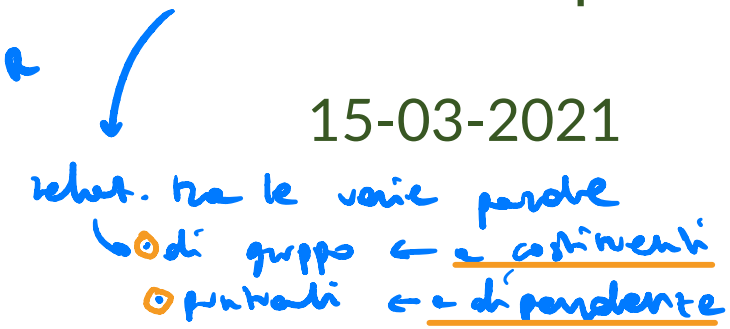
Tecnologie del Linguaggio Naturale

Parte Prima

Lezione n. 04-1

Sintassi: la competence

15-03-2021



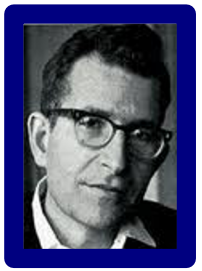
Prologo

This is the cat that caught the rat that stole the cheese.

The man who wrote the book that you told me about

Aspects of the theory of syntax (1957)

N. Chomsky



Prologo

- English Center-Embedding

ricorsione "centrale"

A man that a woman loves

A man that a woman that a child knows loves

A man that a woman that a child that a bird saw knows loves

A man that a woman that a child that a bird that I heard saw knows loves

Prologo

E venne il macellaio,
che uccise il toro,
che bevve l'acqua,
che spense il fuoco,
che bruciò il bastone,
che picchiò il cane,
che morse il gatto,
che si mangiò il topo
che al mercato mio padre comprò.

Alla fiera dell'est (1976)
A. Branduardi

Prologo

Elena e Maria odiano il latte e il caffè

rispettivamente

↑ importante

Outline

1. Sintassi e grammatiche generative
2. La gerarchia di Chomsky e il linguaggio naturale
3. Mildly-Context Sensitive Languages, Tree Adjoining Grammars, Combinatory Categorial Grammars

Outline

- Sintassi e grammatiche generative
- La gerarchia di Chomsky e il linguaggio naturale
- Mildly-Context Sensitive Languages, Tree Adjoining Grammars, Combinatory Categorial Grammars

NLP pipeline

Phonetics acoustic and perceptual elements

Phonology inventory of basic sounds (phonemes) and basic rules for combination, e.g. vowel harmony

Morphology how morphemes combine to form words, relationship of phonemes to meaning

Syntax sentence formation, word order and the formation of constituents from word groupings

Semantics how do word meanings recursively compose to form sentence meanings (from syntax to logical formulas)

Pragmatics meaning that is not part of compositional meaning

Syntax and Semantics

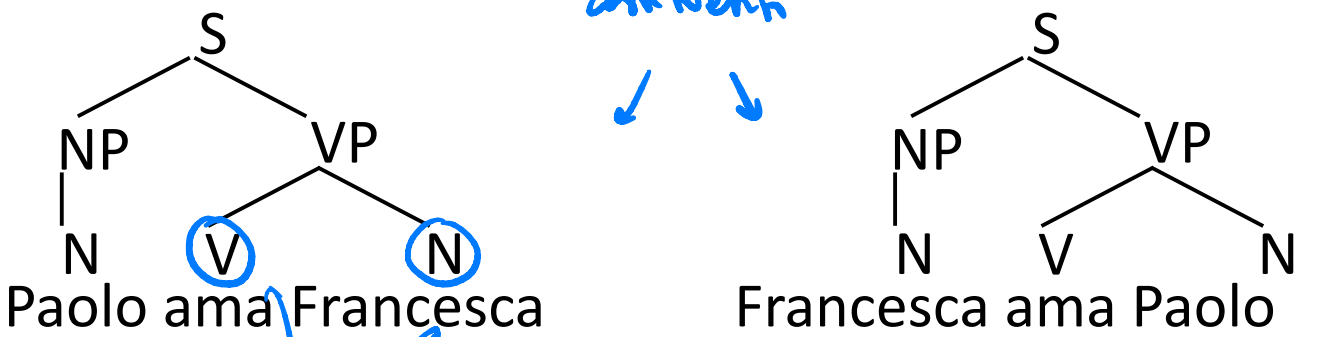
(Paolo (ama Francesca))

vs

Francesca ama Paolo

Syntactic Parsing

Syntactic Parsing



Competence/Performance

"Linguistic theory is concerned primarily with an ideal speaker-listener, in a completely homogeneous speech-communication, who know its (the speech community's) language perfectly and that it is unaffected by such **grammatically irrelevant conditions** as memory limitations, distractions, shifts of attention and interest, and errors (random or characteristic) in applying his knowledge of this language in actual performance."

competence "conoscenza" vs performance "algoritmo"

Competence/Performance in CL

- **Competence** = **Grammatica Formale**

- **Performance** = **Algoritmo di Parsing**

Competence/Performance in CL

- **Competence = Grammatica Formale**

- **Performance = Algoritmo di Parsing**

-> Prox. lezione: “Anatomia di un parser”

