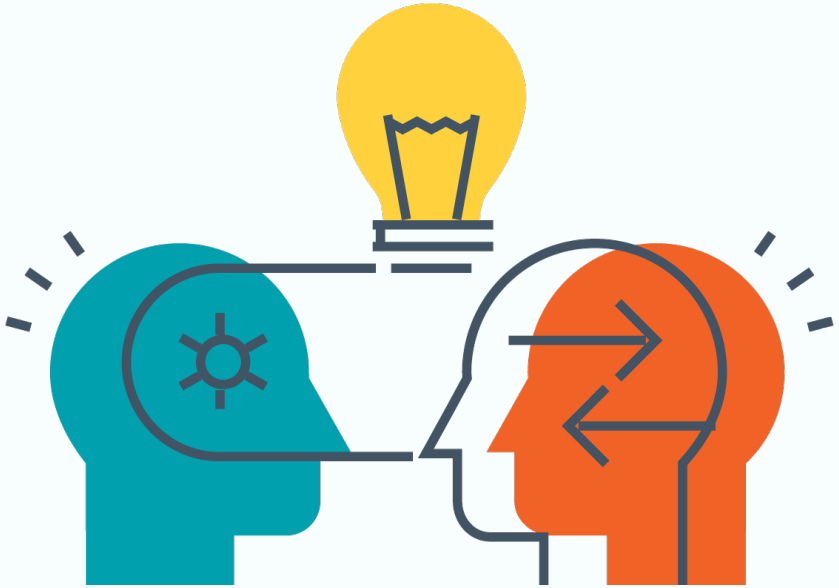


# Introduction to natural language processing

# Agenda

- What is Natural language processing?
- Examples of Natural language processing
- Roadmap to learn natural language processing
- What is python?
    - Importance of Python
    - Important libraries in Python
- What is data?
- What is data pre-processing?
- Types of data pre-processing
    - What is tokenization?
    - What is stemming?
    - What is lemmatization?
    - What are stop words?
- Modelling techniques
    - What is bag of words?
    - What is TF-IDF?
    - What is word embedding?
    - What is sentiment analysis?
- Project on sentiment analysis

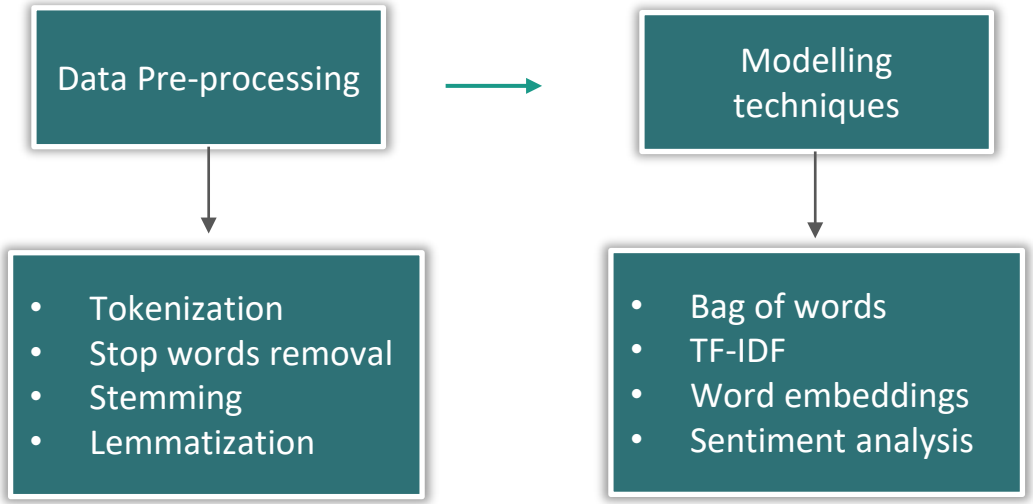
# What is Natural language processing?

