**NLP ASSIGNMENT\_2**

**1.What are Corpora?**

In NLP, a corpus (plural: corpora) is a large and structured collection of texts, used to build and evaluate language models. It represents a sample of the language usage and can be used to extract various linguistic features, such as vocabulary, grammar, and style. Corpora can include written texts, spoken transcripts, or multimedia content and can vary in size, genre, and language. Common examples of corpora are the Brown Corpus, the Reuters Corpus, and the Gutenberg Corpus. The use of corpora enables NLP models to capture the patterns and regularities of language and to generalize to new examples.

**2. What are Tokens?**

In NLP, a token refers to a sequence of characters that represents a single semantic unit in a text. Tokens are the basic building blocks for processing text, and their definition depends on the task and the level of granularity required.

For example, in tokenization, tokens are often defined as individual words or punctuation marks. The process of tokenization involves splitting a text into these smaller units, usually by whitespace and punctuation characters. The resulting list of tokens is the input to many NLP tasks, such as part-of-speech tagging, named entity recognition, and sentiment analysis.

In some cases, tokens can also be defined as subwords or character n-grams, which can help NLP models to handle out-of-vocabulary words and morphological variability. These subword tokens can be obtained through a process called word segmentation or by using a technique like byte-pair encoding.

**3. What are Unigrams, Bigrams, Trigrams?**

Unigrams, bigrams, and trigrams are terms used in NLP to describe different units of text.

A unigram is a single word. For example, in the sentence "the cat sat on the mat", each word is a unigram.

A bigram is a pair of consecutive words in a sentence. For example, the bigrams in the sentence "the cat sat on the mat" are: "the cat", "cat sat", "sat on", "on the", and "the mat".

A trigram is a sequence of three consecutive words in a sentence. For example, the trigrams in the sentence "the cat sat on the mat" are: "the cat sat", "cat sat on", "sat on the", and "on the mat".

These units of text are often used as features in NLP models, for example, to model the probability of a word or sequence of words in a language model or for text classification tasks. The choice of unigram, bigram, or trigram depends on the specific task and the desired level of complexity of the model.

**4. How to generate n-grams from text?**

Generating n-grams from text is a common preprocessing step in NLP tasks. The purpose of n-grams is to capture local word dependencies, which can be useful for various NLP tasks such as language modeling, text classification, and text generation.

An n-gram is a sequence of n consecutive words or characters in a sentence. For example, a bigram is a sequence of two words and a trigram is a sequence of three words. The choice of the value of n depends on the specific NLP task and the desired level of complexity of the model.

**5. Explain Lemmatization**

Lemmatization is a process of reducing words to their base or root form. It's an important preprocessing step in NLP tasks that aims to standardize words to their core meaning, and can help improve the performance of NLP models by reducing the dimensionality of the data.

For example, the words "running," "ran," and "runs" are all forms of the same base verb "run." In lemmatization, these words would be reduced to their base form "run," which is the lemma.

Lemmatization is similar to stemming, which is another preprocessing step that reduces words to their root form, but lemmatization is typically more sophisticated and produces more accurate results. Unlike stemming, which uses simple heuristics to remove suffixes from words, lemmatization uses morphological analysis to determine the correct lemma based on the context of the word in a sentence.

**6. Explain Stemming**

Stemming is a process of reducing words to their root or base form. It's a common preprocessing step in NLP tasks that aims to standardize words to their core meaning, and can help improve the performance of NLP models by reducing the dimensionality of the data.

For example, the words "running," "ran," and "runs" are all forms of the same verb "run." In stemming, these words would be reduced to their root form, such as "run," which is the stem.

Stemming uses simple heuristics, such as removing suffixes, to produce the stem. Unlike lemmatization, which uses morphological analysis to determine the correct lemma based on the context of the word in a sentence, stemming does not consider the context of the word and may produce inaccurate results.

**7. Explain Part-of-speech (POS) tagging**

Part-of-speech (POS) tagging is the process of marking each word in a text with its corresponding grammatical category, such as noun, verb, adjective, adverb, etc. This information can be used to better understand the structure and meaning of a sentence and is a crucial step in many NLP tasks, such as Named Entity Recognition, Parsing, and Text Classification.

POS tagging involves assigning a unique tag to each word in a sentence based on its definition and its context within the sentence. For example, the word "run" could be tagged as a verb in the sentence "She runs every morning" but as a noun in the sentence "The company had a great run last quarter."

There are several standard sets of POS tags, such as the Penn Treebank tags and the Universal Dependencies tags. These tags provide a consistent and well-defined set of categories that can be used to compare and analyze text across different languages and domains.

**8. Explain Chunking or shallow parsing**

Chunking, also known as shallow parsing, is a process in NLP that involves dividing a sentence into smaller parts, called chunks, that have a specific meaning and grammatical structure. Chunks are typically built around phrases, such as noun phrases, verb phrases, adjective phrases, etc.

The goal of chunking is to extract meaningful information from a sentence and represent it in a more structured format, while retaining only the essential elements of the sentence. Chunks are usually represented using a tag that indicates the type of chunk, such as NP (noun phrase), VP (verb phrase), ADJP (adjective phrase), etc.

**9. Explain Noun Phrase (NP) chunking**

Noun Phrase (NP) chunking is a task in NLP that involves identifying and extracting noun phrases from a sentence. A noun phrase is a sequence of words that function as a noun in a sentence and typically include a noun and any words modifying the noun, such as articles, adjectives, and prepositions.

For example, in the sentence "The black cat chased the white mouse.", the noun phrases are "The black cat" and "the white mouse".

NP chunking is a subtask of shallow parsing, also known as chunking, which involves dividing a sentence into smaller parts, called chunks, that have a specific meaning and grammatical structure.

**10. Explain Named Entity Recognition**

Named Entity Recognition (NER) is a task in NLP that involves identifying and classifying named entities in text, such as person names, organizations, locations, dates, times, etc. into predefined categories such as "person", "organization", "location", etc.

The goal of NER is to extract structured information from unstructured text, which can be useful for various NLP applications, such as information retrieval, question answering, text summarization, and more.

NER can be performed using rule-based methods, where predefined rules are used to identify named entities, or machine learning-based methods, where a model is trained on annotated data to identify named entities.