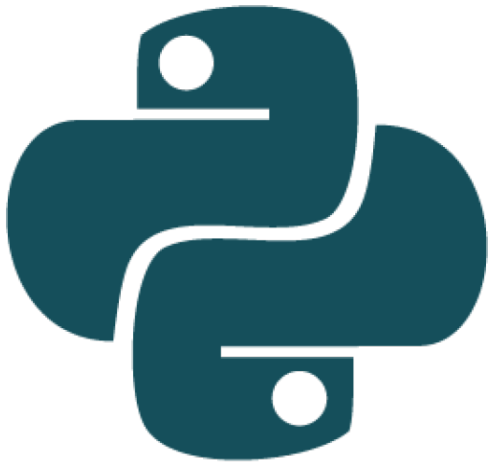
Topic 1
The art of preprocessing



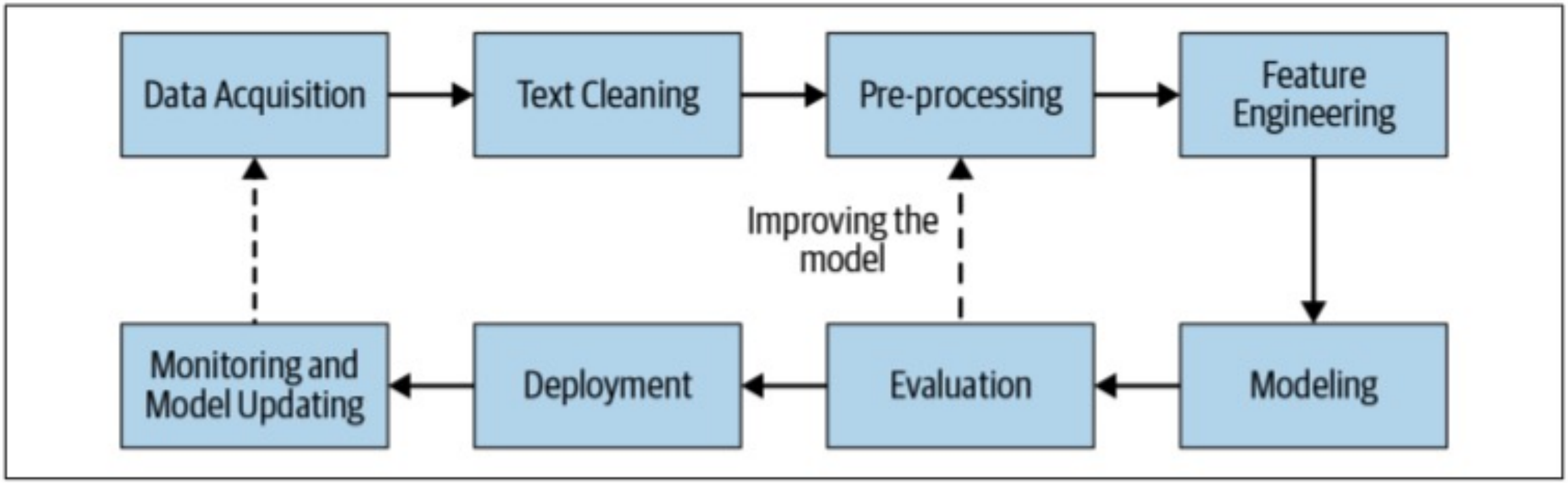
NLTK


GREAT
MODEL
GARBAGE
DATA
GARBAGE
RESULTS

What benefits does preprocessing text have when you are not interested in using ML or DL?

