Strong learning and semantics

Computational Learning of Syntax

Alexander Clark

Department of Philosophy
King's College, London

LSA Summer Institute 2015, Chicago

Topics for today

- Syntactic structure
- Semantic bootstrapping
- Strong learning
- Learning from sound/meaning pairs
- Discussion

Standard view: Semantic Bootstrapping

[Pinker, 1995]

Many models of language acquisition assume that the input to the child consists of a sentence and a representation of the meaning of that sentence, inferred from context and from the child's knowledge of the meanings of the words (e.g. Anderson, 1977; Berwick, 1986; Pinker, 1982, 1984; Wexler & Culicover, 1980). Of course, this can't literally be true – children don't hear every word of every sentence, and surely don't, to begin with, perceive the entire meaning of a sentence from context.

Evolution of language

[Kirby, 2002]

So, an example utterance by an individual in the simulation that happened to know something like English might be the pair:

$$\langle \mathrm{tigereatsjohn}, \textbf{eats}(\textbf{tiger}, \textbf{john}) \rangle$$

Obviously the biggest component of the simulation will be the part that takes sets of pairs such as these and is able to learn from them in some useful way.

[Crain and Pietroski, 2001]

$$
\begin{array}{ccc}
\text{PLD} & \rightarrow \text{LAD} \rightarrow & G_L \\
\langle \text{string}_1, \text{meaning}_1 \rangle & & \langle \text{string}_1, \text{meaning}_1 \rangle \\
\langle \text{string}_2, \text{meaning}_2 \rangle & & \langle \text{string}_2, (\text{meaning}_{2a}, \text{meaning}_{2b}) \rangle \\
\langle \text{string}_3, \text{meaning}_3 \rangle & & \langle \text{string}_3, \text{meaning}_3 \rangle \\
\ldots & & \ldots \\
\langle \text{string}_n, \text{meaning}_n \rangle & & \langle \text{string}_n, \text{meaning}_n \rangle \\
 & & \langle \text{string}_{n+1}, \text{meaning}_{n+1} \rangle \\
 & & \langle \text{string}_{n+2}, (\text{meaning}_{n+2a}, \text{meaning}_{n+2b}) \rangle \\
 & & \ldots
\end{array}
$$

Arguments against this view

1. Impossible for children actually to do this.
2. Assumes precisely the ability which needs to be explained.
3. Language would be unnecessary.
4. Language dependence of the semantic representation.
5. Language acquisition starts before children have a theory of mind. (FALSE)
6. Blind and high-functioning autistic children acquire language with only minor delay.

Why is this view held?

[Steedman, 1996]

As soon as it is recognized that the very earliest stages in acquiring syntax require some language-independent source of information about grammatical categories and grammatical relations, the only plausible source that has ever been identified is the semantic interpretation that underlies the utterance.
. . . sooner or later, the child needs access to semantic interpretations in order to acquire syntactic competence.

[Chomsky, 1966]

For example, it might be maintained, not without plausibility, that semantic information of some sort is essential even if the formalized grammar that is the output of the device does not contain statements of direct semantic nature. Here care is necessary. It may well be that a child given only the inputs of (2) as nonsense elements would not come to learn the principles of sentence formation. This is not necessarily a relevant observation, however, even if true. It may indicate only that meaningfulness and semantic function provide the motivation for language learning, while playing no necessary part in its mechanism, which is what concerns us here.

Running example

Propositional logic

Alphabet

rain, snow, hot, cold, danger	$A_1, A_2, \ldots$
and, or, implies, iff	$\wedge, \vee, \rightarrow, \leftrightarrow$
not	$\neg$
open, close	$(,)$

Running example

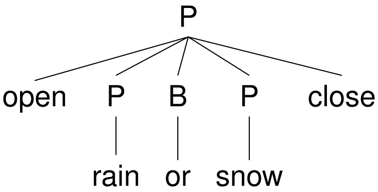
Propositional logic

Alphabet

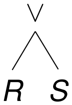
rain, snow, hot, cold, danger	$A_1, A_2, \ldots$
and, or, implies, iff	$\wedge, \vee, \rightarrow, \leftrightarrow$
not	$\neg$
open, close	$(,)$

- rain
- open snow implies cold close
- open snow implies open not hot close close

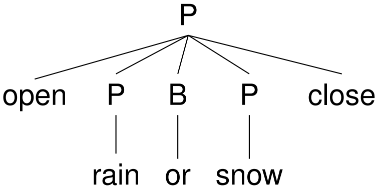
Classic model



open rain or snow close



Classic model



open rain or snow close

Q	R	S	
T	T	T	T
T	T	F	T
T	F	T	T
...	...	...	
F	F	F	F

Models

Two reasonable models

	Inputs	Outputs
Weak	strings	strings
Weak semantic	strings + meanings	strings + meanings

Models

Two reasonable models

	Inputs	Outputs
Weak	strings	strings
Weak semantic	strings + meanings	strings + meanings

An unreasonable model [Wexler and Culicover, 1980]

	Inputs	Outputs
Strong	strings	strings + trees

Structural descriptions as derivation trees of CFGs

Target class of grammars

$\mathcal{G}$ is some set of context-free grammars.
Pick some grammar $G_* \in \mathcal{G}$

Weak learning

We receive examples $w_1, \ldots, w_n, \ldots$
We produce a series of hypotheses $G_1, \ldots, G_n, \ldots$
We want G_n to converge to some grammar $\hat{G}$ such that $L(\hat{G}) = L(G_*)$

Structural descriptions as derivation trees of CFGs

Target class of grammars

$\mathcal{G}$ is some set of context-free grammars.
Pick some grammar $G_* \in \mathcal{G}$

Strong learning

We receive examples $w_1, \ldots, w_n, \ldots$
We produce a series of hypotheses $G_1, \ldots, G_n, \ldots$
We want G_n to converge to some grammar $\hat{G}$ such that $\hat{G} \equiv G_*$

Equivalence

Equality $\hat{G} = G_*$

Isomorphism $\hat{G} \equiv G_*$: a bijection between the nonterminals so that the two grammars have the same productions.

