.plot() to see the g
raph.
Min numerical gradient: 5.75E-02
Min loss divided by 10: 3.63E-02
```



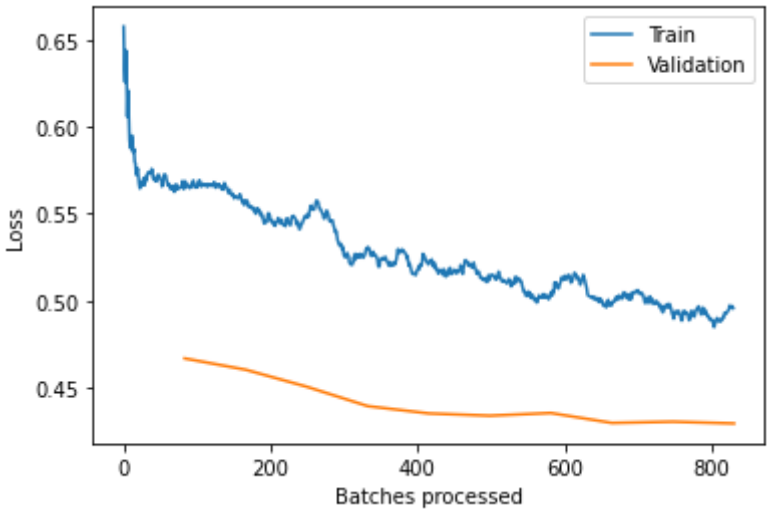
In [34]:
```
learn.fit_one_cycle(5, slice(1e-2/(2.6**4),1e-2), moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	0.626564	0.502791	0.767075	01:17
1	0.603298	0.491204	0.767951	01:04
2	0.588070	0.489713	0.774956	01:11
3	0.573375	0.480227	0.778897	01:06
4	0.566551	0.473863	0.782837	01:05

In [35]:
```
learn.unfreeze()
learn.fit_one_cycle(10, slice(1e-3/(2.6**4),1e-3), moms=(0.8,0.7))
```

epoch	train_loss	valid_loss	accuracy	time
0	0.564141	0.466671	0.779334	03:27
1	0.555309	0.460162	0.787215	03:23
2	0.547579	0.450382	0.791156	03:24
3	0.529618	0.439206	0.790718	02:45
4	0.520763	0.435006	0.793783	02:35
5	0.511774	0.433710	0.799475	02:32
6	0.502777	0.435250	0.799912	02:45
7	0.497663	0.429542	0.801664	02:39
8	0.492418	0.430251	0.802977	02:39
9	0.495556	0.429237	0.801664	02:49

In [36]:
```
learn.recorder.plot_losses()
```



In [37]:
```
def get_preds_as_nparray(ds_type) -> np.ndarray:
    """
    the get_preds method does not yield the elements in order by default
    we borrow the code from the RNNLearner to resort the elements into t
heir correct order
    """
    preds = learn.get_preds(ds_type)[0].detach().cpu().numpy()
    sampler = [i for i in data_classifier.dl(ds_type).sampler]
    reverse_sampler = np.argsort(sampler)
    return preds[reverse_sampler, :]
```

In [38]:
```
test_preds = get_preds_as_nparray(DatasetType.Test)
preds = []
```

In [39]:
```
for i in test_preds:
    preds.append(np.argmax(i))
```

In [77]:
```
sub = pd.read_csv("sample_submission.csv")
sub.head(3)
sub['target'] = preds
sub.to_csv('submission_LSTM.csv', index=False)
sub.head(3)
```

Out[77]:

	id	target
0	0	0
1	2	0
2	3	1

In [ ]:
```
## kaggle score 0.79313 username: Ankita Avadhani
```

In [ ]:
```
#RNN 2 GLoVE
```

In [88]:
```
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras import Sequential, layers, regularizers
from sklearn.model_selection import train_test_split
sent_train, sent_test, labels_train, labels_test = train_test_split(sent
_data, labels_data, test_size=0.2, random_state=42)
```

In [89]:
```
tokenizer = Tokenizer()
tokenizer.fit_on_texts(sent_data)

X_train = tokenizer.texts_to_sequences(sent_data)
# X_test = tokenizer.texts_to_sequences(sent_test)
X_submission = tokenizer.texts_to_sequences(sent_submission)

y_train = labels_data
# y_test = labels_test

vocab_size = len(tokenizer.word_index) + 1  # Adding 1 because of reserv
ed 0 index

print(sent_train[2])
print(X_train[2])
```

```
Tell @BarackObama to rescind medals of 'honor' given to US soldiers at
the Massacre of Wounded Knee. SIGN NOW &amp;amp; RT! https://t.co/u4r8dRiuA
c
[40, 1751, 1620, 7, 6956, 6, 6957, 24, 136, 6958, 20, 1752, 39, 441, 25
6, 57, 2158, 6, 714, 1405, 24, 1106]
```

In [90]:
```
maxlen = 100

X_train = pad_sequences(X_train, padding='post', maxlen=maxlen)
# X_test = pad_sequences(X_test, padding='post', maxlen=maxlen)
X_submission = pad_sequences(X_submission, padding='post', maxlen=maxlen
)

print(X_train[0, :])
```

```
[ 119 4633   24    4 ...    0    0    0    0]
```

In [91]:
```
def create_embedding_matrix(filepath, word_index, embedding_dim):
    vocab_size = len(word_index) + 1  # Adding again 1 because of reserv
ed 0 index
    embedding_matrix = np.zeros((vocab_size, embedding_dim))

    with open(filepath) as f:
        for line in f:
            word, *vector = line.split()
            if word in word_index:
                idx = word_index[word]
                embedding_matrix[idx] = np.array(
                    vector, dtype=np.float32)[:embedding_dim]

    return embedding_matrix
```

In [93]:
```
embedding_dim = 50
embedding_matrix = create_embedding_matrix(
    'glove.twitter.27B.50d.txt',
    tokenizer.word_index, embedding_dim)

drop_out_prob = 0.5

model = Sequential()
model.add(layers.Embedding(input_dim=vocab_size,
                           output_dim=embedding_dim,
                           weights = [embedding_matrix],
                           input_length=maxlen,
                           trainable=False))
# model.add(layers.Conv1D(128, 5, activation='relu'))
model.add(layers.GlobalAveragePooling1D())
model.add(layers.Dense(64, activation='relu',
                kernel_regularizer=regularizers.l2(0.01),
                activity_regularizer=regularizers.l1(0.01)))
model.add(layers.Dropout(drop_out_prob))
model.add(layers.Dense(1, activation='sigmoid'))
model.compile(optimizer='adam',
              loss='binary_crossentropy',
              metrics=['accuracy'])
model.summary()
```