NLP stands for natural language processing which is basically used to understand and interpret human language to the machine. In short it is the automatic way to manipulate the  natural language, like speech and text, by software for further analysis to get the required information from them

# Examples of Natural language processing

- Predictive text
- Email filters
- Data analysis
- Language translation
- Smart assistants

# Roadmap to learn natural language processing



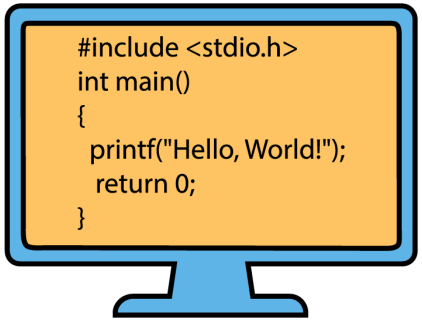
# BUT, How to implement these techniques?


**Python**

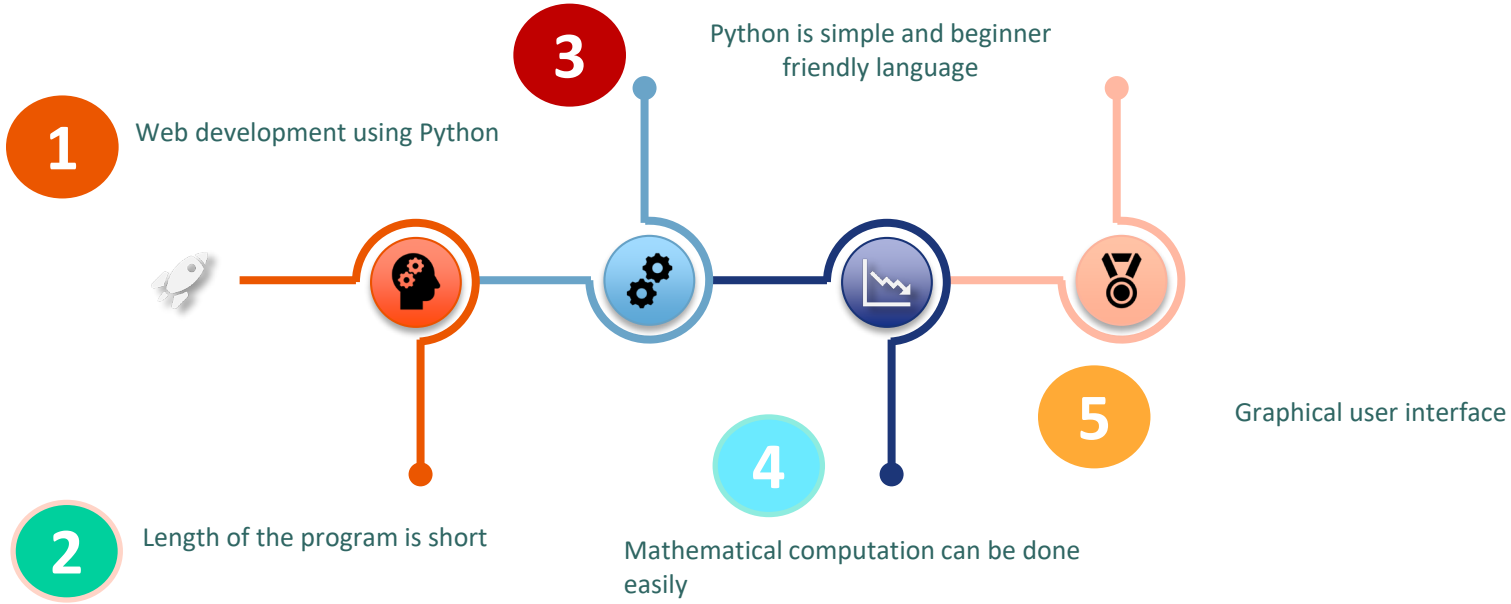
# What is Python?

