Natural Language Processing

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

* How machines process and understand human language
  + Extracting information from words / text – training a model to understand and use language as a human
* Unstructured data

*Use cases*

* Predicting diseases from medical records and patients speech
* Customer feedback – sentimental analysis
* Email spam detecting
* Cognitive assistant
* Identify fake news (deep fakes)
* Financial traders – insight into what people are talking about
* Talent recruitment
* Automate litigation tasks
* Chatbot therapist
* Detecting dementia through speaking patterns

*Basic Models*

* Bag of words – counting word frequency disregarding structure and grammar
  + No room for semantics or context
  + Does not weight words (eg. ‘the’,’a’ common but not critical to meaning)
    - Term frequency – inverse document frequency (determining less meaningful words from their frequency in normal language)
* Tokenization – cutting words up and omitting punctuation
  + Uses blank spaces to chop not always meaningful (eg. ‘New York’)
* Stop words removal – removing common language articles
  + Not universal
* Stemming
  + Removing affixes and suffexes to try get just the overall context
    - Playing – play
    - News – new (doesn’t work as news is different to new)
  + Simple and fast to use
  + Can correct spelling mistakes and typos
* Lemmatisation – moving all words with same base into that base
  + Went – go, best – good
  + Resolves words to dictionary form (similar to stemming but more powerful – eg. caring to care not car)
  + More computationally expensive than stemming
* Topic modelling – clustering text into topics
  + Latent Dirichlet Allocation

<https://medium.com/aubergine-solutions/beginners-guide-to-natural-language-processing-e4981866cb64>

* Use text preprocessing because computers are better with numerical data
* Tokenisation
  + Long strings of texts into smaller tokens
* Stemming
  + Removing affixes
* Lemmatisation
  + Reducing words to their ‘base’
* Corpus = body
  + Collection generated from our text data/ dictionary used for the NLP model
* Stop words
  + Play no part in the meaning of the sentence
  + A, the, and
* Parts of speech (POS)
  + Nouns, verbs, adjectives
  + Helps in understanding the role of a word in a sentence
* Bag of words
  + Representation of sentences that a machine learning model can understand
  + Frequency dictionary for the text
  + Fails to convey the meaning of a sentence as words with high frequency dominate the model
* N-grams
  + Sequence of N items in a text selection
  + More useful for storing the context of sentences
* TF-IDF vectorizer (term frequency – inverse document frequency)
  + Occurrence of a word in a sentence / occurrence of a word in the document
  + Low importance to words with high frequency without removing stop words all together

**Chunking in NLP**

<https://towardsdatascience.com/chunking-in-nlp-decoded-b4a71b2b4e24>

Chunking works on top of POS tagging and extracts phrases from unstructured text by identifying the constituents of a sentence (eg. noun, verbs etc). It does not specify the structure or role in the sentence.

Chunking is important to put on top of tokenization so that phrases can be identified and be more useful to understanding context of a sentence (eg. New York).

A sentence will typically follow the structure: sentence – clauses – phrases – words.

* Noun phrases (NP)
* Verb phrases (VP)
* Adjective phrase (ADJP)
* Adverb phrase (ADVP)
* Prepositional phrase (PP)

Diagram

Description automatically generated

We must introduce some grammar to the noun phrase which is:

* DT? – one or zero determiner
* JJ\* - zero or more adjectives
* NN – noun

This grammar can then be parsed. S

Chart

Description automatically generated