Formal Languages and Linguistics

Pascal Amsili

Sorbonne Nouvelle, Lattice (CNRS/ENS-PSL)

Cogmaster, september 2020

Sorbonne Nouvelle

General introduction

1. Mathematicians (incl. Chomsky) have formalized the notion of **language**

   It might be thought of as an oversimplification,
   always the same story...

2. It buys us:
   1. Tools to think about theoretical issues about language/s (expressiveness, complexity, comparability...)
   2. Tools to manipulate concretely language (e.g. with computers)
   3. A research programme:
      - Represent the syntax of natural language in a fully unambiguously specified way

Now let's get familiar with the mathematical notion of language

Overview

Sorbonne Nouvelle

Alphabet, word

Def. 1 (Alphabet)

An *alphabet* Σ is a finite set of symbols (letters).
The *size* of the alphabet is the cardinal of the set.

Def. 2 (Word)

A *word* on the alphabet Σ is a finite sequence of letters from Σ.
Formally, let $[p] = (1, 2, 3, 4, ..., p)$ (ordered integer sequence).
Then a word is a *mapping*

$$u : [p] \longrightarrow \Sigma$$

p, the length of u, is noted $|u|$.

Examples I

Alphabet {·, —}

Words ———·

·

··—·

...

Alphabet {·— , —··· , —·—· , —·· , · , ... }

Words ··· ——— ···

—·· ·· —·· · ·—· ——— —

·— —··· —··· ·— ·— —··· —··· ·—

...

Examples II

Alphabet $\{0,1,2,3,4,5,6,7,8,9,\cdot\}$
Words $235 \cdot 29$
$007 \cdot 12$
$\cdot 1 \cdot 1 \cdot 00 \cdot \cdot$
$\cancel{3 \cdot 1415962 \ldots}$ (π)
...

Alphabet {a, woman, loves, man}
Words a
a woman loves a woman
man man a loves woman loves a
...

Monoid

Def. 3 (Σ^*)

Let Σ be an alphabet.
The set of all the words that can be formed with any number of letters from Σ is noted Σ^*

Σ^* includes a word with no letter, noted ε

Example:
$$\begin{aligned} \Sigma &= \{a, b, c\} \\ \Sigma^* &= \{\varepsilon, a, b, c, aa, ab, ac, ba, \ldots, bbb, \ldots\} \end{aligned}$$

N.B.: Σ^* is always infinite, except...

Monoid

Def. 3 (Σ^*)

Let Σ be an alphabet.
The set of all the words that can be formed with any number of letters from Σ is noted Σ^*

Σ^* includes a word with no letter, noted ε

Example:
$$\begin{aligned} \Sigma &= \{a, b, c\} \\ \Sigma^* &= \{\varepsilon, a, b, c, aa, ab, ac, ba, \ldots, bbb, \ldots\} \end{aligned}$$

N.B.: Σ^* is always infinite, except...

$$\text{if } \Sigma = \emptyset. \text{ Then } \Sigma^* = \{\varepsilon\}.$$

Sorbonne Nouvelle

Structure of Σ^*

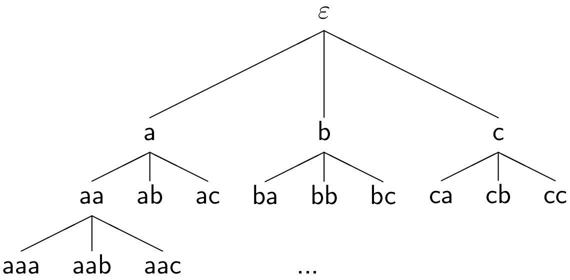
Let k be the size of the alphabet $k = |\Sigma|$.

Then Σ^* contains :

$k^0 = 1$	word(s) of 0 letters (ε)
$k^1 = k$	word(s) of 1 letters
k^2	word(s) of 2 letters
$\dots$	
k^n	words of n letters, $\forall n \geq 0$

Representation of Σ^*

$\Sigma = \{a, b, c\}$



- Words can be enumerated according to different orders
- Σ^* is a *countable* set

Concatenation

Σ^* can be equipped with a binary operation: *concatenation*

Def. 4 (Concatenation)

Let $[p] \xrightarrow{u} X$, $[q] \xrightarrow{w} X$. The concatenation of u and w, noted uw $(u.w)$ is thus defined:

$$\begin{aligned} uw : \quad & [p+q] \longrightarrow X \\ & uw_i = \begin{cases} u_i & \text{for} \quad i \in [1, p] \\ w_{i-p} & \text{for} \quad i \in [p+1, p+q] \end{cases} \end{aligned}$$