Strong equivalence Isomorphic trees

Structural equivalence Same unlabelled parse trees [Paull and Unger, 1968]

Power-series equivalence Same number of derivations [Chomsky and Schützenberger, 1959]

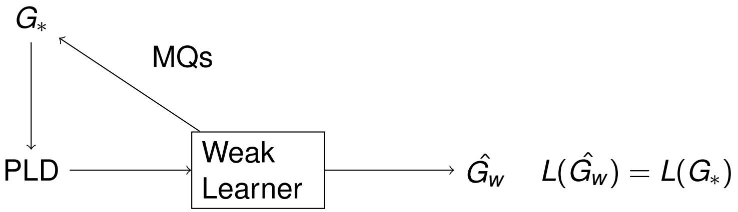
Weak equivalence $L(\hat{G}) = L(G_*)$

Classic model

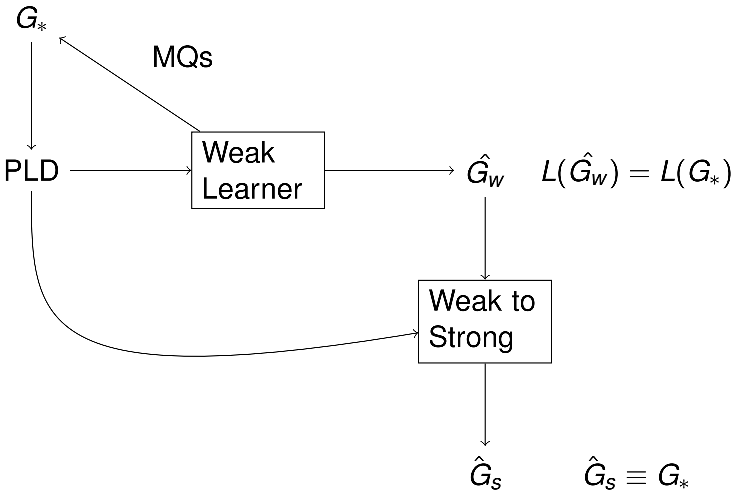
$$\hat{G} \equiv G_*$$

$$\begin{tikzcd}
G_* \arrow[d] & & \\
\mathrm{PLD} \arrow[r] & \boxed{\mathrm{LAD}} \arrow[r] & \hat{G}
\end{tikzcd}$$

Weak to Strong learning



Weak to Strong learning



Canonical Forms

Unlearnable classes

Suppose $\mathcal{G}$ contains G_1 and G_2 such that

$$L(G_1) = L(G_2)$$

$$G_1 \neq G_2$$

Then we cannot strongly learn $\mathcal{G}$.

Canonical Forms

Any learnable class

$\mathcal{G}$ contains at most one structurally distinct grammar for each language:

- A canonical grammar for that language.
- Apart from a change of labels for nonterminals.

Canonical Forms

Any learnable class

$\mathcal{G}$ contains at most one structurally distinct grammar for each language:

- A canonical grammar for that language.
- Apart from a change of labels for nonterminals.

A strong learner will implicitly define a canonical grammar for every language.

Discussion

Is it possible that we could have another language with the same set of grammatical strings as English, but with different syntactic structure/degree of ambiguity?

All CFLs

The minimal grammar will have nonterminals that are closed sets of strings.

All CFLs

The minimal grammar will have nonterminals that are closed sets of strings.

Consequence

- We can work algebraically with the concept lattice.

A general principle

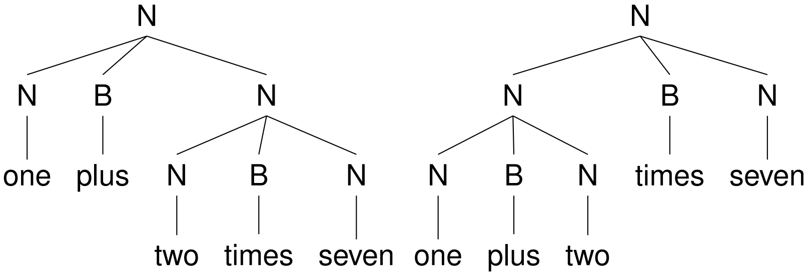
Maximal decompositions will give us minimal grammars.

A general principle

Maximal decompositions will give us minimal grammars.

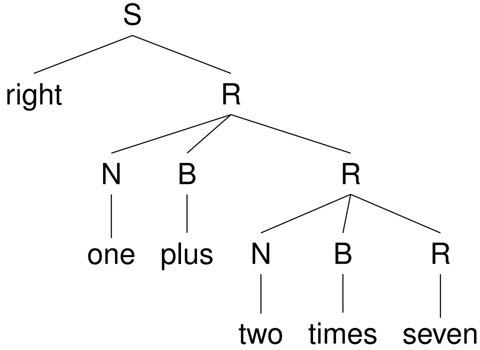
- This result is just talking about a language as a set of strings.
- The minimal grammar for a string-language may not be able to model the string-meaning relation.
- We are therefore making a very strong prediction about the set of allowable languages as sets of string/meaning pairs.

Possible example



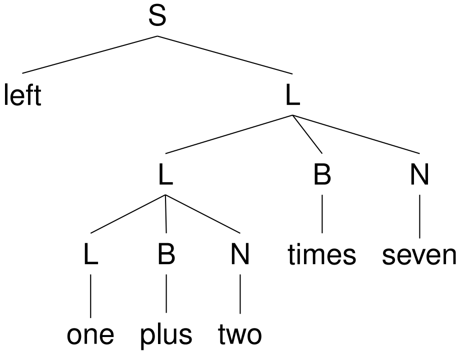
Impossible example

If it starts with “left” then it branches to the left, and if it starts with “right” it branches to the right.



Impossible example

If it starts with “left” then it branches to the left, and if it starts with “right” it branches to the right.



