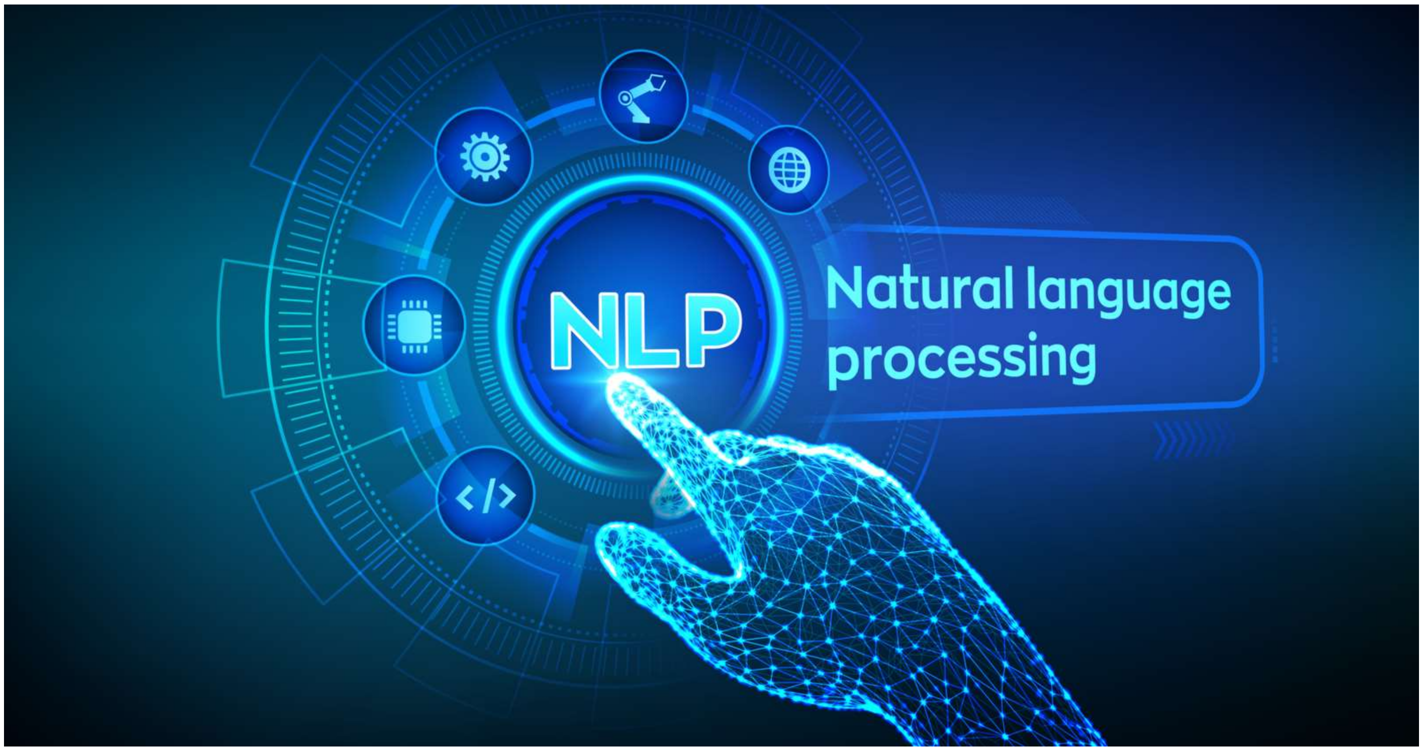
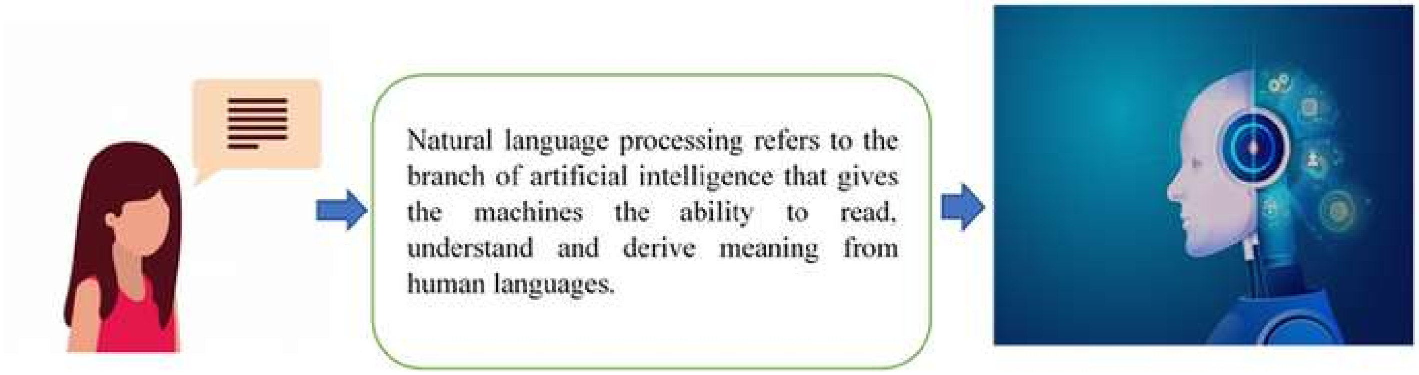
NLP Assignment-1_Introduction to NLP_Practice Sheet