Concatenation

Σ^* can be equipped with a binary operation: *concatenation*

Def. 4 (Concatenation)

Let $[p] \xrightarrow{u} X$, $[q] \xrightarrow{w} X$. The concatenation of u and w, noted uw $(u.w)$ is thus defined:

$$\begin{aligned} uw : \quad & [p+q] \longrightarrow X \\ & uw_i = \begin{cases} u_i & \text{for} \quad i \in [1, p] \\ w_{i-p} & \text{for} \quad i \in [p+1, p+q] \end{cases} \end{aligned}$$

Example :	u	bacba
	v	cca

Sorbonne Nouvelle

Concatenation

Σ^* can be equipped with a binary operation: *concatenation*

Def. 4 (Concatenation)

Let $[p] \xrightarrow{u} X$, $[q] \xrightarrow{w} X$. The concatenation of u and w, noted uw $(u.w)$ is thus defined:

$$uw : \quad [p+q] \longrightarrow X$$
$$uw_i = \begin{cases} u_i & \text{for} \quad i \in [1,p] \\ w_{i-p} & \text{for} \quad i \in [p+1, p+q] \end{cases}$$

Example :	u	bacba
	v	cca
	uv	bacbacca

Factor

Def. 5 (Factor)

A *factor* w of u is a subset of adjascent letters in u.

$-w$ is a factor of u $\Leftrightarrow$ $\exists u_1, u_2$ s.t. $u = u_1 w u_2$

$-w$ is a left factor (*prefix*) of u $\Leftrightarrow$ $\exists u_2$ s.t. $u = w u_2$

$-w$ is a right factor (*suffix*) of u $\Leftrightarrow$ $\exists u_1$ s.t. $u = u_1 w$

Def. 6 (Factorization)

We call *factorization* the decomposition of a word into factors.

Role of concatenation

1. Words have been defined on Σ.
   If one takes two such words, it's always possible to form a new word by concatenating them.
2. Any word can be factorised in many different ways:
   $a\,b\,a\,c\,c\,a\,b$

Role of concatenation

1. Words have been defined on Σ.
   If one takes two such words, it's always possible to form a new word by concatenating them.
2. Any word can be factorised in many different ways:
   $a\,b\,a\,c\,c\,a\,b$
   $(a\,b\,a)(c\,c\,a\,b)$

Role of concatenation

1. Words have been defined on Σ.
   If one takes two such words, it's always possible to form a new word by concatenating them.
2. Any word can be factorised in many different ways:
   $a\ b\ a\ c\ c\ a\ b$
   $(a\ b)(a\ c\ c)(a\ b)$

Role of concatenation

1. Words have been defined on Σ.
   If one takes two such words, it's always possible to form a new word by concatenating them.
2. Any word can be factorised in many different ways:
   $a\ b\ a\ c\ c\ a\ b$
   $(a\ b\ a\ c\ c)(a\ b)$

Role of concatenation

1. Words have been defined on Σ.
   If one takes two such words, it's always possible to form a new word by concatenating them.
2. Any word can be factorised in many different ways:
   $a\,b\,a\,c\,c\,a\,b$
   $(a)(b)(a)(c)(c)(a)(b)$

Role of concatenation

1. Words have been defined on Σ.
   If one takes two such words, it's always possible to form a new word by concatenating them.
2. Any word can be factorised in many different ways:
   $a\,b\,a\,c\,c\,a\,b$
   $(a)(b)(a)(c)(c)(a)(b)$
3. Since all letters of Σ form a word of length 1 (this set of words is called the *base*),
4. any word of Σ^* can be seen as a (unique) sequence of concatenations of length 1 words :
   $a\,b\,a\,c\,c\,a\,b$
   $((((((ab)a)c)c)a)b)$
   $((((((a.b).a).c).c).a).b)$

Properties of concatenation

1. Concatenation is non commutative
2. Concatenation is associative
3. Concatenation has an identity (neutral) element: ε

1. $uv.w \neq w.uv$
2. $(u.v).w = u.(v.w)$
3. $u.\varepsilon = \varepsilon.u = u$

Notation : $a.a.a = a^3$

Overview

Language

Def. 7 ((Formal) Language)

Let Σ be an alphabet.
A language on Σ is a set of words on Σ.

Sorbonne Nouvelle

Language

Def. 7 ((Formal) Language)

Let Σ be an alphabet.
A language on Σ is a set of words on Σ.
or, equivalently,
A language on Σ is a subset of Σ^*

Examples I

Let $\Sigma = \{a, b, c\}$.