- Most NLP software works on the level of sentences or words
  - So at the very least we need sentence tokenization/segmentation and word tokenization
- When scraping data from the web or extracting from and image (pdf) it can require removing non-textual information such as markup and metadata

Generic NLP Pipeline



Natural Language Toolkit (NLTK)

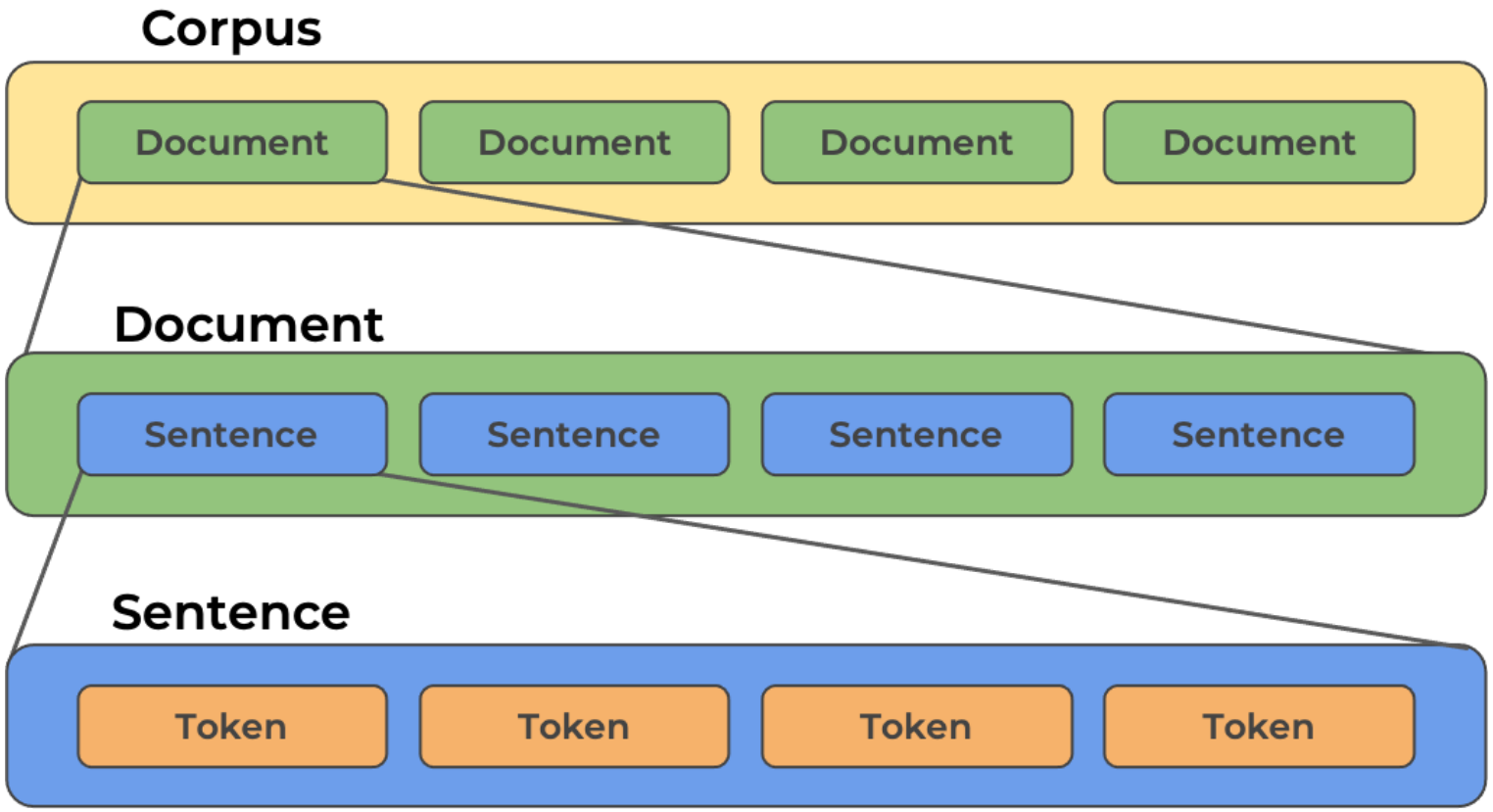
- One of the leading platforms for working with human language data and Python
  - https://www.nltk.org/
- Before preprocessing, we need to first download the NLTK library.

Common cleaning and pre-processing steps

- Removing punctuations e.g. , ! ? .
- Removing URLs
- Removing stop words
- Lower casing
- Tokenization
- Stemming
- Lemmatization

What is a corpus?

- A corpus of documents is the set of all documents in a dataset.