What is NLP?

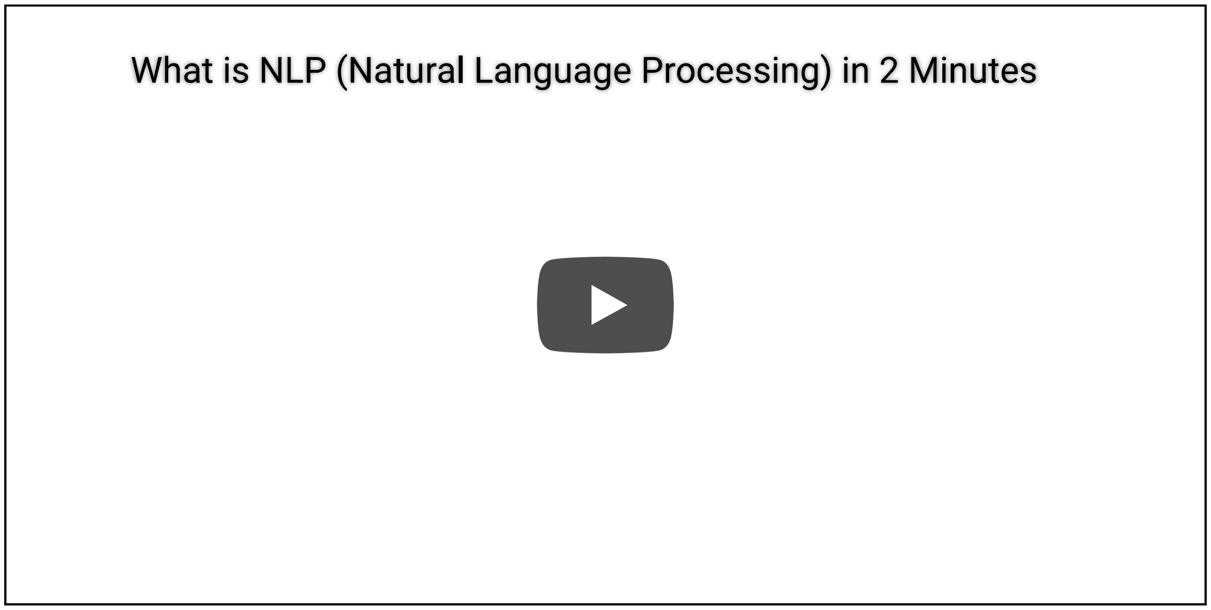
Natural language processing (NLP) is a field of artificial intelligence in which computers analyze, understand, and derive meaning from human language in a smart and useful way.


Natural language processing refers to the branch of artificial intelligence that gives the machines the ability to read, understand and derive meaning from human languages.

It is a discipline that focuses on the interaction between data science and human language, and is scaling to lots of industries. Today NLP is booming thanks to the huge improvements in the access to data and the increase in computational power, which are allowing practitioners to achieve meaningful results in areas like healthcare, media, finance and human resources, among others.

Do refer this amazing video to know what is NLP in an interesting way.