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language
$L_2 = \{a, aa, aaa, aaaa \ldots\}$	

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language
$L_2 = \{a, aa, aaa, aaaa \ldots\}$ or $L_2 = \{a^i \ / \ i \geq 1\}$	infinite language

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language
$L_2 = \{a, aa, aaa, aaaa \ldots\}$ or $L_2 = \{a^i \ / \ i \geq 1\}$	infinite language
$L_3 = \{\varepsilon\}$	finite language, reduced to a singleton

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language
$L_2 = \{a, aa, aaa, aaaa \dots\}$ or $L_2 = \{a^i \ / \ i \geq 1\}$	infinite language
$L_3 = \{\varepsilon\}$	finite language, reduced to a singleton

$\neq$

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language
$L_2 = \{a, aa, aaa, aaaa \ldots\}$ or $L_2 = \{a^i \ / \ i \geq 1\}$	infinite language
$L_3 = \{\varepsilon\}$	finite language, reduced to a singleton
$L_4 = \emptyset$	$\neq$ "empty" language

Examples I

Let $\Sigma = \{a, b, c\}$.

$L_1 = \{aa, ab, bac\}$	finite language
$L_2 = \{a, aa, aaa, aaaa \ldots\}$ or $L_2 = \{a^i \ / \ i \geq 1\}$	infinite language
$L_3 = \{\varepsilon\}$	finite language, reduced to a singleton
	$\neq$
$L_4 = \emptyset$	"empty" language
$L_5 = \Sigma^*$	

Examples II

Let $\Sigma = \{\text{a, man, loves, woman}\}$.

Examples II

Let $\Sigma = \{\text{a, man, loves, woman}\}$.

$L = \{$ a man loves a woman, a woman loves a man $\}$

Examples II

Let $\Sigma = \{\text{a, man, loves, woman}\}$.

$L = \{$ a man loves a woman, a woman loves a man $\}$

Let $\Sigma' = \{\text{a, man, who, saw, fell}\}$.

Examples II

Let $\Sigma = \{\text{a, man, loves, woman}\}$.

$L = \{$ a man loves a woman, a woman loves a man $\}$

Let $\Sigma' = \{\text{a, man, who, saw, fell}\}$.

$$L' = \left\{ \begin{array}{l} \text{a man fell,} \\ \text{a man who saw a man fell,} \\ \text{a man who saw a man who saw a man fell,} \\ \ldots \end{array} \right\}$$

Set operations

Since a language is a set, usual set operations can be defined:

- union
- intersection
- set difference

Set operations

Since a language is a set, usual set operations can be defined:

- union
- intersection
- set difference

$\Rightarrow$ One may describe a (complex) language as the result of set operations on (~~simpler~~) languages:
$\{a^{2k} \;/\; k \geqslant 1\} = \{a, aa, aaa, aaaa, \ldots\} \cap \{ww \;/\; w \in \Sigma^*\}$

Additional operations

Def. 8 (product operation on languages)

One can define the *language product* and its closure *the Kleene star* operation:

- The *product* of languages is thus defined:
$$L_1.L_2 \quad = \quad \{uv \,/\, u \in L_1 \,\&\, v \in L_2\}$$
Notation: $\overbrace{L.L.L \ldots L}^{k \text{ times}} = L^k \; ; \; L^0 = \{\varepsilon\}$
- The Kleene star of a language is thus defined:
$$L^* = \bigcup_{n \geqslant 0} L^n$$

Regular expressions

It is common to use the 3 *rational* operations:

- union
- product
- Kleene star

to characterize certain languages...

Regular expressions

It is common to use the 3 *rational* operations:

- union
- product
- Kleene star

to characterize certain languages...

$$(\{a\} \cup \{b\})^*.\{c\} = \{c, ac, abc, bc, \ldots, baabaac, \ldots\}$$

(simplified notation $(a|b)^*c$ — regular expressions)

Regular expressions

It is common to use the 3 *rational* operations:

- union
- product
- Kleene star

to characterize certain languages...

$$(\{a\} \cup \{b\})^*.\{c\} = \{c, ac, abc, bc, \ldots, baabaac, \ldots\}$$

(simplified notation $(a|b)^*c$ — regular expressions)

... but not all languages can be thus characterized.

