

Dependency

Jean Mark Gawron

Linguistics

San Diego State University

gawron@mail.sdsu.edu

<http://www.rohan.sdsu.edu/~gawron>

Intro

visiting relatives (Chomsky)

1. Different heads

$_1 \text{relatives}_1$ *visiting relatives are annoying.*

your visiting relatives

$_1 \text{visiting}_1$ *visiting relatives is annoying.*

visiting your relatives

2. Visiting relatives can be a nuisance.

3. Different “higher-up” dependencies

attempt wants $_1 \text{visiting}_1$ Never attempt visiting relatives!

hug wants $_1 \text{relatives}_1$ Never hug visiting relatives!

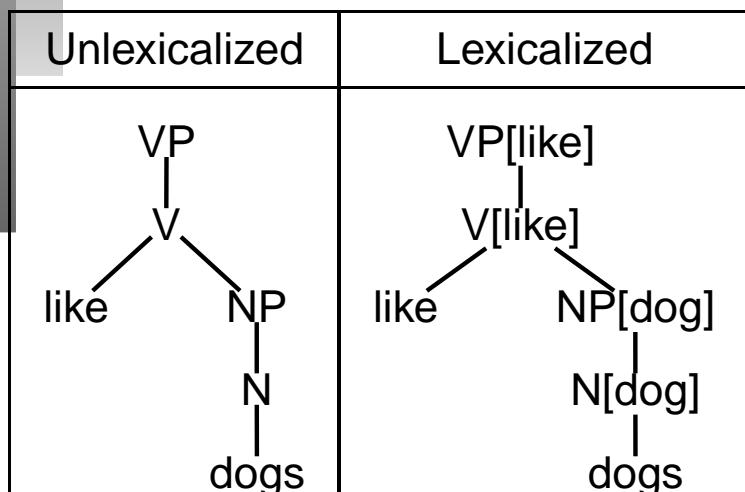
Head words

There is a key word inside constituents which determines a great deal about their distribution:

- What noun phrases refer to: *piano bar*
- Verb agreement:
 - a. Piano **bars** are fun.
 - b. * Piano bars **is** fun.
- I like **eating** beans.
 - * **eat** beans.
 - to eat** beans.
- Finding the head is important in parsing.
- Parsing can be construed as finding the relations between heads.

Lexicalization

Adding information about lexical heads to syntax rules



Schabes et al. (1988) (?,
?)

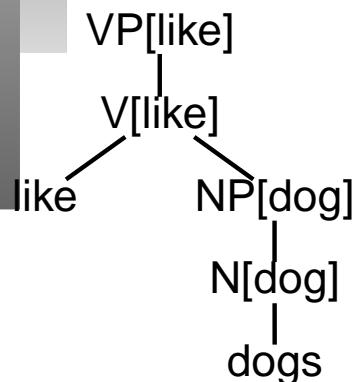
- (1) a. The storm gathered/?convened.
b. I like to solve puzzles/?goats.
c. John *drained/added the skull/sand of/to blood/the bowl.*

Probabilities and Lexicalization

1. Although the idea of lexicalization was explored before statistical methods became widespread in computational linguistics, it seems particularly important in statistical syntactic models
2. Classic probabilistic context-free grammars (PCFGs) are not lexicalized, and this is one reason why they appear to make bad language models.
3. Charniak 1995, Collins 1996, Eisner 1996, Charniak 1997, Collins 1997, Goodman 1997, ?

Bilexicality

Grammars that capture word to word dependencies are called **bilexical**.



If the parameters of the statistical model include bigram dependencies such as:

$$P(VP[like] \mid NP[dog])$$

or

$$P(VP[like] \mid \text{obj}, NP[dog])$$

then the model is bilexical.

Parsing with bilexical models

1. Charniak 1995, Collins 1996, Eisner 1996, Charniak 1997, Collins 1997, Goodman 1997, Collins 1999
2. The work here will focus on Eisner's **dependency grammar** version of a bilexical model (Eisner 2000)
3. I will focus mostly on the question of what a dependency grammar is, how to parse with a dependency grammar, and how Eisner's idea is both different and efficient.

Outline

1. Formalism for dependency grammars.
2. $O(n^5)$ result for “naive” algorithms
3. An $O(n^3)$ algorithm (with several clever tricks)
4. Weights and Polysemy

Formalisms

Dependency Grammar: Simple

?, ?, ?

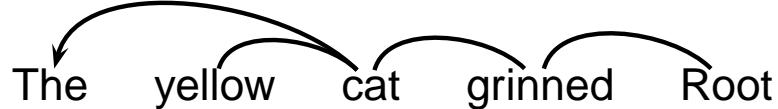
1. V : a set of words (including ROOT)
2. R : a set of dependency relations
3. L : a set of linking triples of the form (w_i, r, w_j) , where w_i and w_j are members of the vocabulary and r is a member of R .