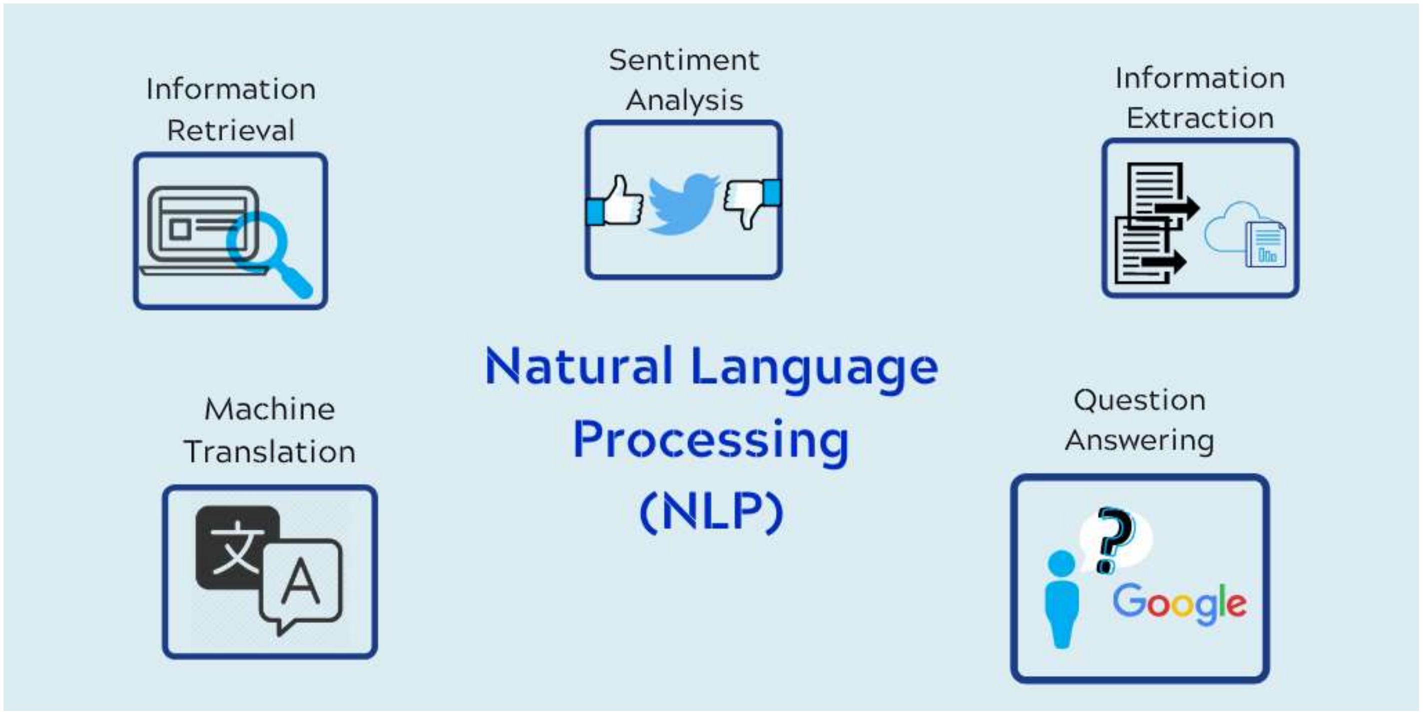
```
from IPython.display import YouTubeVideo
YouTubeVideo('Hbx9bxt7gvc', width=600, height=300)
```



Applications of NLP in the industry

Natural Language Processing technology is especially valuable for businesses. A number of companies have already taken advantage of NLP services from Unicsoft to gain a competitive edge over their rivals.

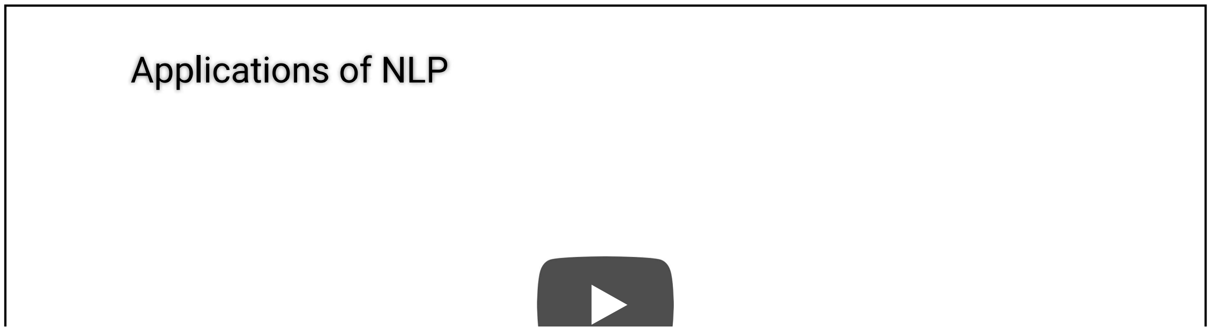
Firstly, this technology helps derive understanding from the multiple unstructured data available online and in call logs. Next, since businesses feel the constant need for enhancing the communication process with their customers, NLP tools are the best way to improve the quality of this interaction. Application of NLP can be found in a range of business contexts, including e-commerce, healthcare and advertising.