Overview

1. Formal Languages
   - Basic concepts
   - Definition
   - Problem
2. Formal Grammars
3. Regular Languages
4. Formal complexity of Natural Languages

Back to "Natural" Languages

English as a formal language:

- **alphabet:** morphemes (often simplified to words —depending on your view on flexional morphology)
  - $\Rightarrow$ Finite at a time t by hypothesis
- **words:** well formed English sentences
  - $\Rightarrow$ English sentences are all finite by hypothesis
- **language:** English, as a set of an infinite number of well formed combinations of "letters" from the alphabet

Discussion I

1. is the alphabet finite?

   closed class morphemes obviously

   open class morphemes what about "new words"?

   morphological derivations can be seen as produced from an unchanged inventory (1)

   other words
   - loan words (rare)
   - lexical inventions (rare)
   - change of category (2) (bounded)

   $\Rightarrow$ negligable

(1) motherese = mother+ese

(2) $\text{american}_A \rightarrow \text{american}_N$

Discussion II

2. is English infinite ?
   - It is supposed that you can always profer a longer sentence than the previous one by adding linguistic material preserving well-formedness.
   - Compatible with the working memory limit

     (Langendoen & Postal, 1984)
3. is language discrete ?

   Well, that's another story

About infinity

Linguists sometimes have trouble with infinity:

> In order for there to be an infinite number of sentences in a language there must either be an infinite number of words in the language (clearly not true) or there must be the possibility of infinite length sentences. The product of two finite numbers is always a finite number. (Mannell, 1999)
>
> *and many others*

About infinity

Linguists sometimes have trouble with infinity:

> ~~In order for there to be an infinite number of sentences in a language there must either be an infinite number of words in the language (clearly not true) or there must be the possibility of infinite length sentences. The product of two finite numbers is always a finite number.~~ (Mannell, 1999)
>
> *and many others*

!! WRONG !!

About infinity

Linguists sometimes have trouble with infinity:

> ~~In order for there to be an infinite number of sentences in a language there must either be an infinite number of words in the language (clearly not true) or there must be the possibility of infinite length sentences. The product of two finite numbers is always a finite number.~~ (Mannell, 1999)
>
> *and many others*

!! WRONG !!

The whole point of formal languages is that they are **infinite** sets of **finite** words on a **finite** alphabet.

About infinity

Linguists sometimes have trouble with infinity:

~~In order for there to be an infinite number of sentences in a language there must either be an infinite number of words in the language (clearly not true) or there must be the possibility of infinite length sentences. The product of two finite numbers is always a finite number.~~ (Mannell, 1999)

and many others

!! WRONG !!

The whole point of formal languages is that they are **infinite** sets of **finite** words on a **finite** alphabet.

von Humbolt: *language is an infinite use of finite means*

(quoted by Chomsky)

Good questions

Why would one consider natural language as a formal language?

- it allows to $\boxed{\text{describe}}$ the language in a formal/compact/elegant way
- it allows to $\boxed{\text{compare}}$ various languages (via classes of languages established by mathematicians)
- it give algorithmic tools to $\boxed{\text{recognize}}$ and to $\boxed{\text{analyse}}$ words of a language.

recognize u	: decide whether $u \in L$
analyse u	: show the internal structure of u

Overview

Introduction

Formal grammars have been proposed by Chomsky as **one of the available means** to characterize a formal language.
Other means include :

- Turing machines (automata)
- λ-terms
- ...

Formal grammar

Def. 9 ((Formal) Grammar)

A **formal grammar** is defined by $\langle \Sigma, N, S, P \rangle$ where

- Σ is an alphabet
- N is a disjoint alphabet (non-terminal vocabulary)
- $S \in V$ is a distinguished element of N, called the *axiom*
- P is a set of « *production rules* », namely a subset of the cartesian product $(\Sigma \cup N)^* N (\Sigma \cup N)^* \times (\Sigma \cup N)^*$.

Examples

$$\langle \Sigma, N, S, P \rangle$$

$$\mathcal{G}_0 = \Bigg\langle$$

Examples

$$\langle \Sigma, N, S, P \rangle$$

$$\mathcal{G}_0 = \Big\langle \{joe, sam, sleeps\},$$

Examples

$$\langle \Sigma, N, S, P \rangle$$

$$\mathcal{G}_0 = \Big\langle \{joe, sam, sleeps\}, \{N, V, S\},$$

Examples

$$\langle \Sigma, N, S, P \rangle$$

$$\mathcal{G}_0 = \Bigg\langle \{joe, sam, sleeps\}, \{N, V, S\}, S,$$

Examples

$$\langle \Sigma, N, S, P \rangle$$

$$\mathcal{G}_0 = \left\langle \{joe, sam, sleeps\}, \{N, V, S\}, S, \left\{ \begin{array}{l} (N, joe) \\ (N, sam) \\ (V, sleeps) \\ (S, N\ V) \end{array} \right\} \right\rangle \}$$