Rewriting Systems

Emil Post , Alan Turing

Rewriting rule

$$\Psi \rightarrow \theta$$

Generative grammar

è una grammatica formale: si basa sulla forma degli oggetti che la compongono

$G=(\Sigma,V,S,P)$

Σ = alphabet ← dizionario

simboli non terminali della grammatica

$V = \{A,B,\ldots\}$

simbolo di start

$S \in V$

insieme di regole di riscrittura/produzioni

insieme delle produzioni

$P = \{\Psi \rightarrow \theta,\ldots\}$

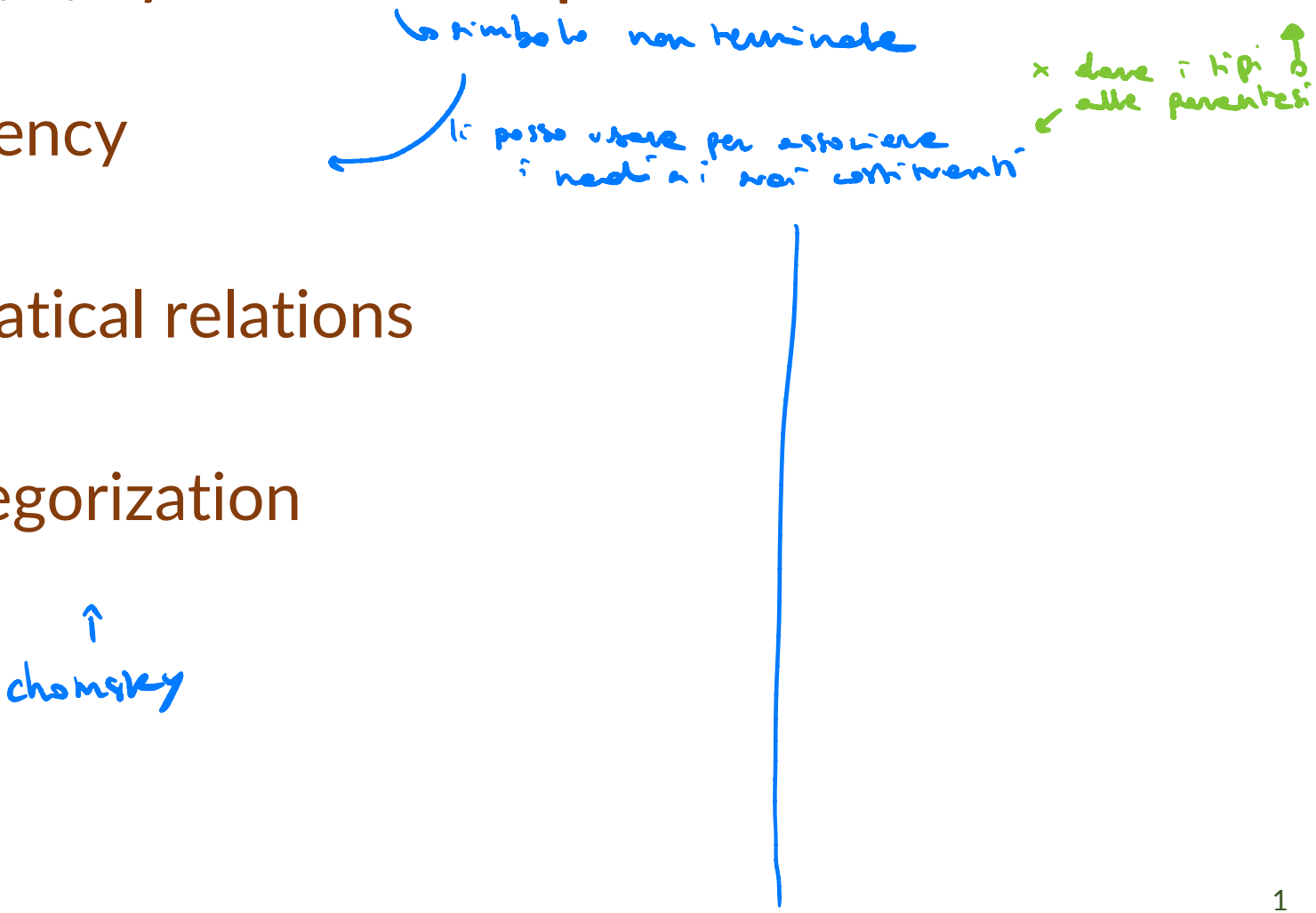
Generative Grammars and Natural Languages

- Generative Grammars models the natural languages as a formal languages
- The derivation tree can model the **syntactic structure** of the sentence

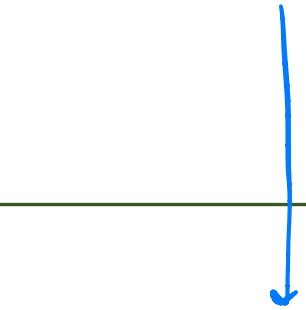
Context-Free Grammars

$G=(\Sigma,V,S,P)$ $\quad A \rightarrow \beta$

- Costituency
- Grammatical relations
- Subcategorization



Constituency



Context-Free Grammars Constituency

Hypothesis:

Constituent <=> non terminal symbols V

Toy Grammar

$G_4=(\Sigma_4,\{S,NP,VP,V_1,V_2\},S,P_4)$

Σ_4 = {I,Anna,John,Harry,saw,see,swimming}

P_4 = {S→ NPVP, VP→V_1S, VP→V_2,

NP→I|John|Harry|Anna,

V_1→saw|see, V_2→swimming}

Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

S

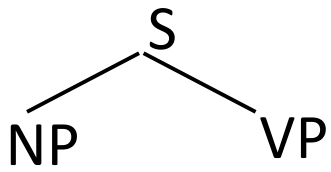
S

Toy Grammar

- S→ NP VP

VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

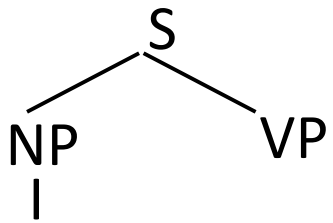
S⇒**NP VP**



Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

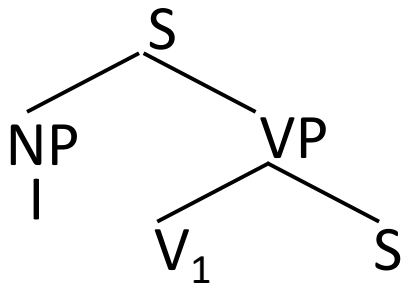
S⇒NP VP⇒**I VP**



Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

S⇒NP VP⇒I VP⇒**I V_1S**

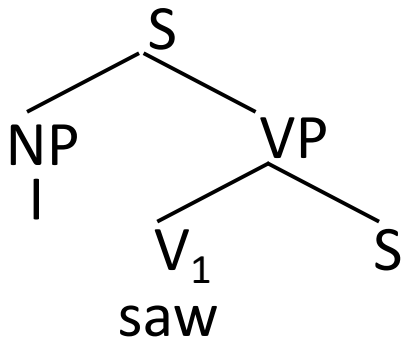


Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

S⇒NP VP⇒I VP⇒I V_1S⇒

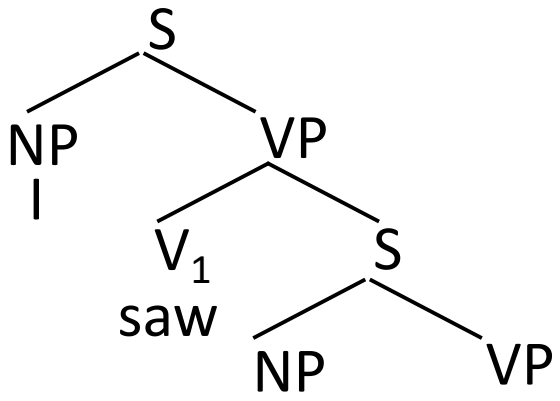
I saw S



Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

S⇒NP VP⇒I VP⇒I V_1S⇒
I saw S ⇒**I saw NP VP**



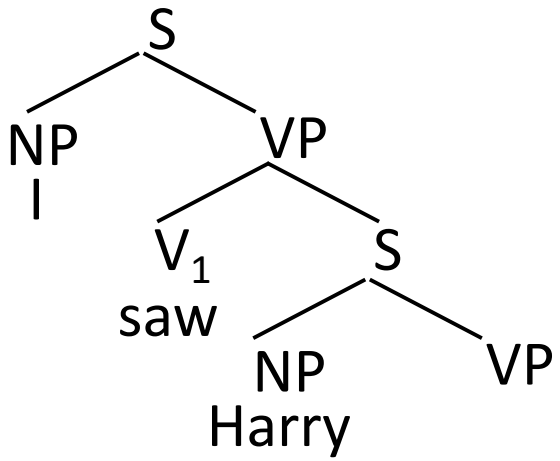
Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

