1. What are Corpora?

A1. Corpora are large collections of written or spoken language used for linguistic research and natural language processing tasks, providing a basis for empirical analysis and development of language-related applications.

1. What are Tokens?

A2. Tokens are individual units of meaning in natural language processing, typically representing words or punctuation marks, that are generated by the process of tokenization.

1. What are Unigrams, Bigrams, Trigrams?

A3. Unigrams, bigrams, and trigrams are types of n-grams, which are contiguous sequences of n items in a given text.

* Unigrams are single words or tokens in a text.
* Bigrams are sequences of two adjacent words in a text.
* Trigrams are sequences of three adjacent words in a text.

For example, in the sentence "The quick brown fox jumps over the lazy dog", the unigrams would be "The", "quick", "brown", "fox", "jumps", "over", "the", "lazy", and "dog". The bigrams would be "The quick", "quick brown", "brown fox", "fox jumps", "jumps over", "over the", "the lazy", and "lazy dog". The trigrams would be "The quick brown", "quick brown fox", "brown fox jumps", "fox jumps over", "jumps over the", "over the lazy", and "the lazy dog".

N-grams are often used in natural language processing tasks such as language modeling and text classification, as they can capture certain linguistic patterns and relationships between words in a text.

1. How to generate n-grams from text?

A4. To generate n-grams from a given text, follow these general steps:

1. Tokenize the text into individual words or tokens using a tokenizer. This step is necessary as n-grams are generated based on tokens.
2. Combine the tokens into sequences of n contiguous tokens, where n is the desired value for n-grams (e.g., unigrams, bigrams, trigrams, etc.).
3. Count the frequency of each n-gram in the text.
4. Explain Lemmatization

A5. Lemmatization is the process of reducing words to their base or dictionary form, which is known as the lemma. The resulting lemmas represent the canonical, or standardized, form of the word and can be used to group together different inflected forms of the same word.

For example, consider the inflected forms of the word "go": "go", "goes", "going", and "gone". By lemmatizing these words, we can reduce them to their base form, "go", which can be used to identify them as variations of the same word.

Lemmatization is a common preprocessing step in natural language processing tasks such as text classification, information retrieval, and machine translation. It can improve the accuracy and efficiency of these tasks by reducing the vocabulary size and improving the consistency of word representation.

Lemmatization is different from stemming, which is another text normalization technique that involves reducing words to their root form by removing suffixes and prefixes. Unlike stemming, lemmatization produces valid words that can be found in a dictionary, which can be useful for tasks such as text generation and language translation where word sense and grammatical correctness are important.

1. Explain Stemming

A6. Stemming is the process of reducing words to their base or root form by removing suffixes and prefixes. The resulting stems may not always be valid words, but they can be used to group together different inflected forms of the same word.

For example, consider the inflected forms of the word "jump": "jump", "jumps", "jumping", and "jumped". By stemming these words, we can reduce them to their root form, "jump", which can be used to identify them as variations of the same word.

Stemming is a common preprocessing step in natural language processing tasks such as text classification, information retrieval, and sentiment analysis. It can improve the efficiency of these tasks by reducing the vocabulary size and improving the speed of text processing.

However, stemming has limitations as it can result in incorrect word stems and can produce stems that are not valid words. This can be a problem for tasks where word sense and grammatical correctness are important, such as text generation and language translation. In such cases, lemmatization, which produces valid words that can be found in a dictionary, may be a better alternative.

1. Explain Part-of-speech (POS) tagging

A7. Part-of-speech (POS) tagging is the process of identifying the grammatical parts of speech of words in a sentence and labeling them accordingly. The parts of speech include nouns, verbs, adjectives, adverbs, pronouns, prepositions, conjunctions, and interjections.

POS tagging is a fundamental task in natural language processing, and it is used in various applications such as text classification, information retrieval, and machine translation. It can help in identifying the role of each word in a sentence, which is essential in determining its meaning and context.

