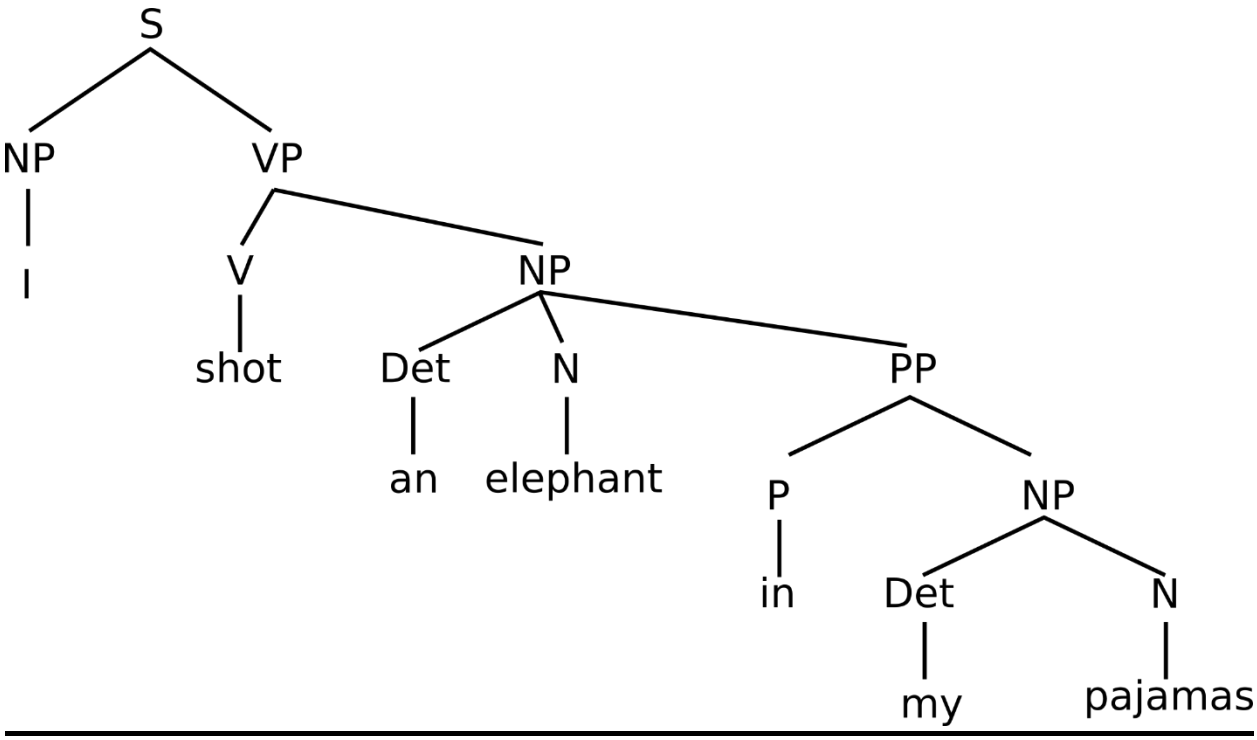
PARSING

The context-free grammars used for parsing so far find the constituent-based structure of the input sentence. This is useful for detecting the phenomenon of "omnipresent ambiguity".