S⇒NP VP⇒I VP⇒I V_1S⇒

I saw S ⇒I saw NP VP ⇒

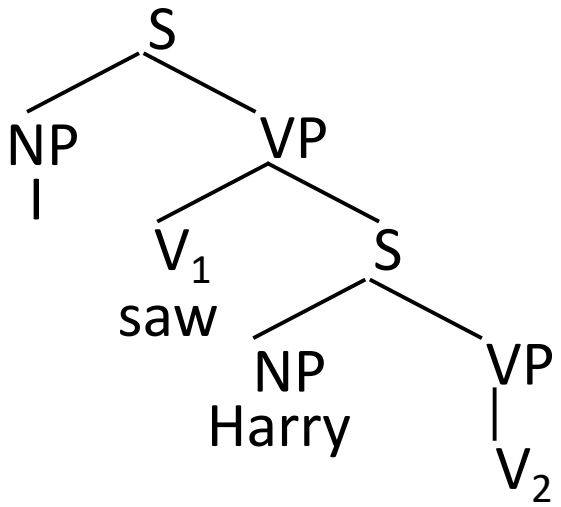
I saw Harry VP



Toy Grammar

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

S⇒NP VP⇒I VP⇒I V_1S⇒
I saw S ⇒I saw NP VP ⇒
I saw Harry VP ⇒
I saw Harry V_2



una grammatica vera è nell'ordine delle 20/30 mila regole

Toy Grammar

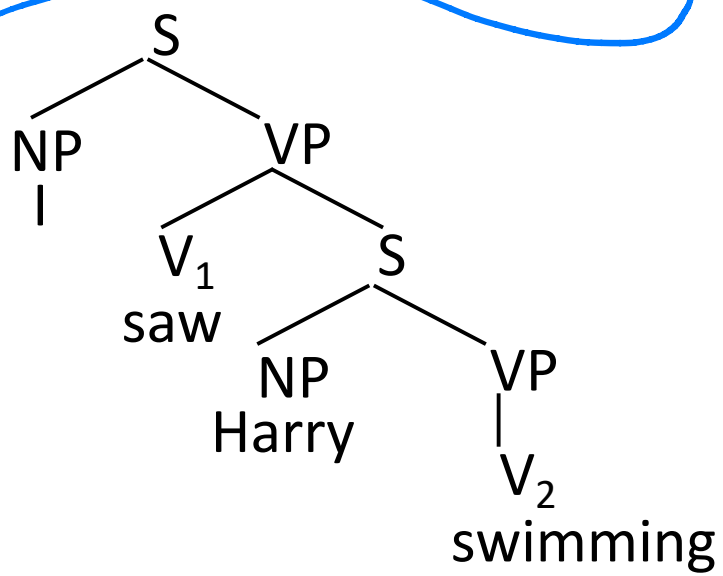
10 regole

S→ NP VP
VP→V_1 S
VP→V_2
NP→I|John|Harry|Anna
V_1→saw|see
V_2→swimming

poi non posso andare più avanti, perché

S⇒NP VP⇒I VP⇒I V_1S⇒
I saw S ⇒I saw NP VP ⇒
I saw Harry VP⇒
I saw Harry V_2⇒
I saw Harry swimming

sono tutti simboli terminali



Outline

- Sintassi e grammatiche generative
- La gerarchia di Chomsky e il linguaggio naturale
- Mildly-Context Sensitive Languages, Tree Adjoining Grammars, Combinatory Categorial Grammars

Languages Chomsky hierarchy

nessuna limitaz. sulla forma di produzione

$A1\phi \rightarrow \phi\phi$

Type 0

$\Psi \rightarrow \theta$

Context-sensitive

$Caa \rightarrow aaCa$

la lunghezza può solo aumentare (dx maggiore sx)

