

## Removing stop words with NLTK in Python

The process of converting data to something a computer can understand is referred to as **pre-processing**. One of the major forms of pre-processing is to filter out useless data. In natural language processing, useless words (data), are referred to as stop words.

### What are Stop words?

**Stop Words:** A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.

We would not want these words taking up space in our database, or taking up valuable processing time. For this, we can remove them easily, by storing a list of words that you consider to be stop words. NLTK(Natural Language Toolkit) in python has a list of stopwords stored in 16 different languages. You can find them in the nltk\_data directory. *home/pratima/nltk\_data/corpora/stopwords* is the directory address.(Do not forget to change your home directory name)

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

**To check the list of stopwords you can type the following commands in the python shell.**

```
import nltk
from nltk.corpus import stopwords
set(stopwords.words('english'))
```

{‘ourselves’, ‘hers’, ‘between’, ‘yourself’, ‘but’, ‘again’, ‘there’, ‘about’, ‘once’, ‘during’, ‘out’, ‘very’, ‘having’, ‘with’, ‘they’, ‘own’, ‘an’, ‘be’, ‘some’, ‘for’, ‘do’, ‘its’, ‘yours’, ‘such’, ‘into’, ‘of’, ‘most’, ‘self’, ‘other’, ‘off’, ‘is’, ‘s’, ‘am’, ‘or’, ‘who’, ‘as’, ‘from’, ‘him’, ‘each’, ‘the’, ‘themselves’, ‘until’, ‘below’, ‘are’, ‘we’,

'these', 'your', 'his', 'through', 'don', 'nor', 'me', 'were', 'her', 'more', 'himself', 'this', 'down', 'should', 'our', 'their', 'while', 'above', 'both', 'up', 'to', 'ours', 'had', 'she', 'all', 'no', 'when', 'at', 'any', 'before', 'them', 'same', 'and', 'been', 'have', 'in', 'will', 'on', 'does', 'yourselves', 'then', 'that', 'because', 'what', 'over', 'why', 'so', 'can', 'did', 'not', 'now', 'under', 'he', 'you', 'herself', 'has', 'just', 'where', 'too', 'only', 'myself', 'which', 'those', 'i', 'after', 'few', 'whom', 't', 'being', 'if', 'theirs', 'my', 'against', 'a', 'by', 'doing', 'it', 'how', 'further', 'was', 'here', 'than'}

**Note:** You can even modify the list by adding words of your choice in the english .txt. file in the stopwords directory.

## Removing stop words with NLTK

The following program removes stop words from a piece of text:

```
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

example_sent = "This is a sample sentence, showing off the stop words filtration."

stop_words = set(stopwords.words('english'))

word_tokens = word_tokenize(example_sent)

filtered_sentence = [w for w in word_tokens if not w in stop_words]

filtered_sentence = []

for w in word_tokens:
    if w not in stop_words:
        filtered_sentence.append(w)

print(word_tokens)
print(filtered_sentence)
```

Run on IDE

Output:

```
['This', 'is', 'a', 'sample', 'sentence', ',', 'showing',
 'off', 'the', 'stop', 'words', 'filtration', '.']
['This', 'sample', 'sentence', ',', 'showing', 'stop',
 'words', 'filtration', '.']
```

## Performing the Stopwords operations in a file

In the code below, text.txt is the original input file in which stopwords are to be removed. filteredtext.txt is the output file. It can be done using following code:

```
import io
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
#word_tokenize accepts a string as an input, not a file.
stop_words = set(stopwords.words('english'))
file1 = open("text.txt")
line = file1.read()# Use this to read file content as a stream:
words = line.split()
for r in words:
    if not r in stop_words:
        appendFile = open('filteredtext.txt','a')
        appendFile.write(" "+r)
        appendFile.close()
```



This is how we are making our processed content more efficient by removing words that do not contribute to any future operations.

This article is contributed by **Pratima Upadhyay**. If you like GeeksforGeeks and would like to contribute, you can also write an article using [contribute.geeksforgeeks.org](https://contribute.geeksforgeeks.org) or mail your article to [contribute@geeksforgeeks.org](mailto:contribute@geeksforgeeks.org). See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Advanced Computer Subject Python Machine Learning

[Login to Improve this Article](#)

Please write to us at [contribute@geeksforgeeks.org](mailto:contribute@geeksforgeeks.org) to report any issue with the above content.

## Recommended Posts:

[Tokenize text using NLTK in python](#)

[How to get synonyms/antonyms from NLTK WordNet in Python?](#)

[Getting started with Machine Learning](#)

[Understanding Logistic Regression](#)

[Implementing Artificial Neural Network training process in Python](#)

[Linear Regression using PyTorch](#)

[Difference between Machine learning and Artificial Intelligence](#)

[Lowest Common Ancestor in a Binary Tree | Set 3 \(Using RMQ\)](#)

[Introduction to Artificial Neural Network | Set 2](#)

[Digital Image Processing Basics](#)

(Login to Rate)

4

Average Difficulty : **4/5.0**  
Based on **3** vote(s)

☐

Add to TODO List

☐

Mark as DONE

Basic

Easy

Medium

Hard

Expert

Writing code in comment? Please use [ide.geeksforgeeks.org](https://ide.geeksforgeeks.org), generate link and share the link here.

Share this post!



1 Comment

GeeksforGeeks

1 Login ▾

♥ Recommend

↗ Share

Sort by Newest ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

**Brent Schneeman** • 4 months ago

consider lowering the words before stopword removal

^ | ▾ • Reply • Share ›

✉ Subscribe   Add Disqus to your siteAdd DisqusAdd   Privacy

A computer science portal for geeks

710-B, Advant Navis Business Park,  
Sector-142, Noida, Uttar Pradesh - 201305  
feedback@geeksforgeeks.org

## COMPANY

About Us  
Careers  
Privacy Policy  
Contact Us

## PRACTICE

Company-wise  
Topic-wise  
Contests  
Subjective Questions

## LEARN

Algorithms  
Data Structures  
Languages  
CS Subjects  
Video Tutorials

## CONTRIBUTE

Write an Article  
GBlog  
Videos

@geeksforgeeks, Some rights reserved