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 100, 50)           1135050
_________________________________________________________________
global_average_pooling1d_1 ( (None, 50)                0
_________________________________________________________________
dense_2 (Dense)              (None, 64)                3264
_________________________________________________________________
dropout_1 (Dropout)          (None, 64)                0
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 65
=================================================================
Total params: 1,138,379
Trainable params: 3,329
Non-trainable params: 1,135,050
_________________________________________________________________
```

In [95]:
```
import matplotlib.pyplot as plt
plt.style.use('ggplot')

def plot_history(history):
    acc = history.history['accuracy']
    val_acc = history.history['val_accuracy']
    loss = history.history['loss']
    val_loss = history.history['val_loss']
    x = range(1, len(acc) + 1)

    plt.figure(figsize=(12, 5))
    plt.subplot(1, 2, 1)
    plt.plot(x, acc, 'b', label='Training acc')
    plt.plot(x, val_acc, 'r', label='Validation acc')
    plt.title('Training and validation accuracy')
    plt.legend()
    plt.subplot(1, 2, 2)
    plt.plot(x, loss, 'b', label='Training loss')
    plt.plot(x, val_loss, 'r', label='Validation loss')
    plt.title('Training and validation loss')
    plt.legend()

history = model.fit(X_train, y_train,
                    epochs=300,
                    verbose=1,
                    validation_split=0.1,
                    batch_size=1000, use_multiprocessing=True)
plot_history(history)
```

```
Epoch 1/300
7/7 [==============================] - 0s 12ms/step - loss: 0.5448 - ac
curacy: 0.7746 - val_loss: 0.5247 - val_accuracy: 0.7887
Epoch 2/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5464 - acc
uracy: 0.7762 - val_loss: 0.5252 - val_accuracy: 0.7861
Epoch 3/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5464 - acc
uracy: 0.7797 - val_loss: 0.5241 - val_accuracy: 0.7940
Epoch 4/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5484 - acc
uracy: 0.7757 - val_loss: 0.5255 - val_accuracy: 0.7835
Epoch 5/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5440 - acc
uracy: 0.7783 - val_loss: 0.5241 - val_accuracy: 0.7887
Epoch 6/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5450 - acc
uracy: 0.7754 - val_loss: 0.5241 - val_accuracy: 0.7861
Epoch 7/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5443 - acc
uracy: 0.7751 - val_loss: 0.5241 - val_accuracy: 0.7874
Epoch 8/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5433 - acc
uracy: 0.7806 - val_loss: 0.5237 - val_accuracy: 0.7874
Epoch 9/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5433 - acc
uracy: 0.7778 - val_loss: 0.5232 - val_accuracy: 0.7874
Epoch 10/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5429 - acc
uracy: 0.7764 - val_loss: 0.5230 - val_accuracy: 0.7861
Epoch 11/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5434 - acc
uracy: 0.7745 - val_loss: 0.5224 - val_accuracy: 0.7874
Epoch 12/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5439 - acc
uracy: 0.7730 - val_loss: 0.5221 - val_accuracy: 0.7913
Epoch 13/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5452 - acc
uracy: 0.7751 - val_loss: 0.5224 - val_accuracy: 0.7887
Epoch 14/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5427 - acc
uracy: 0.7771 - val_loss: 0.5232 - val_accuracy: 0.7874
Epoch 15/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5431 - acc
uracy: 0.7828 - val_loss: 0.5223 - val_accuracy: 0.7874
Epoch 16/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5439 - acc
uracy: 0.7837 - val_loss: 0.5224 - val_accuracy: 0.7887
Epoch 17/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5420 - acc
uracy: 0.7757 - val_loss: 0.5222 - val_accuracy: 0.7861
Epoch 18/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5395 - acc
uracy: 0.7787 - val_loss: 0.5216 - val_accuracy: 0.7874
Epoch 19/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5430 - acc
uracy: 0.7761 - val_loss: 0.5206 - val_accuracy: 0.7900
```

```
Epoch 20/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5421 - acc
uracy: 0.7789 - val_loss: 0.5203 - val_accuracy: 0.7940
Epoch 21/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5413 - ac
curacy: 0.7811 - val_loss: 0.5214 - val_accuracy: 0.7848
Epoch 22/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5455 - acc
uracy: 0.7739 - val_loss: 0.5210 - val_accuracy: 0.7861
Epoch 23/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5432 - acc
uracy: 0.7764 - val_loss: 0.5213 - val_accuracy: 0.7874
Epoch 24/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5405 - acc
uracy: 0.7743 - val_loss: 0.5205 - val_accuracy: 0.7927
Epoch 25/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5385 - acc
uracy: 0.7813 - val_loss: 0.5209 - val_accuracy: 0.7874
Epoch 26/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5409 - acc
uracy: 0.7774 - val_loss: 0.5198 - val_accuracy: 0.7861
Epoch 27/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5417 - acc
uracy: 0.7773 - val_loss: 0.5196 - val_accuracy: 0.7927
Epoch 28/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5436 - acc
uracy: 0.7754 - val_loss: 0.5196 - val_accuracy: 0.7953
Epoch 29/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5417 - acc
uracy: 0.7792 - val_loss: 0.5199 - val_accuracy: 0.7966
Epoch 30/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5403 - acc
uracy: 0.7805 - val_loss: 0.5209 - val_accuracy: 0.7874
Epoch 31/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5418 - acc
uracy: 0.7757 - val_loss: 0.5186 - val_accuracy: 0.7979
Epoch 32/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5391 - acc
uracy: 0.7799 - val_loss: 0.5189 - val_accuracy: 0.7861
Epoch 33/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5419 - acc
uracy: 0.7776 - val_loss: 0.5190 - val_accuracy: 0.7874
Epoch 34/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5425 - acc
uracy: 0.7761 - val_loss: 0.5192 - val_accuracy: 0.7940
Epoch 35/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5430 - acc
uracy: 0.7764 - val_loss: 0.5193 - val_accuracy: 0.7900
Epoch 36/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5398 - acc
uracy: 0.7765 - val_loss: 0.5192 - val_accuracy: 0.7887
Epoch 37/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5366 - acc
uracy: 0.7787 - val_loss: 0.5178 - val_accuracy: 0.7913
Epoch 38/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5390 - acc
uracy: 0.7802 - val_loss: 0.5172 - val_accuracy: 0.7927
```

