preprocess

August 14, 2021

[1]:
```
# !pip install normalise

import nltk
# nltk.download('brown')
# nltk.download('names')
# nltk.download('wordnet')
# nltk.download('averaged_perceptron_tagger')
# nltk.download('universal_tagset')
# nltk.download('stopwords')
# nltk.download('punkt')

from nltk.tokenize import TreebankWordTokenizer
from nltk.stem import PorterStemmer
from normalise import normalise
import numpy as np

porter=PorterStemmer()
tokenizer = TreebankWordTokenizer()
stem_words = np.vectorize(porter.stem)

from tqdm.notebook import tqdm
```

```
/home/behzad/anaconda3/envs/cns/lib/python3.8/site-
packages/sklearn/utils/deprecation.py:143: FutureWarning: The
sklearn.semi_supervised.label_propagation module is  deprecated in version 0.22
and will be removed in version 0.24. The corresponding classes / functions
should instead be imported from sklearn.semi_supervised. Anything that cannot be
imported from sklearn.semi_supervised is now part of the private API.
  warnings.warn(message, FutureWarning)
/home/behzad/anaconda3/envs/cns/lib/python3.8/site-packages/sklearn/base.py:329:
UserWarning: Trying to unpickle estimator LabelPropagation from version 0.18
when using version 0.23.1. This might lead to breaking code or invalid results.
Use at your own risk.
  warnings.warn(
```

[2]:
```
import pandas as pd
df = pd.read_csv('data.csv')
df.head()
```

[2]:
```
   index                                              authors        category  \
0      0  Katherine LaGrave, ContributorTravel writer an…          TRAVEL
1      1                                        Ben Hallman        BUSINESS
2      2                                    Jessica Misener  STYLE & BEAUTY
3      3        Victor and Mary, Contributor\n2Sense-LA.com          TRAVEL
4      4                            Emily Cohn, Contributor        BUSINESS

         date                                           headline  \
0  2014-05-07  EccentriCities: Bingo Parties, Paella and Isla…
1  2014-06-09  Lawyers Are Now The Driving Force Behind Mortg…
2  2012-03-12  Madonna 'Truth Or Dare' Shoe Line To Debut Thi…
3  2013-12-17  Sophistication and Serenity on the Las Vegas S…
4  2015-03-19  It's Still Pretty Hard For Women To Get Free B…

                                                link  \
0  https://www.huffingtonpost.com/entry/eccentric…
1  https://www.huffingtonpost.com/entry/mortgage-…
2  https://www.huffingtonpost.com/entry/madonna-s…
3  https://www.huffingtonpost.com/entry/las-vegas…
4  https://www.huffingtonpost.com/entry/free-birt…

                                  short_description
0  Påskekrim is merely the tip of the proverbial …
1                                               NaN
2  Madonna is slinking her way into footwear now,…
3  But what if you're a 30-something couple that …
4  Obamacare was supposed to make birth control f…
```

[11]:
```python
import json

data = []
targets = []
for (i, row) in tqdm(df.iterrows(), total=df.shape[0]):
    try:
        doc = row.short_description
        if type(doc) is not str:
            continue
        edited = stem_words(
            tokenizer.tokenize(
                ' '.join(
                    normalise(
                        tokenizer.tokenize(
                            doc.lower()
                        ),
                        verbose=False
                    )
                )
```

```
            )
        ).tolist()
        data.append(edited)
        targets.append(row.category)
    except:
        continue
```

```
  0%|          | 0/22925 [00:00<?, ?it/s]
```

[15]:
```
train, test = data[:20000], data[20000:]
train_targets, test_targets = targets[:20000], targets[20000:]

json.dump(train, open('data.json','w'))
json.dump(train_targets, open('targets.json','w'))

json.dump(test, open('test_data.json','w'))
json.dump(test_targets, open('test_targets.json','w'))

words = [word for doc in data for word in doc]
words, counts = np.unique(words, return_counts=True)
words = words[counts>1]
words = ['<UKN>','<s>','</s>']+list(words)
json.dump(words, open('words.json','w'))

vocab = {w:i for i,w in enumerate(words)}
vocab['<PAD>'] = -1
json.dump(vocab, open('vocab.json','w'))
```

[ ]:
```

```