Corpuses

- Gutenberg
- Webtext
- Brown
- Reuters
- Etc.

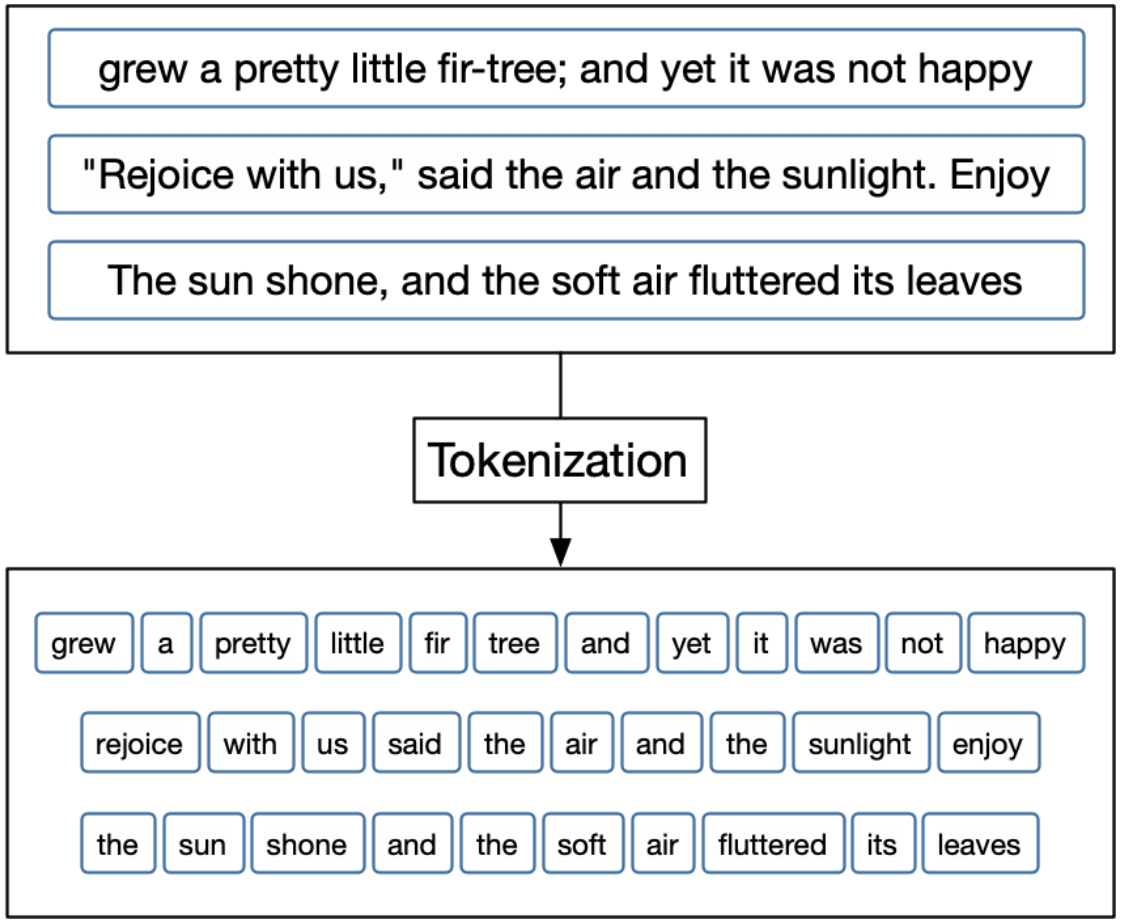
These corpora export several important methods:

- paras
- sents
- words
- raw

NLTK is a string-based library

Tokenization

- Word tokenizers breaks a piece of text into words
  - Unigrams
  - Bigrams
  - Trigrams
  - N-grams
- Sentence tokenizers break a piece of text into sentences



https://smltar.com/tokenization.html

Example Tokenizers

- Whitespace
- Word and Sentence
- Punctuation-based
- Treebank Word
- Tweet tokenizer
- MWETokenizer

Example

```
# pip install nltk
# conda install -c anaconda nltk

import nltk
from nltk.tokenize import word_tokenize

words = word_tokenize(text)
print(words)
```

Limitations?