```
Epoch 39/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5401 - acc
uracy: 0.7806 - val_loss: 0.5168 - val_accuracy: 0.7927
Epoch 40/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5354 - acc
uracy: 0.7778 - val_loss: 0.5173 - val_accuracy: 0.7887
Epoch 41/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5419 - acc
uracy: 0.7789 - val_loss: 0.5173 - val_accuracy: 0.7927
Epoch 42/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5430 - acc
uracy: 0.7780 - val_loss: 0.5177 - val_accuracy: 0.7874
Epoch 43/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5369 - acc
uracy: 0.7776 - val_loss: 0.5185 - val_accuracy: 0.7848
Epoch 44/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5387 - acc
uracy: 0.7794 - val_loss: 0.5169 - val_accuracy: 0.7966
Epoch 45/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5430 - acc
uracy: 0.7781 - val_loss: 0.5173 - val_accuracy: 0.7913
Epoch 46/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5373 - acc
uracy: 0.7808 - val_loss: 0.5164 - val_accuracy: 0.7966
Epoch 47/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5396 - acc
uracy: 0.7805 - val_loss: 0.5167 - val_accuracy: 0.7913
Epoch 48/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5375 - ac
curacy: 0.7812 - val_loss: 0.5163 - val_accuracy: 0.7953
Epoch 49/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5379 - ac
curacy: 0.7834 - val_loss: 0.5175 - val_accuracy: 0.7887
Epoch 50/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5401 - ac
curacy: 0.7767 - val_loss: 0.5160 - val_accuracy: 0.7953
Epoch 51/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5419 - acc
uracy: 0.7806 - val_loss: 0.5161 - val_accuracy: 0.7953
Epoch 52/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5384 - acc
uracy: 0.7784 - val_loss: 0.5170 - val_accuracy: 0.7874
Epoch 53/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5361 - acc
uracy: 0.7790 - val_loss: 0.5153 - val_accuracy: 0.7966
Epoch 54/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5374 - acc
uracy: 0.7780 - val_loss: 0.5151 - val_accuracy: 0.7940
Epoch 55/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5390 - acc
uracy: 0.7792 - val_loss: 0.5143 - val_accuracy: 0.7966
Epoch 56/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5368 - acc
uracy: 0.7813 - val_loss: 0.5156 - val_accuracy: 0.7927
Epoch 57/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5360 - acc
uracy: 0.7794 - val_loss: 0.5147 - val_accuracy: 0.7927
```

```
Epoch 58/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5385 - acc
uracy: 0.7838 - val_loss: 0.5141 - val_accuracy: 0.7940
Epoch 59/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5381 - acc
uracy: 0.7762 - val_loss: 0.5143 - val_accuracy: 0.7927
Epoch 60/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5358 - acc
uracy: 0.7797 - val_loss: 0.5141 - val_accuracy: 0.7927
Epoch 61/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5357 - ac
curacy: 0.7808 - val_loss: 0.5146 - val_accuracy: 0.7900
Epoch 62/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5376 - acc
uracy: 0.7832 - val_loss: 0.5143 - val_accuracy: 0.7927
Epoch 63/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5351 - acc
uracy: 0.7800 - val_loss: 0.5141 - val_accuracy: 0.7913
Epoch 64/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5334 - acc
uracy: 0.7806 - val_loss: 0.5137 - val_accuracy: 0.7927
Epoch 65/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5392 - acc
uracy: 0.7824 - val_loss: 0.5134 - val_accuracy: 0.7953
Epoch 66/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5361 - ac
curacy: 0.7834 - val_loss: 0.5143 - val_accuracy: 0.7913
Epoch 67/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5390 - acc
uracy: 0.7803 - val_loss: 0.5131 - val_accuracy: 0.7979
Epoch 68/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5356 - acc
uracy: 0.7818 - val_loss: 0.5138 - val_accuracy: 0.7940
Epoch 69/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5359 - acc
uracy: 0.7809 - val_loss: 0.5131 - val_accuracy: 0.7940
Epoch 70/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5375 - acc
uracy: 0.7797 - val_loss: 0.5124 - val_accuracy: 0.7979
Epoch 71/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5349 - acc
uracy: 0.7793 - val_loss: 0.5132 - val_accuracy: 0.7927
Epoch 72/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5366 - acc
uracy: 0.7832 - val_loss: 0.5129 - val_accuracy: 0.7927
Epoch 73/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5336 - acc
uracy: 0.7859 - val_loss: 0.5121 - val_accuracy: 0.7940
Epoch 74/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5348 - ac
curacy: 0.7787 - val_loss: 0.5116 - val_accuracy: 0.7979
Epoch 75/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5351 - acc
uracy: 0.7765 - val_loss: 0.5125 - val_accuracy: 0.7913
Epoch 76/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5344 - acc
uracy: 0.7844 - val_loss: 0.5115 - val_accuracy: 0.7927
```

