In [1]:
```
from string import punctuation
from os import listdir
from collections import Counter
from nltk.corpus import stopwords
```

In [2]:
```
# read the doc
def load_doc(filename):
        # open the file as read only
        file = open(filename, 'r')
        # read all text
        text = file.read()
        # close the file
        file.close()
        return text
```

In [3]:
```
# document tokenization
def clean_doc(doc):
        # split into tokens by white space
        tokens = doc.split()
        # remove punctuation from each token
        table = str.maketrans('', '', punctuation)
        tokens = [w.translate(table) for w in tokens]
        # remove remaining tokens that are not alphabetic
        tokens = [word for word in tokens if word.isalpha()]
        # filter out stop words
        stop_words = set(stopwords.words('english'))
        tokens = [w for w in tokens if not w in stop_words]
        # filter out short tokens
        tokens = [word for word in tokens if len(word) > 1]
        return tokens
```

In [4]:
```
# load all docs in a directory
def process_docs(directory, vocab):
	lines = list()
	# walk through all files in the folder
	for filename in listdir(directory):
		# skip files that do not have the right extension
		if not filename.endswith(".txt"):
			continue
		# create the full path of the file to open
		path = directory + '\\' + filename
		# load and clean the doc
		line = doc_to_line(path, vocab)
		# add to list
		lines.append(line)
	return lines
```

In [5]:
```
# load doc, clean and return line of tokens
def doc_to_line(filename, vocab):
	# load the doc
	doc = load_doc(filename)
	# clean doc
	tokens = clean_doc(doc)
	# filter by vocab
	tokens = [w for w in tokens if w in vocab]
	return ' '.join(tokens)
```

In [6]:
```
# save list to file
def save_list(lines, filename):
	data = '\n'.join(lines)
	file = open(filename, 'w')
	file.write(data)
	file.close()
```

In [10]:

```
# load doc and add to vocab
def add_doc_to_vocab(filename, vocab):
        # load doc
        doc = load_doc(filename)
        # clean doc
        tokens = clean_doc(doc)
        # update counts
        vocab.update(tokens)

# load all docs in a directory
def process_docs2(directory, vocab):
        # walk through all files in the folder
        for filename in listdir(directory):
                # skip files that do not have the right extension
                if not filename.endswith(".txt"):
                        continue
                # create the full path of the file to open
                path = directory + '\\' + filename
                # add doc to vocab
                add_doc_to_vocab(path, vocab)

# define vocab
vocab = Counter()
# add all docs to vocab
process_docs2('C:\\Users\\HPW\\Desktop\\txt_sentoken\\neg', vocab)
process_docs2('C:\\Users\\HPW\\Desktop\\txt_sentoken\\pos', vocab)
# print the size of the vocab
print(len(vocab))
# print the top words in the vocab
print(vocab.most_common(50))
# keep tokens with > 5 occurrence
min_occurane = 5
tokens = [k for k,c in vocab.items() if c >= min_occurane]
print(len(tokens))
# save tokens to a vocabulary file
save_list(tokens, 'C:\\Users\\HPW\\Desktop\\txt_sentoken\\vocab.txt')
```

```
46557
[('film', 8860), ('one', 5521), ('movie', 5440), ('like', 3553), ('even', 2555), ('good', 2320), ('time', 228
3), ('story', 2118), ('films', 2102), ('would', 2042), ('much', 2024), ('also', 1965), ('characters', 1947),
('get', 1921), ('character', 1906), ('two', 1825), ('first', 1768), ('see', 1730), ('well', 1694), ('way', 16
68), ('make', 1590), ('really', 1563), ('little', 1491), ('life', 1472), ('plot', 1451), ('people', 1420),
('movies', 1416), ('could', 1395), ('bad', 1374), ('scene', 1373), ('never', 1364), ('best', 1301), ('new', 1
277), ('many', 1268), ('doesnt', 1267), ('man', 1266), ('scenes', 1265), ('dont', 1210), ('know', 1207), ('he
s', 1150), ('great', 1141), ('another', 1111), ('love', 1089), ('action', 1078), ('go', 1075), ('us', 1065),
('director', 1056), ('something', 1048), ('end', 1047), ('still', 1038)]
14803
```

In [12]:
```
# load vocabulary
vocab_filename = 'C:\\Users\\HPW\\Desktop\\txt_sentoken\\vocab.txt'
vocab = load_doc(vocab_filename)
vocab = vocab.split()
vocab = set(vocab)
# prepare negative reviews
negative_lines = process_docs('C:\\Users\\HPW\\Desktop\\txt_sentoken\\neg', vocab)
save_list(negative_lines, 'C:\\Users\\HPW\\Desktop\\txt_sentoken\\negative.txt')
# prepare positive reviews
positive_lines = process_docs('C:\\Users\\HPW\\Desktop\\txt_sentoken\\pos', vocab)
save_list(positive_lines, 'C:\\Users\\HPW\\Desktop\\txt_sentoken\\positive.txt')
```

In [ ]:
```
```