- Punctuation
- Language morphology

Turkish	English
kork(-mak)	(to) fear
korku	fear
korkusuz	fearless
korkusuzlaş (-mak)	(to) become fearless
korkusuzlaşmış	One who has become fearless
korkusuzlaştır(-mak)	(to) make one fearless
korkusuzlaştırıl(-mak)	(to) be made fearless
korkusuzlaştırılmış	One who has been made fearless
korkusuzlaştırılabil(-mek)	(to) be able to be made fearless
korkusuzlaştırılabilecek	One who will be able to be made fearless
korkusuzlaştırabileceklerimiz	Ones who we can make fearless
korkusuzlaştırabileceklerimizden	From the ones who we can make fearless
korkusuzlaştırabileceklerimizdenmiş	I gather that one is one of those we can make fearless
korkusuzlaştırabileceklerimizdenmişçesine	As if that one is one of those we can make fearless
korkusuzlaştırabileceklerimizdenmişçesineyken	when it seems like that one is one of those we can make fearless

Stop words

```
from nltk.corpus import stopwords
nltk.download('stopwords')

language = "english"
stop_words = set(stopwords.words(language))
print(stop_words)
```

```
[nltk_data] Downloading package stopwords to
```

```
{"hasn't", 'hers', 'do', 'is', 'in', 'on', 'been', 'does', 'mightn', "you'd", "couldn't", 'co
uldn', "she's", 'it', 'i', 'own', "mightn't", 'which', 'have', 'or', 'themselves', "needn't",
'has', 'when', 'his', 'further', 'off', 'our', 'of', 'how', "hadn't", 'any', 'are', 'very',
'them', 'into', 'same', 'isn', 'because', 'd', 'wasn', 're', 'each', 'an', 'after', 'agains
t', 'until', 'don', 'll', 'they', 'while', 'under', 'had', 'with', 'here', 'just', "didn't",
'only', 'not', 'now', "you'll", 'having', 'ourselves', 'did', 'hasn', "haven't", 'the', "yo
u're", 't', 'so', 'he', 'too', 'we', 'once', 'hadn', 'that', 'these', "shan't", 'doesn', 'wo
n', 'aren', 'between', "it's", 'few', 'haven', 'she', 'ma', 'by', 've', "mustn't", 'above',
```

Exercise

- NLTK Practice
- Find a corpus in NLTK
- Remove the stop words from your selected text

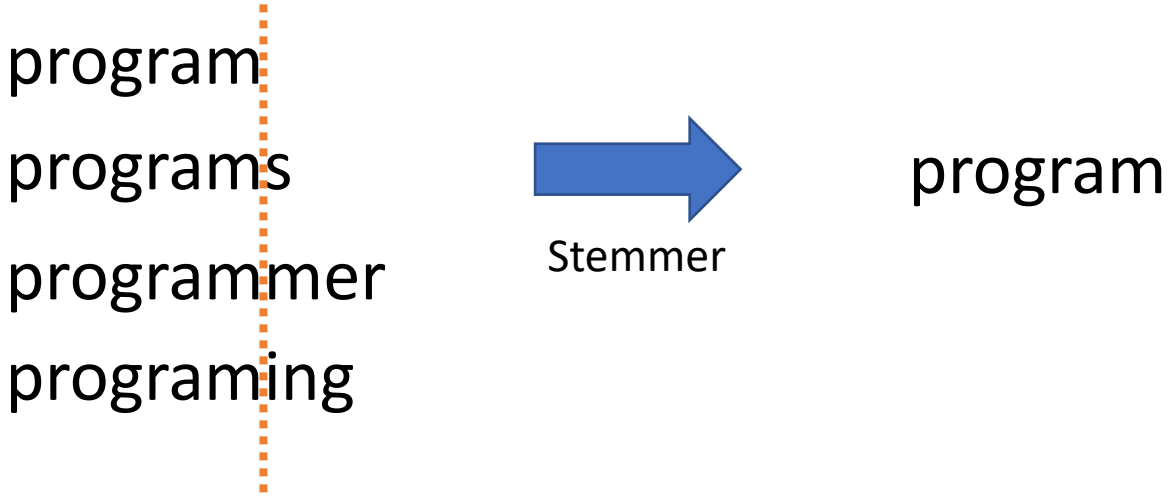
What about Shakespeare?