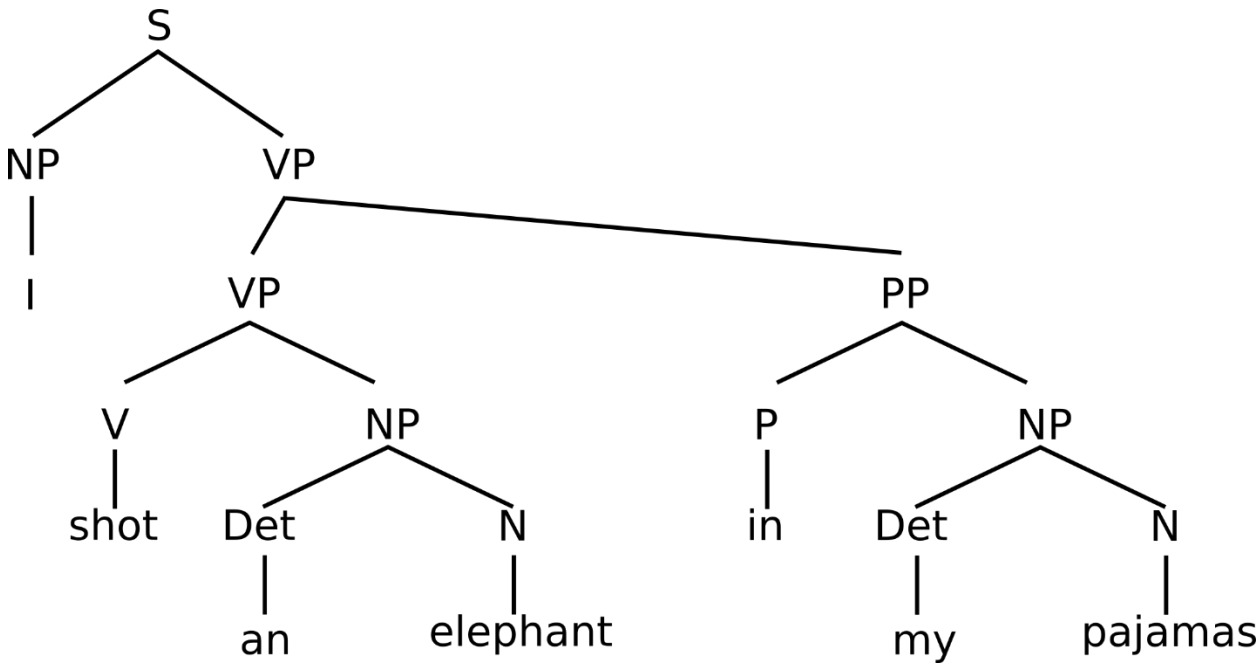
Example: "*While hunting in Africa, I shot an elephant in my pajamas. How an elephant got into my pajamas I'll never know*".

The ambiguity is reflected by the two resulting parse trees. This is structural ambiguity that reflects the existing semantic ambiguity.

Parse tree no.1



Parse tree no. 2



Parse tree no.1 corresponds to the correct sense, which can only be disclosed within the context. Here the elephant is inside the pajamas.

Dependencies and Dependency Grammars

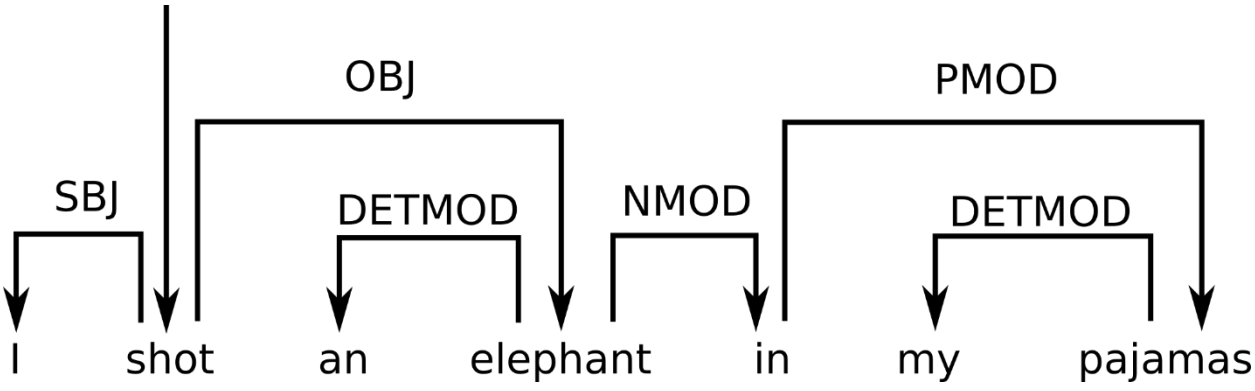
- **Context-free grammars (phrase-structure grammars) illustrate how words and sequences of words *combine* to form constituents.**

- **The dependency grammar approach focuses instead on how words *relate* to other words.**

- ***Dependency* is a binary asymmetric relation that holds between a head and its dependents.**

Example

Here is the dependency structure (parse) of the same sentence:

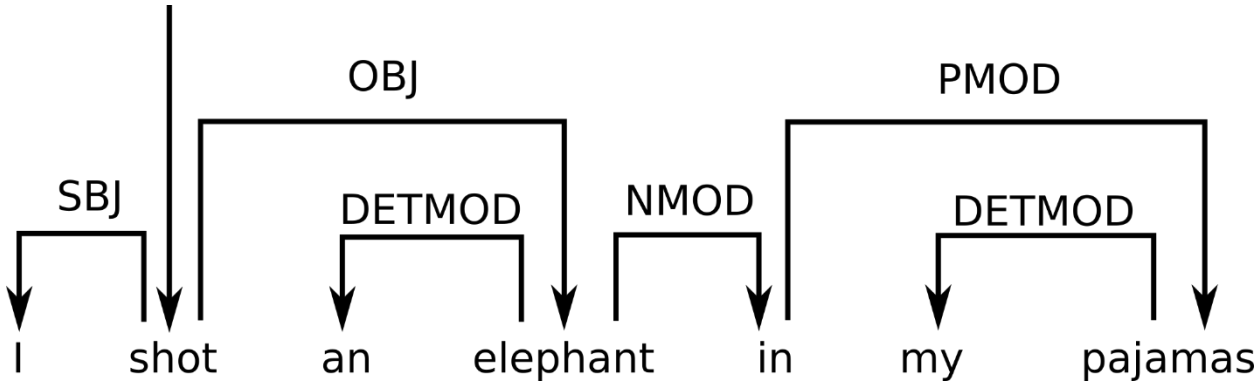


- **The head of a sentence is usually taken to be the tensed verb, and every other word is either dependent on the sentence head or connects to it through a path of dependencies.**

- **Dependency grammar (DG) is a class of modern grammatical theories that are all based on the dependency relation (as opposed to the relation of phrase structure) and that can be traced back primarily to the work of Lucien Tesnière. Dependency is the notion that linguistic units, e.g. words, are connected to each other by directed links. The tensed verb is taken to be the structural center of clause structure. All other syntactic units (words) are either directly or indirectly connected to the verb in terms of the directed links, which are called *dependencies*. (reproduced here from Wikipedia)**

- **A dependency representation of a sentence is a labeled directed graph, where the nodes are the lexical items and the labeled arcs represent dependency relations from heads to dependents.**

  Let's look again at the dependency structure of the same sentence (below):


OBJ
PMOD
SBJ
DETMOD
NMOD
DETMOD
I
shot
an
elephant
in
my
pajamas

Dependency structure:

- **The arcs are labeled with the *grammatical relations* between a dependent and its head. For instance:**
  - **the verb "shot" is the head of the entire sentence;**
  - **the pronoun "I" is in the grammatical relation SBJ (is the subject of) with the verb "shot";**
  - **"in" is in the grammatical relation NMOD (noun modifier) with "elephant".**

  These grammatical functions (given by relations) represent the *dependency type*. (Functiile gramaticale sunt exprimate prin tipurile dependentelor).

- **A dependency graph is projective (*projective analysis*) if, when all the words are written in linear order, the edges can be drawn above the words without crossing. This is equivalent to saying that a word and all its descendants (dependents and dependents of its dependents, etc.) form a contiguous sequence of words within the sentence.**

- **Dependency parsing is non-projective when the edges are allowed to cross.**

- **The dependency grammar theory has initially considered only the dependency syntactic analysis of projective type.**

Main ideas:

- **Dependency grammars are not based on the concept of constituent, but on the direct relations existing among words, viewed as dependency relations.**