POS tagging involves assigning a specific tag or label to each word in a sentence based on its grammatical function. For example, the word "run" can be tagged as a verb, the word "fast" can be tagged as an adverb, and the word "dog" can be tagged as a noun.

There are different approaches to POS tagging, including rule-based methods, statistical methods, and machine learning-based methods. Rule-based methods use predefined rules based on linguistic knowledge to assign tags to words, while statistical and machine learning-based methods learn from annotated data to predict the most likely tag for a given word based on its context.

Overall, POS tagging is an important step in many natural language processing tasks, and its accuracy can significantly impact the performance of downstream applications.

1. Explain Chunking or shallow parsing

A8. Chunking, also known as shallow parsing, is a natural language processing technique that involves identifying and extracting phrases or meaningful chunks from a sentence based on their syntactic structure. Unlike full parsing, which involves analyzing the entire sentence and building a complete parse tree, chunking focuses on identifying smaller units of meaning, such as noun phrases or verb phrases.

Chunking is typically performed after part-of-speech (POS) tagging, as it relies on the POS tags to identify the boundaries of the chunks. It involves applying a set of rules or patterns to the sequence of POS tags to identify the chunks.

For example, consider the following sentence: "The cat sat on the mat". A simple chunking pattern could be to identify noun phrases consisting of an optional determiner (e.g., "the") followed by one or more adjectives (e.g., "cat") and a noun (e.g., "mat"). Using this pattern, we can identify the noun phrase "the cat" and "the mat" in the sentence.

Chunking is useful in natural language processing tasks such as information extraction, named entity recognition, and text classification. By identifying and extracting meaningful chunks from a sentence, we can obtain more structured and semantically meaningful representations of the text, which can be used to extract useful information and insights from unstructured text data.

1. Explain Noun Phrase (NP) chunking

A9. Noun phrase (NP) chunking, also known as noun chunking, is a specific type of chunking that focuses on identifying and extracting noun phrases from a sentence. Noun phrases are grammatical structures that consist of a noun and any accompanying modifiers or determiners.

NP chunking involves applying a set of rules or patterns to the sequence of POS tags in a sentence to identify the boundaries of noun phrases. The process typically starts with POS tagging to label each word in the sentence with its part of speech, and then applies a set of rules or patterns to identify sequences of words that constitute noun phrases.

For example, consider the following sentence: "The big black dog chased the cat". Using NP chunking, we can identify the noun phrases "the big black dog" and "the cat". The rules or patterns used to identify noun phrases can be based on linguistic knowledge or learned from annotated data.

NP chunking is useful in various natural language processing tasks, such as information extraction, sentiment analysis, and machine translation. By identifying and extracting noun phrases from text, we can obtain more meaningful representations of the text that capture its semantic structure and content, which can be used to perform various analyses and predictions.

1. Explain Named Entity Recognition

A10. Named Entity Recognition (NER) is a natural language processing technique that involves identifying and extracting named entities from text, such as people, organizations, locations, and other entities that have specific names. NER involves identifying the boundaries of named entities in a sentence and classifying them into pre-defined categories.

NER typically involves a two-step process:

1. The first step is to identify the entities in the text. This can be done using rule-based approaches or machine learning techniques such as conditional random fields or recurrent neural networks.
2. The second step is to classify the entities into pre-defined categories such as person, organization, location, or other types of entities. This can be done using a pre-defined set of rules or by training a machine learning model on annotated data.

For example, consider the following sentence: "Bill Gates is the founder of Microsoft Corporation, which is located in Redmond, Washington." Using NER, we can identify the named entities "Bill Gates", "Microsoft Corporation", "Redmond", and "Washington", and classify them into the categories "person", "organization", "location", and "location", respectively.

NER is useful in various natural language processing applications such as information extraction, question answering, and text classification. By identifying and extracting named entities from text, we can obtain more structured and semantically meaningful representations of the text, which can be used to extract useful information and insights from unstructured text data.