```
Epoch 77/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5347 - acc
uracy: 0.7792 - val_loss: 0.5120 - val_accuracy: 0.7940
Epoch 78/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5346 - acc
uracy: 0.7799 - val_loss: 0.5121 - val_accuracy: 0.7966
Epoch 79/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5331 - acc
uracy: 0.7799 - val_loss: 0.5118 - val_accuracy: 0.7913
Epoch 80/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5316 - acc
uracy: 0.7830 - val_loss: 0.5104 - val_accuracy: 0.7966
Epoch 81/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5332 - acc
uracy: 0.7837 - val_loss: 0.5112 - val_accuracy: 0.7900
Epoch 82/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5355 - acc
uracy: 0.7770 - val_loss: 0.5105 - val_accuracy: 0.7940
Epoch 83/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5330 - acc
uracy: 0.7851 - val_loss: 0.5122 - val_accuracy: 0.7887
Epoch 84/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5352 - acc
uracy: 0.7832 - val_loss: 0.5109 - val_accuracy: 0.7940
Epoch 85/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5363 - acc
uracy: 0.7802 - val_loss: 0.5105 - val_accuracy: 0.7940
Epoch 86/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5313 - acc
uracy: 0.7863 - val_loss: 0.5106 - val_accuracy: 0.7927
Epoch 87/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5348 - acc
uracy: 0.7793 - val_loss: 0.5106 - val_accuracy: 0.7927
Epoch 88/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5338 - acc
uracy: 0.7792 - val_loss: 0.5097 - val_accuracy: 0.7966
Epoch 89/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5319 - acc
uracy: 0.7783 - val_loss: 0.5121 - val_accuracy: 0.7861
Epoch 90/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5345 - acc
uracy: 0.7806 - val_loss: 0.5099 - val_accuracy: 0.7927
Epoch 91/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5335 - acc
uracy: 0.7811 - val_loss: 0.5098 - val_accuracy: 0.7953
Epoch 92/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5319 - acc
uracy: 0.7844 - val_loss: 0.5092 - val_accuracy: 0.7940
Epoch 93/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5338 - ac
curacy: 0.7822 - val_loss: 0.5091 - val_accuracy: 0.7940
Epoch 94/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5336 - acc
uracy: 0.7840 - val_loss: 0.5096 - val_accuracy: 0.7940
Epoch 95/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5340 - acc
uracy: 0.7808 - val_loss: 0.5092 - val_accuracy: 0.7953
```

```
Epoch 96/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5315 - acc
uracy: 0.7816 - val_loss: 0.5102 - val_accuracy: 0.7927
Epoch 97/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5303 - acc
uracy: 0.7819 - val_loss: 0.5092 - val_accuracy: 0.7966
Epoch 98/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5340 - acc
uracy: 0.7808 - val_loss: 0.5080 - val_accuracy: 0.7966
Epoch 99/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5279 - acc
uracy: 0.7867 - val_loss: 0.5076 - val_accuracy: 0.7940
Epoch 100/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5312 - acc
uracy: 0.7827 - val_loss: 0.5068 - val_accuracy: 0.7927
Epoch 101/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5312 - acc
uracy: 0.7815 - val_loss: 0.5088 - val_accuracy: 0.7927
Epoch 102/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5328 - ac
curacy: 0.7773 - val_loss: 0.5078 - val_accuracy: 0.7953
Epoch 103/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5348 - acc
uracy: 0.7825 - val_loss: 0.5082 - val_accuracy: 0.7966
Epoch 104/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5315 - acc
uracy: 0.7822 - val_loss: 0.5089 - val_accuracy: 0.7953
Epoch 105/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5302 - acc
uracy: 0.7841 - val_loss: 0.5083 - val_accuracy: 0.7953
Epoch 106/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5324 - acc
uracy: 0.7815 - val_loss: 0.5079 - val_accuracy: 0.7953
Epoch 107/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5300 - acc
uracy: 0.7831 - val_loss: 0.5075 - val_accuracy: 0.7940
Epoch 108/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5327 - acc
uracy: 0.7794 - val_loss: 0.5077 - val_accuracy: 0.7940
Epoch 109/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5299 - acc
uracy: 0.7843 - val_loss: 0.5066 - val_accuracy: 0.7953
Epoch 110/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5290 - acc
uracy: 0.7796 - val_loss: 0.5069 - val_accuracy: 0.7953
Epoch 111/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5326 - acc
uracy: 0.7822 - val_loss: 0.5069 - val_accuracy: 0.7940
Epoch 112/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5278 - acc
uracy: 0.7835 - val_loss: 0.5070 - val_accuracy: 0.7979
Epoch 113/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5310 - acc
uracy: 0.7824 - val_loss: 0.5073 - val_accuracy: 0.7927
Epoch 114/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5305 - acc
uracy: 0.7780 - val_loss: 0.5062 - val_accuracy: 0.7953
```

```
Epoch 115/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5325 - acc
uracy: 0.7811 - val_loss: 0.5067 - val_accuracy: 0.7953
Epoch 116/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5297 - acc
uracy: 0.7816 - val_loss: 0.5066 - val_accuracy: 0.7953
Epoch 117/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5305 - acc
uracy: 0.7838 - val_loss: 0.5060 - val_accuracy: 0.7966
Epoch 118/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5255 - acc
uracy: 0.7793 - val_loss: 0.5063 - val_accuracy: 0.7953
Epoch 119/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5319 - acc
uracy: 0.7778 - val_loss: 0.5056 - val_accuracy: 0.7927
Epoch 120/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5281 - acc
uracy: 0.7818 - val_loss: 0.5060 - val_accuracy: 0.7940
Epoch 121/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5332 - acc
uracy: 0.7812 - val_loss: 0.5048 - val_accuracy: 0.7953
Epoch 122/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5264 - acc
uracy: 0.7840 - val_loss: 0.5061 - val_accuracy: 0.7940
Epoch 123/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5281 - acc
uracy: 0.7815 - val_loss: 0.5045 - val_accuracy: 0.7940
Epoch 124/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5347 - acc
uracy: 0.7790 - val_loss: 0.5046 - val_accuracy: 0.7953
Epoch 125/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5288 - acc
uracy: 0.7793 - val_loss: 0.5071 - val_accuracy: 0.7940
Epoch 126/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5310 - acc
uracy: 0.7812 - val_loss: 0.5063 - val_accuracy: 0.7966
Epoch 127/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5243 - acc
uracy: 0.7908 - val_loss: 0.5049 - val_accuracy: 0.7966
Epoch 128/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5262 - acc
uracy: 0.7857 - val_loss: 0.5036 - val_accuracy: 0.7966
Epoch 129/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5294 - acc
uracy: 0.7844 - val_loss: 0.5043 - val_accuracy: 0.7927
Epoch 130/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5261 - acc
uracy: 0.7843 - val_loss: 0.5039 - val_accuracy: 0.7940
Epoch 131/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5278 - acc
uracy: 0.7857 - val_loss: 0.5047 - val_accuracy: 0.7953
Epoch 132/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5293 - acc
uracy: 0.7796 - val_loss: 0.5037 - val_accuracy: 0.7966
Epoch 133/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5269 - acc
uracy: 0.7828 - val_loss: 0.5045 - val_accuracy: 0.7940
```