Here are the top NLP use cases in business.

1. Machine Translation
2. Social Media Monitoring
3. Sentiment Analysis
4. Chatbots and Virtual Assistants
5. Text analysis
6. Speech Recognition
7. Text Extraction
8. Autocorrect, Spell Check

```
###### Refer video:
YouTubeVideo('6Eekv-pZfHY', width=600, height=300)
```

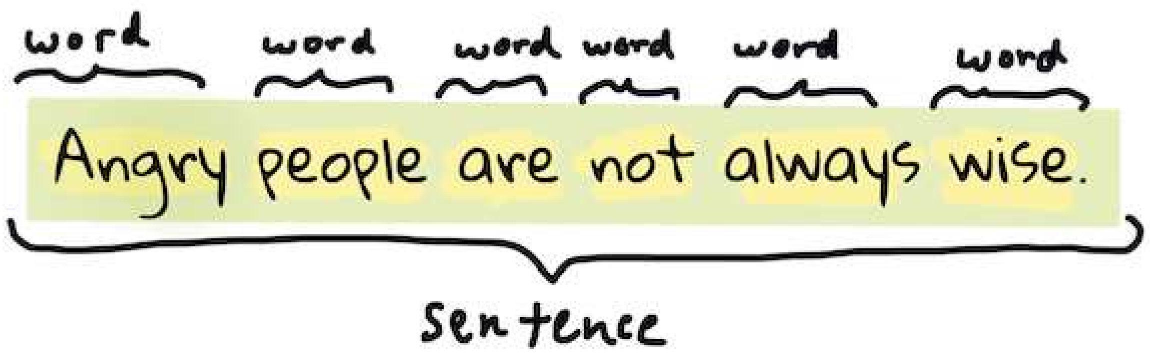


Basics of NLP

1. Tokenization
2. Lemmatization
3. Stemming
4. How to remove stop words

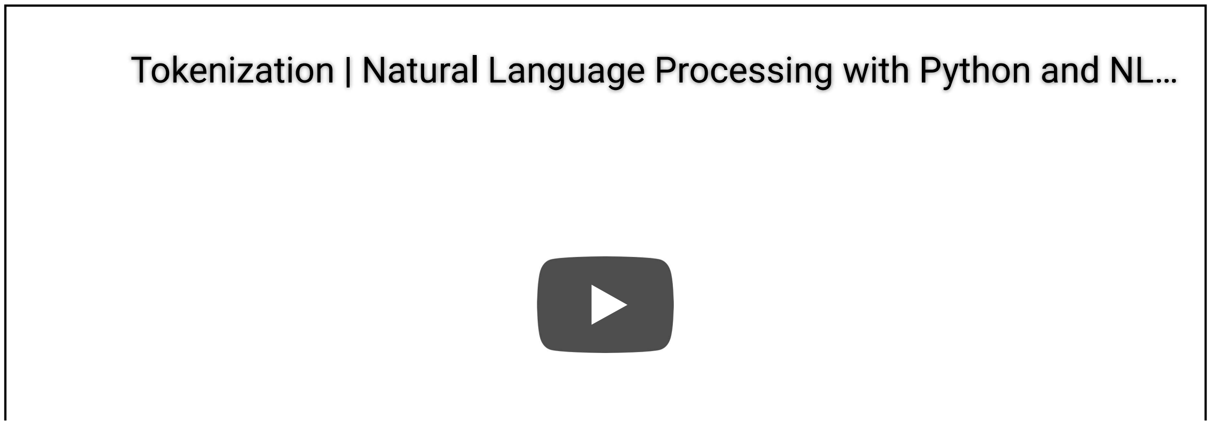
Tokenization

Probably the first thing most NLP algorithms have to do is to split the text into tokens, or words. While this sounds simple, having to account for punctuation and different languages word and sentence delimiters can make it tricky. You might have to use various methods to determine demarcations.