sulla sx ci deve essere un simb. NT

Context-free

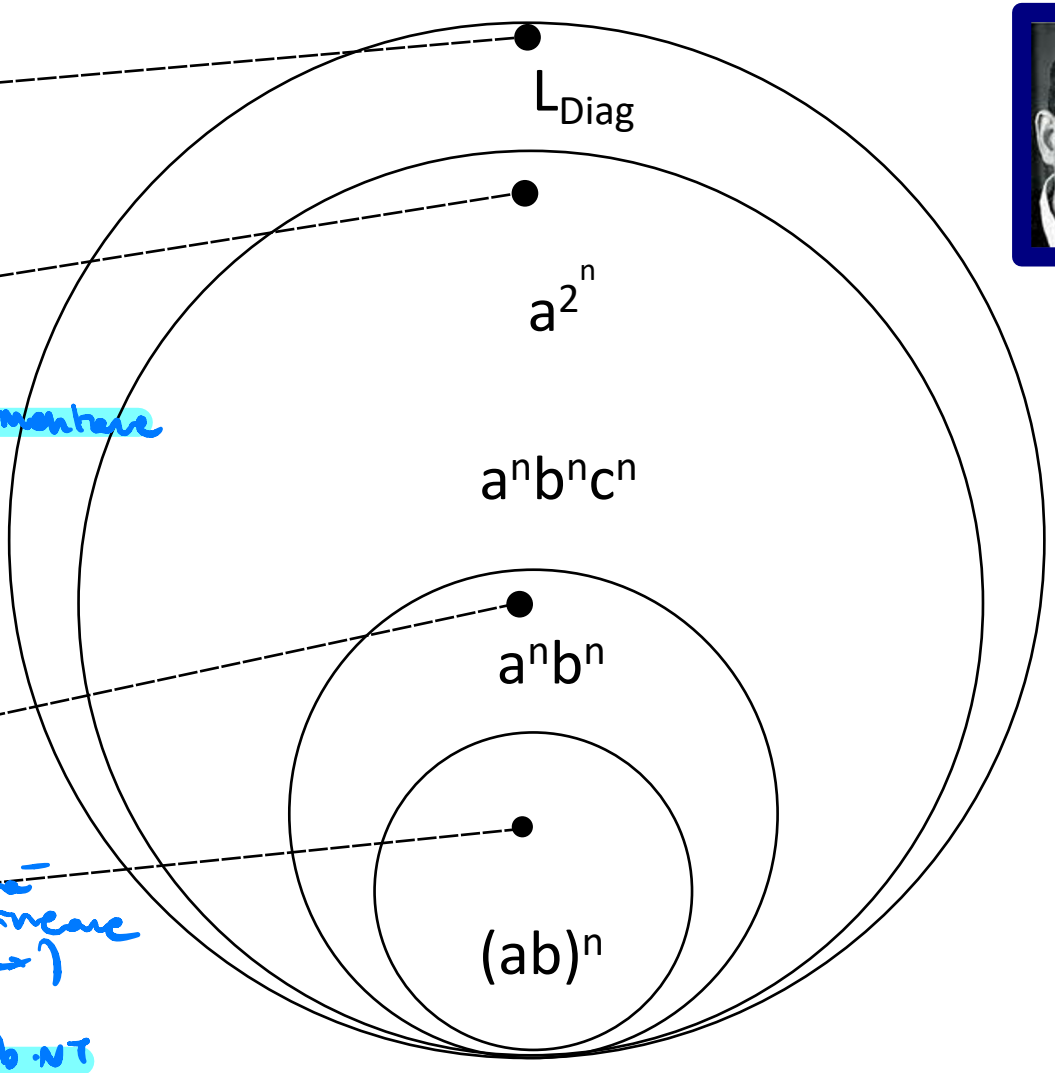
$S \rightarrow aSb$

Linear

$A \rightarrow aB$

simbolo NT → simb. T simb. NT

"non terminale" "terminale"



hanno una complessità comput. lineare (≥ sorgente)

quale di queste grammatiche c'è nel nostro cervello?

Swiss-German is NOT CF

(Shieber,1985)

... das mer em Hans es huus h¨alfed aastriiche

... that we Hans-Dat house-Acc helped paint

'... that we helped Hans paint the house'

... das mer d'chind em Hans es huus l¨ond h¨alfe aastriiche

... that we the childrenAcc HansDat houseAcc let help paint

'... that we let the children help Hans paint the house'

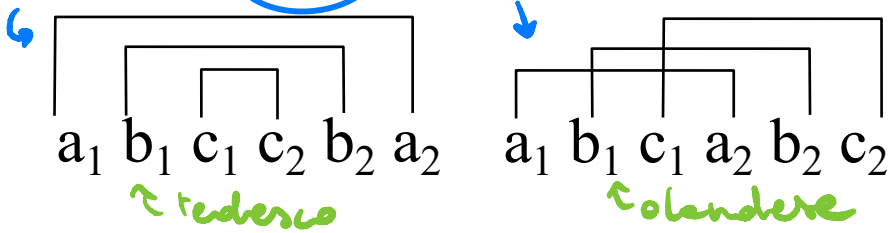
Outline

- Sintassi e grammatiche generative
- La gerarchia di Chomsky e il linguaggio naturale
- Mildly-Context Sensitive Languages, Tree Adjoining Grammars, Combinatory Categorial Grammars

Mildly context sensitive languages

[Joshi, 85]

- Include CFG languages
- Nested and cross-serial dependencies



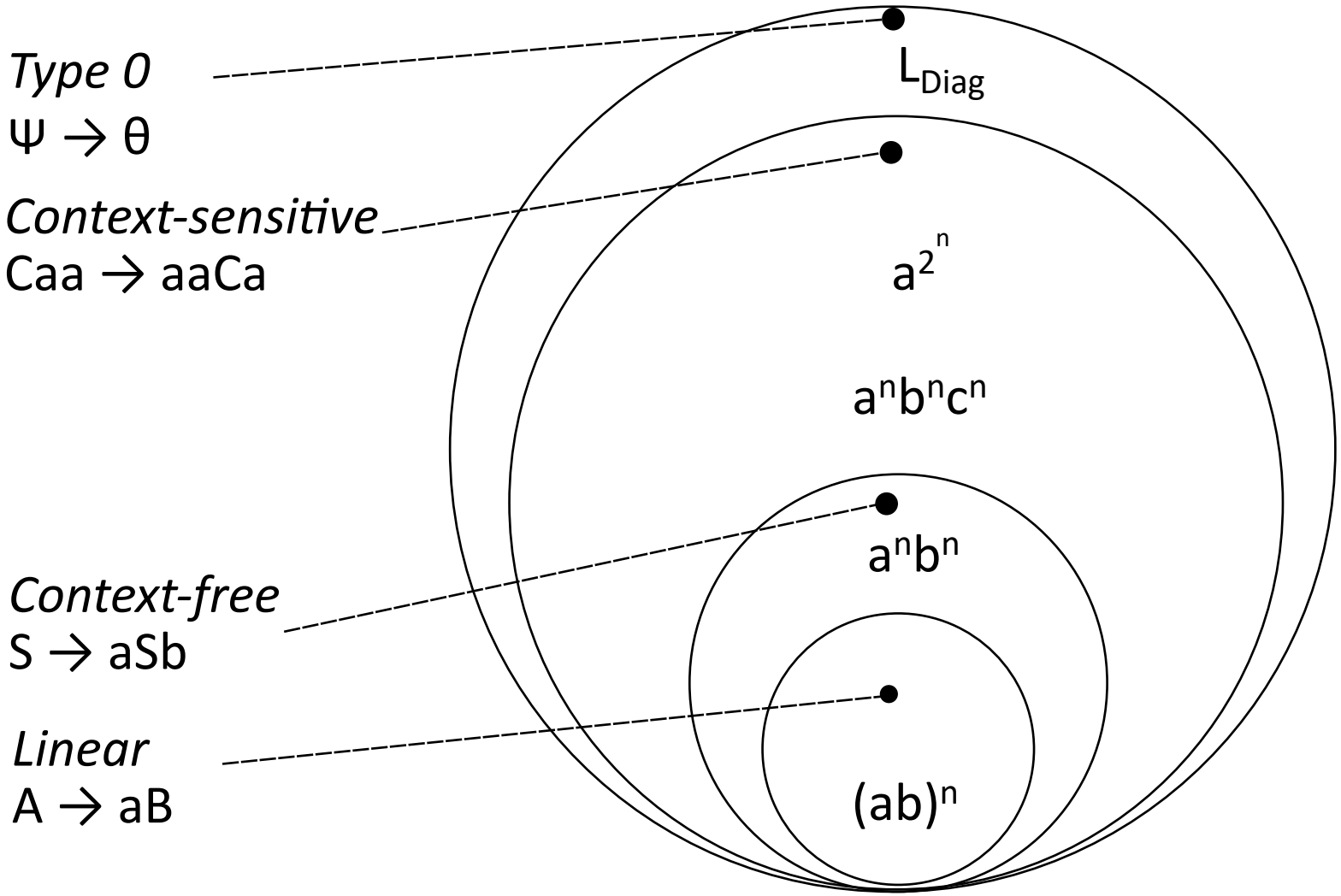
- Polynomially parsable
- Constant growth property

