WEEK 9 - Text Tokenization

nltk - National Language Toolkit - part of Anaconda
Looking into frequency of words
find other significant words in text
Identify words found in Tweets or posts

```
>>> import nltk
>>> text = "I'll never fly Delta again!!  My flight was
supposed to leave MCO for ATL at 6:25pm on Saturday,
March 26, however, due to severe weather, it was
delayed until 8:12pm - no problem. At approx. 8pm we
started boarding.  We just sat there at the gate.  No
explanation etc. until about 9:30pm when the pilot said
we were pushing away from the gate but wouldn't take
off. "
>>> words = text.split()
>>> words
["I'll", 'never', 'fly', 'Delta', 'again!!', 'My',
'flight', 'was', 'supposed', 'to', 'leave', 'MCO',
'for', 'ATL', 'at', '6:25pm', 'on', 'Saturday,',
'March', '26,', 'however,', 'due', 'to', 'severe',
'weather,', 'it', 'was', 'delayed', 'until', '8:12pm',
'-', 'no', 'problem.', 'At', 'approx.', '8pm', 'we',
'started', 'boarding.', 'We', 'just', 'sat', 'there',
'at', 'the', 'gate.', 'No', 'explanation', 'etc.',
'until', 'about', '9:30pm', 'when', 'the', 'pilot',
'said', 'we', 'were', 'pushing', 'away', 'from', 'the',
'gate', 'but', "wouldn't", 'take', 'off.']
>>> import pymongo
>>> client = pymongo.MongoClient()
>>> client.database_names()
['admin', 'bball', 'fbusers', 'lax', 'local',
'peopledb', 'usgs']
Getting data from Facebook
>>> db = client.fbusers
>>> db.collection_names()
['delta']
>>> coll = db.delta
>>> docs = coll.find()
```

```
>>> doclist = list(docs)
>>> msglist = [doc['message'] for doc in doclist if
'message' in doc.keys()]
>>> len(msglist)
397
>>> all_tokens = [tok for msg in msglist for tok in
nltk.word_tokenize(msg)]
>>> len(all_tokens)
29727
>>> all_tokens[:50]
['@', 'Delta', 'my', '63', 'year', 'old', 'father',
'couldn', ''', 't', 'get', 'a', 'drink', 'on', 'your',
'flights', 'and', 'I', ''', 'm', 'calling', 'to',
'complain', 'but', 'hold', 'time', 'ETA', 'is', '2',
'hours', '.', 'For', 'a', '$', '465', 'flight', 'from',
'MYR', 'TO', 'NYC', 'this', 'is', 'Unacceptable', '.',
'If', 'you', 'could', 'kindly', 'help', 'me']
>>> msgFD = nltk.FreqDist(all_tokens)
>>> msgFD.most_common(30)
[('.', 1323), ('to', 1063), ('the', 996), (',', 704),
('and', 691), ('I', 503), ('a', 495), ('Delta', 376),
('for', 359), ('of', 336), ('in', 321), ('on', 284),
('!', 266), ('flight', 241), ('is', 230), (''', 223),
('with', 215), ('you', 209), ('my', 202), ('was', 199),
('that', 169), ('from', 163), ('have', 159), ('are',
153), ('at', 141), ('it', 133), ('this', 131), ('be',
124), ('our', 123), ('?', 119)]

>>> all_tokens = [tok.lower() for msg in msglist for
tok in nltk.word_tokenize(msg)]
>>> all_tokens[:30]
['@', 'delta', 'my', '63', 'year', 'old', 'father',
'couldn', ''', 't', 'get', 'a', 'drink', 'on', 'your',
'flights', 'and', 'i', ''', 'm', 'calling', 'to',
'complain', 'but', 'hold', 'time', 'eta', 'is', '2',
'hours']

>>> nltk_stopwords =
nltk.corpus.stopwords.words('english')
>>> len(nltk_stopwords)
```

```
179
>>> nltk_stopwords
['i', 'me', 'my', 'myself', 'we', 'our', 'ours',
'ourselves', 'you', "you're", "you've", "you'll",
"you'd", 'your', 'yours', 'yourself', 'yourselves',
'he', 'him', 'his', 'himself', 'she', "she's", 'her',