```
Epoch 134/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5294 - acc
uracy: 0.7800 - val_loss: 0.5035 - val_accuracy: 0.7927
Epoch 135/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5272 - acc
uracy: 0.7828 - val_loss: 0.5031 - val_accuracy: 0.7953
Epoch 136/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5295 - acc
uracy: 0.7841 - val_loss: 0.5043 - val_accuracy: 0.7953
Epoch 137/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5290 - acc
uracy: 0.7832 - val_loss: 0.5034 - val_accuracy: 0.7940
Epoch 138/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5261 - acc
uracy: 0.7843 - val_loss: 0.5037 - val_accuracy: 0.7953
Epoch 139/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5288 - acc
uracy: 0.7815 - val_loss: 0.5036 - val_accuracy: 0.7966
Epoch 140/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5269 - acc
uracy: 0.7803 - val_loss: 0.5029 - val_accuracy: 0.7953
Epoch 141/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5229 - acc
uracy: 0.7884 - val_loss: 0.5028 - val_accuracy: 0.7953
Epoch 142/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5272 - acc
uracy: 0.7827 - val_loss: 0.5028 - val_accuracy: 0.7913
Epoch 143/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5263 - acc
uracy: 0.7841 - val_loss: 0.5024 - val_accuracy: 0.7940
Epoch 144/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5290 - acc
uracy: 0.7765 - val_loss: 0.5026 - val_accuracy: 0.7966
Epoch 145/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5267 - acc
uracy: 0.7857 - val_loss: 0.5023 - val_accuracy: 0.7979
Epoch 146/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5288 - acc
uracy: 0.7812 - val_loss: 0.5041 - val_accuracy: 0.7953
Epoch 147/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5269 - acc
uracy: 0.7827 - val_loss: 0.5016 - val_accuracy: 0.7992
Epoch 148/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5276 - acc
uracy: 0.7808 - val_loss: 0.5031 - val_accuracy: 0.7940
Epoch 149/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5260 - acc
uracy: 0.7802 - val_loss: 0.5015 - val_accuracy: 0.7966
Epoch 150/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5233 - acc
uracy: 0.7847 - val_loss: 0.5022 - val_accuracy: 0.7979
Epoch 151/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5284 - acc
uracy: 0.7813 - val_loss: 0.5008 - val_accuracy: 0.7953
Epoch 152/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5240 - ac
curacy: 0.7844 - val_loss: 0.5021 - val_accuracy: 0.7966
```

```
Epoch 153/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5264 - acc
uracy: 0.7837 - val_loss: 0.5025 - val_accuracy: 0.7940
Epoch 154/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5207 - acc
uracy: 0.7870 - val_loss: 0.5003 - val_accuracy: 0.7992
Epoch 155/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5268 - acc
uracy: 0.7813 - val_loss: 0.5012 - val_accuracy: 0.7953
Epoch 156/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5228 - acc
uracy: 0.7834 - val_loss: 0.4996 - val_accuracy: 0.8005
Epoch 157/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5228 - acc
uracy: 0.7851 - val_loss: 0.5003 - val_accuracy: 0.7940
Epoch 158/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5256 - acc
uracy: 0.7854 - val_loss: 0.5013 - val_accuracy: 0.7966
Epoch 159/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5282 - acc
uracy: 0.7816 - val_loss: 0.5009 - val_accuracy: 0.7979
Epoch 160/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5234 - acc
uracy: 0.7848 - val_loss: 0.5020 - val_accuracy: 0.7953
Epoch 161/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5265 - acc
uracy: 0.7812 - val_loss: 0.4995 - val_accuracy: 0.7940
Epoch 162/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5238 - acc
uracy: 0.7847 - val_loss: 0.5013 - val_accuracy: 0.7940
Epoch 163/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5233 - acc
uracy: 0.7841 - val_loss: 0.5000 - val_accuracy: 0.7966
Epoch 164/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5252 - acc
uracy: 0.7835 - val_loss: 0.5007 - val_accuracy: 0.7940
Epoch 165/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5225 - acc
uracy: 0.7859 - val_loss: 0.4994 - val_accuracy: 0.7953
Epoch 166/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5231 - acc
uracy: 0.7894 - val_loss: 0.5003 - val_accuracy: 0.7966
Epoch 167/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5254 - acc
uracy: 0.7856 - val_loss: 0.4996 - val_accuracy: 0.7966
Epoch 168/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5234 - acc
uracy: 0.7857 - val_loss: 0.5002 - val_accuracy: 0.7966
Epoch 169/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5217 - acc
uracy: 0.7831 - val_loss: 0.5003 - val_accuracy: 0.7979
Epoch 170/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5235 - acc
uracy: 0.7834 - val_loss: 0.4989 - val_accuracy: 0.7966
Epoch 171/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5249 - acc
uracy: 0.7816 - val_loss: 0.4997 - val_accuracy: 0.7953
```

```
Epoch 172/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5240 - acc
uracy: 0.7863 - val_loss: 0.4989 - val_accuracy: 0.7966
Epoch 173/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5245 - acc
uracy: 0.7873 - val_loss: 0.4993 - val_accuracy: 0.7992
Epoch 174/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5240 - acc
uracy: 0.7819 - val_loss: 0.4995 - val_accuracy: 0.7979
Epoch 175/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5230 - acc
uracy: 0.7809 - val_loss: 0.4991 - val_accuracy: 0.7953
Epoch 176/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5245 - acc
uracy: 0.7808 - val_loss: 0.4982 - val_accuracy: 0.7940
Epoch 177/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5234 - acc
uracy: 0.7876 - val_loss: 0.4984 - val_accuracy: 0.7966
Epoch 178/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5263 - acc
uracy: 0.7844 - val_loss: 0.4990 - val_accuracy: 0.7966
Epoch 179/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5227 - acc
uracy: 0.7821 - val_loss: 0.5002 - val_accuracy: 0.7940
Epoch 180/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5282 - acc
uracy: 0.7830 - val_loss: 0.4979 - val_accuracy: 0.7953
Epoch 181/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5232 - acc
uracy: 0.7840 - val_loss: 0.4988 - val_accuracy: 0.7966
Epoch 182/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5229 - acc
uracy: 0.7863 - val_loss: 0.4980 - val_accuracy: 0.7966
Epoch 183/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5228 - acc
uracy: 0.7875 - val_loss: 0.4987 - val_accuracy: 0.7979
Epoch 184/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5233 - acc
uracy: 0.7851 - val_loss: 0.4975 - val_accuracy: 0.8005
Epoch 185/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5229 - acc
uracy: 0.7851 - val_loss: 0.4983 - val_accuracy: 0.7979
Epoch 186/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5215 - acc
uracy: 0.7870 - val_loss: 0.4979 - val_accuracy: 0.7966
Epoch 187/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5232 - acc
uracy: 0.7873 - val_loss: 0.4969 - val_accuracy: 0.7992
Epoch 188/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5246 - acc
uracy: 0.7846 - val_loss: 0.4979 - val_accuracy: 0.7979
Epoch 189/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5206 - acc
uracy: 0.7873 - val_loss: 0.4967 - val_accuracy: 0.7953
Epoch 190/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5231 - acc
uracy: 0.7827 - val_loss: 0.4979 - val_accuracy: 0.7966
```