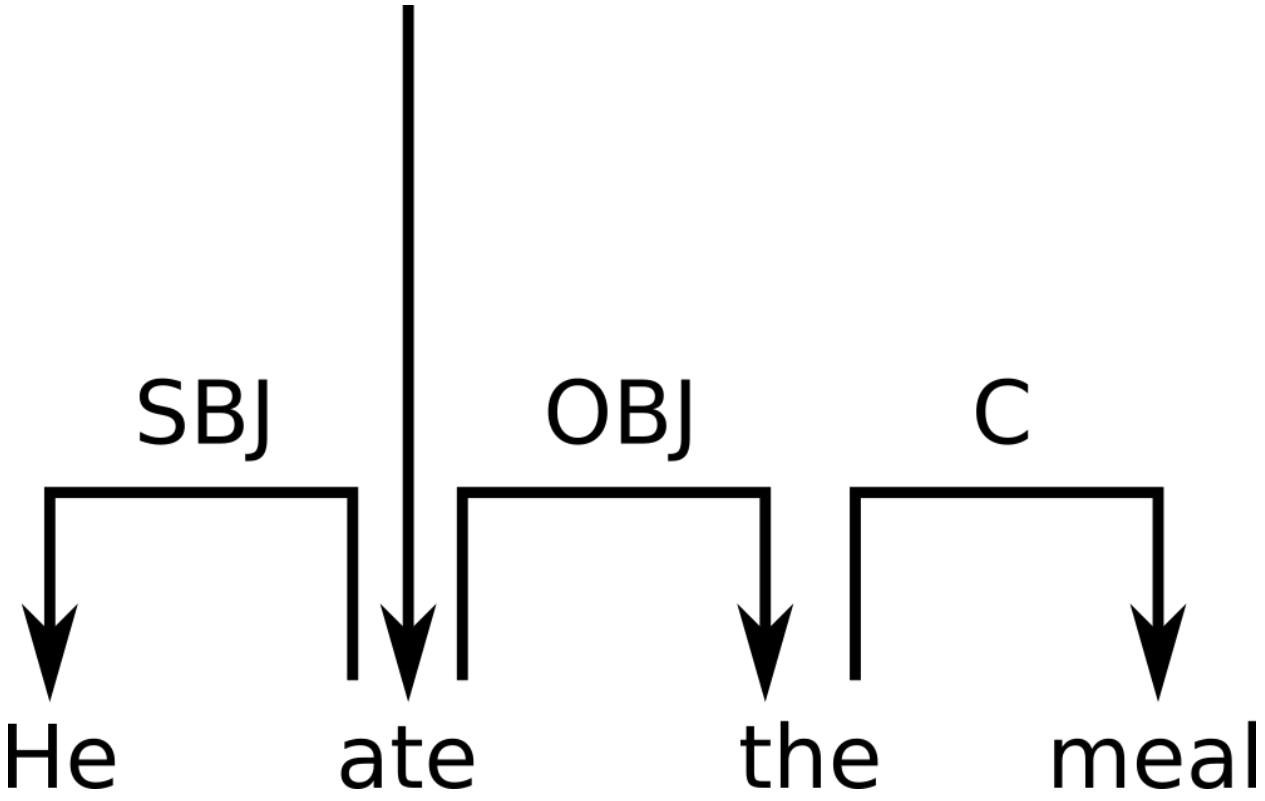
- **At the heart of dependency grammars we have the relation existing between the head word and the dependent word.**

- **In this framework, the syntactic analysis of a sentence (*dependency parse*) consists of describing *all* dependency relations existing among *all* words of a sentence.**

- **The dependency graph that represents the dependency parse of a sentence can be projective or non-projective. Initially, only the projective type of analysis was considered. However, in languages with more flexible word order than English, non-projective dependencies are more frequent.**

- **A major characteristic of this type of syntactic analysis is that it views syntactic phrases (noun phrase, verb phrase etc.) as representing byproducts of the existing dependencies! (byproduct = produs secundar)**

Comparison:



COMPARISON:

1. Syntactic analysis based on constituents

- **the syntactic phrase "the meal" represents a realization of the phrase-structure rule**

$$NP \rightarrow Det\ N$$

_i.e._ it is a noun phrase organized around the noun "meal"– having the head role within the syntactic group

2. Syntactic analysis of dependency type

- **"the" represents the head and "meal" is the dependent (because "the" brings in, inserts "meal" into the sentence).**

REMARKS:

- **Each word depends on the word "linking" it to the rest of the sentence, implicitly explaining why it is used.**

- **Any word of the sentence must depend on a single other word – the head – with the exception of the sentence's predicate, that depends on no other word.**

- **Several words may depend on the same head.**

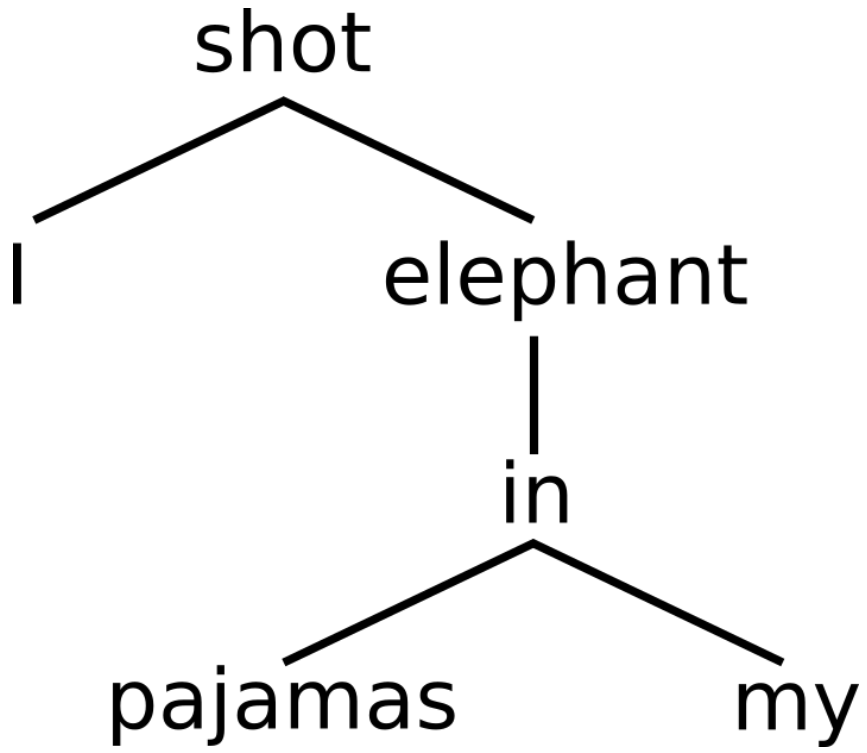
- **The grammatical function (giving the dependency type) is always the function of the dependent word in its relation to the head. (That is why the arrow points towards the dependent word – see the dependency graph).**

