**What is Syntactic analysis?**

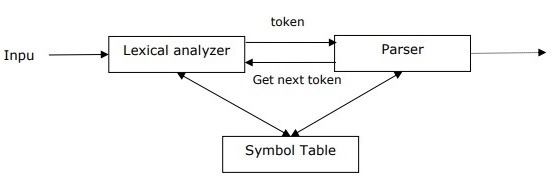
Syntactic analysis or parsing or syntax analysis checks the text for meaningfulness comparing to the rules of formal grammar.

Ex: the sentence like “hot ice-cream” would be rejected by semantic analyzer.

POS tagging helps analyze the sentence structure and relationships between words. This is achieved using Markov models and other algorithms that predict the most likely sequence of POS tags based on the given text.

**Technical Explanation:**

syntactic analysis or parsing may be defined as the process of analyzing the strings of symbols in natural language conforming to the rules of formal grammar.



The main roles of the parse include −

* To report any syntax error.
* To recover from commonly occurring error so that the processing of the remainder of program can be continued.
* To create parse tree.
* To create symbol table.
* To produce intermediate representations (IR).

**Types of Parsing**

Derivation divides parsing into the followings two types −

* Top-down Parsing

In this kind of parsing, the parser starts constructing the parse tree from the start symbol and then tries to transform the start symbol to the input. The most common form of topdown parsing uses recursive procedure to process the input. The main disadvantage of recursive descent parsing is backtracking.

* Bottom-up Parsing

In this kind of parsing, the parser starts with the input symbol and tries to construct the parser tree up to the start symbol.

**Types of Derivation**

In this section, we will learn about the two types of derivations, which can be used to decide which non-terminal to be replaced with production rule −

1. Left-most Derivation

In the left-most derivation, the sentential form of an input is scanned and replaced from the left to the right. The sentential form in this case is called the left-sentential form.

1. Right-most Derivation

In the left-most derivation, the sentential form of an input is scanned and replaced from right to left. The sentential form in this case is called the right-sentential form.

## Grammar

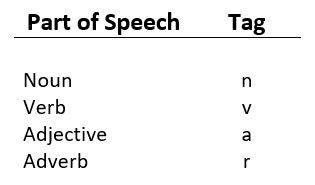
Grammar is very essential and important to describe the syntactic structure of well-formed programs.

**Dependency Grammar**

* In DG, the linguistic units, i.e., words are connected to each other by directed links.
* The verb becomes the center of the clause structure.
* Every other syntactic units are connected to the verb in terms of directed link. These syntactic units are called ***dependencies***.

# **POS(Parts-Of-Speech) Tagging in NLP**

One of the core tasks in **Natural Language Processing (NLP)** is **Parts of Speech (PoS) tagging**, which is giving each word in a text a grammatical category, such as nouns, verbs, adjectives, and adverbs.



**Examples of POS Tagging**

1. Consider the sentence: “The quick brown fox jumps over the lazy dog.”

**After performing POS Tagging:**

* “The” is tagged as determiner (DT)
* “quick” is tagged as adjective (JJ)
* “brown” is tagged as adjective (JJ)
* “fox” is tagged as noun (NN)
* “jumps” is tagged as verb (VBZ)
* “over” is tagged as preposition (IN)
* “the” is tagged as determiner (DT)
* “lazy” is tagged as adjective (JJ)
* “dog” is tagged as noun (NN)

2) In the sentence “She quickly reads a book,” POS tagging assigns tags like “PRON” (pronoun) to “She,” “ADV” (adverb) to “quickly,” “VERB” to “reads,” “DET” (determiner) to “a,” and “NOUN” to “book.” This tagging clarifies the roles and grammatical functions of words, aiding syntactic and semantic analysis in natural language processing tasks.

**Packages:**

Nltk:

'punkt'  
'averaged\_perceptron\_tagger'

from nltk.tokenize import word\_tokenize

from nltk import pos\_tag

Spacy:

pip install spacy  
!python -m spacy download en\_core\_web\_sm

### **Advantages of POS Tagging**

There are several advantages of Parts-Of-Speech (POS) Tagging including:

* **Text Simplification:**Breaking complex sentences down into their constituent parts makes the material easier to understand and easier to simplify.
* **Information Retrieval:**Information retrieval systems are enhanced by point-of-sale (POS) tagging, which allows for more precise indexing and search based on grammatical categories.
* **Named Entity Recognition:**POS tagging helps to identify entities such as names, locations, and organizations inside text and is a precondition for named entity identification.
* **Syntactic Parsing:**It facilitates [syntactic parsing](https://www.geeksforgeeks.org/constituency-parsing-and-dependency-parsing/), which helps with phrase structure analysis and word link identification.

### **Disadvantages of POS Tagging**

some common disadvantages in part-of-speech (POS) tagging include:

* **Ambiguity:**The inherent ambiguity of language makes POS tagging difficult since words can signify different things depending on the context, which can result in misunderstandings.
* **Idiomatic Expressions:**Slang, colloquialisms, and idiomatic phrases can be problematic for POS tagging systems since they don’t always follow formal grammar standards.
* **Out-of-Vocabulary Words:**Out-of-vocabulary words (words not included in the training corpus) can be difficult to handle since the model might have trouble assigning the correct POS tags.
* **Domain Dependence:**For best results, POS tagging models trained on a single domain should have a lot of domain-specific training data because they might not generalize well to other domains.

**EXPLORE Markov models and Viterbi algorithms**

**Syntax Parsing using Context Free Grammars**

A context-free grammar is a means of describing which strings of characters are contained within a particular language. It consists of a set of rules and a start nonterminal symbol. Each rule specifies one way of replacing a nonterminal symbol in the current string with a string of terminal and nonterminal symbols. When the resulting string consists only of terminal symbols, we stop. We say that any such resulting string has been generated by the grammar.

Context-free grammars are used to understand both the syntax and the semantics of many very useful languages, such as mathematical expressions, Java, and English. The syntax of a language indicates which strings of tokens are valid sentences in that language. The semantics of a language involves the meaning associated with strings. In order for a compiler or natural-language recognizers to determine what a string means, it must parse the string. This involves deriving the string from the grammar and, in doing so, determining which parts of the string are noun phrases, verb phrases, expressions, and terms.

Some context-free grammars have a property called look ahead one. Strings from such grammars can be parsed in linear time by what I consider to be one of the most amazing and magical recursive algorithms.