```
Epoch 191/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5243 - acc
uracy: 0.7812 - val_loss: 0.4968 - val_accuracy: 0.8005
Epoch 192/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5216 - acc
uracy: 0.7879 - val_loss: 0.4969 - val_accuracy: 0.7953
Epoch 193/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5210 - acc
uracy: 0.7815 - val_loss: 0.4974 - val_accuracy: 0.7953
Epoch 194/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5188 - acc
uracy: 0.7856 - val_loss: 0.4955 - val_accuracy: 0.7992
Epoch 195/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5212 - acc
uracy: 0.7859 - val_loss: 0.4979 - val_accuracy: 0.7927
Epoch 196/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5236 - acc
uracy: 0.7821 - val_loss: 0.4960 - val_accuracy: 0.8031
Epoch 197/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5221 - acc
uracy: 0.7867 - val_loss: 0.4964 - val_accuracy: 0.7992
Epoch 198/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5237 - acc
uracy: 0.7816 - val_loss: 0.4966 - val_accuracy: 0.7966
Epoch 199/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5227 - acc
uracy: 0.7848 - val_loss: 0.4967 - val_accuracy: 0.7992
Epoch 200/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5226 - acc
uracy: 0.7783 - val_loss: 0.4987 - val_accuracy: 0.7966
Epoch 201/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5203 - acc
uracy: 0.7857 - val_loss: 0.4955 - val_accuracy: 0.7979
Epoch 202/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5228 - acc
uracy: 0.7850 - val_loss: 0.4958 - val_accuracy: 0.7992
Epoch 203/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5206 - ac
curacy: 0.7828 - val_loss: 0.4957 - val_accuracy: 0.7953
Epoch 204/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5205 - acc
uracy: 0.7863 - val_loss: 0.4965 - val_accuracy: 0.7979
Epoch 205/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5225 - ac
curacy: 0.7886 - val_loss: 0.4956 - val_accuracy: 0.7966
Epoch 206/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5231 - ac
curacy: 0.7872 - val_loss: 0.4956 - val_accuracy: 0.7979
Epoch 207/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5212 - acc
uracy: 0.7865 - val_loss: 0.4968 - val_accuracy: 0.7979
Epoch 208/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5206 - acc
uracy: 0.7812 - val_loss: 0.4956 - val_accuracy: 0.7979
Epoch 209/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5210 - acc
uracy: 0.7875 - val_loss: 0.4955 - val_accuracy: 0.7992
```

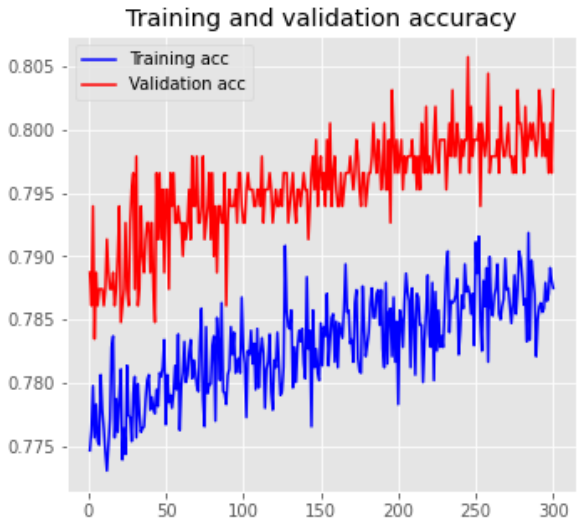
```
Epoch 210/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5232 - ac
curacy: 0.7850 - val_loss: 0.4959 - val_accuracy: 0.7966
Epoch 211/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5223 - acc
uracy: 0.7806 - val_loss: 0.4955 - val_accuracy: 0.7979
Epoch 212/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5204 - acc
uracy: 0.7870 - val_loss: 0.4953 - val_accuracy: 0.7953
Epoch 213/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5184 - acc
uracy: 0.7867 - val_loss: 0.4950 - val_accuracy: 0.7979
Epoch 214/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5187 - acc
uracy: 0.7846 - val_loss: 0.4940 - val_accuracy: 0.7953
Epoch 215/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5174 - acc
uracy: 0.7846 - val_loss: 0.4944 - val_accuracy: 0.8005
Epoch 216/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5237 - acc
uracy: 0.7800 - val_loss: 0.4947 - val_accuracy: 0.7979
Epoch 217/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5208 - ac
curacy: 0.7834 - val_loss: 0.4952 - val_accuracy: 0.7966
Epoch 218/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5208 - ac
curacy: 0.7857 - val_loss: 0.4937 - val_accuracy: 0.8018
Epoch 219/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5192 - acc
uracy: 0.7885 - val_loss: 0.4949 - val_accuracy: 0.7966
Epoch 220/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5186 - acc
uracy: 0.7838 - val_loss: 0.4937 - val_accuracy: 0.7979
Epoch 221/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5170 - acc
uracy: 0.7828 - val_loss: 0.4934 - val_accuracy: 0.7966
Epoch 222/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5179 - acc
uracy: 0.7879 - val_loss: 0.4935 - val_accuracy: 0.7966
Epoch 223/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5242 - acc
uracy: 0.7802 - val_loss: 0.4939 - val_accuracy: 0.7992
Epoch 224/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5214 - acc
uracy: 0.7872 - val_loss: 0.4942 - val_accuracy: 0.8018
Epoch 225/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5216 - acc
uracy: 0.7825 - val_loss: 0.4948 - val_accuracy: 0.7979
Epoch 226/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5169 - acc
uracy: 0.7857 - val_loss: 0.4935 - val_accuracy: 0.7966
Epoch 227/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5189 - acc
uracy: 0.7828 - val_loss: 0.4939 - val_accuracy: 0.7992
Epoch 228/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5182 - acc
uracy: 0.7837 - val_loss: 0.4942 - val_accuracy: 0.7992
```

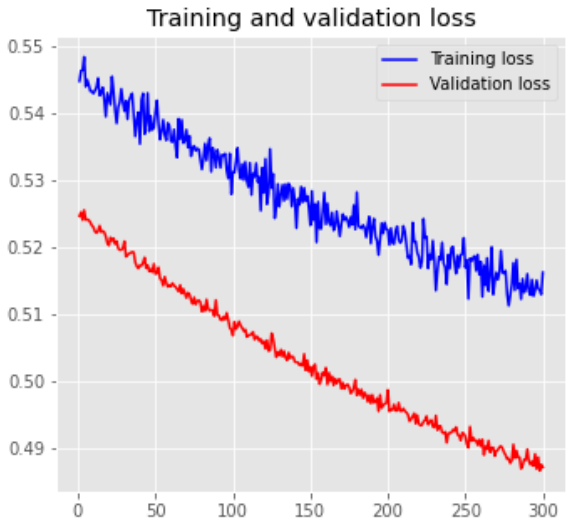
```
Epoch 229/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5210 - ac
curacy: 0.7828 - val_loss: 0.4932 - val_accuracy: 0.7992
Epoch 230/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5180 - acc
uracy: 0.7859 - val_loss: 0.4931 - val_accuracy: 0.7992
Epoch 231/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5135 - acc
uracy: 0.7891 - val_loss: 0.4948 - val_accuracy: 0.7953
Epoch 232/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5164 - acc
uracy: 0.7904 - val_loss: 0.4927 - val_accuracy: 0.7979
Epoch 233/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5186 - acc
uracy: 0.7840 - val_loss: 0.4933 - val_accuracy: 0.8005
Epoch 234/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5181 - acc
uracy: 0.7865 - val_loss: 0.4933 - val_accuracy: 0.7979
Epoch 235/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5207 - acc
uracy: 0.7863 - val_loss: 0.4923 - val_accuracy: 0.8031
Epoch 236/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5176 - acc
uracy: 0.7865 - val_loss: 0.4923 - val_accuracy: 0.7979
Epoch 237/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5170 - acc
uracy: 0.7869 - val_loss: 0.4924 - val_accuracy: 0.7979
Epoch 238/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5145 - acc
uracy: 0.7882 - val_loss: 0.4908 - val_accuracy: 0.8005
Epoch 239/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5189 - acc
uracy: 0.7822 - val_loss: 0.4922 - val_accuracy: 0.7992
Epoch 240/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5168 - acc
uracy: 0.7841 - val_loss: 0.4922 - val_accuracy: 0.7966
Epoch 241/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5199 - acc
uracy: 0.7869 - val_loss: 0.4924 - val_accuracy: 0.7992
Epoch 242/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5201 - acc
uracy: 0.7859 - val_loss: 0.4927 - val_accuracy: 0.7979
Epoch 243/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5177 - acc
uracy: 0.7894 - val_loss: 0.4925 - val_accuracy: 0.7992
Epoch 244/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5161 - acc
uracy: 0.7872 - val_loss: 0.4931 - val_accuracy: 0.7992
Epoch 245/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5187 - acc
uracy: 0.7876 - val_loss: 0.4920 - val_accuracy: 0.8058
Epoch 246/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5170 - acc
uracy: 0.7870 - val_loss: 0.4919 - val_accuracy: 0.7966
Epoch 247/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5188 - acc
uracy: 0.7854 - val_loss: 0.4912 - val_accuracy: 0.8018
```