Prediction

Grammar

$S \rightarrow$ left L, $S \rightarrow$ right R
$N \rightarrow$ one, $N \rightarrow$ two $\ldots$
$B \rightarrow$ plus, $B \rightarrow$ minus, $\ldots$
$L \rightarrow LBN$, $L \rightarrow N$
$R \rightarrow NBR$, $R \rightarrow N$

In this grammar L, R are (weakly) mergeable; as a result this grammar is not structurally possible.

Prediction

Grammar

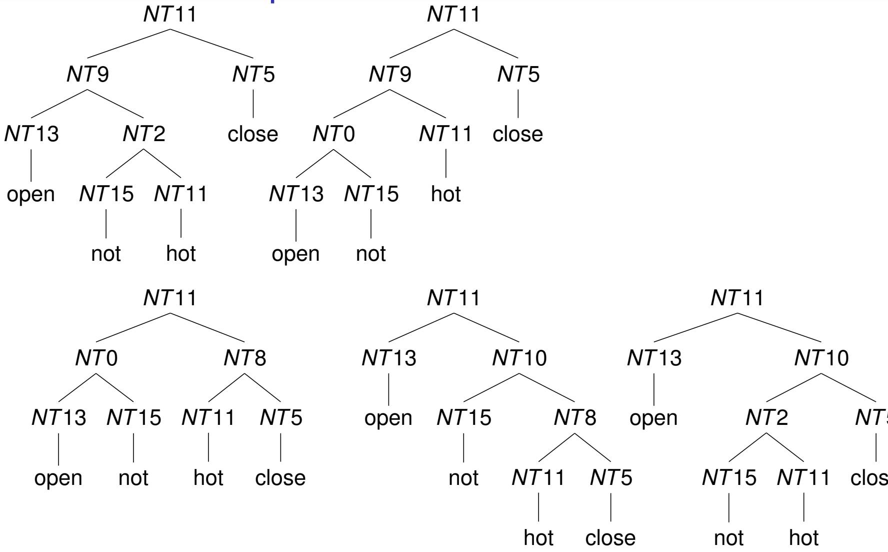
$S \rightarrow$ left L, $S \rightarrow$ right R
$N \rightarrow$ one, $N \rightarrow$ two ...
$B \rightarrow$ plus, $B \rightarrow$ minus, ...
$L \rightarrow LBN$, $L \rightarrow N$
$R \rightarrow NBR$, $R \rightarrow N$

In this grammar L, R are (weakly) mergeable; as a result this grammar is not structurally possible.

Prediction

We predict that all languages use only grammars with minimal nonterminals.

open not hot close



Irreducible elements

Addition

$\{0, 1, 2, 3, \ldots\}$

Irreducible elements are 0 and 1

Irreducible elements

Addition

$\{0, 1, 2, 3, \ldots\}$

Irreducible elements are 0 and 1

Multiplication

$\{0, 1, 2, 3, \ldots\}$

Irreducible elements

Addition

$\{0, 1, 2, 3, \ldots\}$

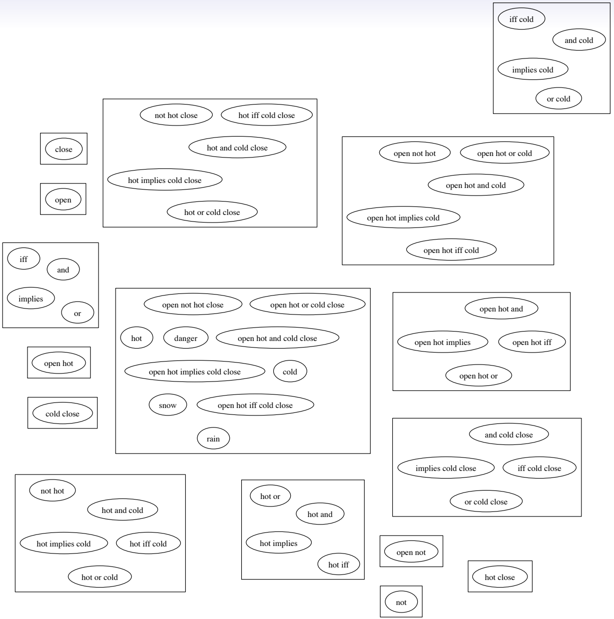
Irreducible elements are 0 and 1

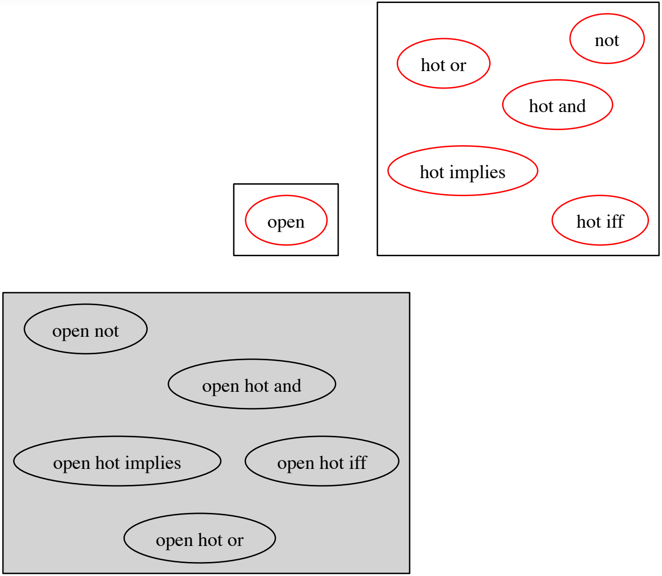
Multiplication

$\{0, 1, 2, 3, \ldots\}$

Irreducible elements are $0, 1, 2, 3, 5, 7, 11 \ldots$

- If the language is not regular, we have an infinite number of possible elements of the lattice, or congruence classes.
- We can select elements that are *irreducible* in a certain sense with respect to the algebraic structure of the lattice.
- To start with we take the irreducible elements with respect to concatenation.


iff cold
and cold
implies cold
or cold
not hot close
hot iff cold close
hot and cold close
hot implies cold close
hot or cold close
close
open
open not hot
open hot or cold
open hot and cold
open hot implies cold
open hot iff cold
iff
and
implies
or
open not hot close
open hot or cold close
hot
danger
open hot and cold close
open hot implies cold close
cold
snow
open hot iff cold close
rain
open hot
cold close
open hot and
open hot implies
open hot iff
open hot or
and cold close
implies cold close
iff cold close
or cold close
not hot
hot and cold
hot implies cold
hot iff cold
hot or cold
hot or
hot and
hot implies
hot iff
open not
not
hot close


not
hot or
hot and
hot implies
open
hot iff
open not
open hot and
open hot implies
open hot iff
open hot or

Definition

Definition

A congruence class X is composite if there are two congruence classes Y, Z such that $X = YZ$.
(and neither Y nor Z is the class $[\lambda]$

Definition

A congruence class X is prime if it is not composite.