- “to be or not to be” – loaded with meaning in English
- What happens when you apply your function to remove stop words?

- Or what about Sentiment Analysis? (we will talk more about this later)
  - What happens to the statement “I am not happy”?

Stemming

- Handcrafted rules to strip endings of words to reduce them to a common form called stems
  - Porter
  - Lancaster
  - Snowball



Lemmatization

- Lemmas are root forms of words
  - How is this different from stemming?
  - Lemmatization requires more linguistic knowledge **better -> good** while better stays the same when stemmed
  - Stemming is removing the suffix of words to produce some form of base word

```
words = ["connects", "connected", "strange", "is", "am"]

stemmed = ["connect", "connect", "strang", "is", "am"]

lemmatized = ["connect", "connect", "strange", "be", "be"]
```

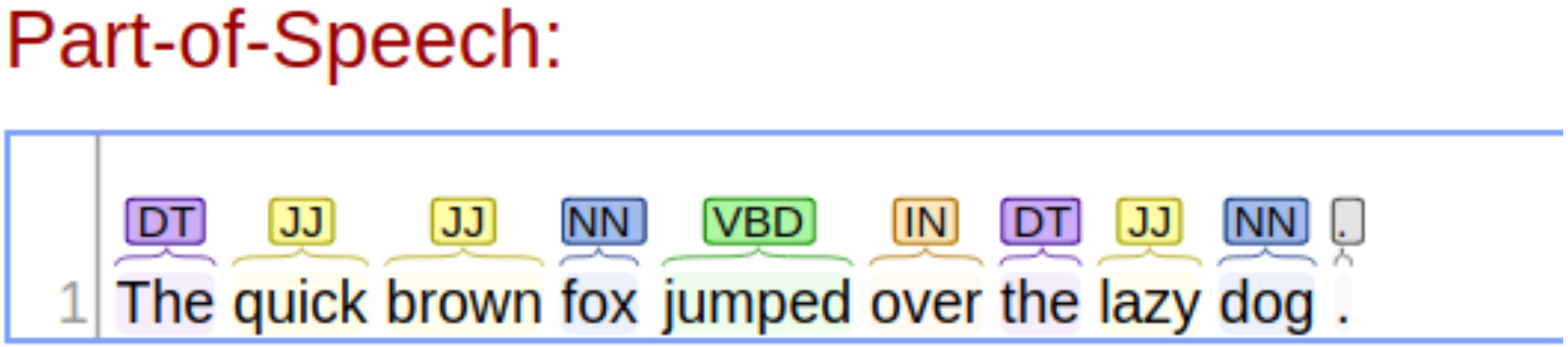
https://byteiota.com/stemming-and-lemmatization/

What good is stemming and lemmatization?

- Information retrieval: using a single word to represent a vector of related words
- Reducing the number of words to reduce the overall size and, thus the complexity of the total data in the system
- Should I do stemming AND lemmatization?