## Python is a popular high level, object oriented and interpreted language

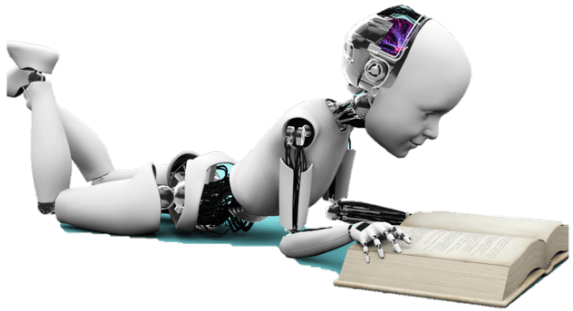
- High level
- Object oriented
- Interpreted



# Benefits of Python



# Important libraries in Python



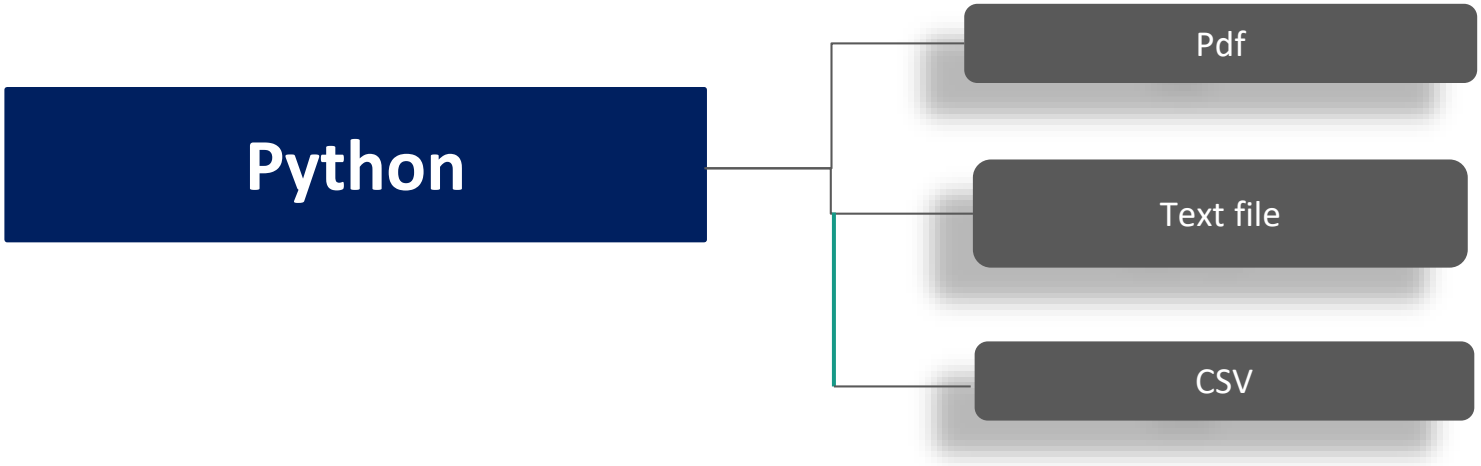
- Numpy
- Pandas
- Matplotlib
- Seaborn
- NLTK
- Spacy

# Data Pre-processing

# What is data?

Bunch of raw information on which operations will be performed

# How to work with different types of document using Python?



# What is data pre-processing?

Data preprocessing is a way to develop informative data from raw data by removing noise and unwanted attributes

**Types of data-preprocessing**

- Remove or fill null values
- Count unique values in the column
- Drop the irrelevant columns

# Types of Data-preprocessing: Imputation techniques

**Remove or fill the null values in the data to get appropriate informative data.**

**Raw data**

| | Sentence # | Word | POS | Tag |
|---|---|---|---|---|
| 0 | Sentence: 1 | Thousands | NNS | O |
| 1 | NaN | of | IN | O |
| 2 | NaN | demonstrators | NNS | O |
| 3 | NaN | have | VBP | O |
| 4 | NaN | marched | VBN | O |

**Converted data**

| | Sentence # | Word | POS | Tag |
|---|---|---|---|---|
| 0 | Sentence: 1 | Thousands | NNS | O |
| 1 | Sentence: 1 | of | IN | O |
| 2 | Sentence: 1 | demonstrators | NNS | O |
| 3 | Sentence: 1 | have | VBP | O |
| 4 | Sentence: 1 | marched | VBN | O |

# Types of Data-preprocessing: Removing/Filling Null Values

**Take the count of unique data to understand and evaluate the dataset.**

**Raw data**

| | |
|---|---|
| Sentence # | 3300 |
| Word | 72706 |
| POS | 72706 |
| Tag | 72706 |