Constant growth property

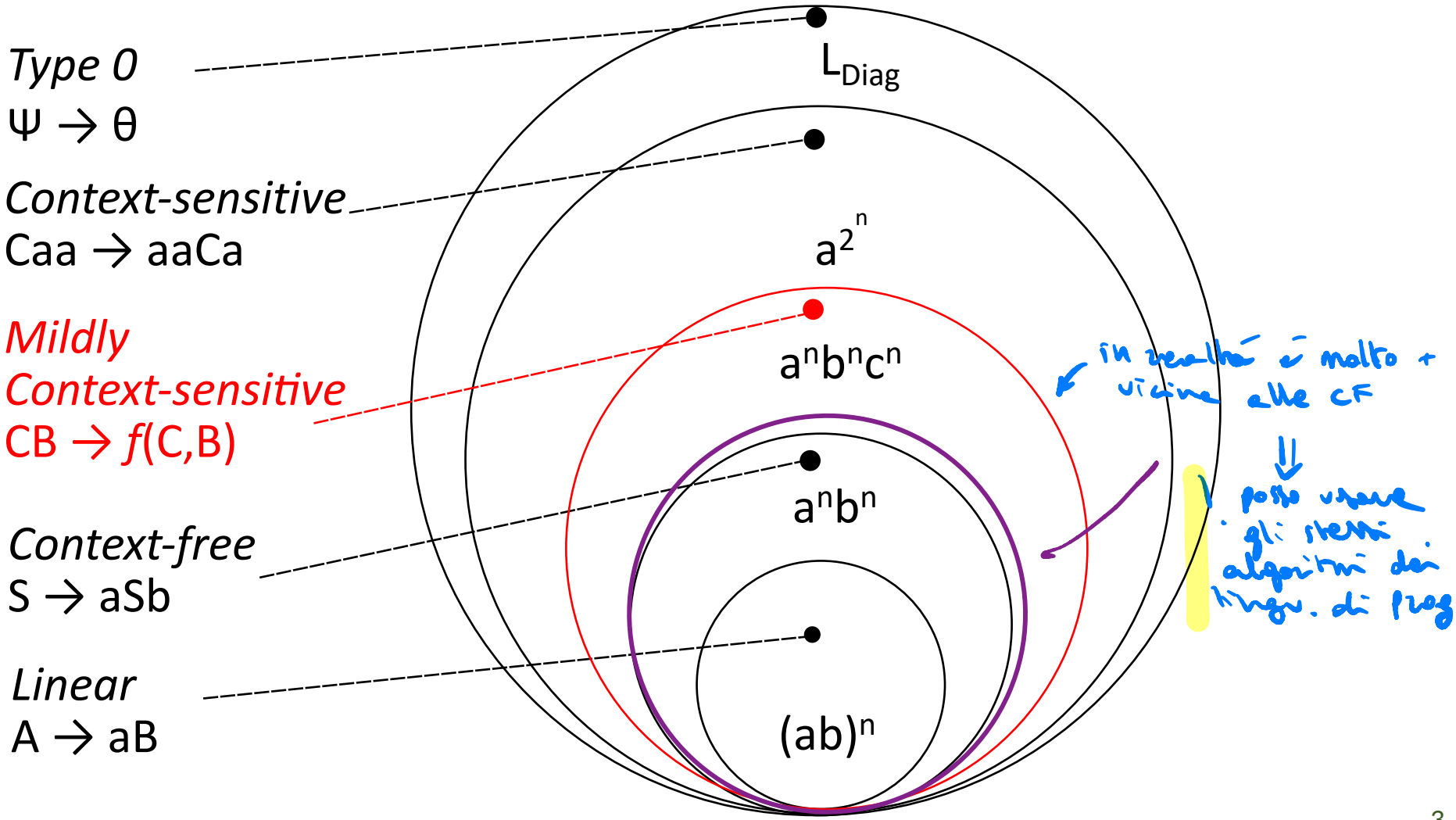
- **Definition** A language L is constant growth if there is a constants c_0 and a finite set of constant C such that for all $w \in L$ where $|w| > c_0$ there is a $w' \in L$ such that $|w| = |w'| + c$ for some $c \in C$.

- This property is the formal version of the linguistic intuition that the sentence belonging to a natural language can be built from a finite set of bounded structures using the same linear operations [Wei88].

Languages Chomsky hierarchy



Languages Chomsky hierarchy



MCL ⇔ TAG,HG,LIG,CCG

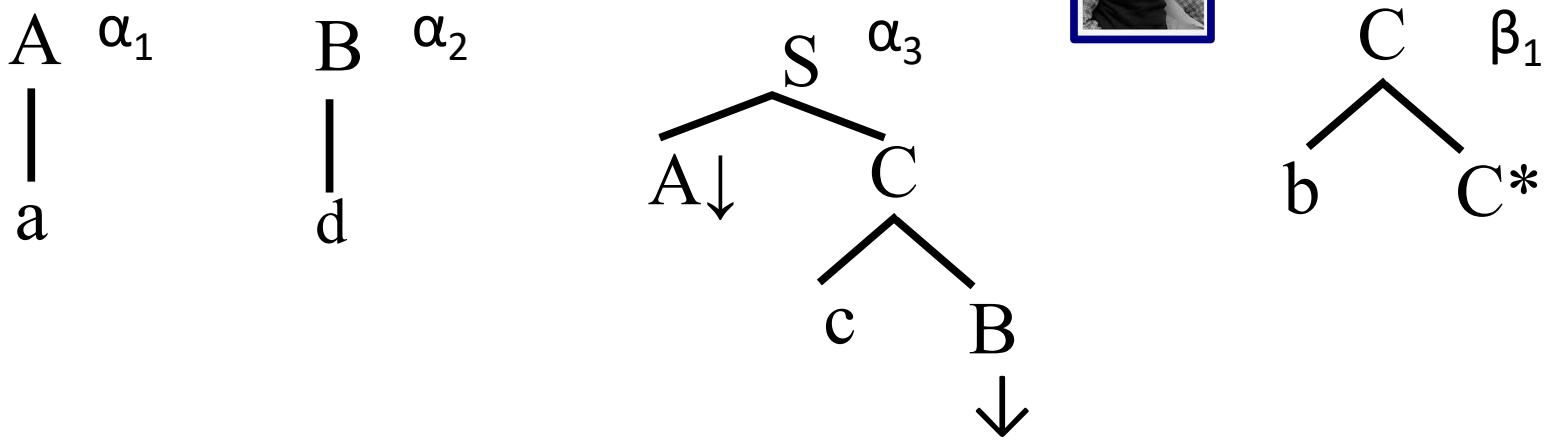
- Tree Adjoining Grammars (Joshi et al. 1975)
- Head Grammars (Pollard 1984)
- Linear Indexed Grammars (Gazdar 1985)
- Combinatory Categorial Grammars (Steedman 1985, Satta 2010)