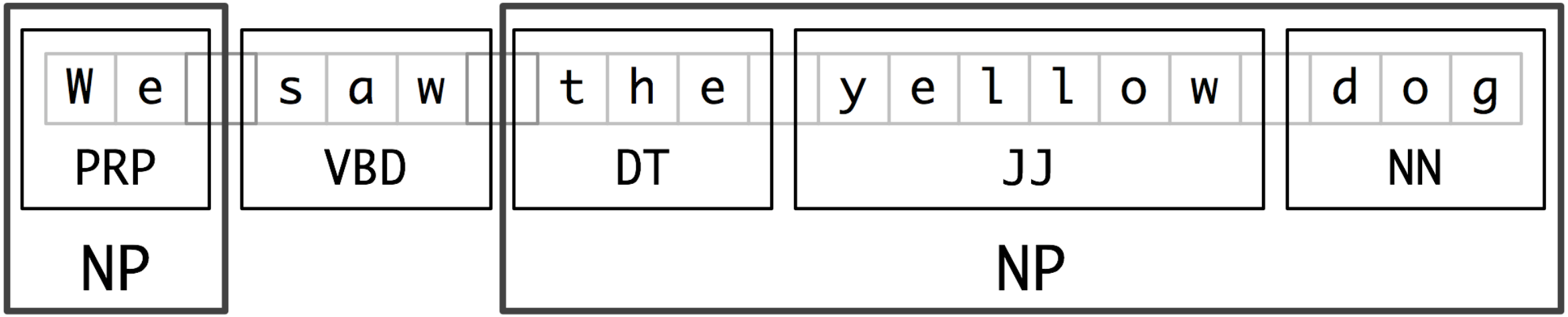
POS tagging

nltk.**help.upenn_tagset()** will get you a list of these tags (in English)



Chunking

- POS tagging combined with regular expressions
- Way to select subsets of tokens

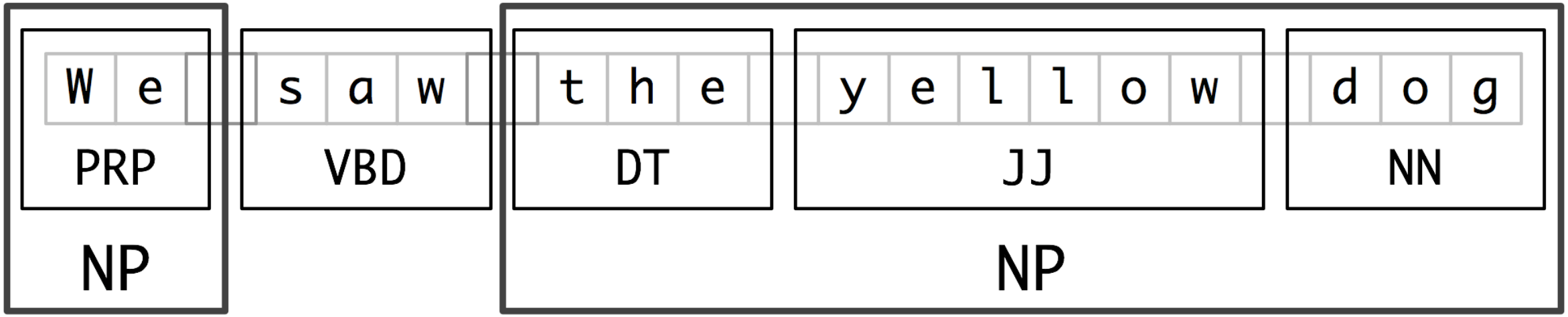


https://www.nltk.org/book/ch07.html

Chunking

- POS tagging combined with regular expressions
- Way to select subsets of tokens

