

DEPENDENCY PARSING

(23/10/23)

① What is it and how it works

⇒ Helps computers to understand grammatical structure of sentence and relationship between words.

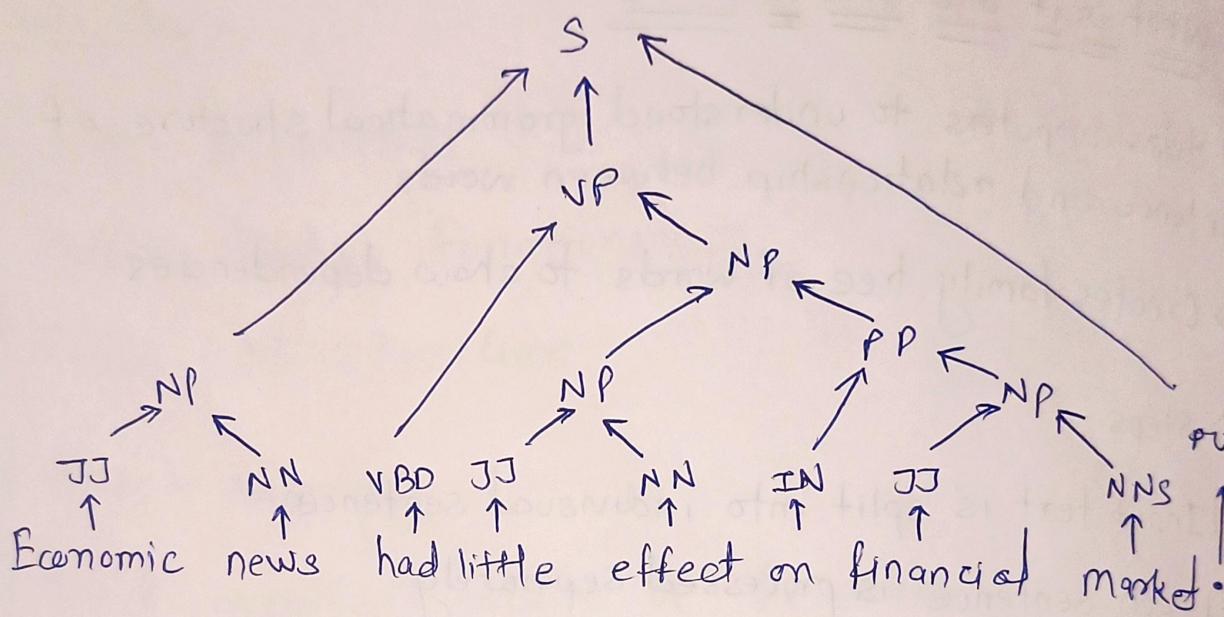
⇒ Creates family tree of words to show dependencies

⇒ Steps

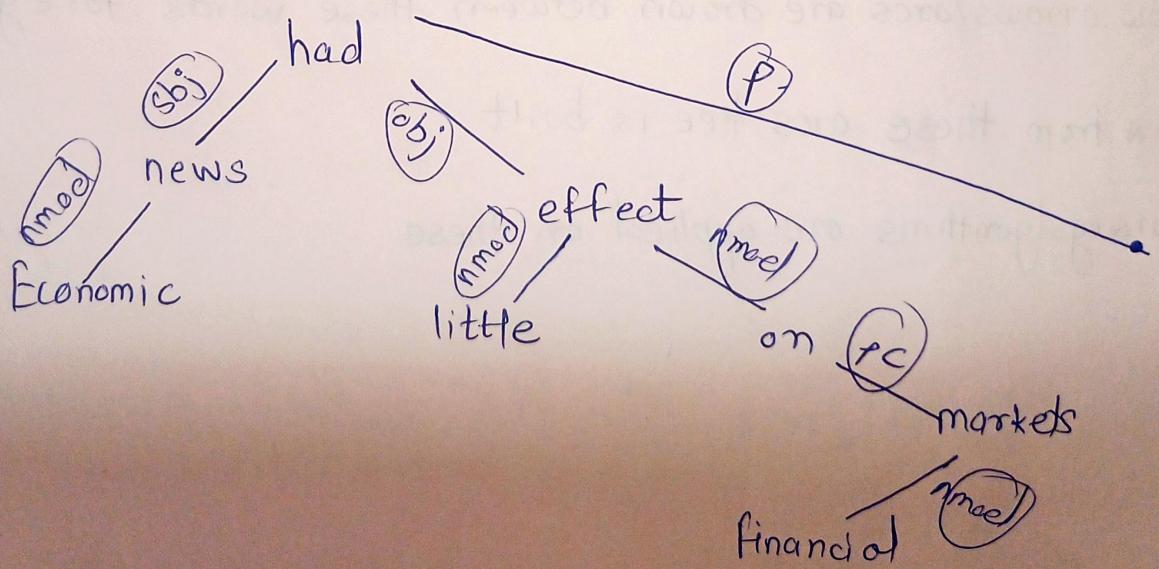
- 1] Input text is split into individual sentences
- 2] Each sentence is processed separately.
- 3] Tokenization
- 4] Now arrows/arcs are drawn between these words. for e.g
- 5] Now from these arcs tree is built.
- 6] parsing algorithms are applied on these