The whole is greater than the sum of the parts

Fundamental theorem of substitutable languages

Every congruence class Q can be uniquely represented as a sequence of primes such that $Q = P_1 \ldots P_n$

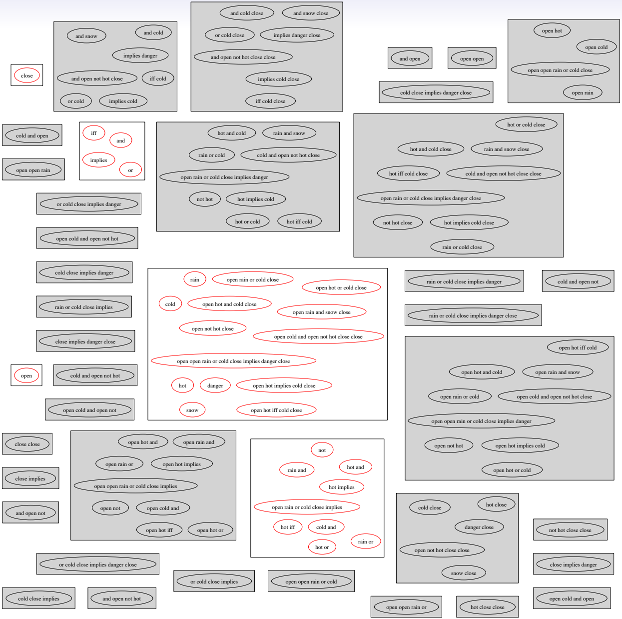
Fundamental theorem of substitutable languages

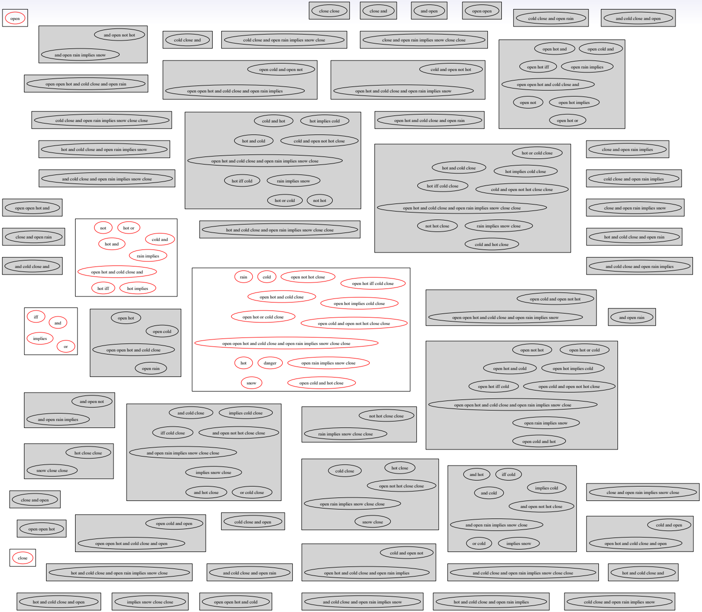
Every congruence class Q can be uniquely represented as a sequence of primes such that $Q = P_1 \dots P_n$

Intuition

If $X = YZ$, and we have a rule $P \to QXR$, then we can change it to $P \to QYZR$


or cold
and cold
implies cold
iff cold
not hot
hot and cold
hot implies cold
hot or cold
hot iff cold
open hot
close
open
not hot close
hot or cold close
hot and cold close
hot implies cold close
hot iff cold close
open not hot
open hot iff cold
open hot and cold
open hot implies cold
open hot or cold
open not hot close
open hot iff cold close
rain
danger
open hot and cold close
open hot implies cold close
cold
snow
open hot or cold close
hot
iff
and
implies
or
open not
open hot and
open hot implies
open hot iff
open hot or
and cold close
implies cold close
iff cold close
or cold close
hot or
not
hot and
hot implies
hot iff
hot close
cold close


close
and snow
and cold
implies danger
and open not hot close
iff cold
or cold
implies cold
and cold close
and snow close
or cold close
implies danger close
and open not hot close close
implies cold close
iff cold close
and open
open open
cold close implies danger close
open hot
open cold
open open rain or cold close
open rain
cold and open
open open rain
iff
and
implies
or
hot and cold
rain and snow
rain or cold
cold and open not hot close
open rain or cold close implies danger
not hot
hot implies cold
hot or cold
hot iff cold
hot or cold close
hot and cold close
rain and snow close
hot iff cold close
cold and open not hot close close
open rain or cold close implies danger close
not hot close
hot implies cold close
rain or cold close
or cold close implies danger
open cold and open not hot
cold close implies danger
rain or cold close implies
close implies danger close
open
cold and open not hot
open cold and open not
rain
open rain or cold close
open hot or cold close
cold
open hot and cold close
open rain and snow close
open not hot close
open cold and open not hot close close
open open rain or cold close implies danger close
hot
danger
open hot implies cold close
snow
open hot iff cold close
rain or cold close implies danger
cold and open not
rain or cold close implies danger close
open hot iff cold
open hot and cold
open rain and snow
open rain or cold
open cold and open not hot close
open open rain or cold close implies danger
open not hot
open hot implies cold
open hot or cold
close close
close implies
and open not
open hot and
open rain and
open rain or
open hot implies
open open rain or cold close implies
open not
open cold and
open hot iff
open hot or
not
rain and
hot and
hot implies
open rain or cold close implies
hot iff
cold and
rain or
hot or
cold close
hot close
danger close
open not hot close close
snow close
not hot close close
close implies danger
open cold and open
or cold close implies danger close
or cold close implies
open open rain or cold
cold close implies
and open not hot
open open rain or
hot close close