Documentation link: https://docs.python.org/3/library/tokenize.html

```
###### Refer video:
YouTubeVideo('Dh4El5MtxpE', width=600, height=300)
```


Tokenization | Natural Language Processing with Python and NL...

```
# Tokenization of paragraphs
# Import nltk and download punkt(enter q in input box)
# your code here
```

```
NLTK Downloader
---------------------------------------------------------------------------
    d) Download   l) List    u) Update   c) Config   h) Help   q) Quit
---------------------------------------------------------------------------
Downloader> q
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
True
```

Punkt:

This tokenizer divides a text into a list of sentences by using an unsupervised algorithm to build a model for abbreviation words, collocations, and words that start sentences. It must be trained on a large collection of plaintext in the target language before it can be used.

Punkt Documentation: https://www.nltk.org/_modules/nltk/tokenize/punkt.html

```
paragraph = '''Welcome to CloudyML Hands-on of every thing you need to know about Cloud & Mac
```

```
# Tokenizing sentences (converting paragraphs into sentences)
# your code here
```

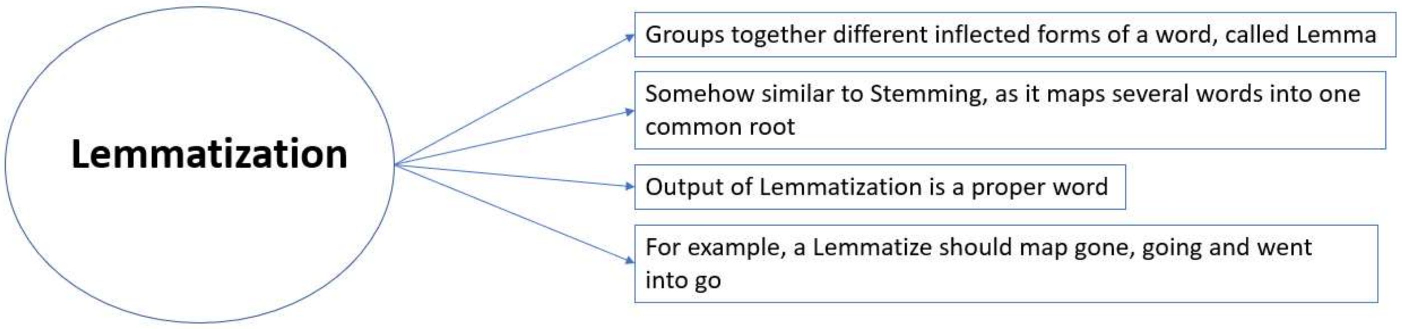
```
['Welcome to CloudyML Hands-on of every thing you need to know about Cloud & Machine Lea
```

```
# Tokenizing words (converting sentences into words)
# your code here
```

```
['Welcome', 'to', 'CloudyML', 'Hands-on', 'of', 'every', 'thing', 'you', 'need', 'to',
```

Lemmatization

It is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. It is similar to stemming but it brings context to the words. So it links words with similar meanings to one word.