Examples

$$\langle \Sigma, N, S, P \rangle$$

$$\mathcal{G}_0 = \left\langle \{joe, sam, sleeps\}, \{N, V, S\}, S, \left\{ \begin{array}{l} N \rightarrow joe \\ N \rightarrow sam \\ V \rightarrow sleeps \\ S \rightarrow N\ V \end{array} \right\} \right\rangle \}$$

Examples (cont'd)

$$\mathcal{G}_1 = \left\langle \{jean, dort\}, \{Np, SN, SV, V, S\}, S, \left\{ \begin{array}{l} S \rightarrow SN\ SV \\ SN \rightarrow Np \\ SV \rightarrow V \\ Np \rightarrow jean \\ V \rightarrow dort \end{array} \right\} \right\rangle \}$$

$$\mathcal{G}_2 = \langle \{(,)\}, \{S\}, S, \{S \longrightarrow \varepsilon \,|\, (S)S\} \rangle$$

Notation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

Notation

$$
\begin{array}{llcl}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

$$\mathcal{G}_3 = \langle \{+, \times, (,), 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}, \{E, F\}, E, \{\ldots\} \rangle$$

Notation

$$
\begin{array}{llll}
\mathcal{G}_3 : & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9
\end{array}
$$

$$\mathcal{G}_3 = \langle \{+, \times, (,), 0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}, \{E, F\}, E, \{\ldots\} \rangle$$

$$G_4 = E \rightarrow E + T \mid T, T \rightarrow T \times F \mid F, F \rightarrow (E) \mid a$$

Immediate Derivation

Def. 10 (Immediate derivation)

Let $\mathcal{G} = \langle X, V, S, P\rangle$ a grammar, $(f, g) \in (X \cup V)^*$ two "words", $r \in P$ a production rule, such that $r : A \longrightarrow u$ $(u \in (X \cup V)^*)$.

- f derives into g (immediate derivation) with the rule r (noted $f \xrightarrow{r} g$) iff
  $\exists v, w$ s.t. $f = vAw$ and $g = vuw$
- f derives into g (immediate derivation) in the grammar $\mathcal{G}$ (noted $f \xrightarrow{\mathcal{G}} g$) iff
  $\exists r \in P$ s.t. $f \xrightarrow{r} g$.

Sorbonne Nouvelle

Derivation

Def. 11 (Derivation)

$$f \xrightarrow{\mathcal{G}*} g \text{ if } \begin{cases} f = g & \text{or} \\ \exists f_0, f_1, f_2, \ldots, f_n \text{ s.t. } & f_0 = f \\ & f_n = g \\ & \forall i \in [1, n] : f_{i-1} \xrightarrow{\mathcal{G}} f_i \end{cases}$$

An example with $\mathcal{G}_0$:
$N\ V\ joe\ N$

Derivation

Def. 11 (Derivation)

$$f \xrightarrow{\mathcal{G}*} g \text{ if } \begin{cases} f = g & \text{or} \\ \exists f_0, f_1, f_2, \ldots, f_n \text{ s.t. } & f_0 = f \\ & f_n = g \\ & \forall i \in [1, n] : f_{i-1} \xrightarrow{\mathcal{G}} f_i \end{cases}$$

An example with $\mathcal{G}_0$:
$N\ V\ joe\ N \longrightarrow sam\ V\ joe\ N$

Derivation

Def. 11 (Derivation)

$$f \xrightarrow{\mathcal{G}*} g \text{ if } \begin{array}{lll} f = g & & \text{or} \\ \exists f_0, f_1, f_2, \ldots, f_n \text{ s.t.} & f_0 = f & \\ & f_n = g & \\ & \forall i \in [1, n] : f_{i-1} \xrightarrow{\mathcal{G}} f_i & \end{array}$$

An example with $\mathcal{G}_0$:

$N\ V\ joe\ N \longrightarrow sam\ V\ joe\ N \longrightarrow\ \ sam\ V\ joe\ joe$ or

Derivation

Def. 11 (Derivation)

$$
f \xrightarrow{\mathcal{G}*} g \text{ if } \begin{array}{ll} f = g & \text{or} \\ \exists f_0, f_1, f_2, \ldots, f_n \text{ s.t. } \begin{array}{l} f_0 = f \\ f_n = g \\ \forall i \in [1, n] : f_{i-1} \xrightarrow{\mathcal{G}} f_i \end{array} & \end{array}
$$