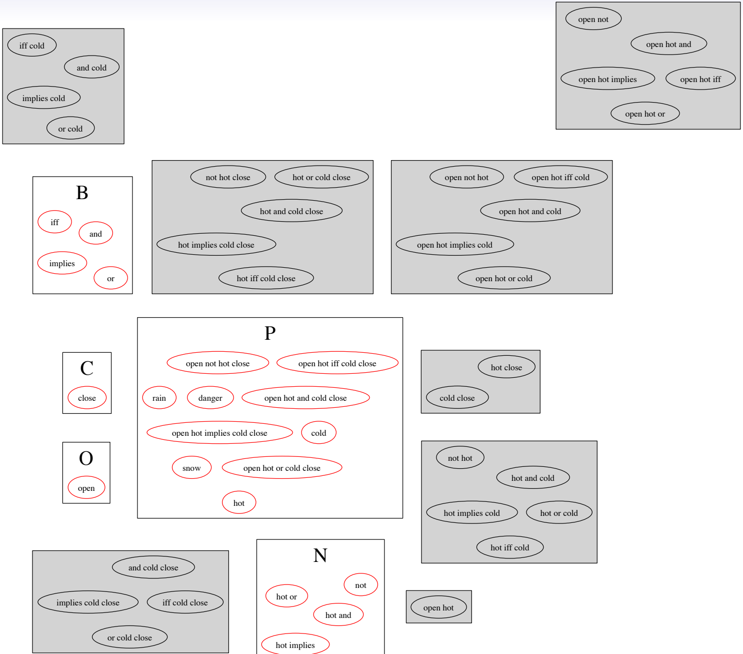


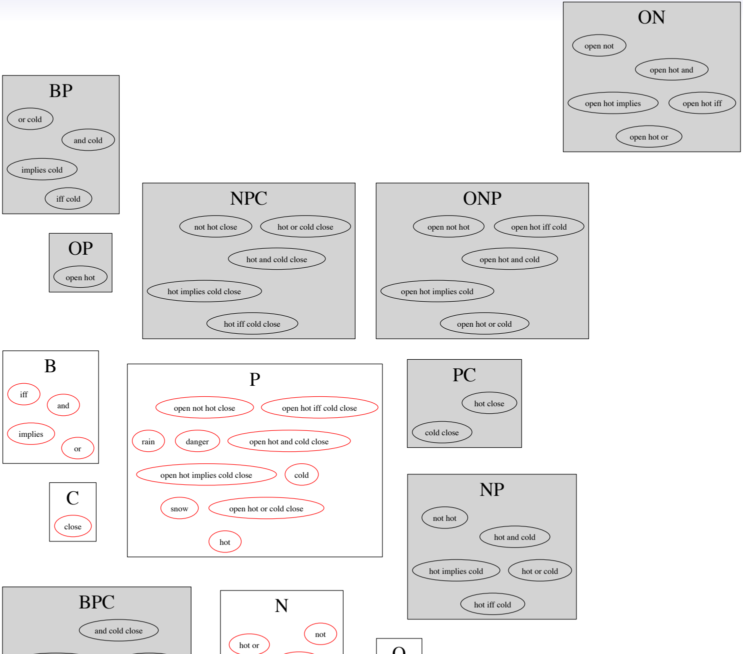
open
close close
close and
and open
open open
cold close and open rain
and cold close and open
and open not hot
and open rain implies snow
cold close and
cold close and open rain implies snow close
close and open rain implies snow close close
open hot and
open cold and
open hot iff
open rain implies
open open hot and cold close and
open not
open hot implies
open hot or
open open hot and cold close and open rain
open cold and open not
open open hot and cold close and open rain implies
cold and open not hot
open hot and cold close and open rain implies snow
cold close and open rain implies snow close close
cold and hot
hot implies cold
hot and cold
cold and open not hot close
open hot and cold close and open rain implies snow close
hot iff cold
rain implies snow
hot or cold
not hot
open hot and cold close and open rain
hot and cold close and open rain implies snow
and cold close and open rain implies snow close
hot or cold close
hot and cold close
hot implies cold close
hot iff cold close
cold and open not hot close close
open hot and cold close and open rain implies snow close close
not hot close
rain implies snow close
cold and hot close
close and open rain implies
cold close and open rain implies
close and open rain implies snow
hot and cold close and open rain
and cold close and open rain implies
open open hot and
close and open rain
and cold close and
not
hot or
cold and
hot and
rain implies
open hot and cold close and
hot iff
hot implies
hot and cold close and open rain implies snow close close
rain
cold
open not hot close
open hot iff cold close
open hot and cold close
open hot implies cold close
open hot or cold close
open cold and open not hot close close
open open hot and cold close and open rain implies snow close close
hot
danger
open rain implies snow close
snow
open cold and hot close
open cold and open not hot
open open hot and cold close and open rain implies snow
and open rain
iff
and
implies
or
open hot
open cold
open open hot and cold close
open rain
open not hot
open hot or cold
open hot and cold
open hot implies cold
open hot iff cold
open cold and open not hot close
open open hot and cold close and open rain implies snow close
open rain implies snow
open cold and hot
and open not
and open rain implies
and cold close
implies cold close
iff cold close
and open not hot close close
and open rain implies snow close close
implies snow close
and hot close
or cold close
not hot close close
rain implies snow close close
hot close close
snow close close
cold close
hot close
open not hot close close
open rain implies snow close close
snow close
and hot
iff cold
implies cold
and cold
and open not hot close
and open rain implies snow close
or cold
implies snow
close and open rain implies snow close
close and open
open open hot
open cold and open
open open hot and cold close and open
cold close and open
cold and open
open hot and cold close and open
close
hot and cold close and open rain implies snow close
and cold close and open rain
cold and open not
open hot and cold close and open rain implies
and cold close and open rain implies snow close close
hot and cold close and
hot and cold close and open
implies snow close close
open open hot and cold
and cold close and open rain implies snow
hot and cold close and open rain implies
cold close and open rain implies snow

Restriction

- We only consider substitutable languages which have a finite number of primes.
- We define nonterminals only for these primes.