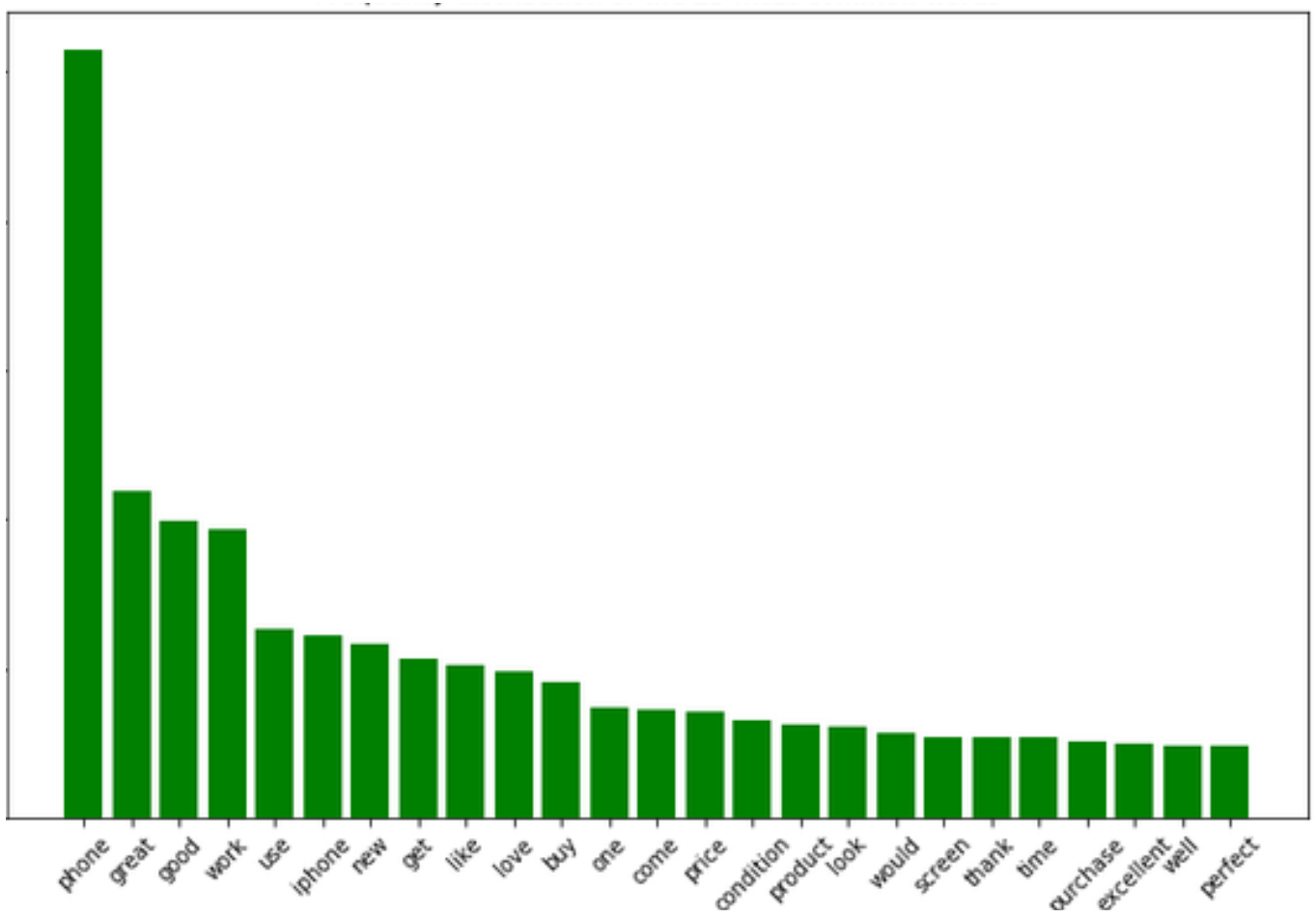
Almost every **sentence** contains at least one noun **phrase**.



https://www.nltk.org/book/ch07.html

Frequency

- For statistical machine learning approaches to NLP, the very first thing we usually need to do is count things
  - FreqDist
  - ConditionalFreqDist