**Converted data**

| | |
|---|---|
| Sentence # | 3300 |
| Word | 9216 |
| POS | 41 |
| Tag | 17 |

# Types of Data-preprocessing: Removing/Filling Null Values

## Remove all the unnecessary columns

**Raw data**

| | Sentence # | Word | POS | Tag | Shape |
|---|---|---|---|---|---|
| 0 | Sentence: 1 | Thousands | NNS | O | NaN |
| 1 | NaN | of | IN | O | NaN |
| 2 | NaN | demonstrators | NNS | O | NaN |
| 3 | NaN | have | VBP | O | NaN |
| 4 | NaN | marched | VBN | O | NaN |

**Converted data**

| | Sentence # | Word | POS | Tag |
|---|---|---|---|---|
| 0 | Sentence: 1 | Thousands | NNS | O |
| 1 | Sentence: 1 | of | IN | O |
| 2 | Sentence: 1 | demonstrators | NNS | O |
| 3 | Sentence: 1 | have | VBP | O |
| 4 | Sentence: 1 | marched | VBN | O |

# Demo: Data Pre-processing using Python

# What is tokenization?

**Tokenization method is used to split a phrase, sentence, paragraph, or an entire text document into smaller units. By doing that we can get the individual words or terms. Each of these smaller units are called tokens.**

# Why do we need tokenization?


- The most basic part of Natural language processing
- It helps to interpret the meaning of the text by analyzing the words present in the text
- Count the number of words in the text

Example:

**“This is a cake.”**

**[‘This’, ‘is’, ‘a’, ‘cake’]**

# Demo: How to implement tokenization using Python?

# What is stemming?

Stemming is the way to reduce a word to its word stem that affixes to suffixes and prefixes. In simple term, This  algorithms work by cutting off the end or the beginning of the word while taking into account a list of common prefixes and suffixes that can be found in an inflected word.

| Form | Suffix | stem |
|---|---|---|
| Cats | -s | cat |
| Birds | -s | bird |

# Why do we need stemming?


- Less input dimensions
- Machine Learning techniques work better with it
- Make training data more dense
- Reduce the size of the dictionary
- Helps to normalize the word in the document

# Demo: How to implement stemming using Python?

# What is Lemmatization in NLP?

Lemmatization helps to do the morphological analysis of the words. It is important to have the knowledge about the detailed dictionaries which the algorithm can refer to link the form back to its lemma.

| Form | Morphological information | Lemma |
|---|---|---|
| Helps | Third person singular number, present tense help | Help |
| Helping | Ing form of the verb | Help |

# Demo: How to implement Lemmatization using Python?

# Difference between stemming and lemmatization

| Topic | Stemming | Lemmatization |
|---|---|---|
| Goal | Reduce inflectional forms (Stemming refers to the crude heuristic process which chops off the ends of the words in order to achieve the goal correctly ) | Reduce inflectional forms(Lemmatization refers to do the things properly with the help of a vocabulary and morphological analysis of words) |
| Implementation | stemmers are typically easier to implement and run faster compare to lemmatization | Lemmatization is difficult to implement |

# What are stop words?

Common words that occur in sentences that add weight to the sentence are known as stop words. These stop words act as a bridge and ensure that sentences are grammatically correct. In simple terms, words that are filtered out before processing natural language data is known as a stop word and it is a common pre-processing method.

# Demo: Remove stop words using Python

# Modelling techniques

# Bag of words

# What is bag of words?

Bag of Words model is used to preprocess the text or documentations. It converts the documents into a bag of words, which keeps a count of the total occurrences of most frequently used words. Bag-of-Words is one of the most used methods to transform tokens into a set of features.