Label	Examples
P	rain, cold, open rain and cold close
O	open
C	close
B	and, or, ...
N	not, hot or, cold and ...


iff cold
and cold
implies cold
or cold
open not
open hot and
open hot implies
open hot iff
open hot or
B
iff
and
implies
or
not hot close
hot or cold close
hot and cold close
hot implies cold close
hot iff cold close
open not hot
open hot iff cold
open hot and cold
open hot implies cold
open hot or cold
P
open not hot close
open hot iff cold close
rain
danger
open hot and cold close
open hot implies cold close
cold
snow
open hot or cold close
hot
C
close
O
open
hot close
cold close
not hot
hot and cold
hot implies cold
hot or cold
hot iff cold
and cold close
implies cold close
iff cold close
or cold close
N
not
hot or
hot and
hot implies
open hot


ON
open not
open hot and
open hot implies
open hot iff
open hot or
BP
or cold
and cold
implies cold
iff cold
NPC
not hot close
hot or cold close
hot and cold close
hot implies cold close
hot iff cold close
ONP
open not hot
open hot iff cold
open hot and cold
open hot implies cold
open hot or cold
OP
open hot
B
iff
and
implies
or
P
open not hot close
open hot iff cold close
rain
danger
open hot and cold close
open hot implies cold close
cold
snow
open hot or cold close
hot
PC
hot close
cold close
C
close
NP
not hot
hot and cold
hot implies cold
hot or cold
hot iff cold
BPC
and cold close
N
not
hot or

Productions

We need non-binary rules.

Correct productions

$P_0 \rightarrow P_1 \dots P_k$
where $P_0 \supsetneq P_1 \dots P_k$

- $N \rightarrow PB$
- $P \rightarrow ONSC$
- Not $P \rightarrow OPBPC$ – correct but too long.
- Not $S \rightarrow ONONSCC$

If there are n primes then there are at most n^2 valid productions.

A Strong Learning Result

Class of grammars

$\mathcal{G}_{sc}$ is the class of canonical grammars for all substitutable languages with a finite number of primes.

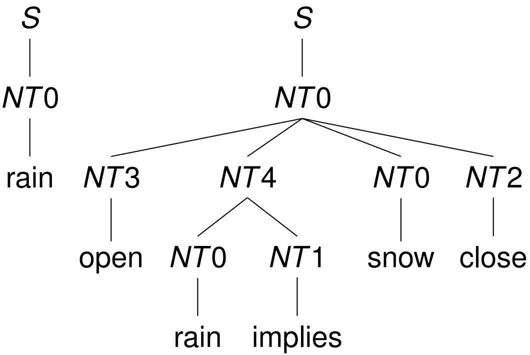
Theorem

There is an algorithm which learns $\mathcal{G}_{sc}$

- From positive examples
- Identification in the limit convergence
- Strongly
- Using polynomial time and data

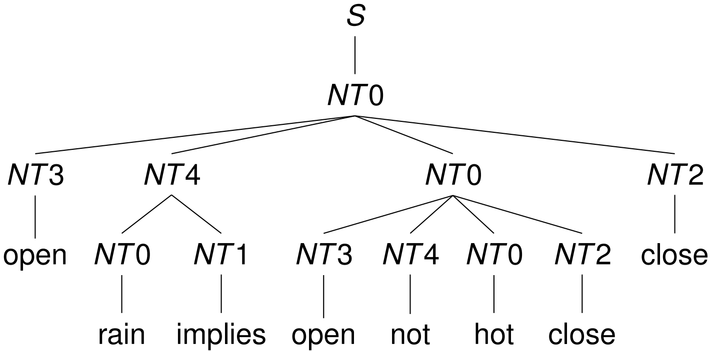
Running example

(verbatim output from implementation)



Running example

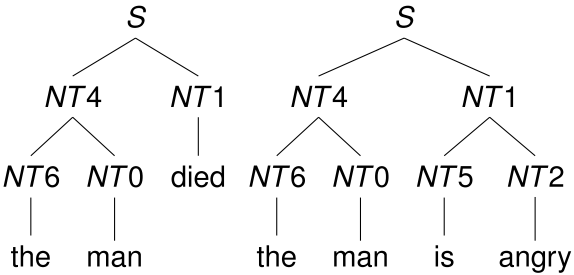
(verbatim output from implementation)



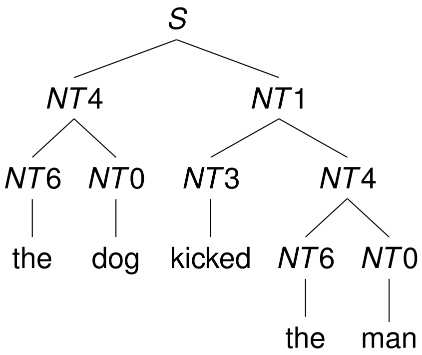
Example 1

1. the man died
2. the dog died
3. the man is angry
4. it died
5. he died
6. the man kicked the dog

Example 1


S
NT4 NT1
NT6 NT0 died
the man
S
NT4 NT1
NT6 NT0 NT5 NT2
the man is angry

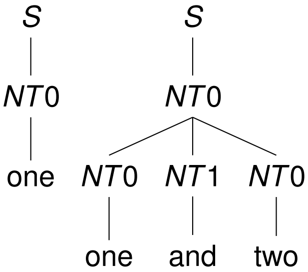
Example 1



Example 2

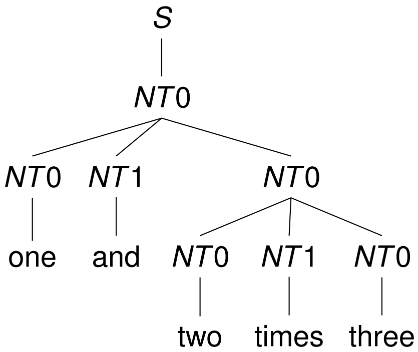
1. one
2. two
3. three
4. one and two
5. two times three
6. one and two times three

Example 2



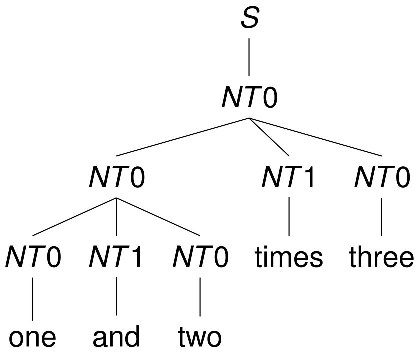
Example 2

Three trees for this string



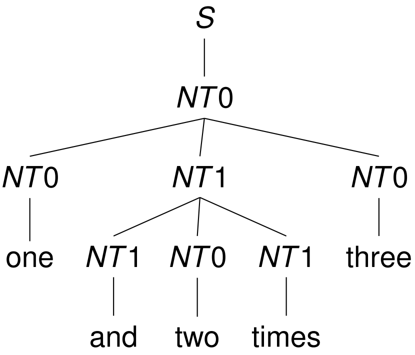
Example 2

Three trees for this string



Example 2

Three trees for this string



Primes

definition

- A closed set of strings, X, is composite if there are closed sets of strings Y, Z such that $X = Y \cdot Z$, (and $X \neq Y$ and $X \neq Z$)
- It is prime if it is not composite.

Primes

definition

- A closed set of strings, X, is composite if there are closed sets of strings Y, Z such that $X = Y \cdot Z$, (and $X \neq Y$ and $X \neq Z$)
- It is prime if it is not composite.

Informally:

- $X \to AB$ and $X \to CD$ then X will be prime.

Toy grammar

S $\rightarrow$ A B
S $\rightarrow$ C D
A $\rightarrow$ a1
A $\rightarrow$ a2
B $\rightarrow$ b1
B $\rightarrow$ b2
C $\rightarrow$ c1
C $\rightarrow$ c2
D $\rightarrow$ d1
D $\rightarrow$ d2

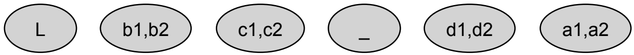
Toy grammar

S $\to$ A B
S $\to$ C D
A $\to$ a1
A $\to$ a2
B $\to$ b1
B $\to$ b2
C $\to$ c1
C $\to$ c2
D $\to$ d1
D $\to$ d2

Generates a finite language

a1 b2, c2 d1, . . .

Motivating example

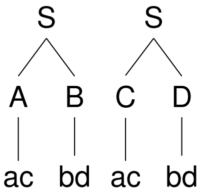


Add some lexical ambiguity

$$\boxed{\begin{array}{c} A \rightarrow ab \\ A \rightarrow ac \\ B \rightarrow ab \\ B \rightarrow bd \\ C \rightarrow ac \\ D \rightarrow bd \end{array}}$$

Add some lexical ambiguity

$$\boxed{\begin{array}{l} A \rightarrow ab \\ A \rightarrow ac \\ B \rightarrow ab \\ B \rightarrow bd \\ C \rightarrow ac \\ D \rightarrow bd \end{array}}$$



English counterexample

Lexical ambiguity

- I can swim
- I may swim
- I want a can of beer

English counterexample

Lexical ambiguity

- I can swim
- I may swim
- I want a can of beer
- *I want a may of beer

English counterexample

Contextual ambiguity

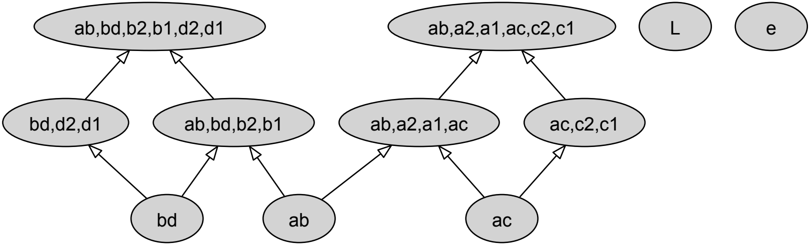
- She is Italian
- She is a philosopher
- She is an Italian philosopher

English counterexample

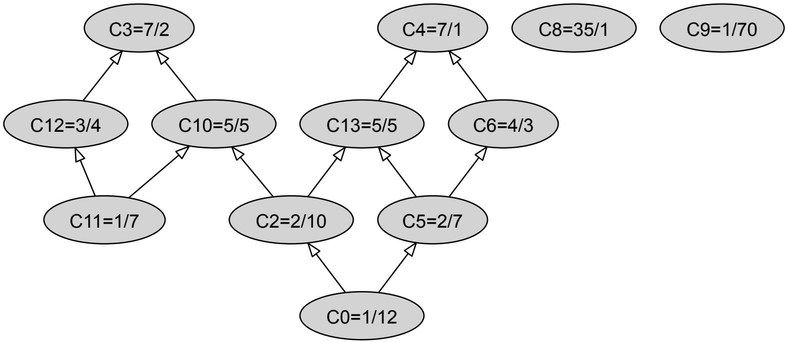
Contextual ambiguity

- She is Italian
- She is a philosopher
- She is an Italian philosopher
- *She is an a philosopher philosopher