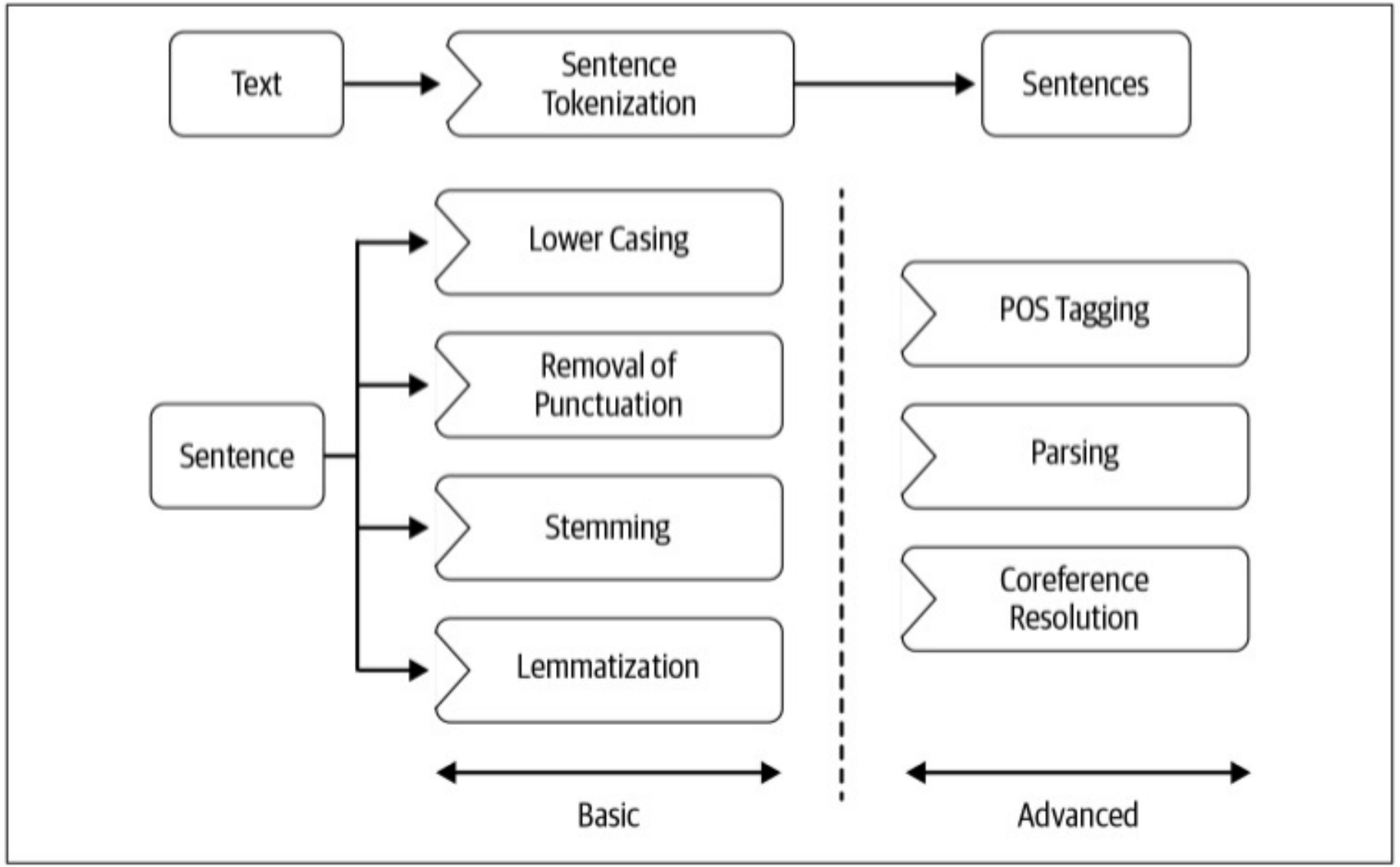


Normalization

- Capture variations of text into a single representation

Raw	Normalized
2moro 2mrrw 2morrow 2mrw tomrw	tomorrow
b4	before
otw	on the way
:) :-) ;-)	smile

Summary of Pre-processing Pipeline



Many, many, many more things you can do with NLTK

- Named entity recognition: Identify and classify named entities (people, organizations, etc.)
- Sentiment analysis: Determine the sentiment (positive, negative, neutral)
- Text classification: Assign a text to a predefined category (e.g., spam or not spam)
- Machine translation: Translate text from one language to another
- Language identification: Determine the language
- Text summarization: Generate a summary of a longer piece of text

Many, many, many more things you can do with NLTK

- **Named entity recognition: Identify and classify named entities (people, organizations, etc.)**
- **Sentiment analysis: Determine the sentiment (positive, negative, neutral)**
- **Text classification: Assign a text to a predefined category (e.g., spam or not spam)**
- Machine translation: Translate text from one language to another
- Language identification: Determine the language
- Text summarization: Generate a summary of a longer piece of text

Activity

- Write a function to find the frequency of the top X words in of the Reuters corpus.
- Report how many of these are stop words for the top 10, 20, and 30 words.

Next class

- We will be working with SpaCy and start comparing to NLTK