Untitled

February 27, 2022

[1]:

```python
import numpy as np
import tensorflow as tf
import ktrain
import pandas as pd
from ktrain import text
```

[2]:

```python
df = pd.read_excel('train.xlsx', dtype=str)
df = df[:1000]
```

[3]:

```python
def clean_text(text):
    import re
    import string

    text = str(text).lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub('https?://\S+|www\.\S+', '', text)
    text = re.sub('<.*?>+', '', text)
    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)
    text = re.sub('\n', '', text)
    text = re.sub('\w*\d\w*', '', text)
    return text
```

[4]:

```python
def preprocess_data(text):
    import nltk
    from nltk.corpus import stopwords

    stop_words = stopwords.words('english')
    stemmer    = nltk.SnowballStemmer("english")

    text = clean_text(text)
    text = ' '.join(word for word in text.split() if word not in stop_words)
    text = ' '.join(stemmer.stem(word) for word in text.split())
    return text
```

[5]:

```python
df['Reviews'].apply(preprocess_data)
```

[5]:

```
0       first tune morn news thought wow final enterta…
1       mere thought go overboard aka babe ahoy make w…
```

```
2      movi fall well standard ultim answer lie poor …
3      wow thought steven segal movi bad everi time t…
4      stori seen matter figur make proper storyboard…
                              …
995    fortun us real mccoy fan like babi boomer grew…
996    mechenoset one beauti romant movi ive ever see…
997    film never receiv attent deserv although one f…
998    absolut love tom robbin book excit interest se…
999    arguabl john thaw finest perform success shake…
Name: Reviews, Length: 1000, dtype: object
```

0.1 Test

[6]:

```
data = pd.read_excel('test.xlsx', dtype=str)
data = data[:1000]
data['Reviews'].apply(preprocess_data)
```

[6]:

```
0      would thought movi man drive coupl hundr mile …
1      realiz go around us news home whole new world …
2      grew watch origin disney cinderella alway love…
3      david mamet wrote screenplay made directori de…
4      admit didnt high expect corki romano howev fel…
                              …
995    ok flick set mexico hitman scott glenn hitch r…
996    dont want go rant butthi worst film ive ever s…
997    love stori somewher poster said famili like on…
998    came nanci drew expect worstbecaus everyon els…
999    saw film televis year ago sever year wake morn…
Name: Reviews, Length: 1000, dtype: object
```

[7]:

```
(X_train, y_train), (X_test, y_test), preprocess = text.
 ↪texts_from_df(train_df=df,
                                                                   ␣
 ↪text_column='Reviews',
                                                                   ␣
 ↪label_columns = 'Sentiment',
                                                                   ␣
 ↪val_df=data,
                                                                   ␣
 ↪maxlen=400,
                                                                   ␣
 ↪preprocess_mode='bert')

# 500 will take 3-4 hour
# 400 will take 1-2 hour
```

```
['neg', 'pos']
```

```
   neg  pos
0  1.0  0.0
1  1.0  0.0
2  1.0  0.0
3  1.0  0.0
4  1.0  0.0
['neg', 'pos']
   neg  pos
0  0.0  1.0
1  0.0  1.0
2  1.0  0.0
3  0.0  1.0
4  1.0  0.0
preprocessing train…
language: en
```

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

```
Is Multi-Label? False
preprocessing test…
language: en
```

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

[8]:

```
X_train[0].shape, X_test[0].shape
```

[8]:

```
((1000, 400), (1000, 400))
```

[9]:

```
model = text.text_classifier(name='bert',
                             train_data=(X_train, y_train),
                             preproc=preprocess)
```

```
Is Multi-Label? False
maxlen is 400
done.
```

0.2 Learning Rate

[10]:

```
learner = ktrain.get_learner(model=model,
                             train_data=(X_train, y_train),
                             val_data=(X_test, y_test),
                             batch_size=6)
```

[11]:

```
# these might take days

# learner.lr_find()
# learner.lr_plot()

# optimal learning rate
```

```
[ ]: learner.fit_onecycle(lr=2e-5, epochs=1)
```

```
begin training using onecycle policy with max lr of 2e-05…
```

1 Prediction

```
[ ]: predictor = ktrain.get_predictor(learner.model, preprocess)
```

```
[ ]: prediction = ['this movie sucks']
     predictor.predict(prediction)
```