-> elementary structures

-> combination rules

Tree Adjoining Grammars

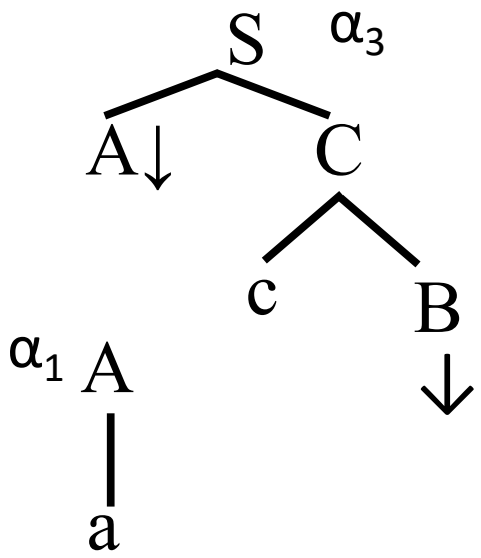
idea: invece di usare stringhe, uso alberi



Elementary structures = **multilevel trees**

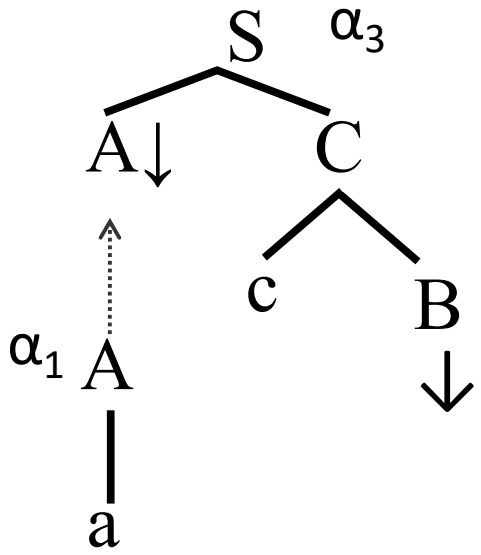
Tree Adjoining Grammars

idea: invece che 1 singola generativa, ho regole + complesse



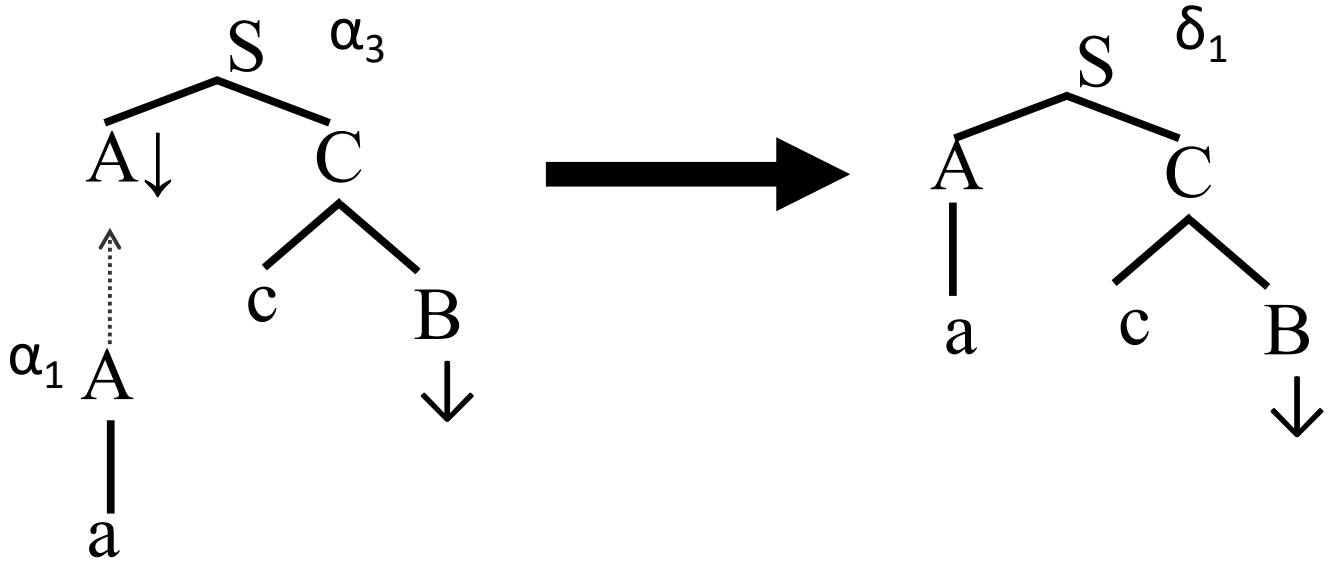
TAG operations: 1) substitution

Tree Adjoining Grammars



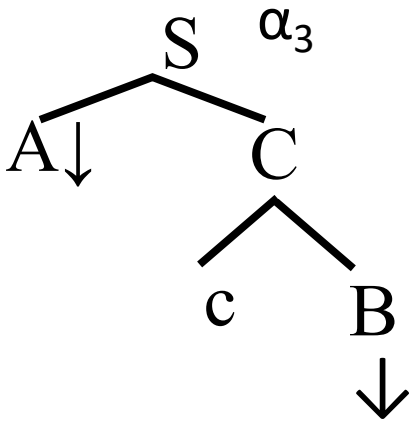
TAG operations: 1) substitution

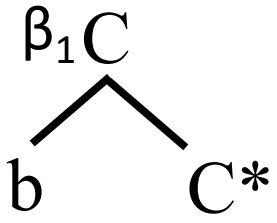
Tree Adjoining Grammars



TAG operations: 1) substitution

Tree Adjoining Grammars

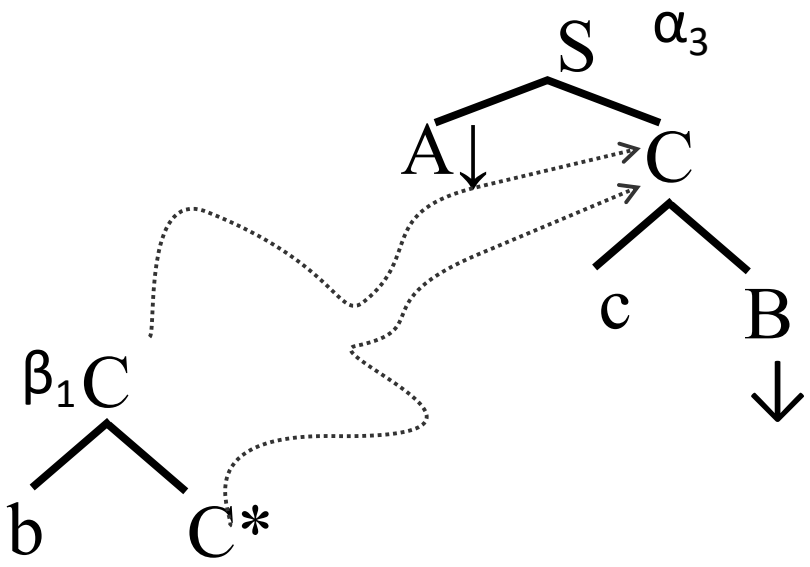