An example with $\mathcal{G}_0$:

$$
\begin{array}{lll} N\ V\ joe\ N \longrightarrow sam\ V\ joe\ N \longrightarrow & sam\ V\ joe\ joe & \text{or} \\ & sam\ V\ joe\ sam & \text{or} \end{array}
$$

Derivation

Def. 11 (Derivation)

$$f \xrightarrow{\mathcal{G}*} g \text{ if } \begin{cases} f = g & \text{or} \\ \exists f_0, f_1, f_2, \ldots, f_n \text{ s.t. } \begin{cases} f_0 = f \\ f_n = g \\ \forall i \in [1, n] : f_{i-1} \xrightarrow{\mathcal{G}} f_i \end{cases} & \end{cases}$$

An example with $\mathcal{G}_0$:

$$N\ V\ joe\ N \longrightarrow sam\ V\ joe\ N \longrightarrow \begin{array}{ll} sam\ V\ joe\ joe & \text{or} \\ sam\ V\ joe\ sam & \text{or} \\ sam\ sleeps\ joe\ N & \text{or} \\ \ldots & \end{array}$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3 : & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$E \times E$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0\,|\,1\,|\,2\,|\,3\,|\,4\,|\,5\,|\,6\,|\,7\,|\,8\,|\,9
\end{array}
$$

An example with $\mathcal{G}_3$:

$E \times E \longrightarrow F \times E$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3 : & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$E \times E \longrightarrow F \times E \longrightarrow 3 \times E$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0\,|\,1\,|\,2\,|\,3\,|\,4\,|\,5\,|\,6\,|\,7\,|\,8\,|\,9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E)$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E) \longrightarrow 3 \times (E + E)$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3 : & E & \longrightarrow & E + E \\
 & & | & E \times E \\
 & & | & (\ E\ ) \\
 & & | & F \\
 & F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E) \longrightarrow 3 \times (E + E) \longrightarrow 3 \times (E + F)$$

Endpoint of a derivation

$$\begin{array}{lccl} \mathcal{G}_3: & E & \longrightarrow & E+E \\ & & | & E \times E \\ & & | & (\ E\ ) \\ & & | & F \\ & F & \longrightarrow & 0\,|\,1\,|\,2\,|\,3\,|\,4\,|\,5\,|\,6\,|\,7\,|\,8\,|\,9 \end{array}$$

An example with $\mathcal{G}_3$:

$$E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E) \longrightarrow 3 \times (E+E) \longrightarrow 3 \times (E+F) \longrightarrow 3 \times (E+4)$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
& & | & E \times E \\
& & | & (\ E\ ) \\
& & | & F \\
& F & \longrightarrow & 0 \,|\, 1 \,|\, 2 \,|\, 3 \,|\, 4 \,|\, 5 \,|\, 6 \,|\, 7 \,|\, 8 \,|\, 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$
E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E) \longrightarrow 3 \times (E + E) \longrightarrow
3 \times (E + F) \longrightarrow 3 \times (E + 4) \longrightarrow 3 \times (F + 4)
$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & \mid & E \times E \\
 & & \mid & (\ E\ ) \\
 & & \mid & F \\
 & F & \longrightarrow & 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$
E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E) \longrightarrow 3 \times (E + E) \longrightarrow 3 \times (E + F) \longrightarrow 3 \times (E + 4) \longrightarrow 3 \times (F + 4) \longrightarrow 3 \times (5 + 4)
$$

Endpoint of a derivation

$$
\begin{array}{llll}
\mathcal{G}_3: & E & \longrightarrow & E + E \\
 & & \mid & E \times E \\
 & & \mid & (\ E\ ) \\
 & & \mid & F \\
 & F & \longrightarrow & 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9
\end{array}
$$

An example with $\mathcal{G}_3$:

$$
E \times E \longrightarrow F \times E \longrightarrow 3 \times E \longrightarrow 3 \times (E) \longrightarrow 3 \times (E + E) \longrightarrow
3 \times (E + F) \longrightarrow 3 \times (E + 4) \longrightarrow 3 \times (F + 4) \longrightarrow 3 \times (5 + 4) \longrightarrow\!\!\!\mid
$$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$:

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S \rightarrow ()$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \to (S)S \to ()S \to ()$
as well as $((()))$, $()()()$, $((()()()))\ldots$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S \rightarrow ()$
as well as $((())), ()()(), ((()()()))\ldots$
but $)()(\notin L_{\mathcal{G}_2}$, even though the following is a licit derivation :