'hers', 'herself', 'it', "it's", 'its', 'itself',
'they', 'them', 'their', 'theirs', 'themselves',
'what', 'which', 'who', 'whom', 'this', 'that',
"that'll", 'these', 'those', 'am', 'is', 'are', 'was',
'were', 'be', 'been', 'being', 'have', 'has', 'had',
'having', 'do', 'does', 'did', 'doing', 'a', 'an',
'the', 'and', 'but', 'if', 'or', 'because', 'as',
'until', 'while', 'of', 'at', 'by', 'for', 'with',
'about', 'against', 'between', 'into', 'through',
'during', 'before', 'after', 'above', 'below', 'to',
'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over',
'under', 'again', 'further', 'then', 'once', 'here',
'there', 'when', 'where', 'why', 'how', 'all', 'any',
'both', 'each', 'few', 'more', 'most', 'other', 'some',
'such', 'no', 'nor', 'not', 'only', 'own', 'same',
'so', 'than', 'too', 'very', 's', 't', 'can', 'will',
'just', 'don', "don't", 'should', "should've", 'now',
'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren',
"aren't", 'couldn', "couldn't", 'didn', "didn't",
'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't",
'haven', "haven't", 'isn', "isn't", 'ma', 'mightn',
"mightn't", 'mustn', "mustn't", 'needn', "needn't",
'shan', "shan't", 'shouldn', "shouldn't", 'wasn',
"wasn't", 'weren', "weren't", 'won', "won't", 'wouldn',
"wouldn't"]
>>> import re
>>> def alpha_filter(w):
...     pattern = re.compile('^[^a-z]+$')
...     if (pattern.match(w)):
...       return True
...     else:
...       return False
...
>>> token_list = [tok for tok in all_tokens if not
```

```
alpha_filter(tok)]
>>> token_list[:30]
['delta', 'my', 'year', 'old', 'father', 'couldn', 't',
'get', 'a', 'drink', 'on', 'your', 'flights', 'and',
'i', 'm', 'calling', 'to', 'complain', 'but', 'hold',
'time', 'eta', 'is', 'hours', 'for', 'a', 'flight',
'from', 'myr']
>>> msgFD = nltk.FreqDist(token_list)
>>> top_words = msgFD.most_common(30)
>>> for word, freq in top_words:
...     print(word, freq)
...
to 1084
the 1066
and 718
a 527
i 513
delta 419
for 373
of 341
in 332
on 298
flight 257
you 251
is 244
my 230
with 226
was 202
we 185
that 179
this 169
from 167
have 166
it 165
are 157
at 151
our 139
be 127
as 126
airport 126
```

```
your 125
not 114
>>> tweet = "RT @OccupySandy: Good Morning NYC.
http://t.co/yRLgrB53 #NotAnotherKatrina#sandy"
>>> tokens = nltk.word_tokenize(tweet)
>>> tokens
['RT', '@', 'OccupySandy', ':', 'Good', 'Morning',
'NYC', '.', 'http', ':', '//t.co/yRLgrB53', '#',
'NotAnotherKatrina', '#', 'sandy']
>>> ttokenizer = nltk.tokenize.TweetTokenizer()
>>> tokens = ttokenizer.tokenize(tweet)
>>> tokens
['RT', '@OccupySandy', ':', 'Good', 'Morning', 'NYC',
'.', 'http://t.co/yRLgrB53', '#NotAnotherKatrina',
'#sandy']
Using Tokenizer on FB Posts
>>> newtokens = [tok.lower() for msg in msglist for tok
in ttokenizer.tokenize(msg)]
>>> newtokens[:50]
['@delta', 'my', '63', 'year', 'old', 'father',
'couldn', "'", 't', 'get', 'a', 'drink', 'on', 'your',
'flights', 'and', 'i', "'", 'm', 'calling', 'to',
'complain', 'but', 'hold', 'time', 'eta', 'is', '2',
'hours', '.', 'for', 'a', '$', '465', 'flight', 'from',
'myr', 'to', 'nyc', 'this', 'is', 'unacceptable', '.',
'if', 'you', 'could', 'kindly', 'help', 'me', 'i']
>>>
```