Simplest Dependency Grammar

One member of R (= “modifies”)

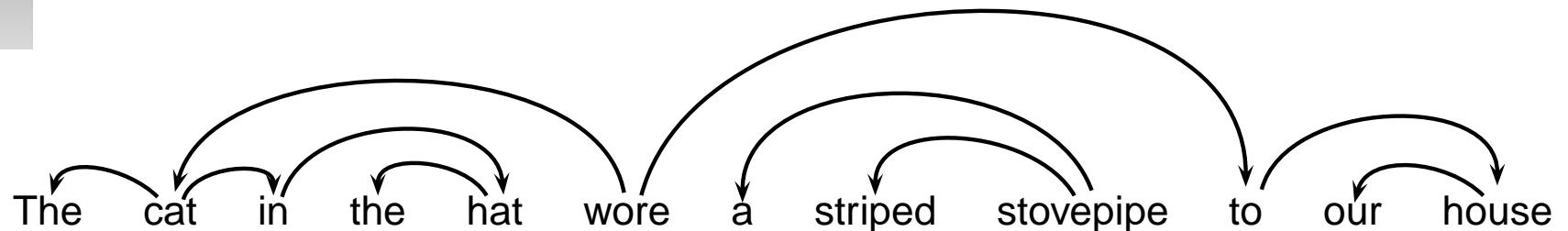
The yellow cat grinned Root

Unlabeled links between words:



Example

The cat in the hat wore a striped stovepipe to our house.



Conditions on dependency structure

1. Every word (except ROOT) finds at least one parent.
2. Every word finds at most one parent.
3. No cycles:

$$a \rightarrow b \rightarrow c \cdots \rightarrow a$$

Dependency Grammar II

?

1. V : a set of words (including ROOT)
2. For each $w \in V$, a pair of Deterministic Finite-State automata, l_w and r_w (left and right automata) accepting some subset of V^* and specifying the possible left and right **dependents** of w .

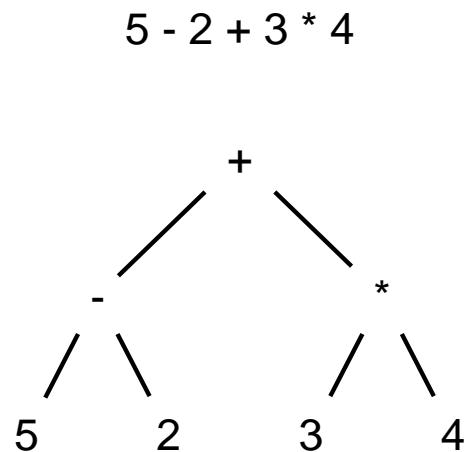
Terminology

- Bounds g, t :
 - g : Max number of states in any automaton
 - t : Max number of lex entries for an ambiguous word [plays a role in polysemy model]
- Dependents
 - Arguments of head word
 - Adjuncts of head word

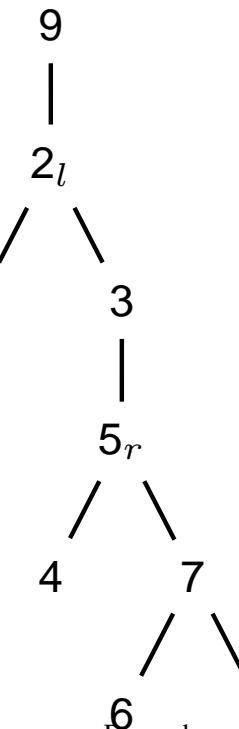
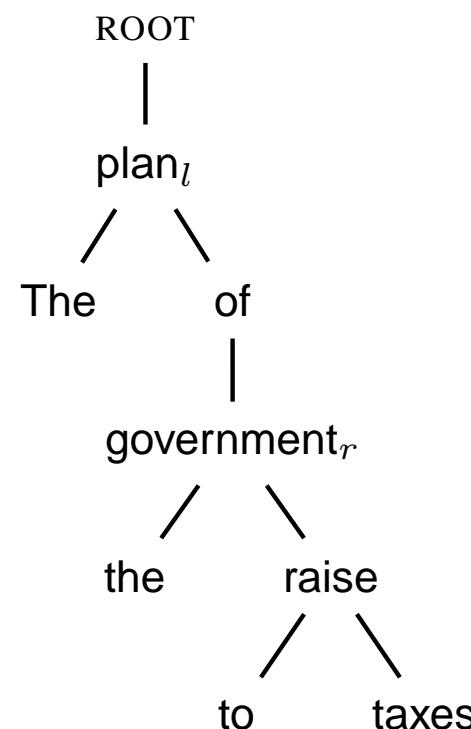
Meaning of grammar

A **dependency tree** is a rooted tree whose nodes are labeled with words from V , and whose root is labeled with the special symbol $\text{ROOT} \in V$. Children are ordered with respect to each other **and with respect to the node**, so there are **left children** and **right children**. [Weird for linguists but not computer scientists.]