```
Epoch 248/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5190 - acc
uracy: 0.7870 - val_loss: 0.4923 - val_accuracy: 0.7979
Epoch 249/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5214 - acc
uracy: 0.7830 - val_loss: 0.4922 - val_accuracy: 0.7992
Epoch 250/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5170 - acc
uracy: 0.7911 - val_loss: 0.4917 - val_accuracy: 0.7992
Epoch 251/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5181 - acc
uracy: 0.7886 - val_loss: 0.4906 - val_accuracy: 0.7992
Epoch 252/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5123 - acc
uracy: 0.7916 - val_loss: 0.4901 - val_accuracy: 0.8005
Epoch 253/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5183 - acc
uracy: 0.7848 - val_loss: 0.4932 - val_accuracy: 0.7940
Epoch 254/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5179 - acc
uracy: 0.7825 - val_loss: 0.4911 - val_accuracy: 0.8005
Epoch 255/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5185 - acc
uracy: 0.7881 - val_loss: 0.4916 - val_accuracy: 0.7992
Epoch 256/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5187 - acc
uracy: 0.7860 - val_loss: 0.4911 - val_accuracy: 0.7979
Epoch 257/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5165 - ac
curacy: 0.7891 - val_loss: 0.4912 - val_accuracy: 0.7992
Epoch 258/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5189 - ac
curacy: 0.7816 - val_loss: 0.4905 - val_accuracy: 0.8045
Epoch 259/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5159 - ac
curacy: 0.7900 - val_loss: 0.4912 - val_accuracy: 0.7966
Epoch 260/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5178 - acc
uracy: 0.7860 - val_loss: 0.4903 - val_accuracy: 0.7979
Epoch 261/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5126 - acc
uracy: 0.7853 - val_loss: 0.4913 - val_accuracy: 0.7979
Epoch 262/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5173 - acc
uracy: 0.7869 - val_loss: 0.4890 - val_accuracy: 0.7979
Epoch 263/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5143 - acc
uracy: 0.7875 - val_loss: 0.4902 - val_accuracy: 0.7979
Epoch 264/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5184 - acc
uracy: 0.7894 - val_loss: 0.4899 - val_accuracy: 0.7992
Epoch 265/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5135 - acc
uracy: 0.7854 - val_loss: 0.4911 - val_accuracy: 0.7966
Epoch 266/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5158 - acc
uracy: 0.7850 - val_loss: 0.4894 - val_accuracy: 0.8018
```