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \to (S)S \to ()S \to ()$
as well as $((())), ()()(), ((()()()))\ldots$
but $)()(\notin L_{\mathcal{G}_2}$, even though the following is a licit derivation :
$)S(\to$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S \rightarrow ()$
as well as $((()))$, $()()()$, $((()()()))$...
but $)()(\notin L_{\mathcal{G}_2}$, even though the following is a licit derivation :
$)S(\rightarrow)(S)S(\rightarrow$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_\mathcal{G}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_\mathcal{G} = L_\mathcal{G}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S \rightarrow ()$
as well as $((()))$, $()()()$, $((()()()))\ldots$
but $)()(\notin L_{\mathcal{G}_2}$, even though the following is a licit derivation :
$)S(\rightarrow)(S)S(\rightarrow)()S(\rightarrow$

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S \rightarrow ()$
as well as $((()))$, $()()()$, $((()()()))\ldots$
but $)()(\notin L_{\mathcal{G}_2}$, even though the following is a licit derivation :
$)S(\rightarrow)(S)S(\rightarrow)()S(\rightarrow)()($

Engendered language

Def. 12 (Language engendered by a word)

Let $f \in (\Sigma \cup N)^*$.

$L_{\mathcal{G}}(f) = \{g \in X^* / f \xrightarrow{\mathcal{G}*} g\}$

Def. 13 (Language engendered by a grammar)

The *language engendered by a grammar* $\mathcal{G}$ is the set of words of Σ^* derived from the axiom.

$L_{\mathcal{G}} = L_{\mathcal{G}}(S)$

For instance $() \in L_{\mathcal{G}_2}$: $S \rightarrow (S)S \rightarrow ()S \rightarrow ()$
as well as $((()))$, $()()()$, $((()()()))\ldots$
but $)()(\notin L_{\mathcal{G}_2}$, even though the following is a licit derivation :
$)S(\rightarrow)(S)S(\rightarrow)()S(\rightarrow)()($
for there is no way to arrive at $)S($ starting with S.

Example

$$G_4 = E \rightarrow E + T \mid T, T \rightarrow T \times F \mid F, F \rightarrow (E) \mid a$$

$a + a$, $a + (a \times a)$, ...

Proto-word

Def. 14 (Proto-word)

A proto-word (or proto-sentence) is a word on $(\Sigma \cup N)^* N (\Sigma \cup N)^*$ (that is, a word containing at least one letter of N) produced by a derivation from the axiom.

$$E \to E + T \to E + T * F \to T + T * F \to T + F * F \to T + a * F \to F + a * F \to a + a * F \to \cancel{a + a * a}$$

Multiple derivations

A given word may have several derivations:
$E \rightarrow E + E \rightarrow F + E \rightarrow F + F \rightarrow 3 + F \rightarrow 3 + 4$

Multiple derivations

A given word may have several derivations:

$E \to E + E \to F + E \to F + F \to 3 + F \to 3 + 4$

$E \to E + E \to E + F \to E + 4 \to F + 4 \to 3 + 4$

Multiple derivations

A given word may have several derivations:

$E \rightarrow E + E \rightarrow F + E \rightarrow F + F \rightarrow 3 + F \rightarrow 3 + 4$

$E \rightarrow E + E \rightarrow E + F \rightarrow E + 4 \rightarrow F + 4 \rightarrow 3 + 4$

... but if the grammar is not ambiguous, there is only one **left** derivation:

Multiple derivations

A given word may have several derivations:
$E \rightarrow E + E \rightarrow F + E \rightarrow F + F \rightarrow 3 + F \rightarrow 3 + 4$
$E \rightarrow E + E \rightarrow E + F \rightarrow E + 4 \rightarrow F + 4 \rightarrow 3 + 4$
… but if the grammar is not ambiguous, there is only one **left** derivation:
$\underline{E} \rightarrow \underline{E} + E \rightarrow \underline{F} + E \rightarrow 3 + \underline{E} \rightarrow 3 + \underline{F} \rightarrow 3 + 4$

Multiple derivations

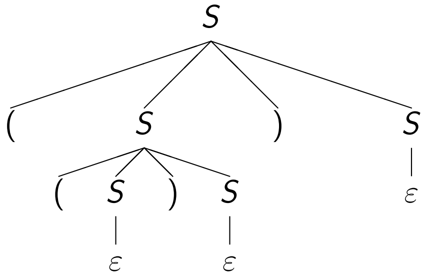
A given word may have several derivations:
$E \rightarrow E+E \rightarrow F+E \rightarrow F+F \rightarrow 3+F \rightarrow 3+4$
$E \rightarrow E+E \rightarrow E+F \rightarrow E+4 \rightarrow F+4 \rightarrow 3+4$
... but if the grammar is not ambiguous, there is only one **left** derivation:
$\underline{E} \rightarrow \underline{E}+E \rightarrow \underline{F}+E \rightarrow 3+\underline{E} \rightarrow 3+\underline{F} \rightarrow 3+4$

parsing: trying to find the/a left derivation (resp. right)

Derivation tree

For context-free languages, there is a way to represent the set of equivalent derivations, via a derivation tree which shows all the derivation independantly of their order.