Infix order:

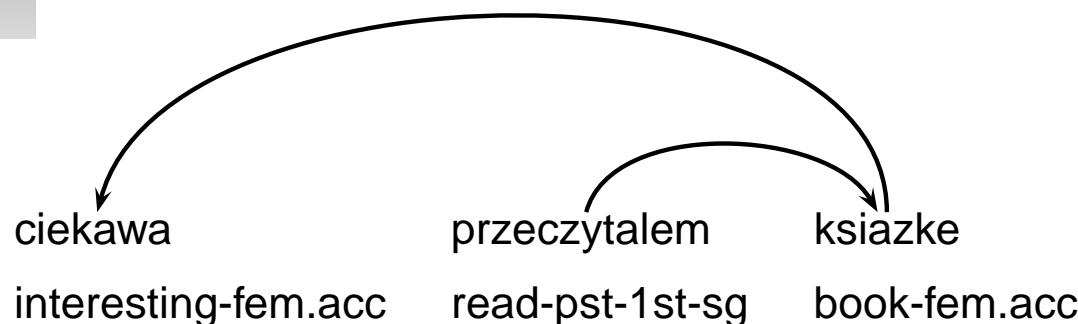


For all w , left children admitted by w_l , right children by w_r



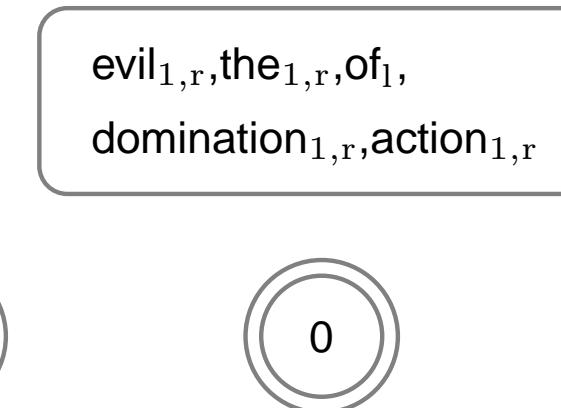
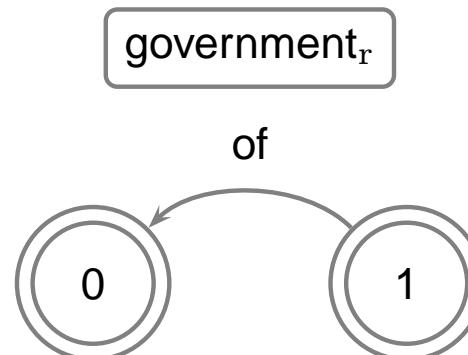
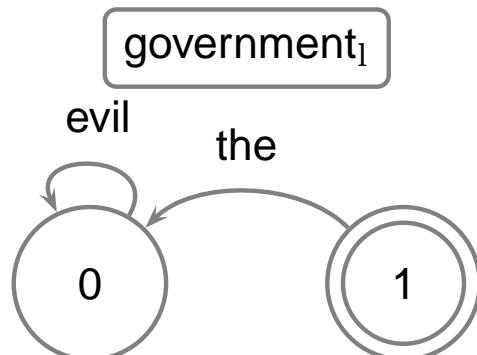
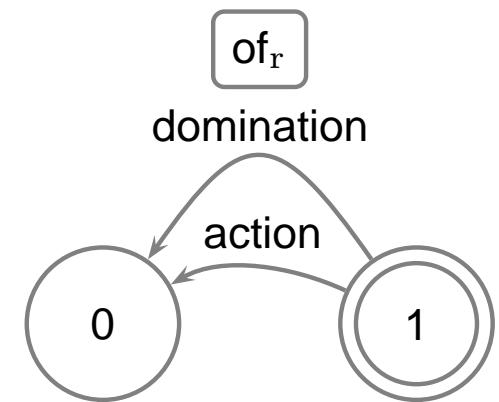
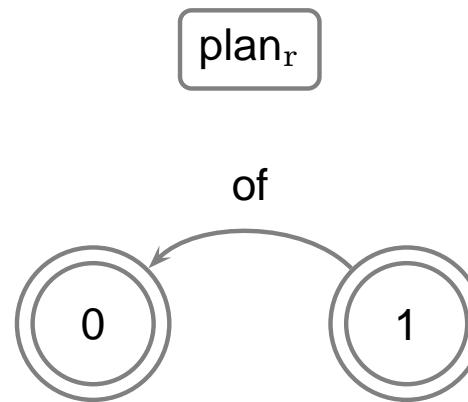
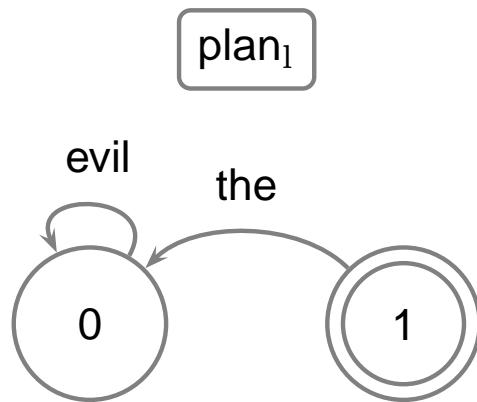
Projectivity

