Context

- Context is how various parts in a language come together to convey a particular meaning. Context includes long-term references, world knowledge and common sense along with the literal meaning of words and phrases.
- The meaning of a sentence can change based on the context as words and phrases can sometimes have multiple meanings. Generally, context is composed from semantics and pragmatics.
- Semantics is the direct meaning of the words and sentences without external context.
- Pragmatics adds world knowledge and external context of the conversation to enable us to infer implied meaning.

Pre-processing

- Why do we have to pre-process text?
- NLP software typically works at the sentence level and expects a separation of words at the minimum.
- So, we need some way to split a text into words and sentences before proceeding further in a processing pipeline.
- Some times, we need to remove special characters and digits, and sometimes, we don't care whether a word is in upper or lowercase and want everything in lowercase.

Tokenization

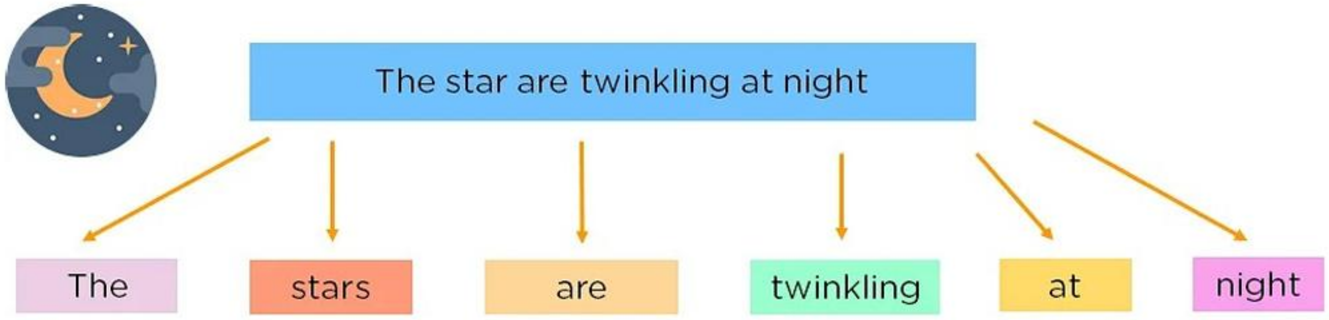
- A commonly used library is Natural Language Tool Kit (NLTK)

•word_tokenize: Each token is a single word, punctuation character, or number.

•word_tokenize: Each token is a single word, punctuation character, or number.

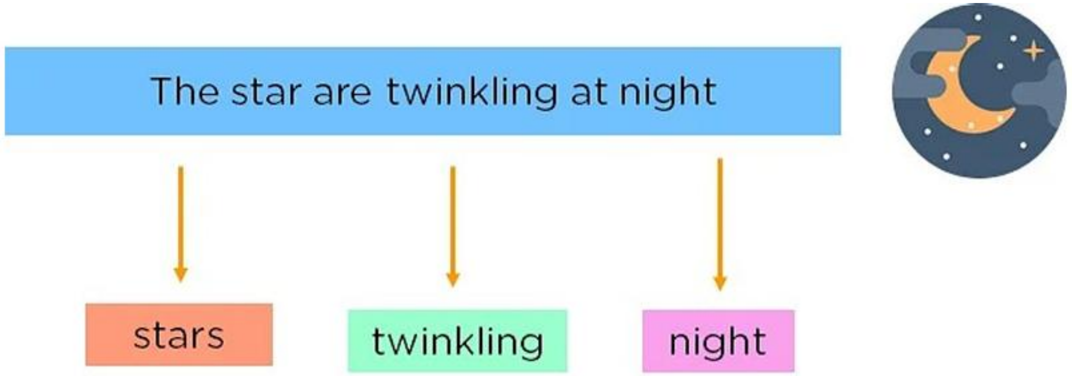
Tokenization

- It is the process of splitting text into meaningful units called *tokens*.
- We will import from NLTK two functions that split texts into tokens:
  - **word_tokenize**: Each token is a single word, punctuation character, or number.
  - **sent_tokenize**: Each token is an entire sentence extracted from the text. It is particularly useful for processing large chunks of text by identifying sentence boundaries based on punctuation, capitalization, and other language patterns.
    - **Punctuation Detection**
    - **Abbreviation Handling**