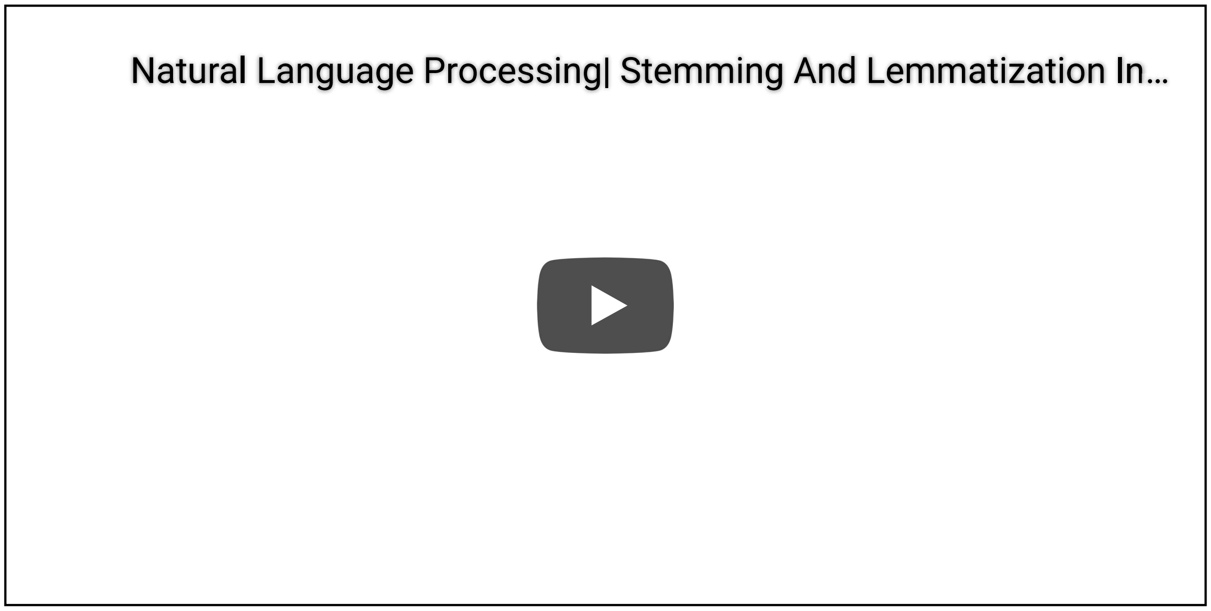

Lemmatization
Groups together different inflected forms of a word, called Lemma
Somehow similar to Stemming, as it maps several words into one common root
Output of Lemmatization is a proper word
For example, a Lemmatize should map gone, going and went into go

Applications of lemmatization are:

Used in comprehensive retrieval systems like search engines and compact indexing

Documentation : https://www.kite.com/python/docs/nltk.WordNetLemmatizer

```
###### Refer video:
YouTubeVideo('JpxCt3kvbLk', width=600, height=300)
```


Natural Language Processing| Stemming And Lemmatization In...

```
# import WordNetLemmatizer and download wordnet
# your code here
```

```
    [nltk_data] Downloading package wordnet to /root/nltk_data...
    [nltk_data]   Unzipping corpora/wordnet.zip.
    True
```

WordNet is a database of words, synonyms, antonyms and many other details for every word in many different languages. It is incredibly useful when attempting to build translations, spell checkers, or language tools of any type.

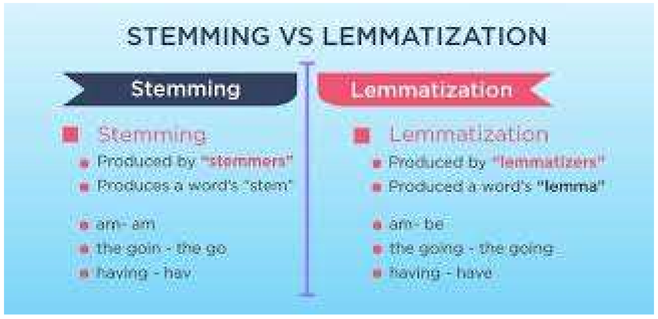
Reference: https://www.nltk.org/howto/wordnet.html

```
# lemmatize the given words or you can use your own words :)
# your code here
```

```
rocks : rock
corpora : corpus
```

Stemming

It is the process of producing morphological variants of a root/base word. Stemming programs are commonly referred to as stemming algorithms or stemmers. A stemming algorithm reduces the words “chocolates”, “chocolatey”, “choco” to the root word, “chocolate” and “retrieval”, “retrieved”, “retrieves” reduce to the stem “retrieve”. Stemming is an important part of the pipelining process in Natural language processing. The input to the stemmer is tokenized words.


STEMMING VS LEMMATIZATION

Stemming	Lemmatization
Stemming	Lemmatization
Produced by "stemmers"	Produced by "lemmatizers"
Produces a word's "stem"	Produced a word's "lemma"
am- am	am- be
the goin - the go	the going - the going
having - hav	having - have

Applications of stemming :

1. Stemming is used in information retrieval systems like search engines.
2. It is used to determine domain vocabularies in domain analysis.

```
# import PorterStemmer and word_tokenize
```

The Porter stemming algorithm (or 'Porter stemmer') is a process for removing the commoner morphological and inflexional endings from words in English. Its main use is as part of a term normalisation process that is usually done when setting up Information Retrieval systems.

