CS224n: NLP with Deep Learning **Winter 2019**

Lecture 5

Course Coordinator: Prof. Chris Manning *Scribes: Akash Gupta*

Two views of linguistic structures -

- Phrase Structure/Phrase Structure Grammars
- Dependency Structures

Phrase Structure Grammars - Idea is that sentences are built out of units (words) that progressively nest. E.g. - *the (Det) cat (Noun), a (Det) dog (Noun)* – PSG rule: NP− >Det N.

− >Can make big sentences using this grammar. E.g. *the large cat in a crate on the table by the door* – PSG rule: NP− >Det (Adj) N PP; PP− >Prep NP

Dependency Structure Grammars - Shows which words depend on (modify or are arguments of) which other words.

Q) Why is it important to understand sentence structure?
A) Some ambiguities in natural languages:

- Prepositional phrase attachment ambiguity. For E.g. {*Scientists count whales*} *from space* **OR** *Scientists count* {*whales from space*}.
- Coordination scope ambiguity. For e.g. {*Shuttle veteran and longtime NASA executive*} *Fred Gregory appointed to board* **OR** {*Shuttle veteran*} *and* {*longtime NASA executive Fred Gregory*} *appointed to board* − > Confused if one or two persons.
- Adjectival Modifier Ambiguity. For e.g. *Students get first hand job experience.*
- Verb Phrase (VP) attachment ambiguity. For e.g. *Mutilated body washes up on Rio beach to be used for Olympics beach volleyball*

− > Dependency syntax tells that syntactic structure consists of relations between lexical items, normally binary asymmetric relations (”arrows”) called dependencies.

How to draw dependency structures? -

1. Draw arrows from head to dependent (Tesniére's method)
2. Add a fake ROOT node so every word is a dependent of precisely 1 other node.

Dependency Parsing - A sentence s parsed by choosing for each word what other word (including ROOT) is it a dependent of.

- Usually some constraints:
  - Only one word is a dependent of ROOT.
  - Don't want cycles, A− >B, B− >A.
- Form a tree of dependencies.
- Final issue of whether arrows can cross (non-projective) or not - Not common in English.

Arc-standard transition based parser - Stack and buffer method.

Evaluation of Dependency parsing - UAS (Unlabeled Attachment Score) - Ignore the labels, LAS (Labeled Attachment Score)

A neural dependency parser - *Chen and Manning 2014* - Solves the problems with traditional parsers like sparsity, incompleteness, expensive computation.