- **This formalism presents the advantage of being able to treat discontinuous constituents, as well as the variation of word order. (English presents a quite rigid word order. However, many languages, such as Russian, Finnish, have variable word order). Non-projective dependency parsing especially, is useful for languages displaying variable word order.**

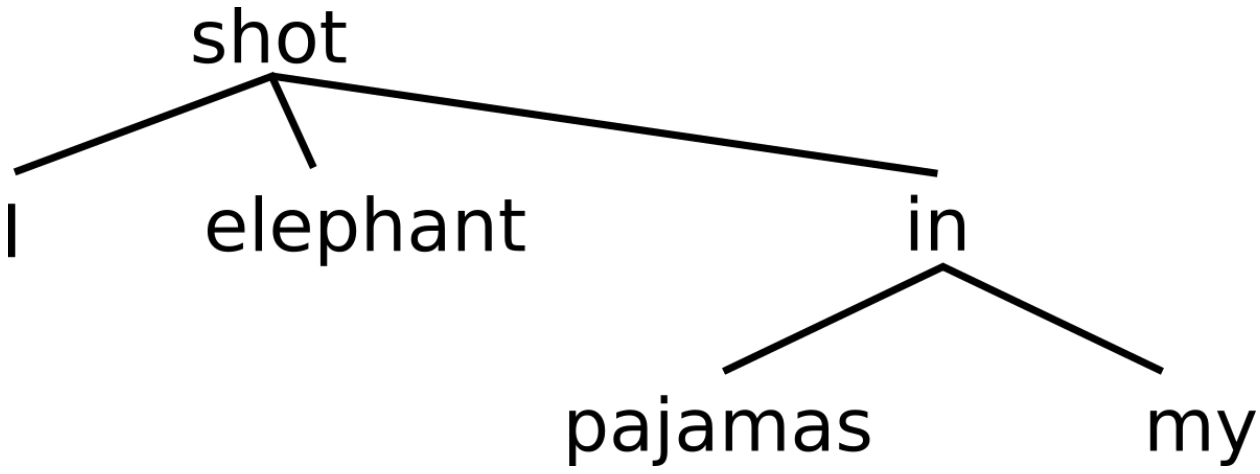
Note that: A great variety of dependency relations may exist among the words of a sentence. The role of dependency grammars is that of specifying the restrictions that the dependency relations must meet in order for the structure they define to be correct from the linguistic point of view.

The dependency structure of a sentence can also be represented as a tree, one in which the dependents present themselves as sons of the corresponding head words.

The ambiguity displayed by our previous example leads to two different dependency structures having the following corresponding parse trees:



Here the elephant is inside the pajamas. Note that “in” depends on “elephant”. The analysis is a <u>projective</u> one.



Here I was dressed in my pajamas when I shot the elephant. Note that "in" depends on "shot". The dependency type is different. The analysis is again a projective one.

CRITERIA for deciding what is the head H and what is the dependent D in a construction C (most important ones):

1. **H determines the distribution class of C; or alternatively, the external syntactic properties if C are due to H.**

2. **H determines the semantic type of C.**

3. **H is obligatory while D may be optional.**

4. **H selects D and determines whether it is obligatory or optional.**

5. **The morphological form of D is determined by H (e.g., agreement or case government).**

Main TOOL for dependency parsing:

STANFORD DEPENDENCY PARSER

Stanford dependencies are triplets consisting of:

- **name of the relation**
- **governor**
- **dependent**

See dependency parsing with NLTK at the lab !