Reference: https://www.nltk.org/_modules/nltk/stem/porter.html

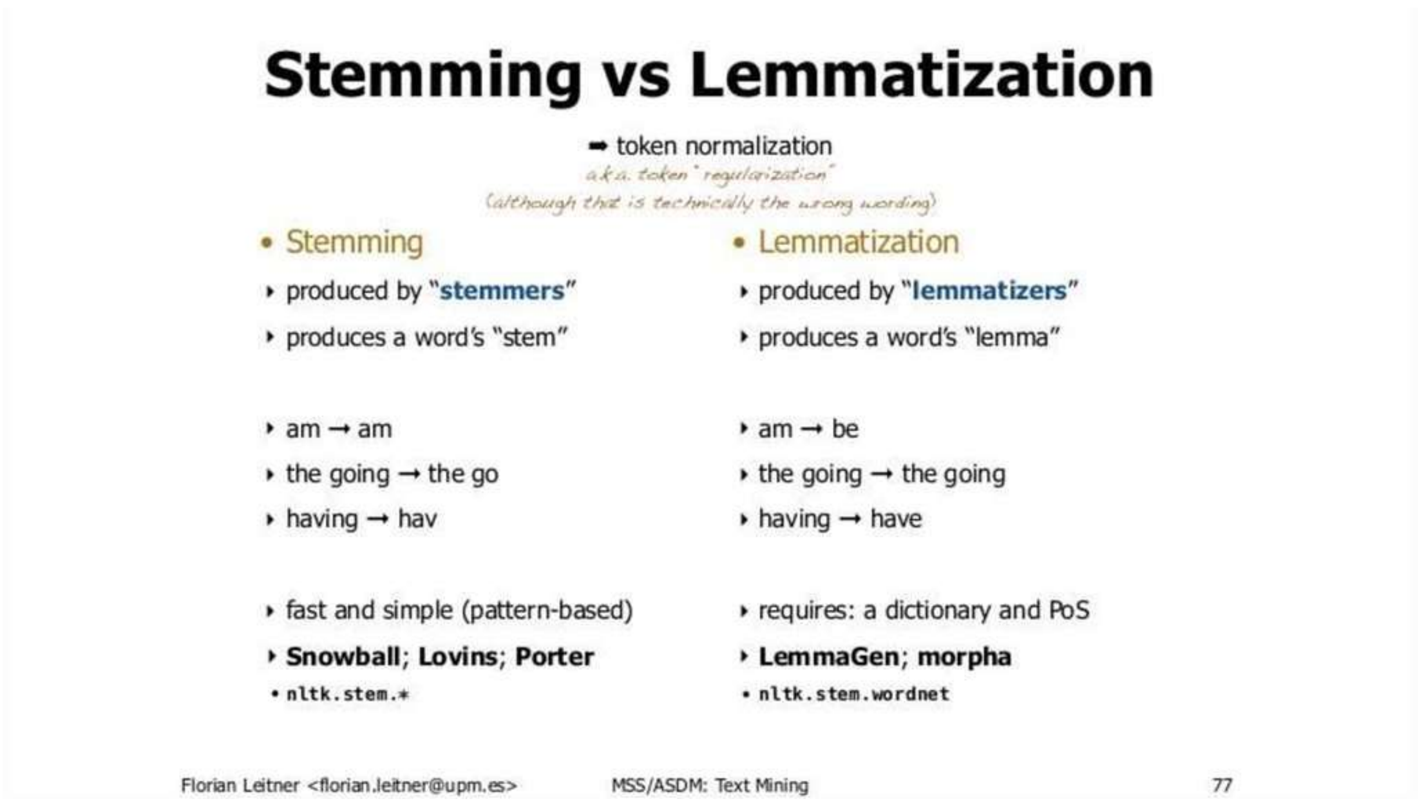
```
# create the PorterStemmer object
# your code here


# stem the words given below
words = ["program", "programs", "programmer", "programming", "programmers"]
# your code here
```

```
program  :  program
programs  :  program
programmer  :  programm
programming  :  program
programmers  :  programm
```

Stemming VS Lemmatization

Stemming just removes or stems the last few characters of a word, often leading to incorrect meanings and spelling.

