# 0102_tokenization_stemming_and_stop_word_removal _compressed

December 15, 2018

## 1 Pre-processing data: tokenization, stemming, and removal of stop words (compressed code)

In the previous lesson we went through each of the steps of cleaning text, showing what each step does. Below is compressed code that does the same, and can be applied to any list of text strings. Here we import the imdb data set, extract the review text and clean it, and put the cleaned reviews back into the imdb DataFrame.

In [2]:
```
import nltk
import pandas as pd
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords

# If not previously performed:
# nltk.download('stopwords')

stemming = PorterStemmer()
stops = set(stopwords.words("english"))

def apply_cleaning_function_to_list(X):
    cleaned_X = []
    for element in X:
        cleaned_X.append(clean_text(element))
    return cleaned_X


def clean_text(raw_text):
    """This function works on a raw text string, and:
        1) changes to lower case
        2) tokenizes (breaks down into words
        3) removes punctuation and non-word text
        4) finds word stems
        5) removes stop words
        6) rejoins meaningful stem words"""
```

```
    # Convert to lower case
    text = raw_text.lower()

    # Tokenize
    tokens = nltk.word_tokenize(text)

    # Keep only words (removes punctuation + numbers)
    # use .isalnum to keep also numbers
    token_words = [w for w in tokens if w.isalpha()]

    # Stemming
    stemmed_words = [stemming.stem(w) for w in token_words]

    # Remove stop words
    meaningful_words = [w for w in stemmed_words if not w in stops]

    # Rejoin meaningful stemmed words
    joined_words = ( " ".join(meaningful_words))

    # Return cleaned data
    return joined_words


### APPLY FUNCTIONS TO EXAMPLE DATA

# Load data example
imdb = pd.read_csv('imdb.csv')

# If you do not already have the data locally you may download (and save) by
# uncommenting and running the following lines

# file_location = 'https://gitlab.com/michaelallen1966/00_python_snippets' +\
#     '_and_recipes/raw/master/machine_learning/data/IMDb.csv'
# imdb = pd.read_csv(file_location)
# save to current directory
# imdb.to_csv('imdb.csv', index=False)

# Truncate data for example
imdb = imdb.head(100)

# Get text to clean
text_to_clean = list(imdb['review'])

# Clean text
cleaned_text = apply_cleaning_function_to_list(text_to_clean)

# Show first example
print ('Original text:',text_to_clean[0])
```

```
        print ('\nCleaned text:', cleaned_text[0])

        # Add cleaned data back into DataFrame
        imdb['cleaned_review'] = cleaned_text
```

```
Original text: I have no read the novel on which "The Kite Runner" is based. My wife and daught

Cleaned text: read novel kite runner base wife daughter thought movi fell long way short book
```

In [ ]: