Natural Language Processing (NLP)

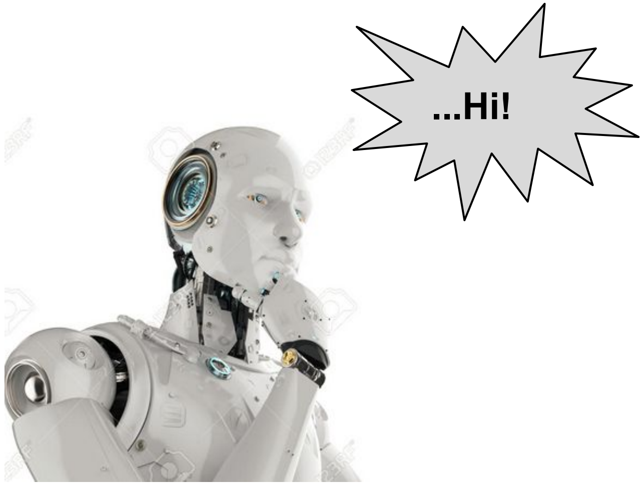
by M.Z.

Outline

- NLP Intro
- Examples
- Basic Commands
  - Tokenization
  - Stemming & Lemmatization
  - Stop Words
  - POS-Tag
- Advanced
  - Sentiment Analysis
- Limitations & Future
- Conclusion

What is NLP?

Definition: "a subfield of linguistics, computer science, information engineering, and artificial intelligence where the goal is to be able to get machines (computers) to understand the meaning of human (natural) language. " (Wikipedia)



High Level Structure

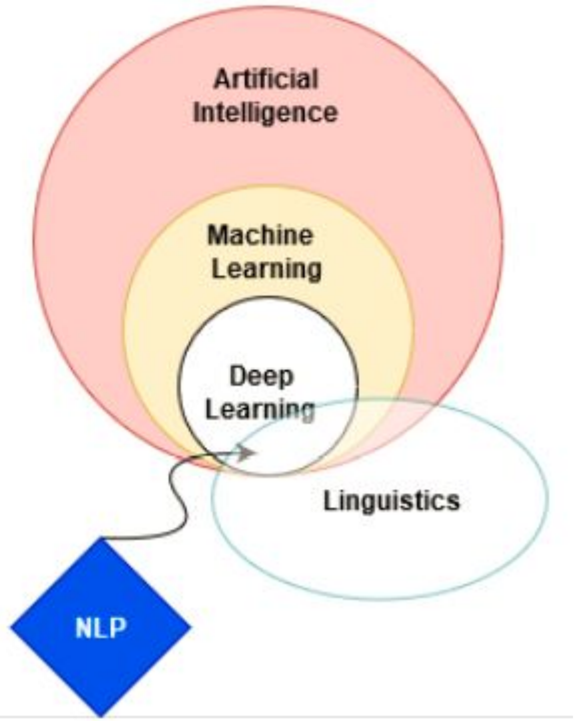
Artificial Intelligence: concerned with making computers intelligent (less reliant on human instruction).

Machine Learning: set of tools for making inference and predictions (computer science and statistics).

Deep Learning: special area of ML where a lot more data is required to establish relationships strengths (text/pics) uses multiple layers to progressively extract higher level features from the raw input (edges⤜ shape ⤜ identify object).

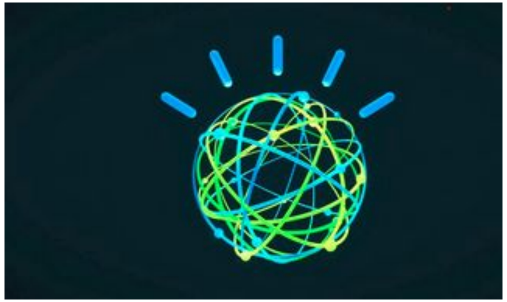
- ❖ We have complex problems (language)
- ❖ Lots of data that we can feed into the model

Linguistics: study of language through sound and meaning.



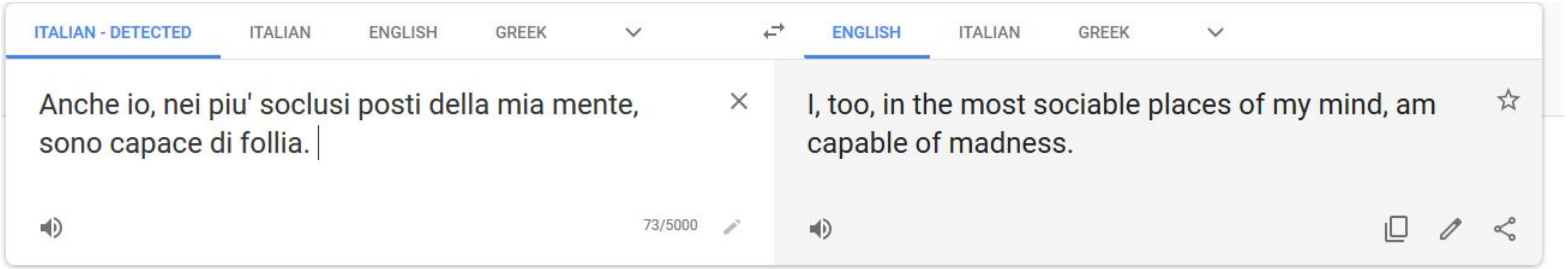
Problems Solved with NLP

❖ Voice Driven Interfaces



Problems Solved with NLP

❖ Machine Translation



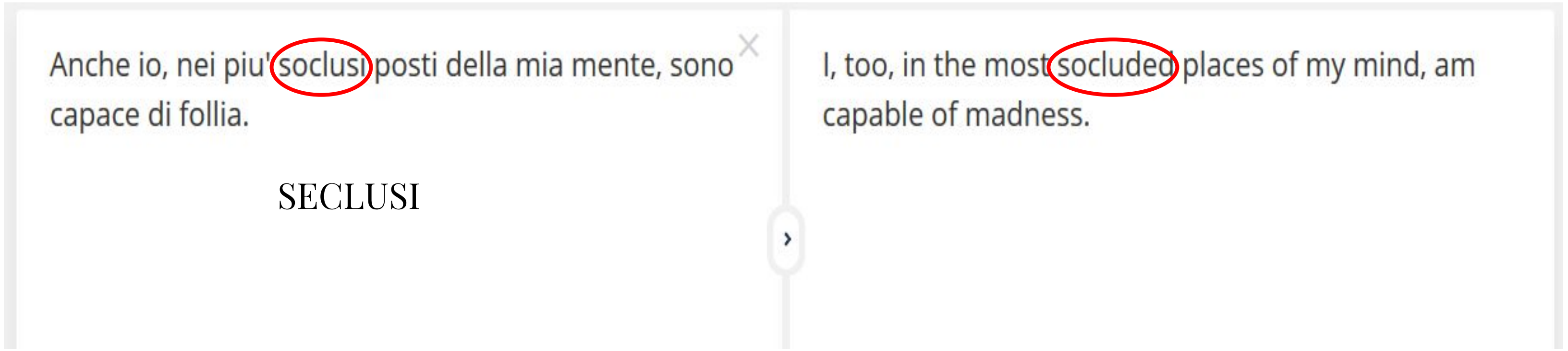
ITALIAN - DETECTED ITALIAN ENGLISH GREEK

ENGLISH ITALIAN GREEK

Anche io, nei piu' soclusi posti della mia mente, sono capace di follia.

73/5000

I, too, in the most sociable places of my mind, am capable of madness.



Anche io, nei piu' soclusi posti della mia mente, sono capace di follia.

I, too, in the most socluded places of my mind, am capable of madness.

SECLUSI

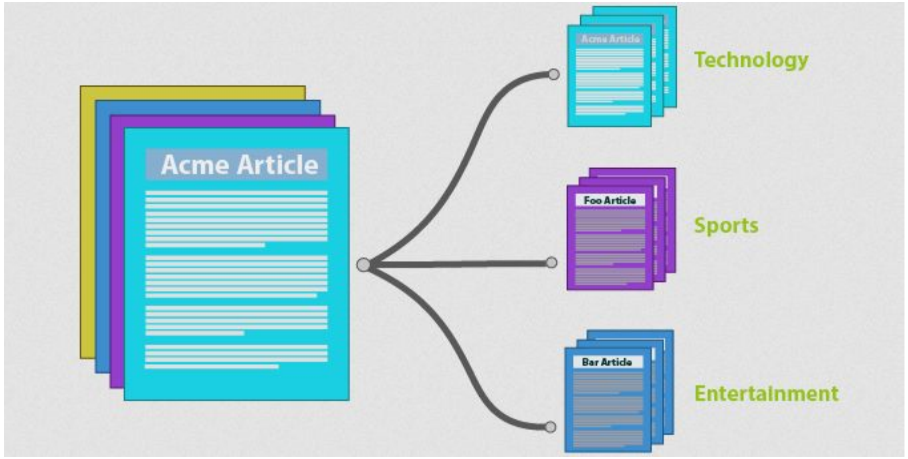
Why is it important?

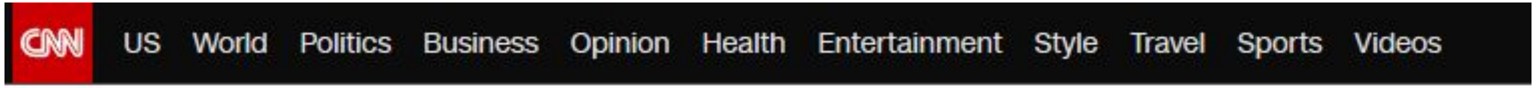
- ❖ NLP processing helps computers communicate with humans in their own language and scales other language-related tasks.
  - ➢ read text, hear speech, interpret it, measure sentiment and determine which parts are important.

- ❖ Machines can analyze more language-based data than humans, without fatigue and in a consistent, unbiased way.
  - ➢ More efficient & less biased

Problems Solved with NLP

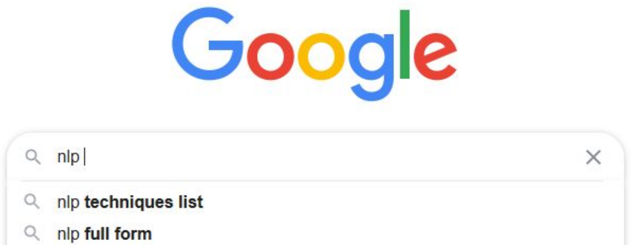
❖ Document Categorization





Problems Solved with NLP

- ❖ Search Engines

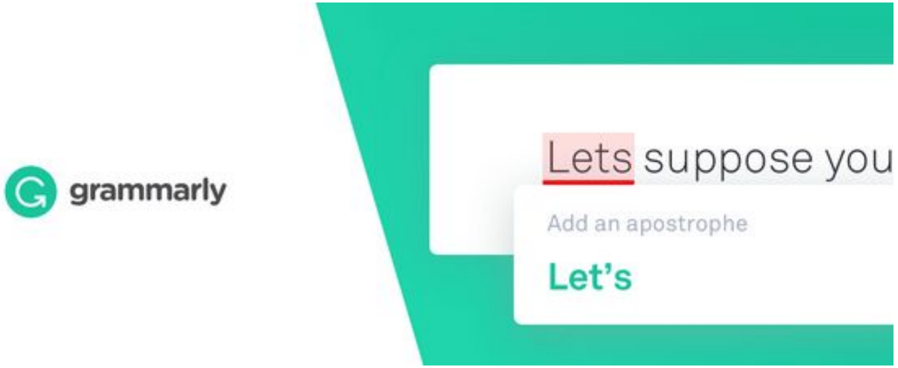

nlp
nlp techniques list
nlp full form

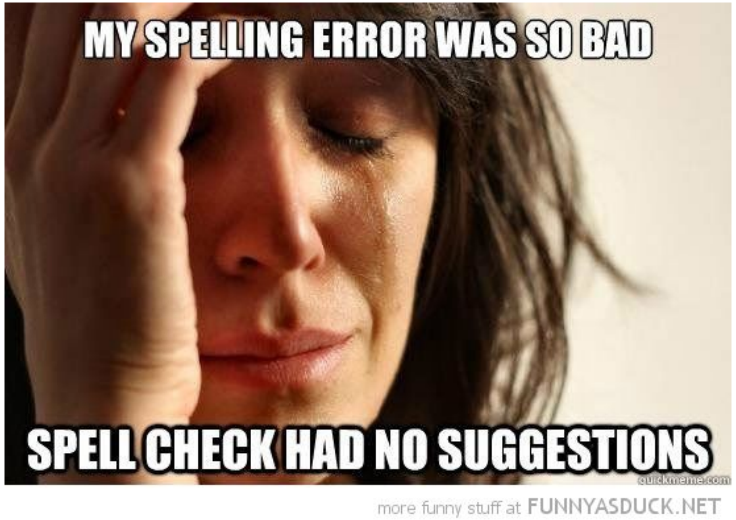
The search engine that doesn't track you. Help Spread DuckDuckGo!

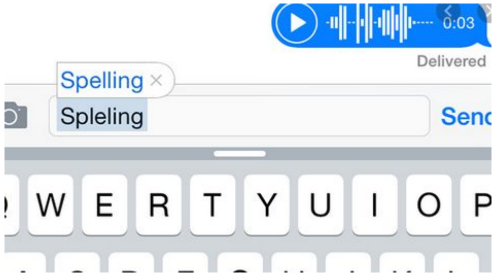
many more....

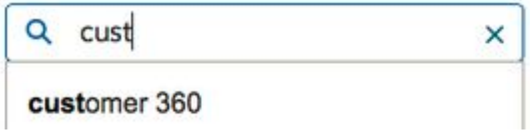
Problems Solved with NLP

❖ Spell Checker (*autocomplete*, *autocorrect*)



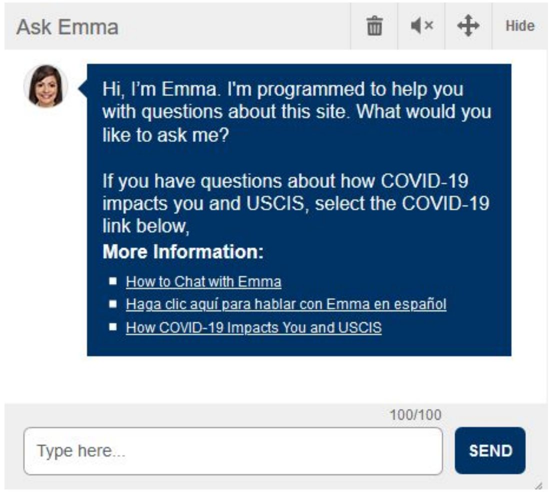

MY SPELLING ERROR WAS SO BAD
SPELL CHECK HAD NO SUGGESTIONS
quickmeme.com
more funny stuff at FUNNYASDUCK.NET





Problems Solved with NLP

❖ Robot Chat



Ask Emma

Hide

Hi, I'm Emma. I'm programmed to help you with questions about this site. What would you like to ask me?

If you have questions about how COVID-19 impacts you and USCIS, select the COVID-19 link below,

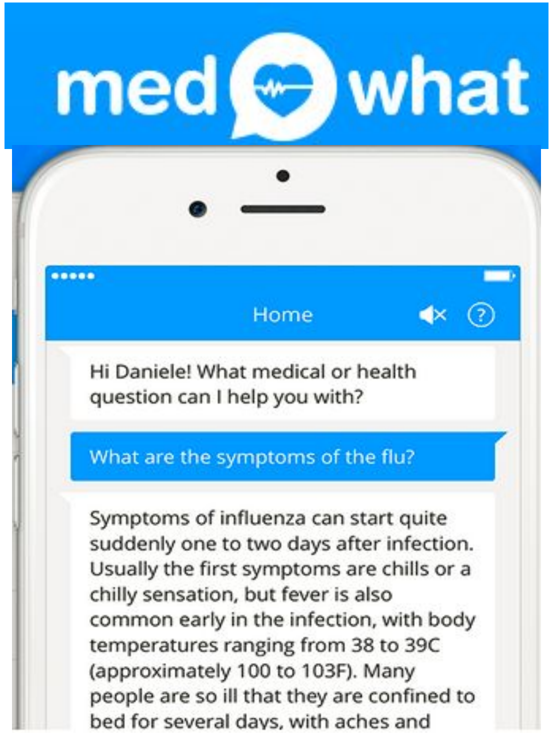
More Information:

- How to Chat with Emma
- Haga clic aquí para hablar con Emma en español
- How COVID-19 Impacts You and USCIS

100/100

Type here...

SEND



med what

Home

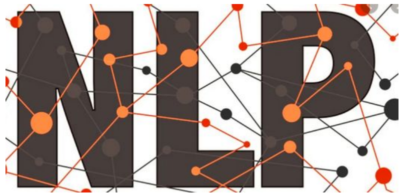
Hi Daniele! What medical or health question can I help you with?

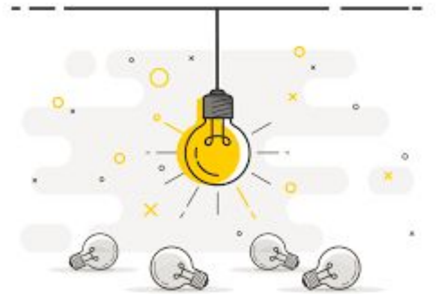
What are the symptoms of the flu?

Symptoms of influenza can start quite suddenly one to two days after infection. Usually the first symptoms are chills or a chilly sensation, but fever is also common early in the infection, with body temperatures ranging from 38 to 39C (approximately 100 to 103F). Many people are so ill that they are confined to bed for several days, with aches and

Major Pillars of NLP

- **Content categorization**. A linguistic-based document summary, including search and indexing, content alerts and duplication detection.
- **Topic discovery and modeling.** Accurately capture the meaning and themes in text collections, and apply advanced analytics to text, like optimization and forecasting.
- **Contextual extraction.** Automatically pull structured information from text-based sources.
- **Sentiment analysis.** Identifying the mood or subjective opinions within large amounts of text, including average sentiment and opinion mining.
- **Speech-to-text and text-to-speech conversion.** Transforming voice commands into written text, and vice versa.
- **Document summarization.** Automatically generating synopses of large bodies of text.
- **Machine translation.** Translating file/voice input from one language to the other





Basic Functions

- Packages
- Tokenization
- Stemming
- Lemmatization
- Stop Words
- Part-of-speech tagging
- Sentiment Analysis

Packages

- Data Frame Manipulation (selection, deletion, sorting…)
    - Numpy
    - Pandas
- Data Processing (specific operation pertinent to NLP procedures)
    - NLTK
    - TextBlob
- Modeling
    - Sklearn
        - KNN
        - Linear Regression

Tokenization

“The process of segmenting running text into sentences and words. It is the task of cutting a text into pieces called **tokens**.” Such tokens can vary according to user needs: split text into individual words, sentences, or **n-grams** (collections of n words).

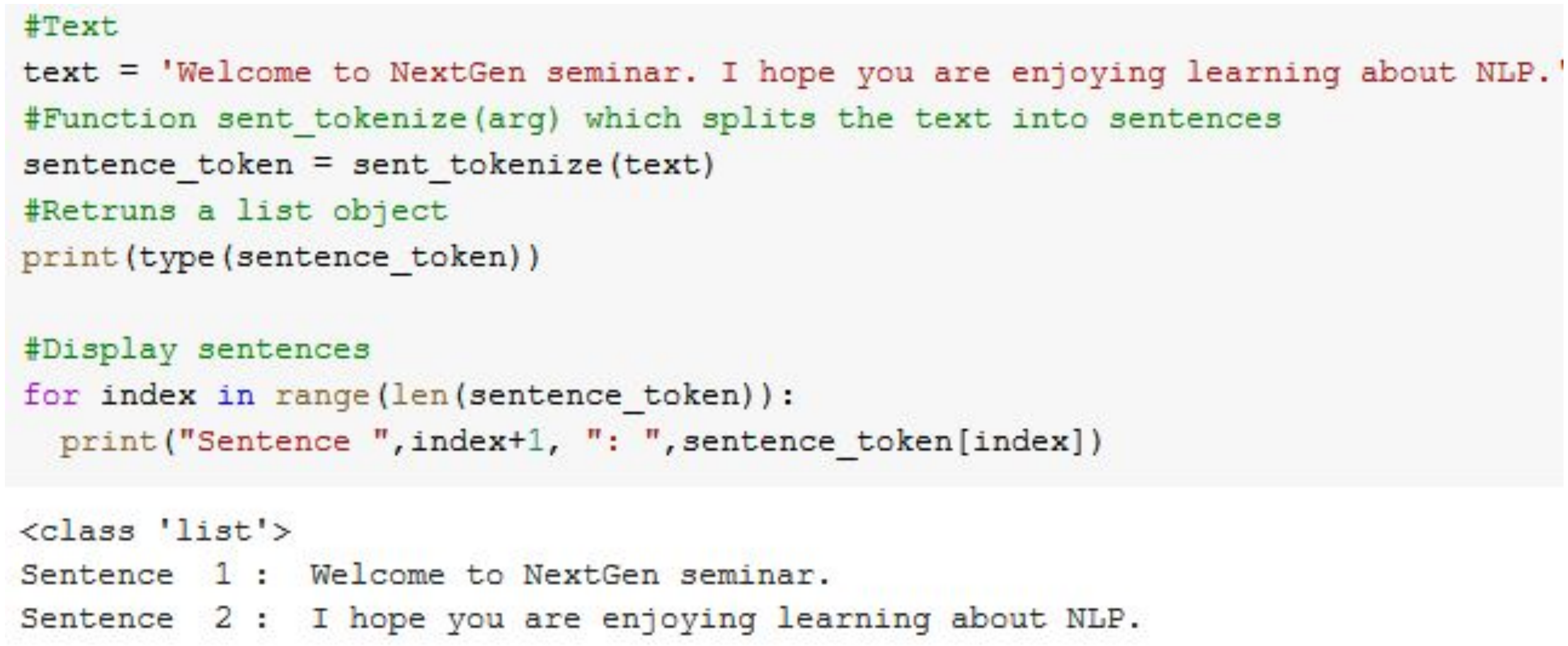
Example:

Word tokenization: “Hello Everyone” ↣ “Hello”, “Everyone”

Sentence tokenization: “Hello Everyone. How are you doing today?” ↣ “Hello”, “Everyone”, “.”, “How”, “are”, “doing”, “today” ,“?”

Bi-gram: “Today is a nice day” ↣ “Today is”, “ is a ”, “a nice”, “nice day”

Tokenization ...



```
#Text
text = 'Welcome to NextGen seminar. I hope you are enjoying learning about NLP.'
#Function sent_tokenize(arg) which splits the text into sentences
sentence_token = sent_tokenize(text)
#Retruns a list object
print(type(sentence_token))

#Display sentences
for index in range(len(sentence_token)):
  print("Sentence ",index+1, ": ",sentence_token[index])
```

```
<class 'list'>
Sentence  1 :  Welcome to NextGen seminar.
Sentence  2 :  I hope you are enjoying learning about NLP.
```

Stemming

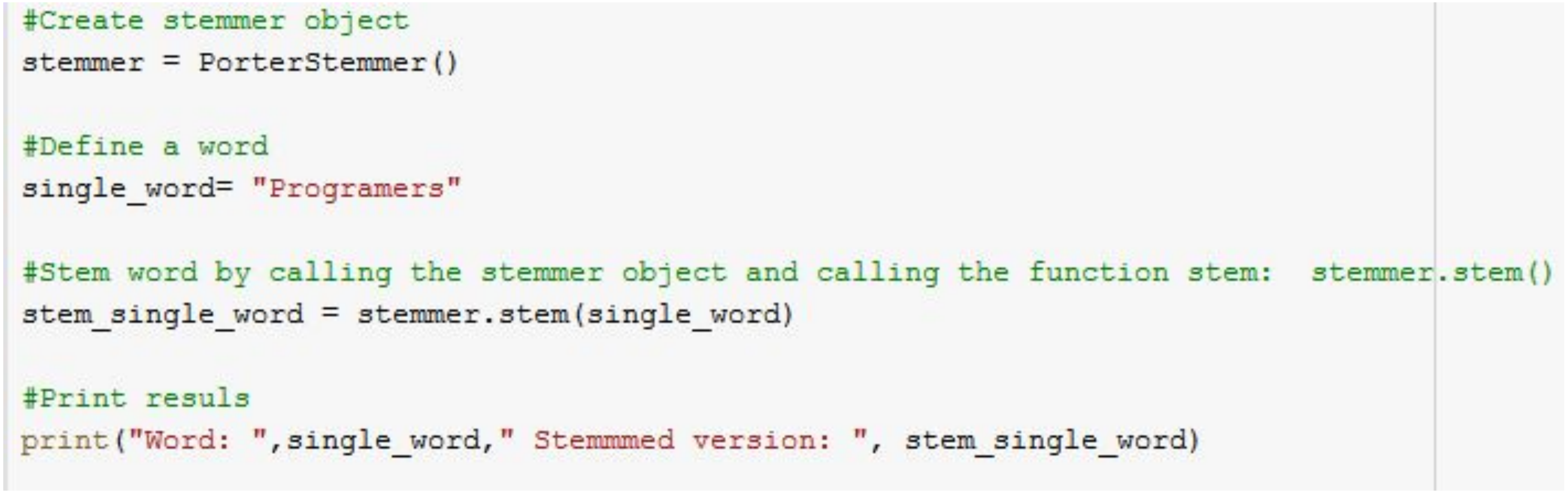
"Stemmers remove morphological **affixes** from words, leaving only the word stem." In short, removing *'extras'* from a word and reduce it to its basic root.

Example: beautiful ⤐ beauty , dies ⤐ die, flying ⤐ fly, mules ⤐ mule

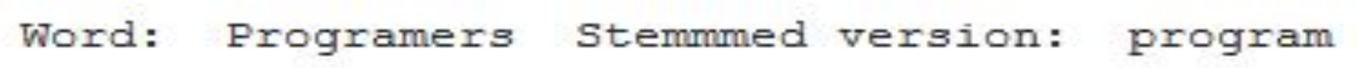
OverStemming: stemming a word when non-necessary (frequent problem).
Universal⤐ Universe ,   University ⤐ Universe

Porter Stemmer: common stemmer used, although it is not the most precise, better stemmers exist, but it is useful for basic stemming needs.

Stemming ...



```
#Create stemmer object
stemmer = PorterStemmer()

#Define a word
single_word= "Programers"

#Stem word by calling the stemmer object and calling the function stem:  stemmer.stem()
stem_single_word = stemmer.stem(single_word)

#Print resuls
print("Word: ",single_word," Stemmmed version: ", stem_single_word)
```



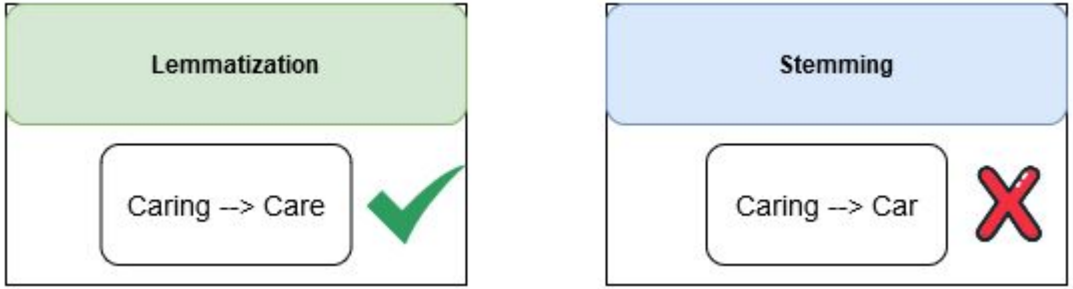
```
Word:  Programers  Stemmmed version:  program
```

Lemmatization

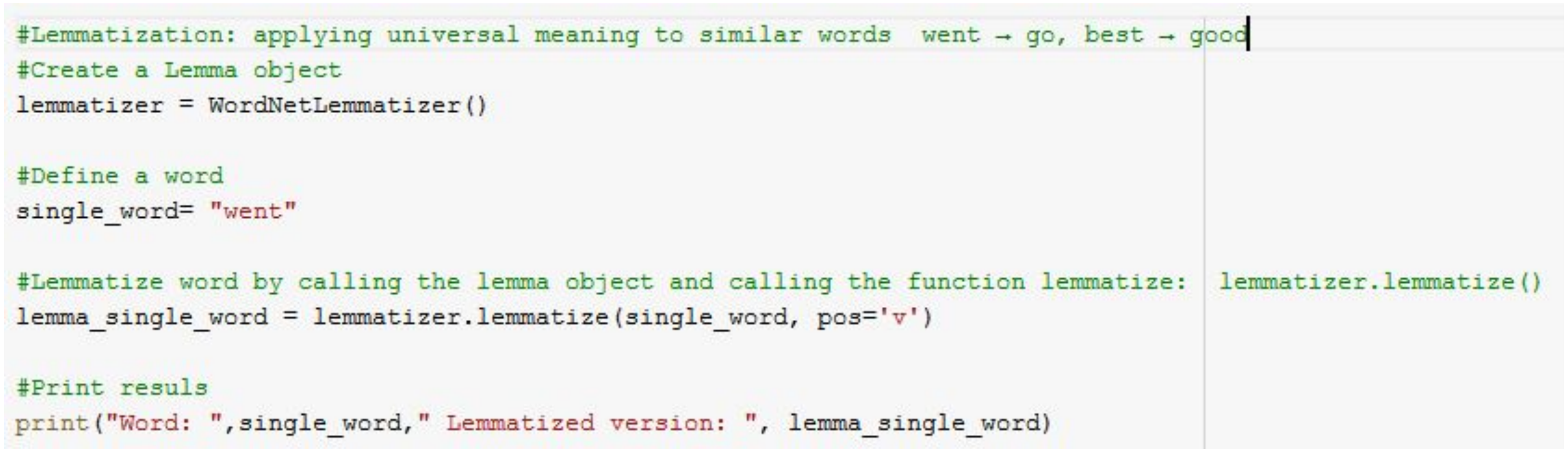
Similar task to stemming, in the sense that we are modifying a word to its root, but with the difference that in this scenario the change is based on "**meaning**" rather than affixes. Similar words that have similar meaning are considered equal.

Example: verbs in the past tense are changed to the present (went ↠ go), synonyms are unified (best ↠ good).

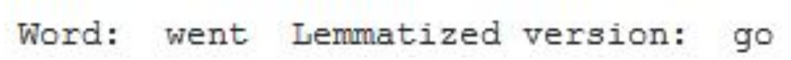
Efficiency of this libraries is based on the dictionaries (vocabulary) that they use.



Lemmatization...



```
#Lemmatization: applying universal meaning to similar words  went → go, best → good
#Create a Lemma object
lemmatizer = WordNetLemmatizer()

#Define a word
single_word= "went"

#Lemmatize word by calling the lemma object and calling the function lemmatize:  lemmatizer.lemmatize()
lemma_single_word = lemmatizer.lemmatize(single_word, pos='v')

#Print resuls
print("Word: ",single_word," Lemmatized version: ", lemma_single_word)
```



```
Word:  went  Lemmatized version:  go
```

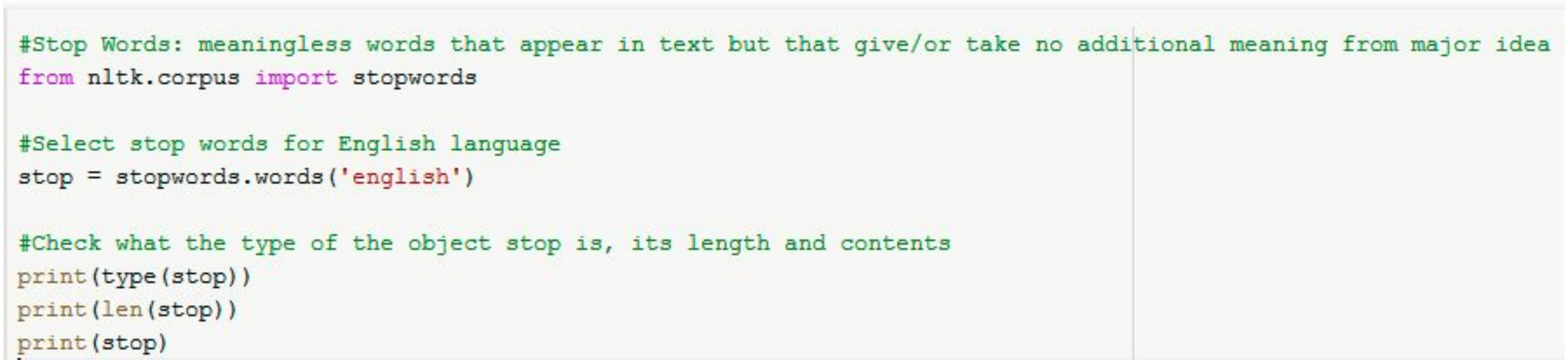
Stop Words

“Stop Words are words which **do not** contain important significance to be used in search queries. Usually, these words are filtered out from search queries because they return a vast amount of unnecessary information.

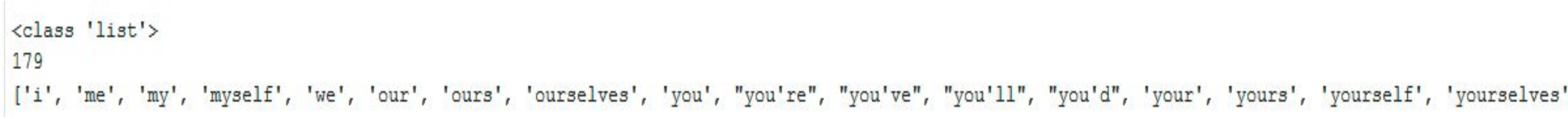
Example: words that are commonly used in the English language such as 'as, the, be, are' etc.

Each programming language will give its own list of stop words to use, and you can always add more stop words, or create your own customized version.

Stop Words...



```
#Stop Words: meaningless words that appear in text but that give/or take no additional meaning from major idea
from nltk.corpus import stopwords

#Select stop words for English language
stop = stopwords.words('english')

#Check what the type of the object stop is, its length and contents
print(type(stop))
print(len(stop))
print(stop)
```

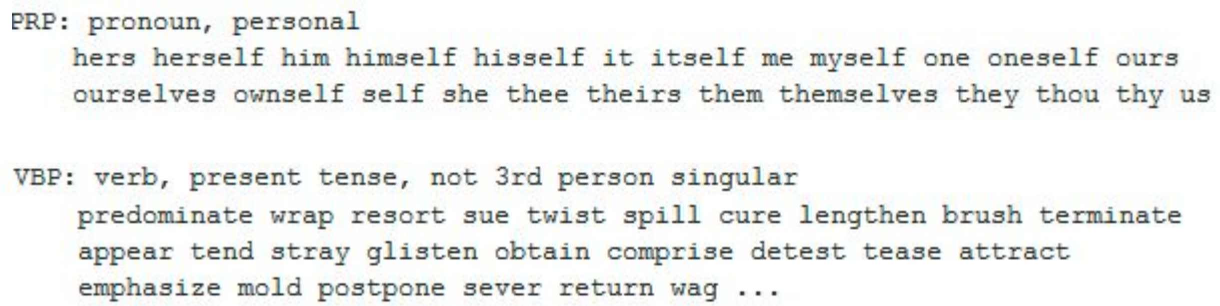


```
<class 'list'>
179
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves'
```

Part of Speech Tagging (POS)

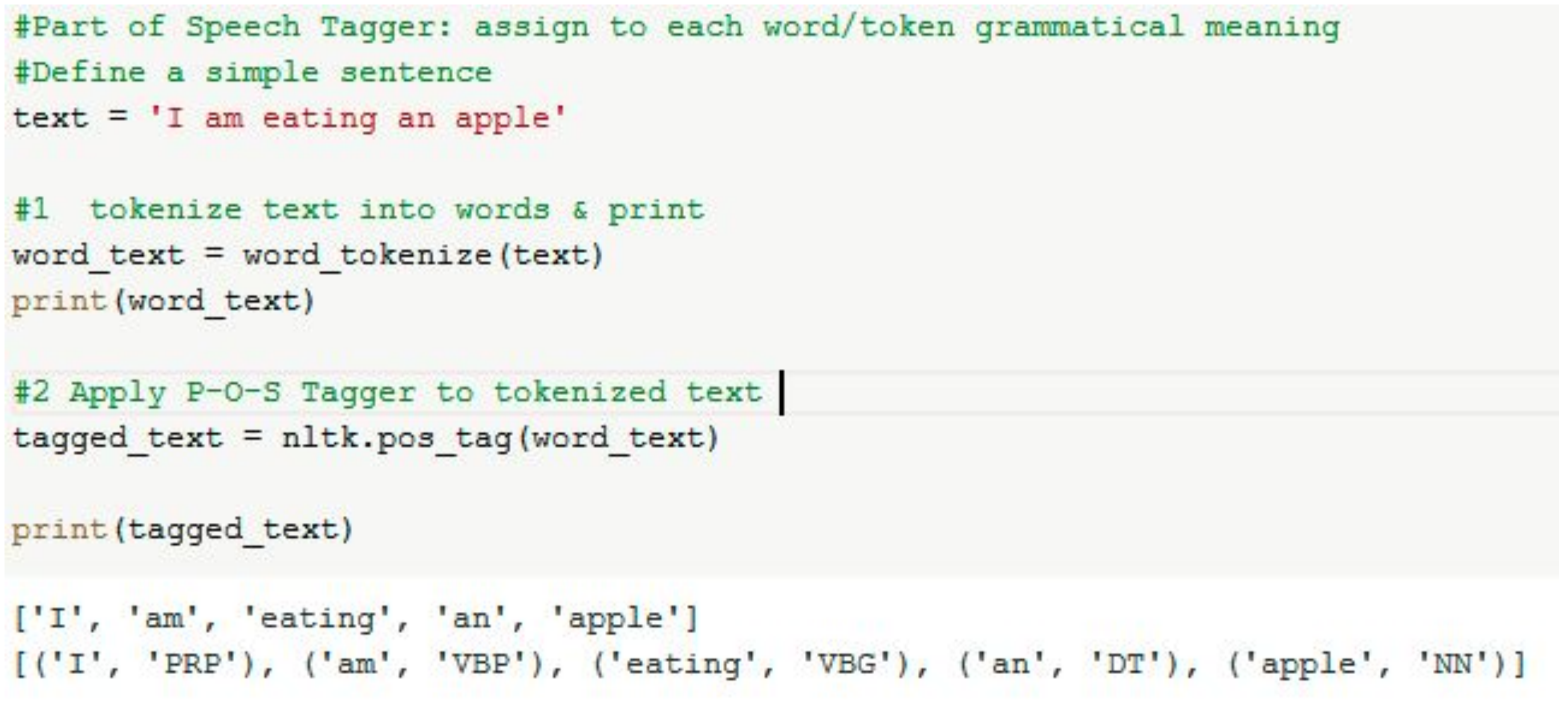
“A Part-Of-Speech Tagger (POS Tagger) is a piece of software that reads text in some language and assigns parts of speech to each word (and other token), such as noun, verb, adjective, etc.,”

<u>Example</u>: “I am eating an apple.” ↣ [(I,PRP), (am, VPB), (eating, VBG),(an,DT), (apple, NN)]



```
PRP: pronoun, personal
    hers herself him himself hisself it itself me myself one oneself ours
    ourselves ownself self she thee theirs them themselves they thou thy us

VBP: verb, present tense, not 3rd person singular
    predominate wrap resort sue twist spill cure lengthen brush terminate
    appear tend stray glisten obtain comprise detest tease attract
    emphasize mold postpone sever return wag ...
```

Part of Speech Tagging (POS)...



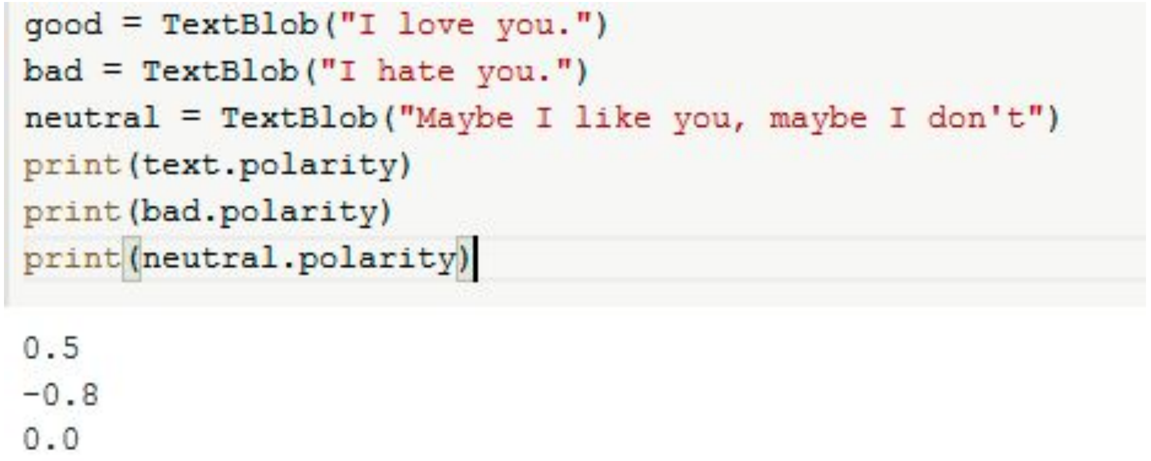
```
#Part of Speech Tagger: assign to each word/token grammatical meaning
#Define a simple sentence
text = 'I am eating an apple'

#1  tokenize text into words & print
word_text = word_tokenize(text)
print(word_text)

#2 Apply P-O-S Tagger to tokenized text
tagged_text = nltk.pos_tag(word_text)

print(tagged_text)
```

```
['I', 'am', 'eating', 'an', 'apple']
[('I', 'PRP'), ('am', 'VBP'), ('eating', 'VBG'), ('an', 'DT'), ('apple', 'NN')]
```

Sentiment Analysis

“Sentiment analysis is a machine learning technique that detects polarity (e.g. a *positive* or *negative* opinion) within text, whether a whole document, paragraph, sentence, or clause.”

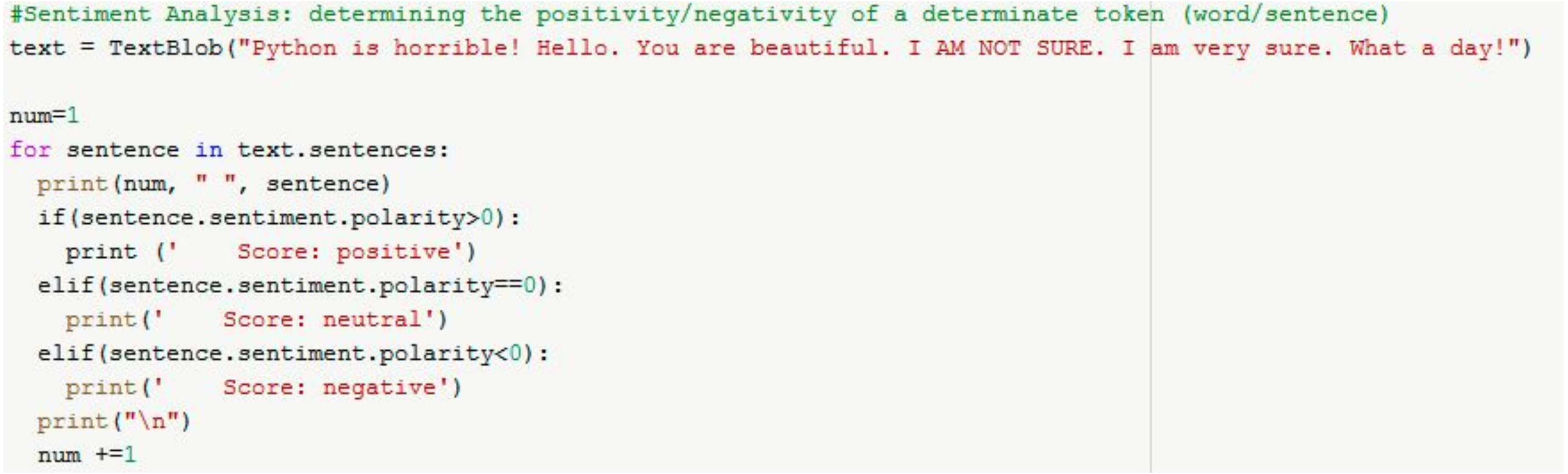
<u>Examples</u>



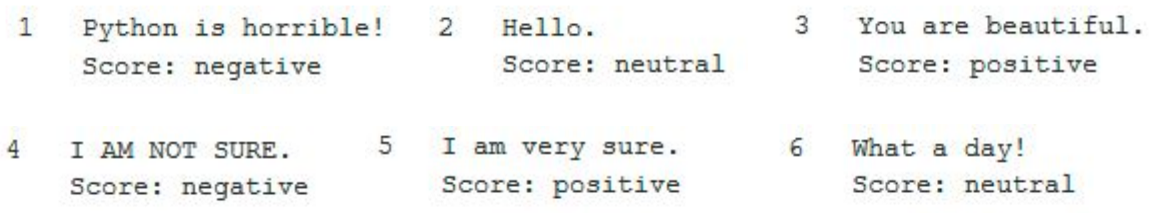
```
good = TextBlob("I love you.")
bad = TextBlob("I hate you.")
neutral = TextBlob("Maybe I like you, maybe I don't")
print(text.polarity)
print(bad.polarity)
print(neutral.polarity)
```

```
0.5
-0.8
0.0
```

Sentiment Analysis...



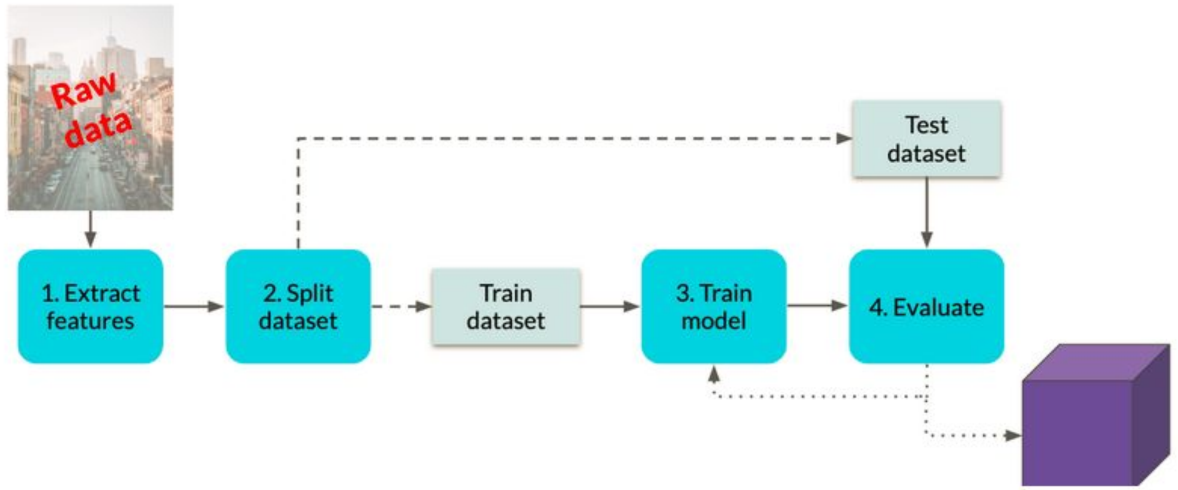
```
#Sentiment Analysis: determining the positivity/negativity of a determinate token (word/sentence)
text = TextBlob("Python is horrible! Hello. You are beautiful. I AM NOT SURE. I am very sure. What a day!")

num=1
for sentence in text.sentences:
  print(num, " ", sentence)
  if(sentence.sentiment.polarity>0):
    print ('    Score: positive')
  elif(sentence.sentiment.polarity==0):
    print('    Score: neutral')
  elif(sentence.sentiment.polarity<0):
    print('    Score: negative')
  print("\n")
  num +=1
```



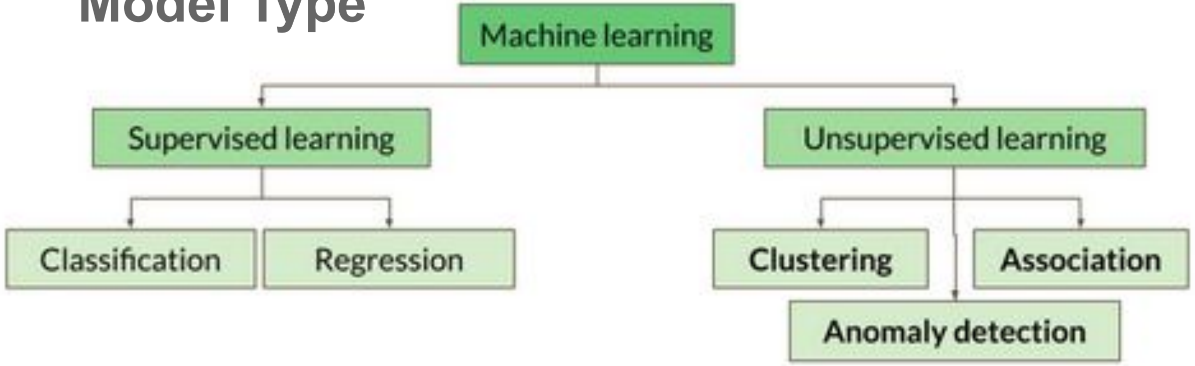
```
1   Python is horrible!   2   Hello.             3   You are beautiful.
    Score: negative           Score: neutral         Score: positive

4   I AM NOT SURE.        5   I am very sure.    6   What a day!
    Score: negative           Score: positive        Score: neutral
```

ML & NLP

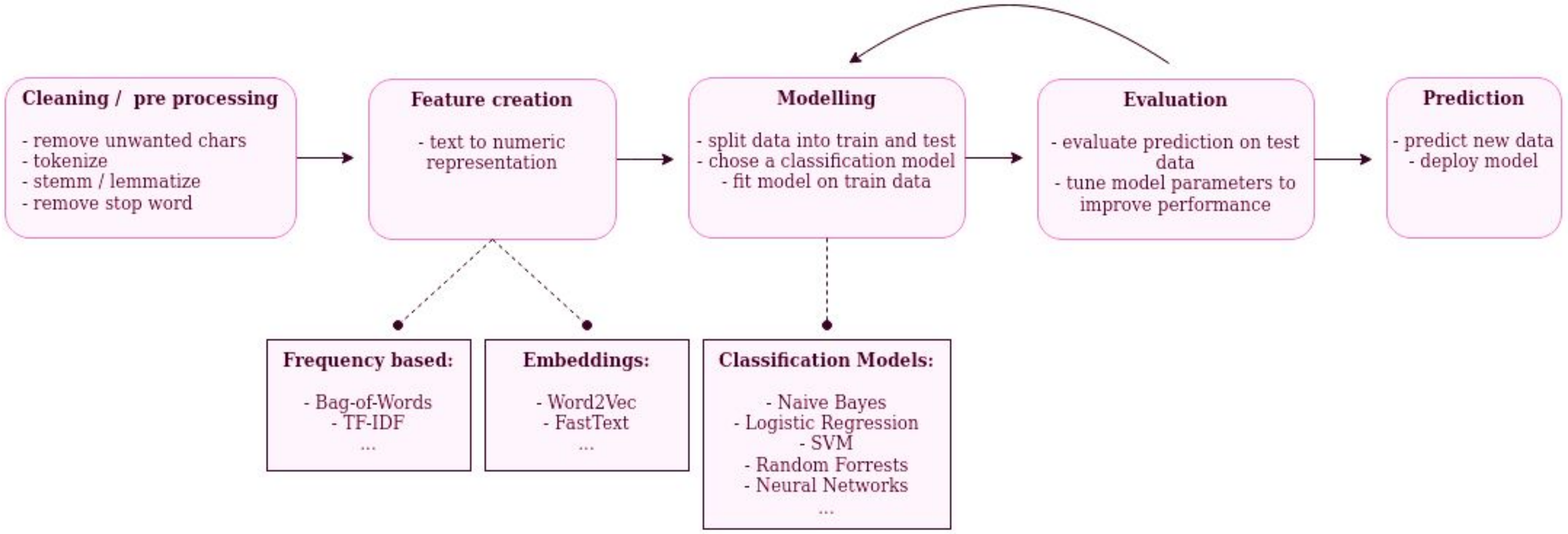
Machine learning workflow



Model Type



Typical NLP Workflow



Source: https://data-dive.com/german-nlp-binary-text-classification-of-reviews-part1

The challenge

Data quality

Machine Learning Model

- Garbage in garbage out
- Output quality depends on input quality

Slide Source: datacamp.com

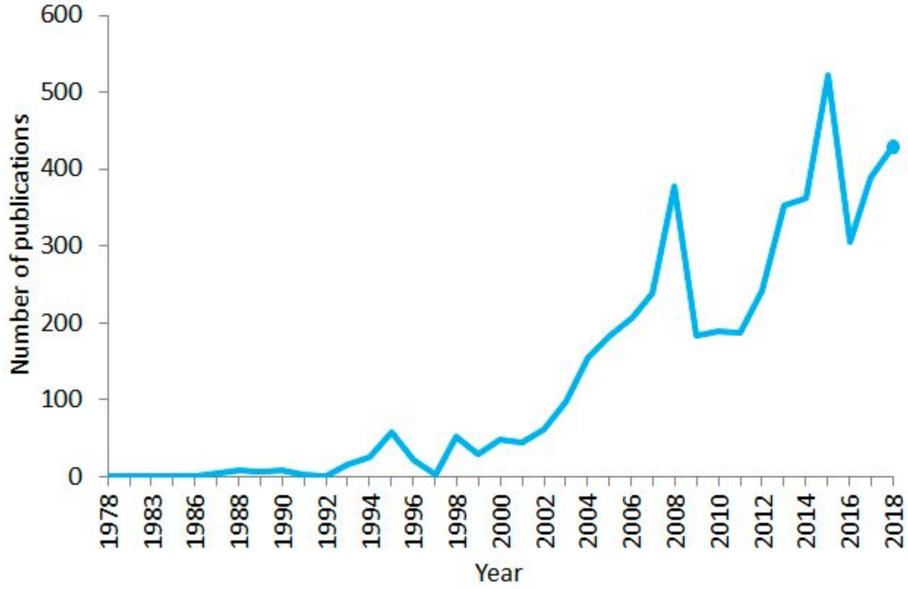
TayTweets the broken A.I.

The Future

PubMed: 30 million biomedical citations/articles

-Big Data ↣ need for structure and meaning ↣ NLP.

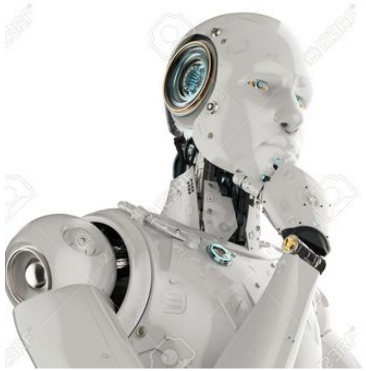
-Safe to conclude that as long as we will be dependent on Data, NLP is part of the process that allows for mass discoveries in an effortless manner.



Number of publications containing the sentence "natural language processing" in PubMed in the period 1978–2018. As of 2018, PubMed comprised more than 29 million citations for biomedical literature

Conclusion

- NLP is a subfield of ML that overlaps with other fields such as linguistics & C.S.
- NLP techniques are very powerful and used in a variety of different fields/scenarios: ……
- NLP aims at solving complex problems, and as a result big challenges and responsibilities comes as well. We should ensure quality control at every step.
- NLP is not just today’s reality, but tomorrow’s destiny as well.



Thank You!

References

TowardDS-https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1

SAS-https://www.sas.com/en_us/insights/analytics/what-is-natural-language-processing-nlp.html

NLPFlow-https://data-dive.com/german-nlp-binary-text-classification-of-reviews-part1

MonkeyLearn- https://monkeylearn.com/sentiment-analysis/

YT-https://www.youtube.com/watch?v=Lr4yi9onykg

Book-https://www.nltk.org/book/