merge di un albero dentro un altro

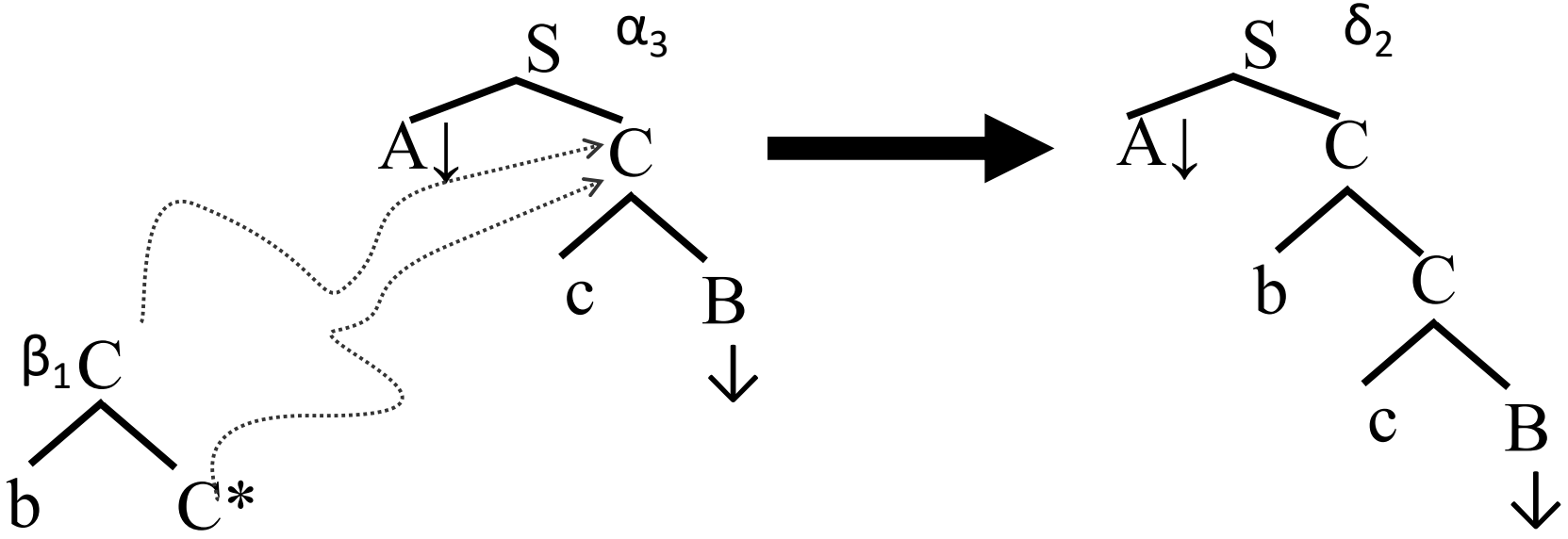
TAG operations: 2) adjoining

Tree Adjoining Grammars



TAG operations: 2) adjoining

Tree Adjoining Grammars



TAG operations: 2) adjoining

	CF standard	Tree Adjoining
struttura dati	stringa	alberi
regola di riscrittura	sostituzione	sostituzione + joining

perché farlo?: PRO: riesco a modellare certi fenomeni linguistici conosciuti in maniera molto + semplice

TAG and MCSL

- TAG properly contains all context-free languages (finitely ambiguous). Theorem (Schabes 1990)
- TAG is polynomially parsable}: **$O(n^6)$**
  - Embedded Push Down Automata, CKY (Vijay-Shanker 1987)
  - Left-to-right parser (Schabes 1990)