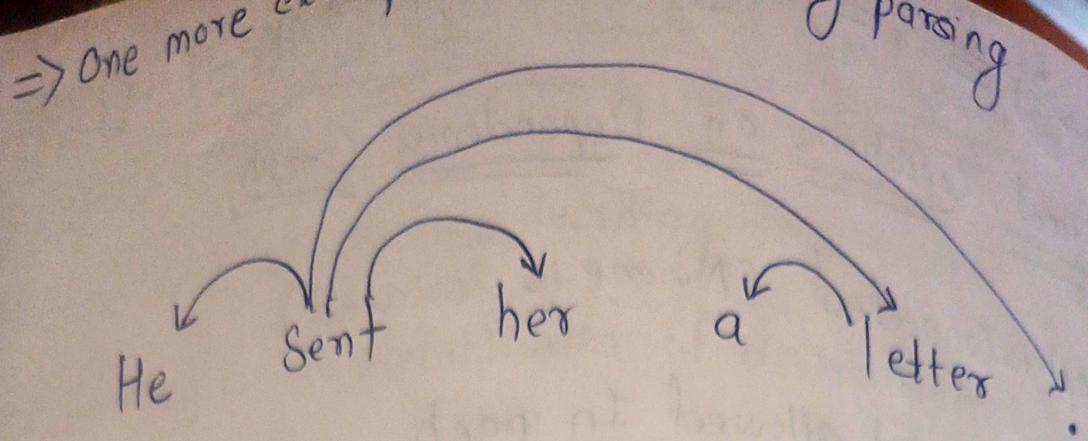
⇒ We will mainly find relationship between 2 words
⇒ lets look difference between dependency structure and phrase structure

Following is phrase structure



Following dependency structure





Some clear cases where we can easily find the dependencies

⇒ Exocentric & Endocentric constructions

Endocentric constructions are those construction in which there is one word which carries whole meaning and other parts serve to modify or describe the Head.

Exocentric construction

meaning of whole phrase is different from the meaning of individual parts.

① Dependency Graph

⇒ 1] Directed graph

2] Set of Terminals Nodes (labelled as wordname & pos tag)

3] Set of Arcs (labelled as dependency type)

⇒ Arc(w_i, d, w_j)

$$w_i \xrightarrow{d} w_j \iff (w_i, d, w_j) \in A$$

$$i \rightarrow j = (i, j) \in A$$

$i \rightarrow^* j \equiv i = j \vee \exists k : i \rightarrow k, k \rightarrow^* j$

① formal Conditions on Dependency Graph

- 1] Graph is connected (No one is isolated)
- 2] Cycles are not allowed In graph
- 3] obeys single head constraint
for any node; there must be atmost one head
- 4] There should be no (non) projectivity / crossing of dependency

In profession of words

- 1] Connectedness \Rightarrow Syntactic structure is complete
- 2] Acyclicity \Rightarrow Syntactic structure is hierarchical
- 3] Single-Head \Rightarrow Every word has atmost one syntactic head
- 4] Projectivity \Rightarrow No crossing of dependency