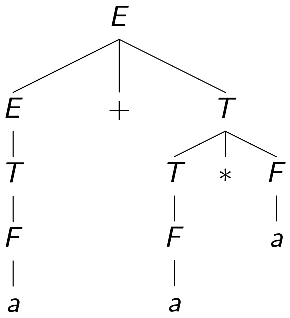
Grammar $\mathcal{G}_2$: $S \longrightarrow \varepsilon \mid (S)S$



$$S \rightarrow (S)S \rightarrow ((S)S)S \rightarrow ((S)S) \rightarrow ((S)) \rightarrow (())$$

Structural analysis

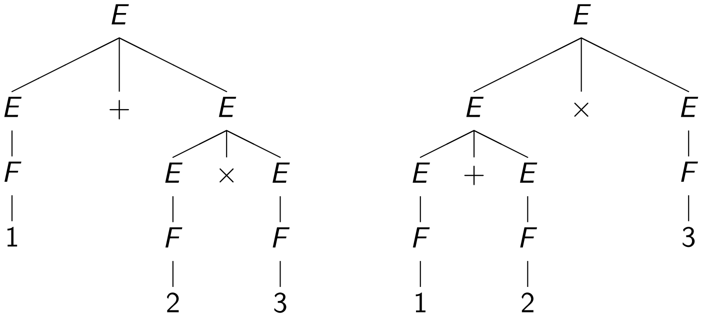
Syntactic trees are precious to give access to the semantics



Ambiguity

When a grammar can assign more than one derivation tree to a word $w \in L(G)$ (or more than one left derivation), the grammar is *ambiguous*.

For instance, $\mathcal{G}_3$ is ambiguous, since it can assign the two follwing trees to $1 + 2 \times 3$:



About ambiguity

- Ambiguity is not desirable for the semantics
- Useful artificial languages are rarely ambiguous
- There are context-free languages that are intrinsequely ambiguous (3)
- Natural languages are notoriously ambiguous...

(3) $\{a^n ba^m ba^p ba^q | (n \geqslant q \wedge m \geqslant p) \vee (n \geqslant m \wedge p \geqslant q)\}$

Comparison of grammars

- different languages generated $\Rightarrow$ different grammars
- same language generated by $\mathcal{G}$ and $\mathcal{G}'$:
  $\Rightarrow$ same weak generative power
- same language generated by $\mathcal{G}$ and $\mathcal{G}'$,
  and same structural decomposition :
  $\Rightarrow$ same strong generative power

References I

Bar-Hillel, Yehoshua, Perles, Micha, & Shamir, Eliahu. 1961. On formal properties of simple phrase structure grammars. *STUF-Language Typology and Universals*, 14(1-4), 143–172.

Chomsky, Noam. 1957. *Syntactic Structures*. Den Haag: Mouton & Co.

Gazdar, Gerald, & Pullum, Geoffrey K. 1985 (May). *Computationally Relevant Properties of Natural Languages and Their Grammars*. Tech. rept. Center for the Study of Language and Information, Leland Stanford Junior University.

Gibson, Edward, & Thomas, James. 1997. The Complexity of Nested Structures in English: Evidence for the Syntactic Prediction Locality Theory of Linguistic Complexity. *Unpublished manuscript, Massachusetts Institute of Technology*.

Joshi, Aravind K. 1985. *Tree Adjoining Grammars: How Much Context-Sensitivity is Required to Provide Reasonable Structural Descriptions?* Tech. rept. Department of Computer and Information Science, University of Pennsylvania.

Langendoen, D Terence, & Postal, Paul Martin. 1984. *The vastness of natural languages*. Basil Blackwell Oxford.

Mannell, Robert. 1999. *Infinite number of sentences*. part of a set of class notes on the Internet. `http://clas.mq.edu.au/speech/infinite_sentences/`.

Schieber, Stuart M. 1985. Evidence against the Context-Freeness of Natural Language. *Linguistics and Philosophy*, 8(3), 333–343.

Stabler, Edward P. 2011. Computational perspectives on minimalism. *Oxford handbook of linguistic minimalism*, 617–643.