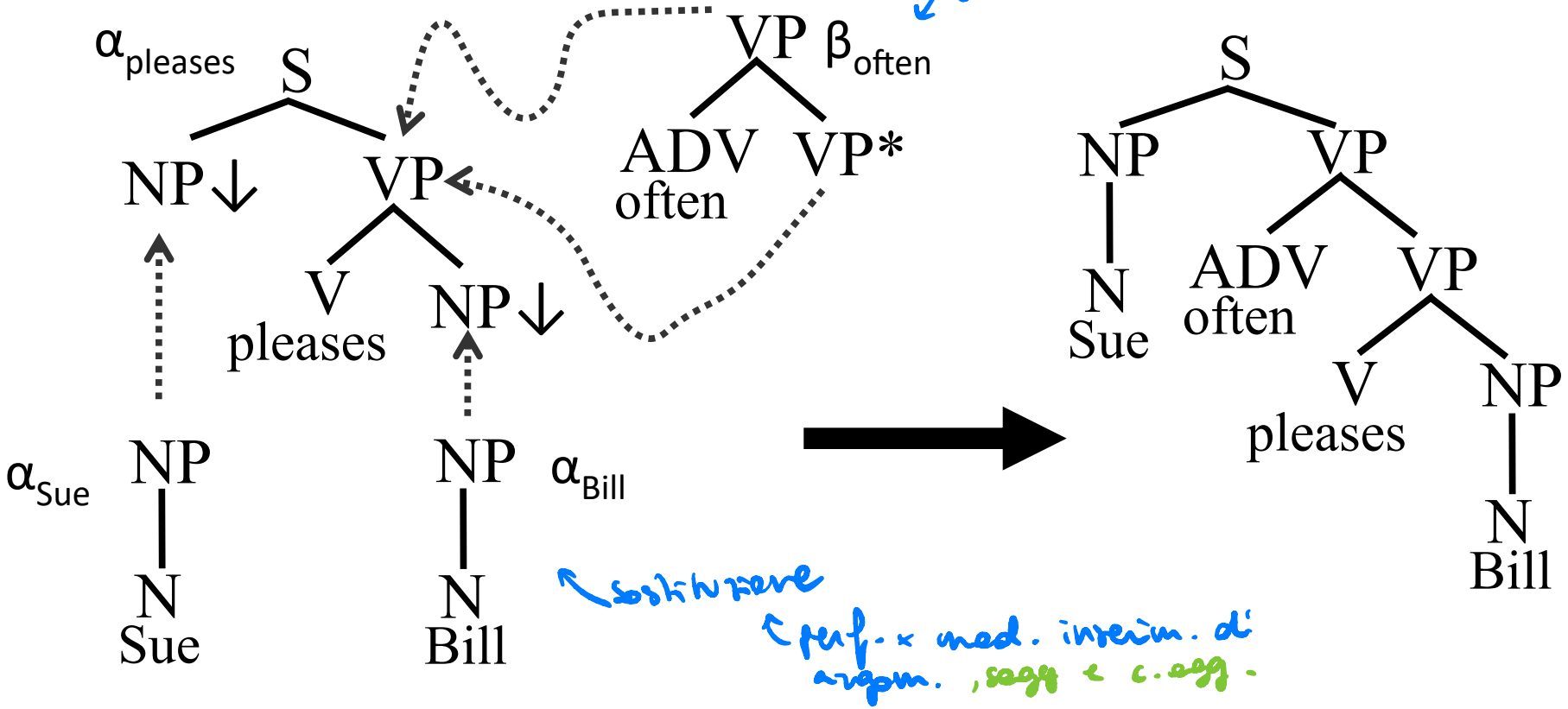
TAG and MCSL

- TAG captures only certain types of dependencies
  - **Cross-serial dependencies**: verb-raised analisys (Kroch Santorini 1991)
  - No mix-languages
- TAG has the constant--growth property: (Weir 1988)

Lexicalized Tree Adjoining Grammars

- Extended domain of locality
- Recursion Factorization by adjoining operation
- Lexicalization

LTAG



- structures = multilevel trees
- operations = substitution, adjoining

Combinatory Categorial Grammar

- Generative -> top-down
- Categoriali ->bottom-up

The Syntactic Process, *Mark Steedman*

Categories, Lexicon, Rules

Combinatory Categorial Grammar

- Generative -> top-down
- Categoriali ->bottom-up

è la struttura dati

Category

-> $A \subseteq C$, where A is a given set of atomic elements

-> (X/Y) , $(X\backslash Y) \in C$, if $X, Y \in C$

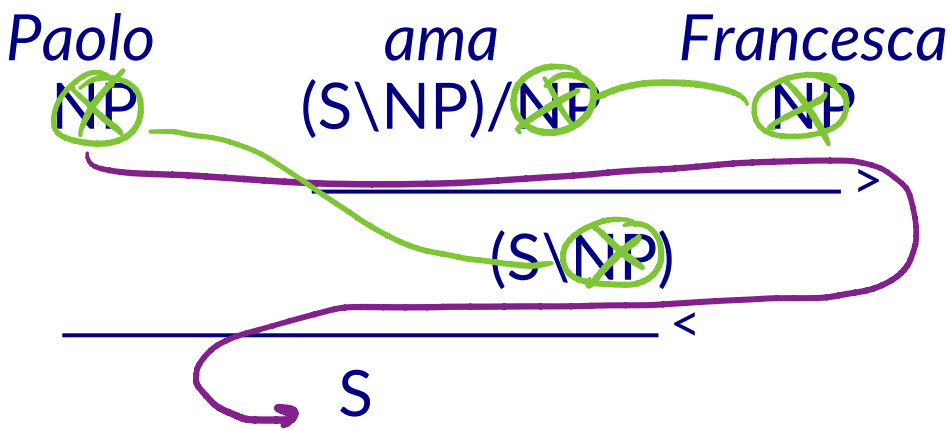
CCG

- Lexicon
  - Paolo : NP
  - Francesca : NP
  - amare : (S\NP)/NP
- Rules
  - X/Y Y ⇒ X
    ->
  - Y X\Y ⇒ X
    -<
  - …

invece delle reg. di riscrittura, ho regole di combinatore

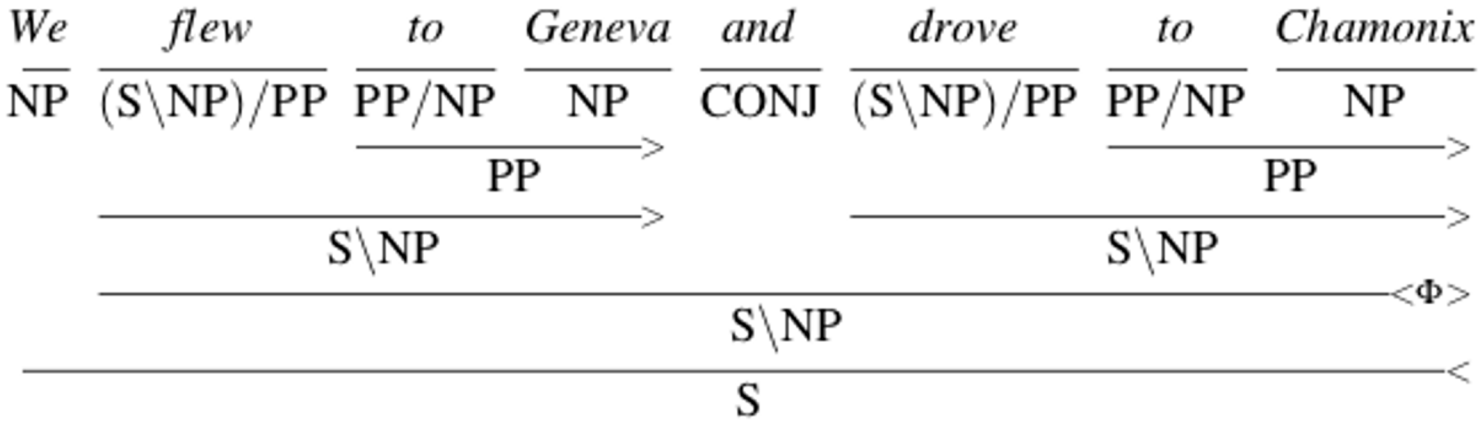
CCG

- Lexicon
  - Paolo : NP
  - Francesca : NP
  - amare : (S\NP)/NP
- Rules
  - X/Y Y ⇒ X
    ->
  - Y X\Y ⇒ X
    -<
  - …



CCG → sono utili in casi particolari

- Coordinazione: X CONJ X ⇒ X



R ⇓ (noi scegliamo questa)

ho 2 strade: 1. il NL ≅ CF (xkè la di-H. è poca)

2. usare una TAG o una CCG o altre x modellare una mildly blabla (e non fare l'approssimazione NL≅CF)