Removing Stop Words

- Some words, typically called **stop-words**, are so common in all texts that they add little to none information for tasks like text classification and others. Typical examples are articles and prepositions.
- The NLTK library provides a list of stopwords for many languages.
- Stopwords are typically filtered as a preprocessing step before many NLP tasks.
- Typical approach:
  - Import NLTK
  - Tokenize
  - Remove stop words
  - Count remaining

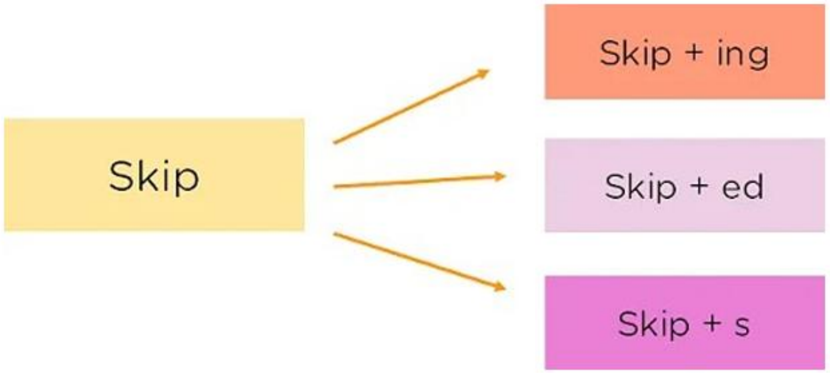


Removing Duplicates

- Tokenize
- Convert to lower case
- Remove duplicates
- Build vocab
- Find size of the vocab

Stemming

- **Stemming** is the process of reducing inflected words to their stem
- i.e. their base or root form.
- The stem need not be identical to the root form of the word or may not be an existing word in the first place, but it’s usually sufficient that related words reduce to the same stem.

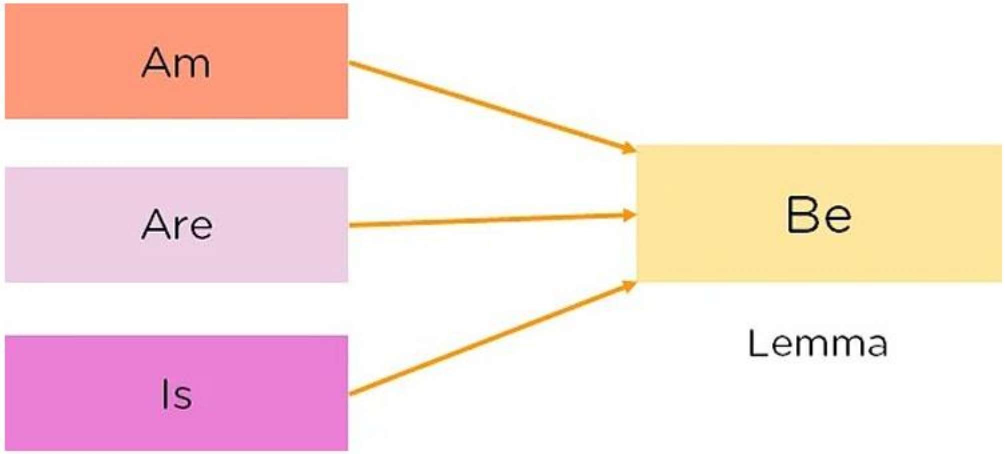


Stemming

- A popular stemmer for the English language is the **PorterStemmer**, which performs suffix stripping to produce stems.
- **Lancaster Stemmer**: More aggressive than Porter Stemmer.
- Steps:
  - **Import the necessary stemmer**
  - **Tokenize the text** to break it into individual words.
  - **Apply stemming** to each word using the chosen stemmer.
  - **Reconstruct or process the stemmed words.**

Lemmatization

- **Lemmatization** is used to group together the inflected forms of a word so that they can be analyzed as a single item, i.e. their **lemma**.
- The process of obtaining the Root Stem of a word. Root Stem gives the new base form of a word that is present in the dictionary and from which the word is derived.
- You can also identify the base words for different words based on the tense, mood, gender,etc


Am
Are
Is
Be
Lemma