Because a dependency tree is a tree, there can be no crossing links. [free word languages, e.g., Polish, a problem]

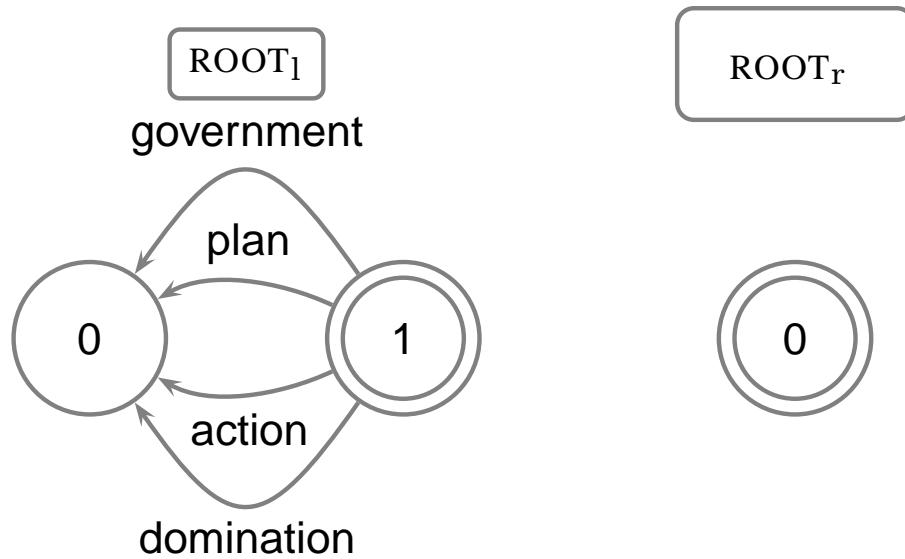


There is no dependency tree capturing this word order and these dependency links.

A grammar



Grammar ROOT



Some issues about dependency grammar

There are a number of phenomena whose treatment under this view of grammar is unclear or controversial:

1. Conjunction: The lobster and the kettle walked. [How to preserve blexical selection?]
2. Auxiliaries: If subject selection by the main verb is to be preserved, these must be treated as obligatory modifiers of non finite verbs. [Why can't auxiliaries be heads?] So the notion *head* doesn't line up with locus of finiteness, agreement.
3. How do we get subject-aux inversion? Is this compatible with the answer to the previous question? How do we distinguish inverted clauses from uninverted clauses?

In general, the treatment of grammatical phenomena that aren't projections of lexical properties (**constructions**) is problematic. [Categorial grammar: a richer version of dependency grammar]

Example

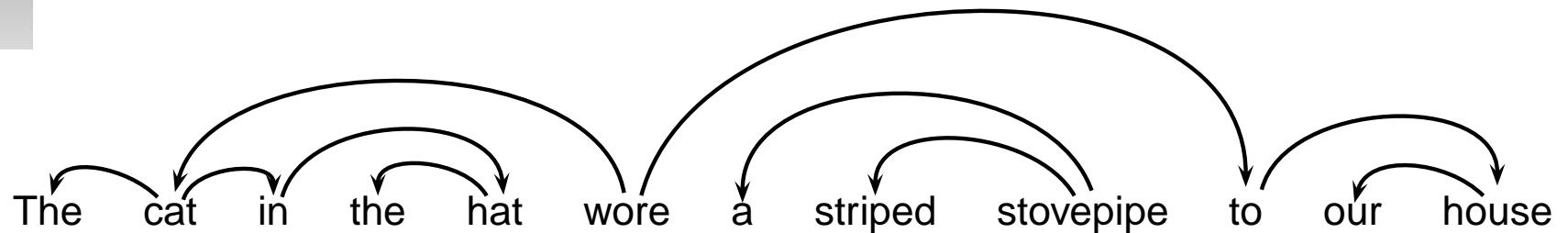
The yellow cat grinned.

The yellow cat grinned



Example

The cat in the hat wore a striped stovepipe to our house.



(Decomposing an) Analysis tree