Lemmatization considers the context and converts the word to its meaningful base form, which is called Lemma. Sometimes, the same word can have multiple different Lemmas.



Stemming vs Lemmatization

➡ token normalization
a.k.a. token "regularization"
(although that is technically the wrong wording)

Stemming	Lemmatization
produced by "**stemmers**"	produced by "**lemmatizers**"
produces a word's "stem"	produces a word's "lemma"
am → am	am → be
the going → the go	the going → the going
having → hav	having → have
fast and simple (pattern-based)	requires: a dictionary and PoS
Snowball; Lovins; Porter	**LemmaGen; morpha**
`nltk.stem.*`	`nltk.stem.wordnet`

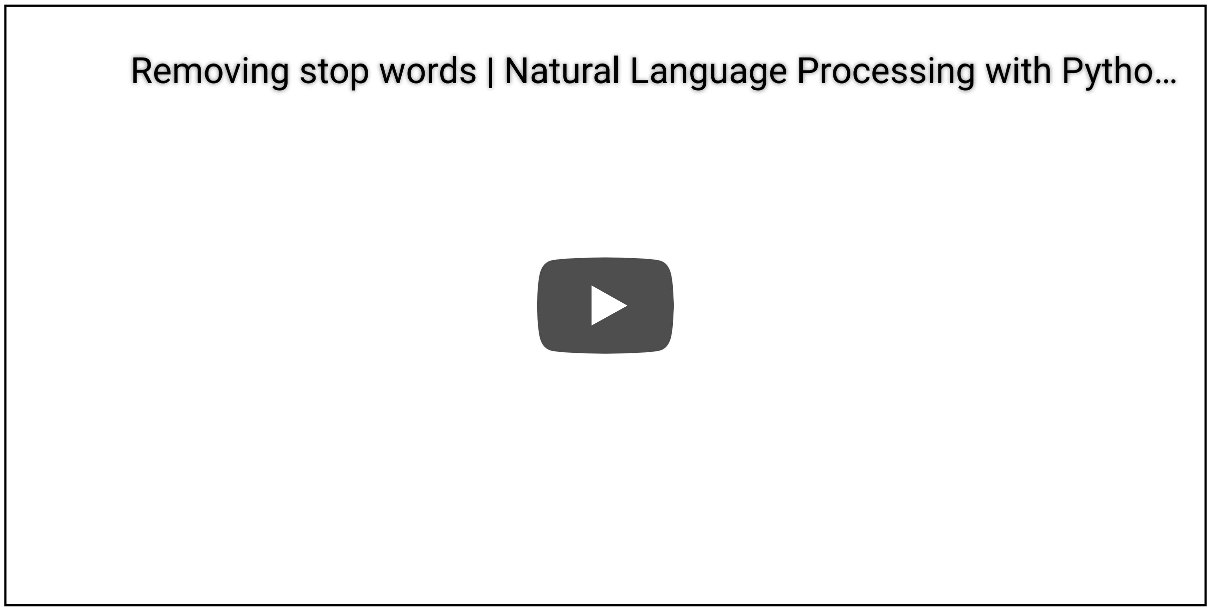
Florian Leitner <florian.leitner@upm.es> MSS/ASDM: Text Mining 77

How to remove stop words

The stopwords in nltk are the most common words in data. They are words that you do not want to use to describe the topic of your content. They are pre-defined and cannot be removed.

Documtation: https://pythonspot.com/nltk-stop-words/

```
###### Refer video:
YouTubeVideo('ob6IlbV13IM', width=600, height=300)
```


Removing stop words | Natural Language Processing with Pytho…

```
# import nltk and download stopwords
# your code here
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
True
```

```
#import english Stopwords
# your code here
```

```
 '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't",
      'shouldn',
      "shouldn't",
      'wasn',
      "wasn't",
      'weren',
      "weren't",
      'won',
      "won't",
      'wouldn',
```

```
#Remove Stops words in a sentence
words = ['There', 'is', 'a', 'Deep','Learning','Course','in','CLoudyML']
# your code here
```

```
There
Deep
Learning
Course
CLoudyML
```

Great job!! You have come to the end of this practice sheet. Treat yourself for this :))

Do fill this feedback form

You may head on to the next assignment.