# Demo: Implement bag of words using Python

# TF-IDF

# What is TF-IDF?

- TF-IDF stands for Term Frequency and Inverse Document Frequency,
- This helps to measure the score in order to get the information retrieval (IR) or summarization.
- TF-IDF is also used to reflect how relevant a term is in a given document
- Procedure to calculate TF-IDF by multiplying two metrics:
  - How many times a word appears in a document,
  - And the inverse document frequency of the word across a set of documents

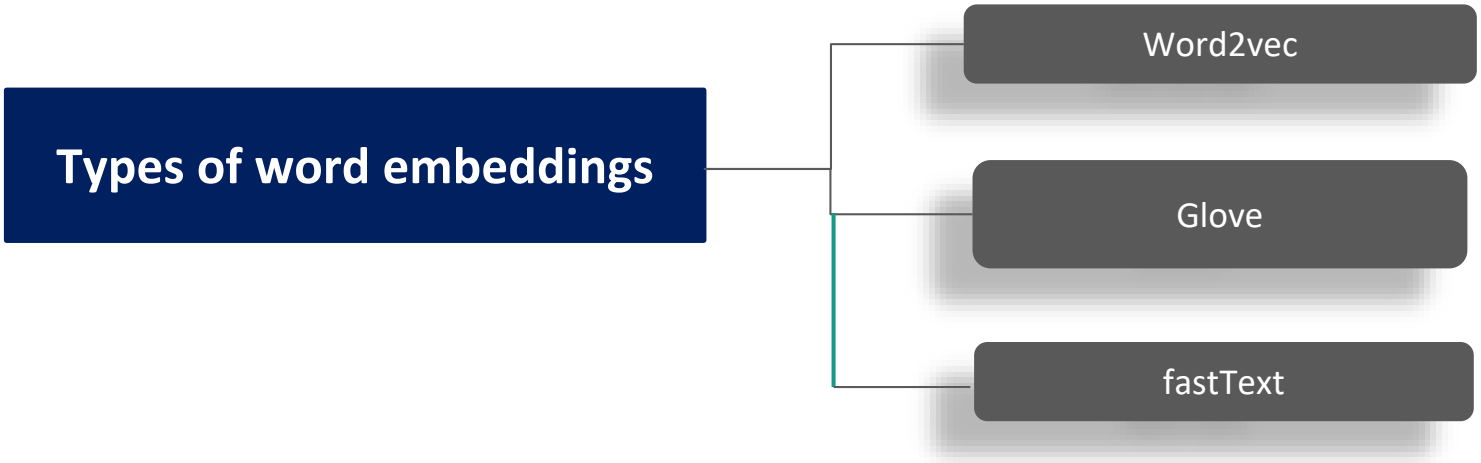
# Why do we need TF-IDF?

- TF-IDF helps to establish how important a particular word is in the context of the document corpus. TF-IDF takes into account the number of times the word appears in the document and offset by the number of documents that appear in the corpus.
- TF is the frequency of term divided by a total number of terms in the document.
- IDF is obtained by dividing the total number of documents by the number of documents containing the term and then taking the logarithm of that quotient.
- Tf.idf is then the multiplication of two values TF and IDF.

# Demo: Implement TF-IDF using Python

# What is word embedding?

Word Embeddings vectors are one of the most common way to encode words as vectors of numbers those vectors can ben fed in into the Machine Learning models for inference and also it helps to establish the distance between two tokens

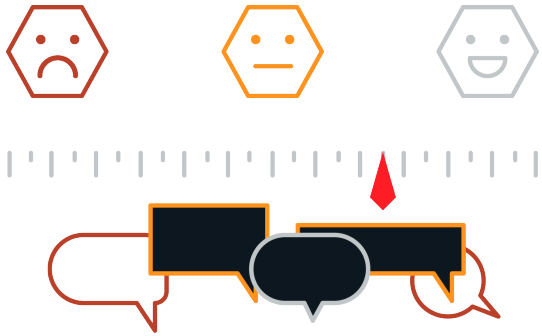


# Demo: Implement word embeddings using Python

# What is sentiment analysis?

Sentiment Analysis is a technique which is commonly used to understand the positive, negative or neutral sentiment about a particular topic

- Sentiments in texts are represented as a value between -1 (negative sentiment) and 1 (positive sentiment) referred to as polarity
- It is an unsupervised Machine Learning technique



# Demo: Project on sentiment analysis

greatlearning
Power Ahead

# Thank You!