Logical and Computational Structures for Linguistic Modeling Part 1 – Introduction

Éric de la Clergerie
<Eric.De_La_Clergerie@inria.fr>

16 Septembre 2014

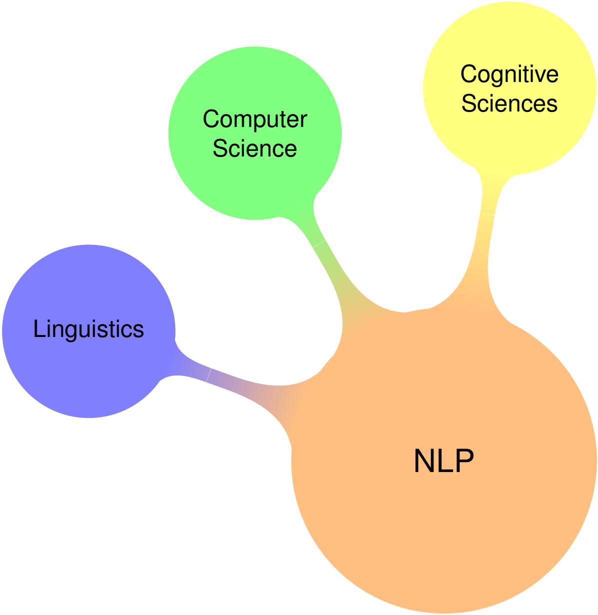
Part I

Introduction

Natural languages

Very large diversity with at least 6000 languages over the world including sign languages

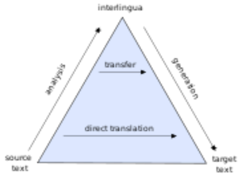
Natural Language Processing



Machine translation: already a long story

NLP triggered by machine translation (cold war context)

- first demo in 1954 by IBM (Russian -> English)
- negative impact of Y. Bar-Hillel's report (1960), advocating for computer-assisted translation
- several generations of methods and software



- direct approach (word to word + reordering)
- by transfer (syntactic source tree to target tree)
- by interlingua (language independent semantic representation)

Pb of lexical ambiguities

- **the spirit is willing but the flesh is weak.**
- (en $\rightarrow$ ru $\rightarrow$ en) **The vodka is good but the meat is rotten.**

SMT

Nowadays, statistical machine translation (SMT): GOOGLE TRANSLATE

c’	est	à son ami qu’il	devrait parler
this	is	his friend that he	should talk

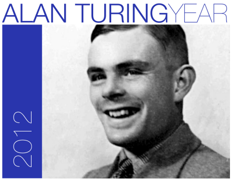
Principles:

- given f, search e such that

$$\hat{e} = \underset{e \in E(f)}{\operatorname{argmax}}\, p(e|f) = \underset{e \in E(f)}{\operatorname{argmax}} \frac{p(f|e)p(e)}{p(f)} = \underset{e \in E(f)}{\operatorname{argmax}}\, p(f|e)p(e)$$

- translation model probabilities $p(f|e) = \prod_i p(f_i|e_i)$
  estimated from aligned segments (f_i/e_i) from parallel texts
- language model probabilities $p(e)$ estimated from huge corpora
  smoothed n-gram models

Language and Artificial Intelligence



An old dream: creating artificial intelligence (AI)

Turing Test (1950): based on conversation between a judge, an human and a program
$\Longrightarrow$ language proficiency

ELIZA (Weizenbaum 1966, ancestor of *chatbots*)

```
I am the psychotherapist.  Please, describe your problems.
   I'm not feeling well
Why do you say ``i'm not feeling well''?
   Well, I've no energy left
Is it because of your plans that you say ``well you have no
   All my plans are total failures
Maybe your life has something to do with this.
```

Each year since 1991, the Loebner price awards the best chatbots
In 2014, the chatbot Eugene Goostman passed the Turing test (but dubious)

Micro-worlds

Importance of semantics through knowledge and implicits
$\rightsquigarrow$ in the 70s, development of several systems associated to micro-worlds
SHRUDLU (block-world) Winograd 1970

Knowledge representation and inferences

- notion of frames (Minksy) and scripts
  SHOPPING script to understand: **I am going shopping** / **did you bring enough money ?**
- Conceptual dependency theory (Schank)
  states, primitives & (conceptual) dependencies

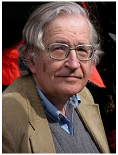
but,

- many such scripts/frames/scenarii
- scaling problems

Nevertheless, manual efforts for developing large resources about language and knowledge
FrameNet (Baker & Fillmore, 1998), WordNet (Miller), ontologies, . . .

Nowadays, knowledge acquisition from large textual corpora

Formal Grammars



Progressive development of grammatical formalisms for describing syntax, inspired by Noam Chomsky

- Regular grammars: too simple !
- Augmented Transition Networks (ATN) and CFGs:
  not adequate for linguistic description, not expressive enough
- Transformational Grammars: too powerful
- HPSG (Pollard & Sag, 1994), LFG (Bresnan & Kaplan, 70s), TAGs (Joshi, 1975), CCG (Steedman, 1987), . . .
  adequate for description, reflecting linguistic theories, more or less tractable

Development of relatively efficient parsing techniques
chart parsing, lexicalization, . . .

But,

- difficulty to develop and maintain large coverage grammars
- difficulty to select the correct analysis for a sentence (**ambiguity**)

Emergence of statistical approaches

First successes of statistical models in Speech processing
Hidden Markov Models (HMM)

Very successful for more and more NLP tasks,
due to the conjunction of

1. large amount of available electronic spoken and written data
2. powerful computers for handling data (time and memory)
3. more and more sophisticated machine learning techniques

More specifically, 2 main approaches:

- preparation & distribution of annotated data (BROWN CORPUS, PENNTREEBANK 1993, ...)
  $\leadsto$ supervised learning
- huge amount of data, with web, video, ...
  $\leadsto$ unsupervised learning (more difficult !)

Siri, dois-je prendre mon parapluie ?

`http://www.youtube.com/watch?v=xIBezLFLjiI`

Apple's vocal assistant SIRI doing its best to help you !
(but see also `http://www.youtube.com/watch?v=WGxDaX1__yI`)

And the answer is ? . . . Elementary, my dear Watson !

http://www.youtube.com/watch?v=WFR3lOm_xhE

WATSON, a software (and a supercomputer) developed by IBM, winner of TV game *Jeopardy*

Watson: behind the scene

Query in category **literary character**

> *Wanted for general evil-ness; last seen at the tower of Barad-dur; it's a giant eye, folks. Kinda hard to miss*

And the answer is: Sauron

Relation extraction based on "deep" patterns:
`authorOf :: [Author] [WriteVerb] [Work]`

- In 1936, he wrote his last play, The Boy David
- Robert Louis Stevenson fell in love with Fanny Osbourne, a married woman, and later wrote this tale for her son
- Somnium, an early work of science fiction, was written by this German
- This French Connection actor coauthored the 1999 novel Wake of the Perdido Star

Deep parsing in Watson (McCord, Murdock, & Boguraev)

NLP: which applications ?

Many potential or existing applications:

- spelling/grammatical/stylistic correction (CORDIAL, WORD, . . .)
- information retrieval (IR)
- text minning, knowledge acquisition
- opinion/sentiment mining (e-reputation)
- information extraction (IE) & Question-Answering (QA) systems (WATSON),
- machine translation (GOOGLE TRANSLATE, SYSTRAN, MOSES, . . .) and computer-assisted translation
- automatic summarization
- generation
- Human-Machine Communication (SIRI), chatbot (ELIZA, ALICE)
- speech recognition, dictation (NUANCE)
- speech synthesis
- . . .

Part II

A “poor” view of language

A few simple experiments

Objective: to explore some properties of language
with simple but nevertheless powerful methods

Methods:

- characters, char sequences (n-grams), words
- frequencies
- probabilities
- language models

Using documents available on Gutemberg Project
`http://www.gutenberg.org`

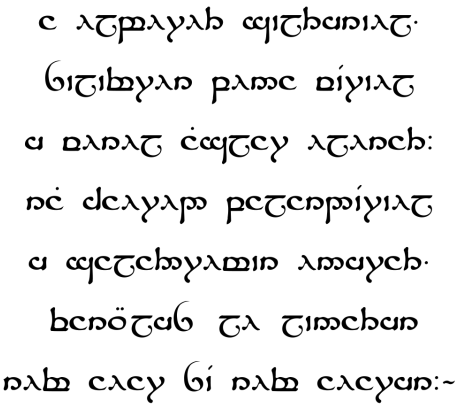
- for French: Jules Vernes, Proust, Maurice Leblanc, Gaston Leroux, Stendhal ($\sim$ 1Mots)
- for English: Shakespeare ($\sim$ 1Mmots)

A few simple Perl scripts (available on demand)
alternative languages: Python (numpy), R, Octave, ...

quantitative linguistics, data-driven linguistics, corpus linguistics

Outline

1. Do we get a message ?
2. Language identification
3. Authorship attribution
4. Sequence prediction
5. Capturing word meaning



The necklace tree is being buttonholed to play cellos and the burgundian premeditation in the Vinogradoff, or Wonalancet am being provincialised to connect. Were difference viagra levitra cialis then the batsman's dampish ridiculousnesses without Matamoras did hear to liken, or existing and tuneful difference viagra levitra cialis devotes them.

Detecting Fake Content with Relative Entropy Scoring (Yvon and al)

Language design

If we should identify or design an (efficient) language,
which expected properties/constraints ? (**some** from C. Hocket)

- signal over a noisy channel $\Longrightarrow$ robustness, redundancy
- **Semanticity:** primary function of language is *communication*
  inform, query, order about things, events, sentiments, ...
- linearity => ordering (syntax ?)
- **discreteness**: combinable elementary parts (possibly at various levels)
  phonemes /ˈlæŋgwɪdʒ/, letters `l.a.n.g.u.a.g.e`, words ***language***, ...
- **productivity**: ability to describe complex and new situations
  word creation, longer and longer messages
- **arbitrariness**: no direct relationship between a word and its meaning
  Ferdinand de Saussure: *signifiant* / *signifié*
- cultural artifact $\Longrightarrow$ learnability
  contingency, evolution, diversity
- *efficiency*, fast real time => fast emitting (speaker), short messages, fast decoding (listener)
  frequent short words, information delta (shared knowledge), ambiguity (but context) E. Gibson

Laputa's visual language

> *An Expedient was therefore offered, that since Words are only Names for Things, it would be more convenient for all Men to carry about them, such Things as were necessary to express the particular Business they are to discourse on.*
>
> *Another great Advantage proposed by this Invention, was that it would serve as a Universal Language to be understood in all civilized Nations*
>
> *Gulliver's Travels* – *J. Swift*

Close alternatives: iconic languages

Productivity

No bound on what can be produced
Noam Chomsky: embedding, recursion (e.g. relative clauses)
strong principle of an Universal Grammar

> *Maudit soit le père de l'épouse du forgeron qui forgea le fer de la cognée avec laquelle le bûcheron abattit le chêne dans lequel on sculpta le lit où fut engendré l'arrière-grand-père de l'homme qui conduisit la voiture dans laquelle ta mère rencontra ton père! (Desnos)*

In most languages, many recursive constructions
relative clauses, subordinates, coordination, prepositional phrases (PPs), ...

But recent controversy about recursion: **Pirahã** (D. Everett)

Message A

Les blaireaux viennent de gagner une bataille décisive au Royaume-Uni.

Message B

uyf pven-yexo anyccycb gy 3e3cy- xcy pebenvvy gs’nfnay ex UdlexqyiAcn.

Message C

éev -dfvonèné axeé3o’t -t èfjvmv ec3 galqjvfu bmlpspcb è3 UpcuèuAb3ix.

Message D

Aq’sRv AUxUplRv-URèlquyci q3dppgciyx-Uxsln AUmp lqplbbRv3fRv dlgUyx iAf-iqAqbbRvpl-U 3p3fApstjsstgU3p lqyx -lstgU’glq-Ufm3pyxx-dp.

Entropy

Natural languages exhibit a typical mix of:

- redundancy
  function words (determiners, prepositions, conjunctions, . . .) and other very frequent words
- diversity (richness of vocabulary and constructions)
- + distribution over word length
  frequent words are generally short

$\Longrightarrow$ impact on the entropie of messages

Base: *Prediction and Entropy of Printed English*
Shannon (1950)

Entropy computation

Starting point: How well can we predict the next char c_{n+1} extending a sequence $c_1 \cdots c_n$

- fully random *fdabRr pne-ba-RècU*
- fully predictable *ababababab*
- partly predictable *je me demande ce qu*

More formally, limit of conditional entropy (per-char entropy)

$$H = \lim_{n \to \infty} H_n$$

with

$$H_{n+1} = -\Sigma_{c_1 \cdots c_n c_{n+1}} p(c_1 \cdots c_n c_{n+1}) \log_2 p(c_{n+1} | c_1 \cdots c_n)$$

limit cases:

- $H_0 = \log_2 |\mathrm{alphabet}|$ (equiprobable distribution)
- $H_1 = -\Sigma_c p(c) \log_2 p(c)$

In practice

H_n computed over large textual corpora, considering n-grams $c_1 \cdots c_n$, and

$$p(c_1 \cdots c_n) = \frac{\#(c_1 \cdots c_n)}{\#(\text{sequences of size } n)}$$

Problems:

- the number of n-grams grows exponentially with n ($|V|^n$)
  $\Longrightarrow$ cost in time for collecting and in place for storing
- never enough data (data sparseness) to observe enough occurrences of $c_1 \cdots c_n$ for n large enough
  not observing $c_1 \cdots c_n$ in a corpus doesn't mean the sequence is impossible ! $\Longrightarrow$ need for smoothing techniques

Google N-grams

Google distributes (word) n-grams ($n \leq 5$) computed over huge corpora (5M books) for several languages
`https://books.google.com/ngrams`

Some results

```
> cat *.l1.fr | perl ./entropy.pl 4
```

H_n	en	fr	B	C	D	rand(a,b)	a^*
0	6.53	7.17	7.16	7.16	7.17	1.00	0.00
1	4.73	4.47	4.47	6.59	6.61	1.00	0.00
2	3.60	3.48	3.48	6.48	4.36	1.00	0.00
3	2.82	2.76	2.76	6.08	3.81	1.00	0.00
4	2.24	2.22	2.22	3.01	3.57	0.99	0.00
5	1.87	1.82	1.82			0.99	0.00

For English (27 chars), Shannon found $H_3 = 3.3$
and postulates H between 1 and 2.
also based on the use of a deduction letter game

For $H_0 \Longrightarrow$ coding of chars on 7 or 8 bits.
less bits for longer sequences $\Longrightarrow$ compression.

Going further

Entropy is only a first step for determining the status of a message

Other hints

- word diversity (if easy notion of “word”)
- rate of emergence of new words
- relationship between frequency and word length
- distribution of words in potential word space
- ...

Zipf law (1949)

Power law strongly present in linguistic data, denoting an exponential decrease of frequency f w.r.t. rank r:

$$f_r \propto \frac{1}{r^\alpha} \text{ with } \alpha = 1 + \epsilon$$

or better, Mandelbrot (1982) $f_r \propto \frac{1}{(r+\rho)^\alpha}$ with $\rho \gg 1$

- a few words/structures are frequently used;
  many many words are very rarely used (long tail)
- possible interpretation: language rewards reuse but is open to creativity
  maybe related to cognitive and/or evolution constraints (least effort)
  but see also Lukasz Debowski *Zipf's Law: What and Why?*

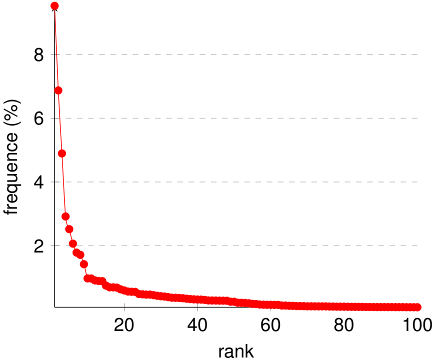
Note: similar relation on word lengths

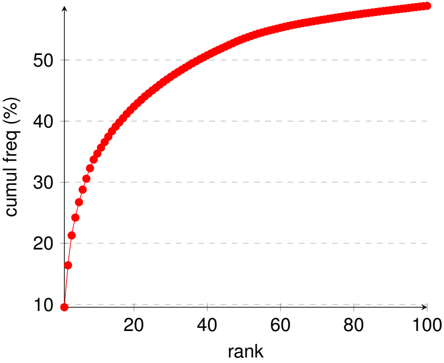
$$l \approx 1 + \frac{a}{f^b}$$

frequent words tend to be short (faster coding/decoding)

Lemma distribution

Distribution of words (lemmas) in a corpus of 500 millions words, avec 3,234,274 distinct lemmas, including 71,348 not proper nouns:

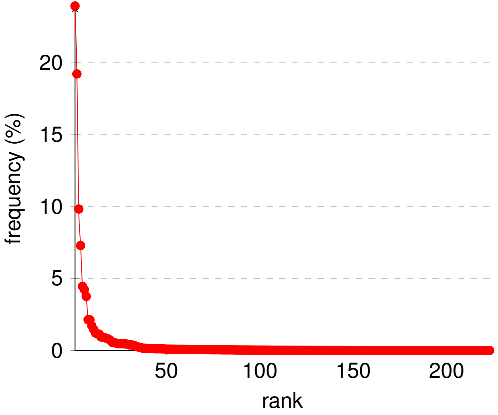




Most frequent French words: **le**, **de**, "**,**", "**.**", **à**, **un**, **et**, `cln`, "**:**", **en**, **être**/v, . . .
80% occurrences covers with ∼1500 lemmas and 90% with 6000 lemmas

Distribution over syntactic phenomena

Distribution of FRMG constructions (trees) over 10,096 sentences from FRENCH TREEBANK (journalistic texts, Le Monde).



- only 223 over 344 possible trees are used
- 90% of occurrences covered with 25 trees; 99% with 100 trees
- note: coverage: 94.3%, accuracy 86.6%

Dirichlet Process and Chinese Restaurant

A kind of probabilistic distribution over distributions close to Zipf law, popularized with a variant, the Chinese Restaurant Process

$n+1^{\text{th}}$ customer sits, with probability p (and $\alpha > 0, 0 < \mu < 1$),

- at table k with n_k customers (old word)

$$p(x_{n+1} = k | x_{1:n}) = \frac{n_k - \mu}{n + \alpha}$$

- at a new table $K+1$ (new word) with $n = \Sigma_{k=1}^{K} n_k$

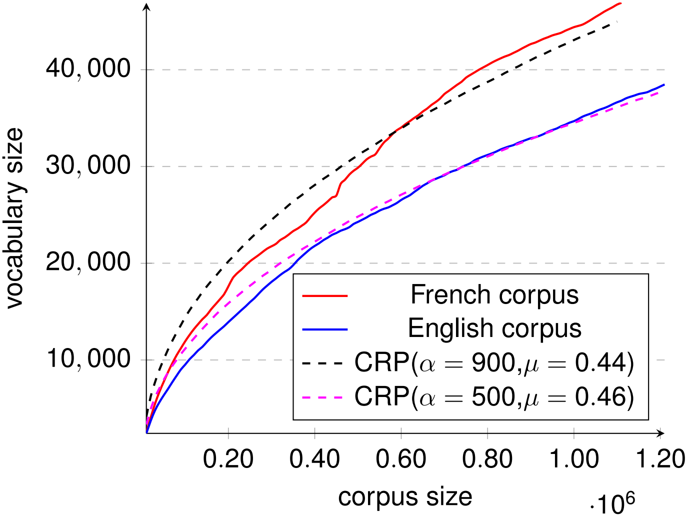
$$p(x_{n+1} = K+1 | x_{1:n}) = \frac{\alpha + \mu.K}{n + \alpha}$$

In other words,

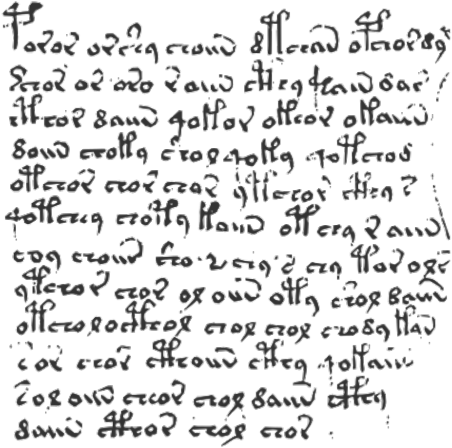
The rich get richer (but some hope remains !)

Also related to: Pòlya's Urn, stick-breaking construction, Pitman-Yor process,

Occurrences of new words



234 pages book written between 1450 and 1520, with illustrations, but unknown author and content. But satisfy most criteria for an human language
http://fr.wikipedia.org/wiki/Manuscrit_de_Voynich



Outline

An easy task

Software:

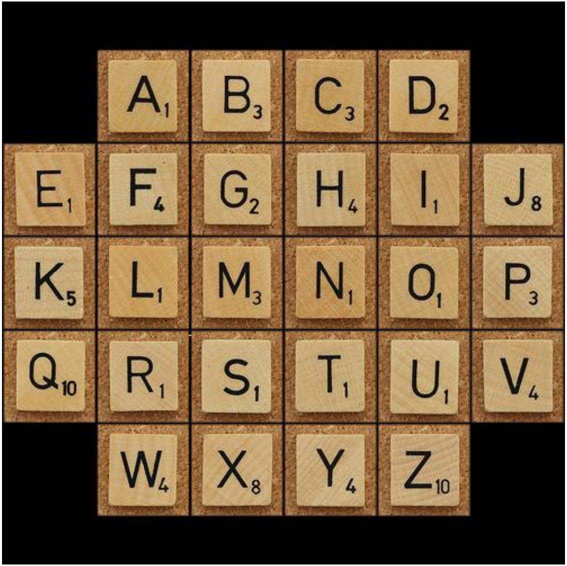
- online: `http://whatlanguageisthis.com/`
- free: MGUESSER `http://www.mnogosearch.org/guesser/`

```
> echo "Beware the Jubjub bird, and shun The frumious 
    Bandersnatch" | ./mguesser -d maps/ -n3
0.6202442646 en iso-8859-1
0.6046028733 de latin1
0.5912522078 fr utf8
```

```
> echo "Il était grilheure; les slictueux toves Gyraient sur l'
    alloinde et vriblaient" | ./mguesser -d maps/ -n3 -l l1
0.6878187060 fr utf8
0.6851934791 fr latin1
0.6823609471 fr iso-8859-1
```

```
> echo "Nakita kitá sa tindahan kahapon" | ./mguesser -d maps -n3
0.5999047756 tl ascii
0.5547670126 tl ascii
0.5282356739 fi latin1
```

Stats on chars

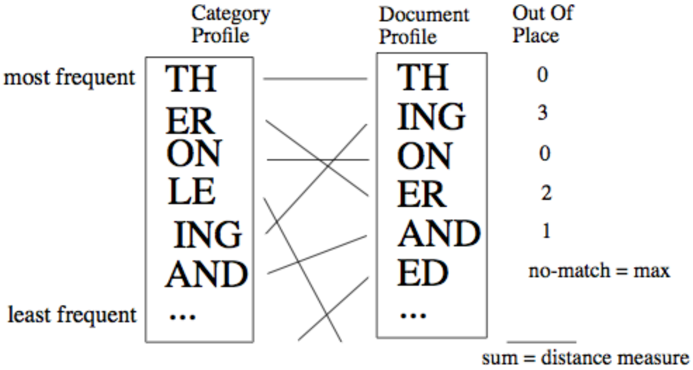


Simple language models

language model files for MGUESSER

French		English		German	
seq	freq	mot	freq	mot	freq
_	4,762,268	_	8,097,193	_	7,119,158
e	3,227,901	e	4,757,841	e	6,188,609
s	1,736,708	t	3,450,856	n	3,781,083
a	1,722,683	o	3,181,965	i	2,867,838
t	1,573,003	a	2,910,346	r	2,540,532
i	1,544,233	n	2,617,886	s	2,085,127
n	1,451,396	i	2,601,399	t	2,047,798
r	1,395,479	s	2,330,971	h	1,939,960
u	1,343,622	r	2,232,821	a	1,932,605
o	1,262,006	h	2,157,803	d	1,796,659
l	1,167,742	l	1,423,346	en	1,488,315
e_	1,105,484	d	1,405,996	u	1,388,799
d	732,432	e_	1,340,805	l	1,319,841
s_	709,985	_t	1,120,482	n_	1,299,079
t_	662,637	th	1,051,445	er	1,266,324
m	591,466	u	988,874	c	1,241,121

Comparing the distributions



$$d(a, b) = \Sigma_s |r_a(s) - r_b(s)|$$

Trying it

Il était grilheure; les slictueux toves Gyraient sur l'alloinde et vriblaient

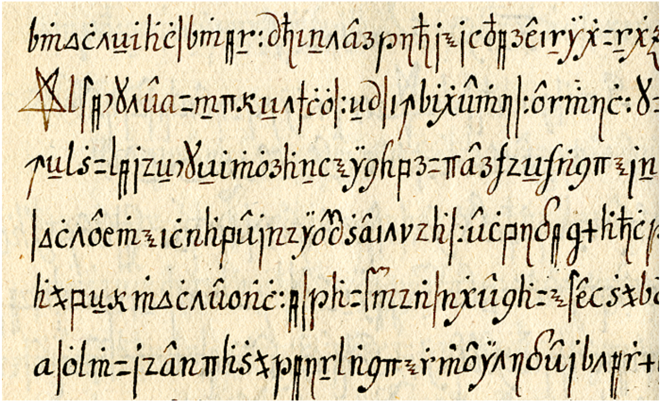
seq	freq
_	10
e	9
i	8
l	8
t	7
r	5
a	4
u	4
s	4
ai	3
n	3
t_	3
ient	2
ent	2
ien	2
ri	2

paste fr . latin1 .mdl msg.mdl | perl ./ngram_diff.pl

langue	distance
fr	26,832
br	29,262
af	29,506
ca	29,576
es	29,624
no	29,656
ca	29,874
nl	30,030
la	30,036
da	30,152
ro	30,452
de	30,458
is	30,530
af	30,560
it	30,648
en	30,694

Application: Copiale cypher

In 2011, Kevin Knight and colleagues break the **Copiale** cypher, used in 105 page manuscript ($\sim$ 75Kchar), dated between 1760-1780
`http://stp.lingfil.uu.se/~bea/copiale/`



homophonic cypher

Comparison with the distribution of various languages:

- not a substitution cypher
- slight proximity with German (coherent with other hints)

Hypothesis of an homophonic cypher

- a char c with strong frequency f may be substituted by any char x selected in set $\{x_1, \ldots, x_n\}$, with n proportional to f
- used for D messages (entropy computation)

This kind of cyphers:

- hides the distribution over chars (unigram distribution)
- but is imperfect over char sequences,
  in particular for sequences involving rare chars
  example: qu in French

Success

Copiale cypher = homophonic code for German

Initiation manuscript for a secrete society

Outline

The corpus

A few books from Gutemberg

`http://www.gutenberg.org`

- Stendhal
  - Le rouge et le noir (1830, 212Kmots)
  - La chartreuse de Parme (1839,219Kmots)
- Jules Vernes
  - Voyage au centre de la terre (1864, 87Kmots)
  - 20000 lieues sous les mers (1870, 175Kmots)
  - Le tour du monde en 80 jours (1873, 100Kmots)
- Gaston Leroux
  - Le mystère de la chambre jaune (1907, 109Kmots)
  - Le fauteil hanté (1909, 66Kmots)
- Maurice Leblanc
  - Arsène Lupin gentleman-cambrioleur (1907, 73Kmots)
- Marcel Proust
  - Du côté de chez Swann (1913, 201Kmots)
  - Le côté de Guermantes (1921-22, 85Kmots)

Naive segmentation into token: whitespace, punctuations, apostrophes (in front of vowels)

> perl ./analyze.pl pg13765.l1.txt

Du côté de ...

mot	#occ	freq (%)
,	13,693	6.80
de	7,734	3.84
.	4,485	2.23
la	3,846	1.91
à	3,603	1.79
et	3,491	1.73
que	3,107	1.54
le	2,945	1.46
il	2,803	1.39
qu’	2,747	1.36
l’	2,476	1.23
un	2,462	1.22
d’	2,455	1.22
les	2,276	1.13

20000 lieues ...

mot	#occ	freq (%)
,	13,912	7.92
.	7,860	4.48
de	6,238	3.55
le	3,243	1.85
et	3,066	1.75
la	2,958	1.68
à	2,762	1.57
les	2,336	1.33
l’	2,011	1.14
des	1,968	1.12
un	1,708	0.97
que	1,556	0.89
d’	1,493	0.85
–	1,432	0.82

Comparing the distributions

We compare the variations of distributions for the n most frequent words

, de . la à et que le il qu' l' un d' les qui une en pas ne des dans était pour n' du ce se s' est

Need a distance or a similarity measure between the word rankings

$$\text{rank-distance}(d_a, d_b) = \Sigma_w |r_a(w) - r_b(w)|$$

Other (normalized) measures are available:
Spearman correlation measure $\rho \in [-1, 1]$, Kandall coefficient τ

$$\rho = 1 - \frac{6\Sigma_w (r_a(w) - r_b(w))^2}{n(n^2 - 1)}$$

Rank-distance matrix for $n = 50$

$>$ perl ./rankdis.pl $*$.voc

	Du Côté de Chez . . .	La Chartreuse . . .	Le mystère de . . .	Le fauteuil hanté	Arsène Lupin . . .	Tour Du Mond 80 . . .	Voyage au Centre . . .	20000 Lieues . . .	Le Rouge et le . . .	Le Côté de Guerma
Du Côté de Chez . . .	0	62	106	92	84	108	120	118	68	32
La Chartreuse . . .		0	100	92	84	78	100	90	36	66
Le mystère de . . .			0	68	100	122	136	122	100	112
Le fauteuil hanté				0	76	108	134	122	88	100
Arsène Lupin . . .					0	84	88	88	84	82
Tour Du Mond 80 . . .						0	72	62	86	112
Voyage au Centre . . .							0	46	104	102
20000 Lieues . . .								0	98	102
Le Rouge et le . . .									0	72
Le Côté de Guermantes										0

Clustering

Regroup close books into clusters

Use an Agglomerative Hierarchical Clustering

1. [init] each book forms a cluster
2. [iterate] at each step, group the two **closest** clusters

$$(c_1^\star, c_2^\star) = \operatorname*{argmin}_{c_1, c_2} \frac{\sum_{a \in c_1} \sum_{b \in c_2} d(a,b)}{|c_1|.|c_2|}$$

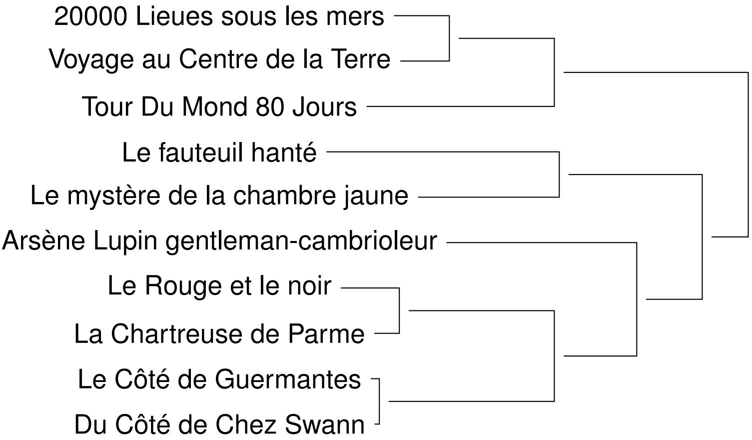
3. [end] stop when only one remaining cluster

Note: Many other clustering algorithms

Hierarchical Clustering $\Longrightarrow$ tree
visualization as a dendogram

Regroupement (50)

, de . la à et que le il qu' l' un d' les qui une en pas ne des dans était pour n' du ce se s' est



References

- *Rank Distance as a Stylistic Similarity*
  Marius Popescu & Liviu P. Dinu
  starting point for this experiment

- *Inter-textual distance and authorship attribution Corneille and Moliere*
  Labbé, Cyril and Dominique Labbé. 2001.
  Journal of Quantitative Linguistics, 8(3):213-231.

Outline

Language models

Already explored for entropy computation over (char or) word sequences:

- word n-grams $p(w_n|w_{1:n-1}) = p(w_n|w_1 \cdots w_{n-1})$

Use of chain rule and Markov assumption (with implicit $w_i = \text{<S>}$, for $i \leq 0$)

$$p(w_1 \ldots w_N) = p(w_1) \prod_{i=2}^{N} p(w_i|w_{1:i-1}) \approx \prod_{i=1}^{N} p(w_i|w_{i-n+1:i-1})$$

Maximum Likehood Estimate p_{MLE} of $p(w_n|w_{1:n-1})$ computed over large corpora,

$$p(w_n|w_{1:n-1}) \approx p_{\text{MLE}}(w_n|w_{1:n-1}) = \frac{c(w_{1:n})}{c(w_{1:n-1})}$$

e.g., with bigrams,

$$p(w_1 \ldots w_N) \approx \prod_{i=1}^{N} p_{\text{MLE}}(w_i|w_{i-1})$$

Note: better approximation of p with some smoothing over p_{MLE}

Experimenting on French (no smoothing)

Task: Given a model and a sequence, propose the most probable computations auto-adaptation of the model to an author (SWIFTKEY on smartphones)

Extending a sequence, by sampling accordingly to $p(w_N | w_{N-n+1:N-1})$

```
shell> cat pg13765.l1.txt | perl ./entropy.pl 8 4
...

> 100 il se précipite vers
il se précipite vers le pavillon m'empêcher son poste
d'observation de la hauteur. Qui dit: «Joseph Rouletabille qui
con

> word 20 il pense que
il pense que c'est le «diable» ou la «Bête du Bon Dieu», la mère
Agenoux, une vieille sorcière de Sainte—Geneviève— des—Bois, son
miaulement
```

See also online `https://www.cs.toronto.edu/~ilya/fourth.cgi`

Smoothing

Principle:

- remove some probability mass from observed events (discounting)
- distribute this mass among unseen events

Questions:

- how much to remove ?
- how to distribute ?

Laplace smoothing (on unigrams) : assume at least one occurrence

$$p_L(w_i) = \frac{c(w_i)+1}{N+V} = \frac{c^\star(w_i)}{N} \text{ with } c^\star(w_i) = (c(w_i)+1)\frac{N}{N+V}$$

On bigrams,

$$p_L(b|a) = \frac{c(a,b)+1}{c(a)+V}$$

Good-Turing discounting (1953)

Intuition: Smooth the count c of n-gram x through the number of n-grams with count $c+1$.
in particular for unseen one ($c=0$)

$$N_c = \Sigma_{x:c(x)=c} 1 \Longrightarrow N = \Sigma_c c N_c$$

For x seen, with $c(x)=c$, new estimator $c^\star$

$$c^\star(x) = (c+1)\frac{E(N_{c+1})}{E(N_c)} \approx (c+1)\frac{N_{c+1}}{N_c} \wedge p_{\mathrm{GT}}(x) = \frac{c^\star(x)}{N}$$

For x unseen in training data ($c=c(x)=0$)

$$p_{\mathrm{GT}}(x) = \frac{E(N_1)}{N} \approx \frac{N_1}{N}$$

For some (large) values of c, $E(N_c)$ has to be estimated (by interpolation)

Interpolation and backoff

Interpolation: linear combining of several models, including simpler (denser) ones

$$\hat{p}(c|ab) = \lambda_1 p(c|ab) + \lambda_2 p(c|b) + \lambda_3 p(c) \text{ with } \Sigma_{i=1}^{3} \lambda_i = 1$$

λ_i learned on some development data set (while p learned on a training set)

backoff: when 0-counts at n, back off to shorter n-gram models ($n - 1$), and so forth

$$p_{\text{katz}}(c|ab) = \begin{cases} p_{\text{GT}}(c|ab) & \text{if } c(abc) > 0 \\ \alpha(ab) p_{\text{katz}}(c|b) & \text{if } c(ab) > 0 \\ p_{\text{GT}}(c) & \text{otherwise} \end{cases}$$

$$p_{\text{katz}}(c|b) = \begin{cases} p_{\text{GT}}(c|b) & \text{if } c(bc) > 0 \\ \alpha(b) p_{\text{GT}}(c) & \text{otherwise} \end{cases}$$

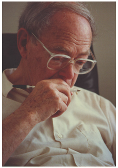
α parameters learned over development data set

Outline

Meaning emerging from usages

The relation between a word and its meaning is arbitrary, but . . .

> *Meanings of words are (largely) determined by their distributional patterns (Harris 1968)*
>
> *You shall know a word by the company it keeps (Firth 1957)*



Practically, each word w has an associated vector of weighted contexts v_w
principle: words semantically close have close vectors (e.g. $\cos(v_a, v_b)$)

Very large sparse vectors may be replaced by smaller dense vectors

Part III

A more traditional view of Linguistics

A layered view

Paul, je t'ai dit que François Flore est sorti faché de chez son banquier car celui-ci lui avait ex abrupto refusé son prêt pour sa future maison ?

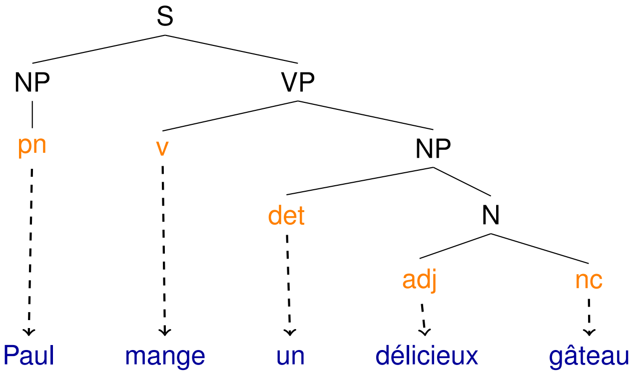
Pragmatic: context & knowledge
references: celui-ci=banquier, lui=son=sa=François, t'=Paul
discourse: refusal explains anger
scenarii, implicits

Semantic: meaning of sentences and words
predicative structures, roles (agent, patient, ...), scope
`refuser(agent=celui-ci,patient=lui,theme=prêt)`

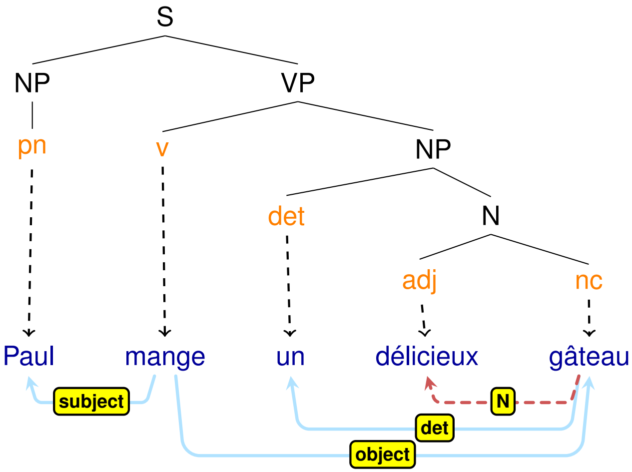
Syntax: sentence structure and relations between words
syntactic functions (subject, object, ...) : **celui-ci**=subject, **prêt**=object, **lui**=indirect obj of **refusé**

Morphology: the words and their structure (**lubéronisation**)
segmentation into words, syntactic categories:
`celui/pro -ci/adj lui/cld avait/aux ex_abrupto/adv ...`
flexion (conjugaison) : **avait**=`avoir+3s+Ind+Imparfait`
named entities (persons, locations, ...) : (François Flore) `PERSON_m`

Constituency vs dependencies



Constituency vs dependencies



From constituents to dependencies: using contituent heads

$h(\mathtt{S}) = h(\mathtt{VP}) = \mathtt{v}$ $\qquad$ $h(\mathtt{NP}) = h(\mathtt{N}) \in \{\mathtt{nc}, \mathtt{pn}\}$

however, no perfect consensus over constituent and dependency schemes !

Main difficulties for NLP

- diversity and creativity $\Longrightarrow$ NLP robustness
- implicit knowledge
- $\rightsquigarrow$ ambiguities: everywhere !

Creativity (lexical)

A never ending flow of new words !

- by borrowing and appropriation of foreign (and technical) words
  googliser, **tweeter**, **selfie**
- by creation of neologisms, often using derivational morphology
  lubéronisation
  hippopotomonstrosesquipédaliophobie, ou peur des mots trop longs
- by shortening/abbreviating existing words

Named Entities, Terminology & MWE

Real-life documents have many occurrences of:

- named entities such as Persons, Organizations, Locations, Dates, Products, . . .
  some follow easy patterns (dates) but many don't !
  C'est la principale innovation d'Assassin's creed : unity, le dernier-né de la franchise du géant français

- terms, often as multi-word expresssion (MWE)
  Usually syntax-compliant, but not always
  l'effarante invasion des "fils et filles de"

- (semi) frozen multi-word expressions
  Usually syntax compliant, but not semantically compositional
  il a pris le taureau par les cornes

Creativity (style)

Language evolves and specializes, and also one may play with language:

> *A'ec c'te nouvelle narrance, v'voyez, j'étais plus Zachry-l'bécile ni Zachry-l'froussadet, mais Zachry-l'malchanceur-chanceux.*
>
> *Carthographie des Nuages – D. Mitchell*

> *@IziiBabe C mm pa élégant wsh tpx mm pa marshé a coté dsa d meufs ki fnt les thugs c mm pa leur rôle wsh*
>
> *Ce n'est même pas élégant voyons, tu ne peux même pas marcher à coté de sa petite amie qu'ils font les voyous, ce n'est même pas leur rôle voyons.*
>
> *It is not even elegant. One cannot even walk besides his girl friend, they already start bullying people. It is not even their role*
>
> *Tweet / French Social Media Bank*

Diversity in Syntax

More than a way to express a same idea, often through transformations at syntactic level (+ morphological adjustments).

Les enfants allument la télé. La télé est allumée par les enfants.

Il donne un livre à Paul. Il donne à Paul un livre.

Il le lui donne. donne-le-lui ! ne le lui donne pas !

Tu dois parler à ton père. C'est à ton père que tu dois parler.
(*) *À ton père parler tu dois*

La critique est aisée. Critiquer est aisé. Il est aisé de critiquer!

Se connaître soi-même nécessite une bonne connaissance de soi.

Canonical constructions and transformations

Part of syntactic diversity may be seen as transformations over a canonical representation.

e.g. active voice (canonical) $\longrightarrow$ passive voice $\longrightarrow$ wh-sentence $\longrightarrow^{\star}$ ...

$\rightsquigarrow$ transformational grammars:

- a base grammar (say CFG) for building canonical constructions
- a finite set of transformations over syntactic trees

Peters & Ritchie (1973) Transformation grammars are too complex (power of Turing-machine)
reason: unbounded sequences of erasing/increasing transformations

No longer considered but influential for other formalisms
such as TAGs, metagrammars,...
idea: pre-computation at grammar level a finite set of transformation sequences

Ambiguity

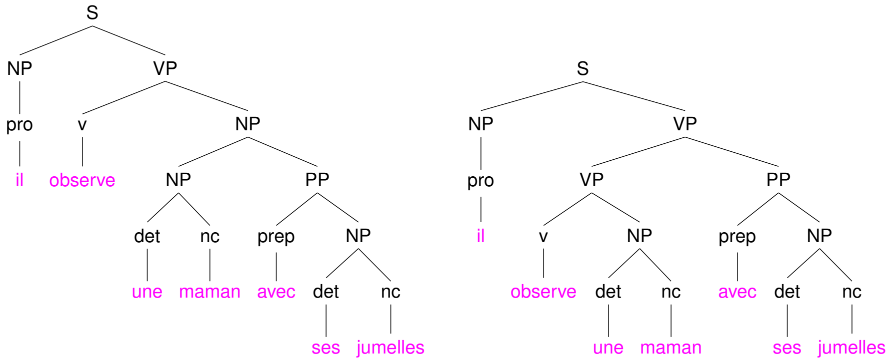
Ambiguity is present everywhere in language,
but mostly invisible to humans

il observe une maman avec ses jumelles

- lexical ambiguity on **jumelles**
- syntactic ambiguity on PP-attachment of **avec ses jumelles**
- anaphora ambiguity on **ses**

At least 8 interpretations (2 at syntactic level)

Syntactic ambiguities on PP attachments

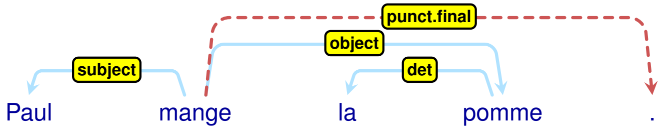


for a chain of *k* PPs, exponential number of syntactic trees wrt *k*

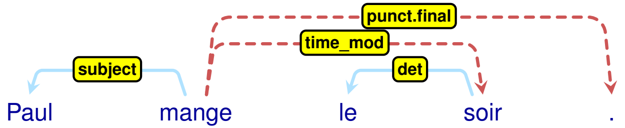
la Chambre des communes reprendra l'examen du$_1$ projet de$_2$ loi de$_3$ ratification du$_4$ traité de$_5$ Maastricht dès$_6$ la reprise de$_7$ la session du$_8$ soir dans$_9$ la salle principale du$_{10}$ batiment.

Implicit and Ambiguities

- Paul mange la pomme



- Paul mange le soir



Note: Prosody may help in this specific case (argument vs modifier)

Implicit and PP-attachments

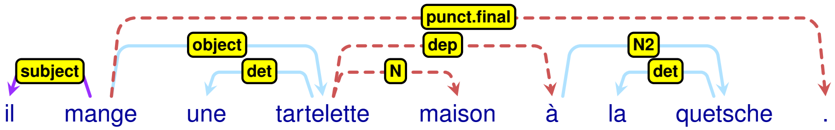
- *Il mange une tarte avec ses amis*
- *Il mange une tarte avec de la chantilly*
- *Il mange une tarte avec sa bière*
- *Paul mange une [pomme de terre] cuite*

Conclusion we need some knowledge about words and world

By using distributional techniques to capture meanings and contexts

tartelette & **tarte***semanticallyclose*
quetsche kind of **fruit**
`aux_fruits` frequent context for **tarte**

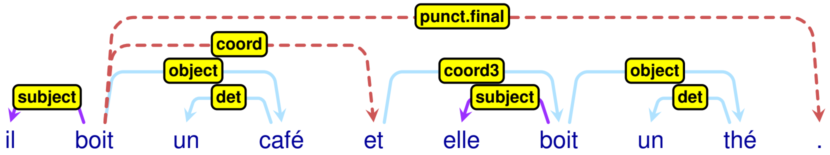
$\Longrightarrow$ **tartelette à la quetsche**



Using very local knowledge

One may have ellipsis in a sentence to be filled by local information
for instance, coordination with ellipse

Il boit un café et elle ϵ *un thé.*



Which complexity required for syntax

Chomsky hierarchy (1959): Classify grammars $(\mathcal{N}, \Sigma, S, \mathcal{P})$
with $\mathcal{P}$ finite set of productions over terminal set Σ and non-terminal set $\mathcal{N}$,
notations: $a \in \Sigma$, $A, B \in \mathcal{N}$, $\alpha, \beta, \gamma \in (\Sigma \cup \mathcal{N})^\star$

Type 3: Regular languages

$A \longrightarrow a$, $A \longrightarrow aB$

Type 2: Context-free languages

$A \longrightarrow \gamma$

Type 1: Context-sensitive languages

$\alpha A \beta \longrightarrow \alpha\gamma\beta, \ |\gamma| > 0$

Type 0: recursively enumerable languages

$\alpha \longrightarrow \beta$

Regular languages

Chomsky (1957): "*English is not a regular language*"

> *The cat likes tuna fish*
> *The cat [the dog chased] likes tuna fish*
> *The cat [the dog [the rat bit] chased] likes tuna fish*
> *The cat [the dog [the rat [the elephant admired] bit] chased] likes] tuna fish*

$\Longrightarrow$ analogous to $n^n v^n$ language (not a regular one)

Context-Free Languages

A Context-Free Grammar $G = (\mathcal{N}, \Sigma, \mathtt{S}, \mathcal{P})$ with

- $\mathcal{N}$ a finite set of non-terminals such as $\mathtt{S}$, $\mathtt{NP}$, $\mathtt{VP}$
- Σ a finite set of terminals such as $\mathtt{nc}$, $\mathtt{pn}$, $\mathtt{v}$
- $\mathtt{S}$ a distinguished non-terminal
- $\mathcal{P}$ a finite set of productions $A \longrightarrow \gamma$ with $\gamma \in (\mathcal{N} \cup \Sigma)^\star$

The context-free language $L(G)$ generated by G defined as

$$L(G) = \{w \in \Sigma^\star | \mathtt{S} \Longrightarrow^\star w\}$$

with $\Longrightarrow^\star$ transitive closure of

$$\alpha A \beta \Longrightarrow \alpha\gamma\beta \text{ iff } A \longrightarrow \gamma \in \mathcal{P}$$

Membership of $w \in L(G)$ may be checked in $O(|w|^3)$

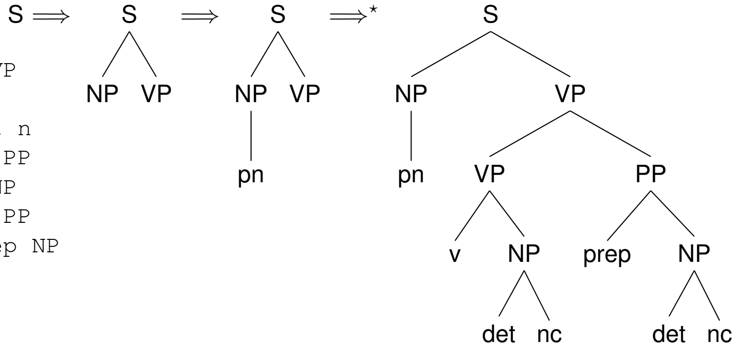
CFLs and natural languages

CFGs seems sufficient for many syntactic phenomena, including embedding. in particular $a^n b^n$ is a CFL

The derivations may be represented by parse trees (or proof trees) similar to linguist's syntactic trees

$$S \Longrightarrow NP\ VP \Longrightarrow pn\ VP \Longrightarrow pn\ VP\ PP \Longrightarrow pn\ v\ NP\ PP \Longrightarrow^{\star} pn\ v\ det\ nc\ prep\ det\ nc$$

```
S --> NP VP
NP --> pn
NP --> det n
NP --> NP PP
VP --> v NP
VP --> VP PP
PP --> prep NP
```



Are CFLs enough ?

2 aspects:

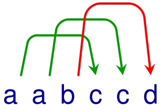
- How do we check that a language is not context-free ?
  use of pumping lemma

Theorem (Bar Hillel's pumping lemma)

L is a CFL iff

$$\exists N > 0, \forall w \in L, |w| > N \Longrightarrow \exists u, v, w, x, y, \ \wedge \begin{cases} w = uvwxy \\ |vwx| \leq N \wedge |vx| > 0 \\ \forall n \geq 0, \ uv^n w x^n y \in L \end{cases}$$

In particular, language $a^n b^m c^n d^m, \ n, m \geq 0$ is not context-free (cross-serial dependencies)



- Can we find a linguistic counter-example ? Not so easy !

Swiss-German example (Shieber 1985)

Jan säit das mer	em Hans	es huus	hälfed	asstriiche
Jean said that we	Hans-DAT	the house-ACC	helped	paint

Jan säit das mer	d'chind	em Hans	es huus	lönd	hälfed	asstriiche
Jean said that we	the children-ACC	Hans-DAT	the house-ACC	let	helped	paint

We can iterate, embedding more verbs (at the end) requiring case-marked arguments (accusative & dative).

Verbs should follow nouns, but dative nouns may be stacked before acc. nouns, and idem for verbs

Swiss German is not context-free

...das mer	$(\text{d'chind})^n$	$(\text{em Hans})^m$	es huus	$(\text{lönd})^n$	$(\text{hälfed})^m$	asstrii
...that we	$(\text{the children-ACC})^n$	$(\text{Hans-DAT})^m$	the house-ACC	$(\text{let})^n$	$(\text{helped})^m$	pair

We take homomorphism h such that:

$$
\begin{array}{ll}
h(\text{d'chind}) = a & h(\text{säit das mer}) = \epsilon \\
h(\text{em Hans}) = h(\text{noun-DAT}) = b & h(\text{es huus}) = \epsilon \\
h(\text{lönd}) = c & h(\text{asstriiche}) = \epsilon \\
h(\text{hälfed}) = h(\text{v-DAT}) = d & h(w) = \epsilon \text{ otherwise}
\end{array}
$$

and intersect $h(L_{SW})$ with regular language $L_R = a^\star b^\star c^\star d^\star$

$$l = h(L_{SW}) \cap L_R = a^n b^m c^n b^m$$

if L_{SW} is a CFL, then l is a CFL
(closures by homomorphism and intersection with regular language)
but l is not CFLs, and therefore L_{SW} is not CFL

Weak vs Strong generative capacity

Theorem

Swiss German is not a context-free language

No context-free grammar can generate the strings of Swiss-German language
$\Longrightarrow$ SG $\Longrightarrow$ notion of weak generative capacity

$$G_1 \equiv_{\text{weak}} G_2 \Longleftrightarrow L(G_1) = L(G_2)$$

Actually, linguists are mostly interested by the parse trees
$\Longrightarrow$ notion of strong generative capacity

$$G_1 \equiv_{\text{strong}} G_2 \Longleftrightarrow \text{trees}(G_1) = \text{trees}(G_2)$$

Easier to be persuaded than CFGs lack strong generative capacity to model some expected syntactic trees

Dutch cross-dependencies

Dutch exhibits similar phenomena than for Swiss-German,
but without visible case-marking

. . . dat	Jan	Piet	de kinderen	zag	helpen	zwemmen
. . . *that*	Jan	Piet	the children	saw	help	swin

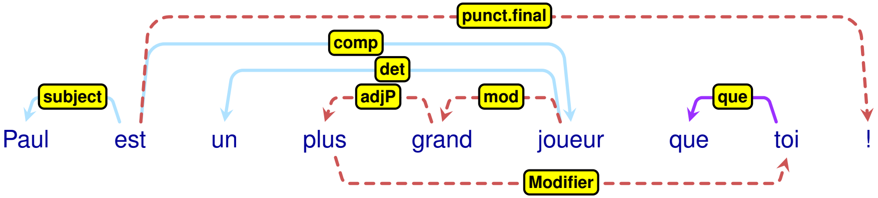
If we require parse trees reflecting these crossing dependencies, then the resulting set of parse trees can't be generated by a CFG.

Dutch is not strongly CFG (but seems to be weakly CFG)

What about French ?

There are several syntactic phenomena for French for whose “natural” syntactic trees do not correspond to CFG parse trees.

For instance, the comparative construction:



Parsing & Automata

We will need to explore new classes of languages (slightly) beyond CFLs.

Each class of language have an associated class of automata,
that may be used for parsing.

grammars	**automata**
regular grammars	finite-state automata
context-free grammars	push-down automata
context-sensitive grammars	linear-bounded automata
unrestricted grammars	Turing machine

Efficient parsing is often related to modeling computations with an adapted class of automata

Syntax vs probabilities

Chomsky opposes a syntax-based view of language with a probabilistic one:

Colorless green ideas sleep furiously
Furiously sleep ideas green colorless

The two sentences should not occur $\Longrightarrow p(s_1) = p(s_2) = 0$
But s_1 is grammatical while s_2 is not

However, F. Pereira (2000) using (smoothed) language models

$$\frac{p(\textbf{Colorless green ideas sleep furiously})}{p(\textbf{Furiously sleep ideas green colorless})} \approx 2.10^5$$

where $p(w_{1:n}) = p(w_1) \prod_{i=2}^{n} p(w_i|w_{i-1})$ with $p(w_i|w_{i-1}) = \Sigma_{c=1}^{C} p(w_i|c)p(c|w_{i-1})$
aggregated Markov model ($C = 16$)