```
Epoch 267/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5200 - acc
uracy: 0.7860 - val_loss: 0.4900 - val_accuracy: 0.7992
Epoch 268/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5129 - acc
uracy: 0.7866 - val_loss: 0.4901 - val_accuracy: 0.7979
Epoch 269/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5149 - acc
uracy: 0.7898 - val_loss: 0.4888 - val_accuracy: 0.7979
Epoch 270/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5162 - acc
uracy: 0.7873 - val_loss: 0.4896 - val_accuracy: 0.7992
Epoch 271/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5152 - acc
uracy: 0.7875 - val_loss: 0.4897 - val_accuracy: 0.8005
Epoch 272/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5163 - acc
uracy: 0.7860 - val_loss: 0.4901 - val_accuracy: 0.7979
Epoch 273/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5195 - acc
uracy: 0.7847 - val_loss: 0.4903 - val_accuracy: 0.7979
Epoch 274/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5162 - ac
curacy: 0.7862 - val_loss: 0.4900 - val_accuracy: 0.7966
Epoch 275/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5161 - acc
uracy: 0.7885 - val_loss: 0.4894 - val_accuracy: 0.7979
Epoch 276/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5165 - acc
uracy: 0.7854 - val_loss: 0.4890 - val_accuracy: 0.7966
Epoch 277/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5129 - acc
uracy: 0.7873 - val_loss: 0.4884 - val_accuracy: 0.8031
Epoch 278/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5113 - acc
uracy: 0.7904 - val_loss: 0.4884 - val_accuracy: 0.8005
Epoch 279/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5128 - acc
uracy: 0.7898 - val_loss: 0.4878 - val_accuracy: 0.8005
Epoch 280/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5144 - acc
uracy: 0.7884 - val_loss: 0.4882 - val_accuracy: 0.7992
Epoch 281/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5177 - acc
uracy: 0.7862 - val_loss: 0.4905 - val_accuracy: 0.7966
Epoch 282/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5136 - acc
uracy: 0.7866 - val_loss: 0.4885 - val_accuracy: 0.8018
Epoch 283/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5182 - acc
uracy: 0.7832 - val_loss: 0.4896 - val_accuracy: 0.7979
Epoch 284/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5139 - ac
curacy: 0.7919 - val_loss: 0.4886 - val_accuracy: 0.7992
Epoch 285/300
7/7 [==============================] - 0s 11ms/step - loss: 0.5146 - ac
curacy: 0.7834 - val_loss: 0.4886 - val_accuracy: 0.8005
```

```
Epoch 286/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5123 - ac
curacy: 0.7897 - val_loss: 0.4869 - val_accuracy: 0.7992
Epoch 287/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5153 - ac
curacy: 0.7882 - val_loss: 0.4875 - val_accuracy: 0.8005
Epoch 288/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5133 - ac
curacy: 0.7870 - val_loss: 0.4883 - val_accuracy: 0.7992
Epoch 289/300
7/7 [==============================] - 0s 11ms/step - loss: 0.5139 - ac
curacy: 0.7821 - val_loss: 0.4883 - val_accuracy: 0.7979
Epoch 290/300
7/7 [==============================] - 0s 10ms/step - loss: 0.5160 - ac
curacy: 0.7850 - val_loss: 0.4898 - val_accuracy: 0.7992
Epoch 291/300
7/7 [==============================] - 0s 14ms/step - loss: 0.5134 - ac
curacy: 0.7860 - val_loss: 0.4881 - val_accuracy: 0.8031
Epoch 292/300
7/7 [==============================] - 0s 12ms/step - loss: 0.5129 - ac
curacy: 0.7863 - val_loss: 0.4875 - val_accuracy: 0.8018
Epoch 293/300
7/7 [==============================] - 0s 11ms/step - loss: 0.5152 - ac
curacy: 0.7856 - val_loss: 0.4880 - val_accuracy: 0.7979
Epoch 294/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5127 - acc
uracy: 0.7859 - val_loss: 0.4874 - val_accuracy: 0.8005
Epoch 295/300
7/7 [==============================] - 0s 11ms/step - loss: 0.5135 - ac
curacy: 0.7879 - val_loss: 0.4891 - val_accuracy: 0.7979
Epoch 296/300
7/7 [==============================] - 0s 13ms/step - loss: 0.5151 - ac
curacy: 0.7865 - val_loss: 0.4869 - val_accuracy: 0.7992
Epoch 297/300
7/7 [==============================] - 0s 15ms/step - loss: 0.5138 - ac
curacy: 0.7870 - val_loss: 0.4886 - val_accuracy: 0.7966
Epoch 298/300
7/7 [==============================] - 0s 19ms/step - loss: 0.5136 - ac
curacy: 0.7891 - val_loss: 0.4866 - val_accuracy: 0.8005
Epoch 299/300
7/7 [==============================] - 0s 9ms/step - loss: 0.5130 - acc
uracy: 0.7881 - val_loss: 0.4875 - val_accuracy: 0.7966
Epoch 300/300
7/7 [==============================] - 0s 8ms/step - loss: 0.5162 - acc
uracy: 0.7875 - val_loss: 0.4872 - val_accuracy: 0.8031
```





In [98]:
```
#adding bert
```

In [100]:
```
import tensorflow as tf
import tensorflow_hub as hub
from tensorflow.keras.callbacks import ModelCheckpoint
```

In [113]:
```
%%time
module_url = 'https://tfhub.dev/google/universal-sentence-encoder-large/4'
embed = hub.KerasLayer(module_url, trainable=False, name='USE_embedding')
```

```
CPU times: user 10.4 s, sys: 2.58 s, total: 13 s
Wall time: 13.3 s
```

In [114]:
```
sub1 = pd.read_csv("sample_submission.csv")
sub1.head(3)
sub1['target'] = preds
sub1.to_csv('submission_BERT.csv', index=False)
sub1.head(3)
```

Out[114]:

	id	target
0	0	0
1	2	0
2	3	1

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: