TASK 4 STOP WORDS

In [1]:
```
from nltk.corpus import stopwords
```

In [2]:
```
stopwords.words('english')
```

Out[2]:
```
['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"]
```

In [3]:
```
stopwords.words('french')
```

Out[3]:
```
['au',
 'aux',
 'avec',
 'ce',
 'ces',
 'dans',
 'de',
 'des',
 'du',
 'elle',
 'en',
 'et',
 'eux',
 'il',
 'ils',
 'je',
 'la',
 'le',
 'les',
 'leur',
 'lui',
 'ma',
 'mais',
 'me',
 'même',
 'mes',
 'moi',
 'mon',
 'ne',
 'nos',
 'notre',
 'nous',
 'on',
 'ou',
 'par',
 'pas',
 'pour',
 'qu',
 'que',
 'qui',
 'sa',
 'se',
 'ses',
 'son',
 'sur',
 'ta',
 'te',
 'tes',
 'toi',
 'ton',
 'tu',
 'un',
 'une',
 'vos',
 'votre',
 'vous',
 'c',
 'd',
 'j',
 'l',
 'à',
 'm',
 'n',
 's',
 't',
 'y',
 'été',
 'étée',
 'étées',
 'étés',
 'étant',
 'étante',
 'étants',
 'étantes',
 'suis',
 'es',
 'est',
 'sommes',
 'êtes',
 'sont',
 'serai',
 'seras',
 'sera',
 'serons',
 'serez',
 'seront',
 'serais',
 'serait',
 'serions',
 'seriez',
 'seraient',
 'étais',
 'était',
 'étions',
 'étiez',
 'étaient',
 'fus',
 'fut',
 'fûmes',
 'fûtes',
 'furent',
 'sois',
 'soit',
 'soyons',
 'soyez',
 'soient',
 'fusse',
 'fusses',
 'fût',
 'fussions',
 'fussiez',
 'fussent',
 'ayant',
 'ayante',
 'ayantes',
 'ayants',
 'eu',
 'eue',
 'eues',
 'eus',
 'ai',
 'as',
 'avons',
 'avez',
 'ont',
 'aurai',
 'auras',
 'aura',
 'aurons',
 'aurez',
 'auront',
 'aurais',
 'aurait',
 'aurions',
 'auriez',
 'auraient',
 'avais',
 'avait',
 'avions',
 'aviez',
 'avaient',
 'eut',
 'eûmes',
 'eûtes',
 'eurent',
 'aie',
 'aies',
 'ait',
 'ayons',
 'ayez',
 'aient',
 'eusse',
 'eusses',
 'eût',
 'eussions',
 'eussiez',
 'eussent']
```

In [5]:
```
import nltk
```

In [6]:
```
entries = nltk.corpus.cmudict.entries()
```

In [7]:
```
len(entries)
```

Out[7]: 133737

In [8]:
```
for entry in entries[500:515]:
    print(entry)
```

```
('accessories', ['AE0', 'K', 'S', 'EH1', 'S', 'ER0', 'IY0', 'Z'])
('accessorize', ['AE0', 'K', 'S', 'EH1', 'S', 'ER0', 'AY2', 'Z'])
('accessorized', ['AE0', 'K', 'S', 'EH1', 'S', 'ER0', 'AY2', 'Z', 'D'])
('accessory', ['AE0', 'K', 'S', 'EH1', 'S', 'ER0', 'IY0'])
('accetta', ['AA0', 'CH', 'EH1', 'T', 'AH0'])
('accident', ['AE1', 'K', 'S', 'AH0', 'D', 'AH0', 'N', 'T'])
('accidental', ['AE2', 'K', 'S', 'AH0', 'D', 'EH1', 'N', 'T', 'AH0', 'L'])
('accidental', ['AE2', 'K', 'S', 'AH0', 'D', 'EH1', 'N', 'AH0', 'L'])
('accidentally', ['AE2', 'K', 'S', 'AH0', 'D', 'EH1', 'N', 'T', 'AH0', 'L', 'IY0'])
('accidentally', ['AE2', 'K', 'S', 'AH0', 'D', 'EH1', 'N', 'AH0', 'L', 'IY0'])
('accidently', ['AE1', 'K', 'S', 'AH0', 'D', 'AH0', 'N', 'T', 'L', 'IY0'])
("accident's", ['AE1', 'K', 'S', 'AH0', 'D', 'AH0', 'N', 'T', 'S'])
('accidents', ['AE1', 'K', 'S', 'AH0', 'D', 'AH0', 'N', 'T', 'S'])
('accion', ['AE1', 'CH', 'IY0', 'AH0', 'N'])
('accival', ['AE1', 'S', 'IH0', 'V', 'AA2', 'L'])
```

In [ ]:
```
#  3. wordnet
```

In [1]:
```
from nltk.corpus import wordnet as wn
id = wn.synsets('motorcar')     # we get an id from this function and we must it
wn.synset(id).lemma_names()     # gives the lemmas in the subset
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-1-855c42fe64b1> in <module>
      1 from nltk.corpus import wordnet as wn
      2 id = wn.synsets('motorcar')     # we get an id from this function and we must it
----> 3 wn.synset(id).lemma_names()     # gives the lemmas in the subset

~\Anaconda3\lib\site-packages\nltk\corpus\reader\wordnet.py in synset(self, name)
   1327     def synset(self, name):
   1328         # split name into lemma, part of speech and synset number
-> 1329         lemma, pos, synset_index_str = name.lower().rsplit('.', 2)
   1330         synset_index = int(synset_index_str) - 1
   1331 

AttributeError: 'list' object has no attribute 'lower'
```

In [13]:
```
from nltk.corpus import wordnet as wn
wn.synsets('motorcar')     # we get an id from this function and we must it
wn.synset(id).lemma_names()     # gives the lemmas in the subset
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-13-eb1000bcb832> in <module>
      1 from nltk.corpus import wordnet as wn
      2 wn.synsets('motorcar')     # we get an id from this function and we must it
----> 3 wn.synset(id).lemma_names()     # gives the lemmas in the subset

~\Anaconda3\lib\site-packages\nltk\corpus\reader\wordnet.py in synset(self, name)
   1327     def synset(self, name):
   1328         # split name into lemma, part of speech and synset number
-> 1329         lemma, pos, synset_index_str = name.lower().rsplit('.', 2)
   1330         synset_index = int(synset_index_str) - 1
   1331 

AttributeError: 'list' object has no attribute 'lower'
```

TEXT tokenisation

In [ ]:
```
# NLTK PIPELINING
```

In [14]:
```
import nltk
texts = ["""Robert John Downey Jr. (born April 4, 1965) is an American actor, producer, and singer.
 His career has been characterized by critical and popular success in his youth, followed by a perio
d of substance abuse and legal troubles, before a resurgence of commercial success in middle age. In
2008, Downey was named by Time magazine among the 100 most influential people in the world,[2][3] an
d from 2013 to 2015, he was listed by Forbes as Hollywood's highest-paid actor. His films have gross
ed over $14.4 billion worldwide, making him the second highest-grossing box-office star of all time.

At the age of five, he made his acting debut in Robert Downey Sr.'s film Pound in 1970. His subseque
ntly worked with the Brat Pack in the teen films Weird Science (1985) and Less Than Zero (1987). In
 1992, Downey portrayed the title character in the biopic Chaplin, for which he was nominated for th
e Academy Award for Best Actor and won a BAFTA Award. Following a stint at the Corcoran Substance Ab
use Treatment Facility on drug charges, he joined the TV series Ally McBeal, for which he won a Gold
en Globe Award; however in the wake of two drug charges, one in late 2000 and one in early 2001, he
 was fired and his character terminated. He stayed in a court-ordered drug treatment program shortly
after and has maintained his sobriety since 2003."""]


# text regaring robert downey jr
```

In [15]:
```
for text in texts:
    sentences  = nltk.sent_tokenizer(text)
    for sentence in sentences:
        words = nltk.word_tokenizer(sentance)
        tagged_words = nltk.pos_tag(words)
        print(tagged_words)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-15-114b96ec51e7> in <module>
      1 for text in texts:
----> 2     sentences  = nltk.sent_tokenizer(text)
      3     for sentence in sentences:
      4         words = nltk.word_tokenizer(sentance)
      5         tagged_words = nltk.pos_tag(words)

AttributeError: module 'nltk' has no attribute 'sent_tokenizer'
```

In [ ]:
```
#twitter tokenizer
```

In [16]:
```
import nltk
from nltk.tokenizer import tweetTokenizer
text = 'hey there its fun to do nlp :) #laughoutloud'
twtkn = TweetTokenizer()
twtkn.tokenizer(text)
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-16-953928bdd41d> in <module>
      1 import nltk
----> 2 from nltk.tokenizer import tweetTokenizer
      3 text = 'hey there its fun to do nlp :) #laughoutloud'
      4 twtkn = TweetTokenizer()
      5 twtkn.tokenizer(text)

ModuleNotFoundError: No module named 'nltk.tokenizer'
```

In [ ]:
```
#frequency distribution
```

In [ ]:
```
#brown Corpus
```

In [17]:
```
from nltk.corpus import brown
```

In [18]:
```
news_text = brown.words(categories='news')
```

In [2]:
```
news_text = brown.words(categories='news')
fdist = nltk.FreqDist(w.lower()from w in news_text)
modals = ['can','could','may','might','must']
```

```
  File "<ipython-input-2-324e36422171>", line 2
    fdist = nltk.FreqDist(w.lower()from w in news_text)
                                      ^
SyntaxError: invalid syntax
```