Tokenization

Tokenization: The process of splitting a text into smaller units called tokens. These tokens can be words, subwords, or characters.

Cleaning : Focuses on removing noise and irrelevant information from the corpus. This can include:

Removing HTML tags: If your corpus is scraped from the web, it might contain HTML tags that need to be removed.

Removing special characters: Characters like emojis, symbols, and non-alphanumeric characters that don't contribute to the meaning of the text. The decision to remove or keep these depends on the task. For sentiment analysis, emojis might be important.

Removing punctuation: Punctuation marks can be removed or handled specially depending on the task. For example, you might remove punctuation for keyword extraction but keep it for sentence boundary detection.

Removing stop words: Common words like "the," "a," "is," etc., that often don't carry much meaning and can be removed to reduce the size of the vocabulary. However, be careful, as stop words can be important in some contexts (e.g., question answering).

Handling whitespace: Removing extra spaces, tabs, and newlines to ensure consistent formatting.

Removing numbers: Whether to remove or keep numbers depends on the task. For some tasks, numbers are important (e.g., extracting dates or prices).

Normalization : Aims to transform words into a standard form so that variations of the same word are treated as a single token. This includes:

Lowercasing: Converting all text to lowercase. This ensures that "The" and "the" are treated as the same word. However, be aware that lowercasing can sometimes be detrimental, especially when dealing with proper nouns or tasks where case is important (e.g., named entity recognition).

Stemming: Reducing words to their **root** form by removing suffixes (e.g., "running" -> "run"). Stemming can be aggressive and sometimes produces non-words. Common stemming algorithms include Porter stemmer and Snowball stemmer.

Lemmatization: Reducing words to their dictionary form (lemma) using morphological analysis (e.g., "better" -> "good"). Lemmatization is generally more accurate than stemming but also more computationally expensive. It requires a dictionary and part-of-speech tagging.

Handling contractions: Expanding contractions (e.g., "can't" -> "cannot").

- **Spelling correction:** Correcting misspelled words.
- **Replacing synonyms:** Replacing words with their synonyms to reduce vocabulary size.