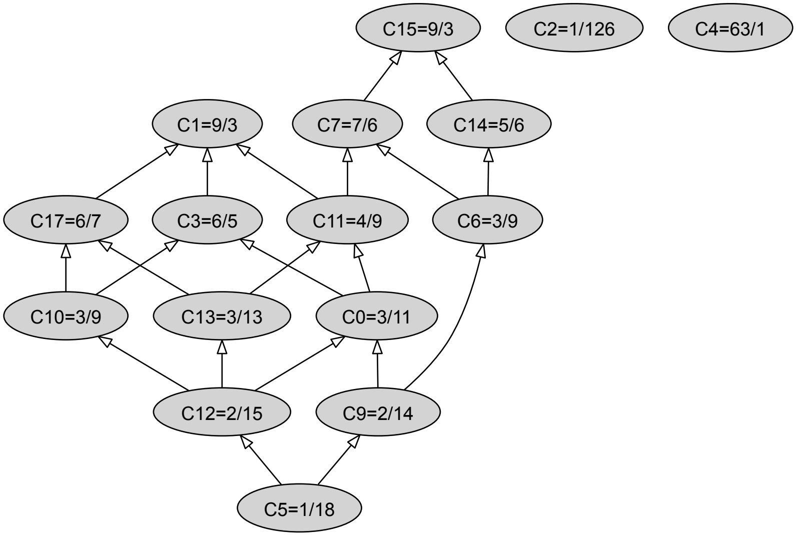
Motivating example



Motivating example



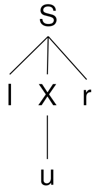
Motivating example



Suppose you know:

$l(u)r$

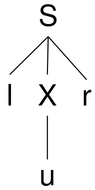
Tree



Suppose you know:

$l(u)r$

Tree



What is the label X?

$$\{l\Box r\}^{\triangleleft} \geq X$$

$$X \geq \{u\}^{\triangleright\triangleleft}$$

$\{l\Box r\}^{\triangleleft}$

|

X

|

$\{u\}^{\triangleright\triangleleft}$

$$\{l\Box r\}^{\lhd}$$

$$|$$

$$X$$

$$|$$

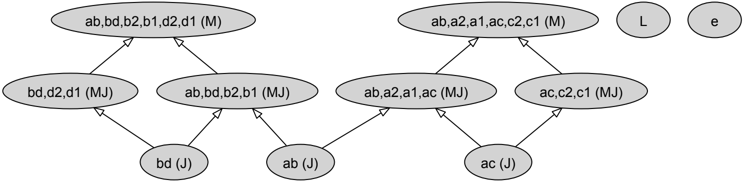
$$\{u\}^{\rhd\lhd}$$

If $\{l\Box r\}^{\lhd} = \{u\}^{\rhd\lhd}$ then X is fixed.

- If there are $l, r \in \Sigma^*$ such that $X = \{l \Box r\}^{\lhd}$, then X is meet semi-irreducible (MSI)
- If there is a w such that $X = \{w\}^{\rhd\lhd}$ then X is join semi-irreducible (JSI)

We are interested in MSI-JSI-primes: these are (semi)-irreducible wrt the three operations $\circ, \vee, \wedge$.

Motivating example



Conditions

- A finite number of MJ-primes.
- For every $a \in \Sigma$, and every context $l \Box r \in a^{\triangleright}$ there is a MJ-prime A such that $\{l \Box r\}^{\triangleleft} \geq A \geq a^{\triangleright\triangleleft}$
- Two technical conditions that guarantees a finite number of productions.

Conditions

- A finite number of MJ-primes.
- For every $a \in \Sigma$, and every context $l\Box r \in a^{\triangleright}$ there is a MJ-prime A such that $\{l\Box r\}^{\triangleleft} \geq A \geq a^{\triangleright\triangleleft}$
- Two technical conditions that guarantees a finite number of productions.

Theorem: [Clark, 2015]

This defines:

- A class of languages
- A class of grammars
- And a bijection between them.

Auxiliary Inversion in Polar Questions

Chomsky (1971), Crain and Nakayama (1987)

1. The student is hungry.
2. Is the student hungry?
3. The student who is in the garden is hungry.

Auxiliary Inversion in Polar Questions

Chomsky (1971), Crain and Nakayama (1987)

1. The student is hungry.
2. Is the student hungry?
3. The student who is in the garden is hungry.
4. Is the student who is in the garden hungry?
5. * Is the student who in the garden is hungry?

Auxiliary Fronting

Simple experiment with substitutable language learning algorithm.

Training data

the man who is hungry died .
the man ordered dinner .
the man died .
the man is hungry .
is the man hungry ?
the man is ordering dinner .

Auxiliary Fronting

Simple experiment with substitutable language learning algorithm.

Training data

the man who is hungry died .
the man ordered dinner .
the man died .
the man is hungry .
is the man hungry ?
the man is ordering dinner .

Test data

is the man who is hungry ordering dinner ?
∗is the man who hungry is ordering dinner ?

Counter-example

Berwick, Coen and Niyogi

Some data sets will give the wrong answer:

- Does he think [well] ?
- Does he think [hitting is not fighting] ?

Counter-example

Berwick, Coen and Niyogi

Some data sets will give the wrong answer:

- Does he think [well] ?
- Does he think [hitting is not fighting] ?
- I know [the man] likes to fight .
- I know [the man John] likes to fight .

Counter-example

Berwick, Coen and Niyogi

Some data sets will give the wrong answer:

- Does he think [well] ?
- Does he think [hitting is not fighting] ?
- I know [the man] likes to fight .
- I know [the man John] likes to fight .
- Is [the man] [well] ?
- * Is the man John hitting is not fighting ?

Critique

[Berwick et al., 2011]

Put another way, language acquisition is not merely a matter of acquiring a capacity to associate word strings with interpretations. Much less is it a mere process of acquiring a (weak generative) capacity to produce just the valid word strings of a language. Idealizing, one can say that each child acquires a procedure that generates boundlessly many meaningful expressions, and that a single string of words can correspond to more than one expression.

Example

[Berwick et al., 2011]

- Eagles that fly eat.
- Can eagles that fly eat?

S $\rightarrow$ NP VP ., S $\rightarrow$ CAN NP VI ?
NP $\rightarrow$ EAGLES (RC), NP $\rightarrow$ THEY
RC $\rightarrow$ THAT VP
VP $\rightarrow$ (CAN) VI
VP $\rightarrow$ DIED
VI $\rightarrow$ EAT, VI $\rightarrow$ FLY

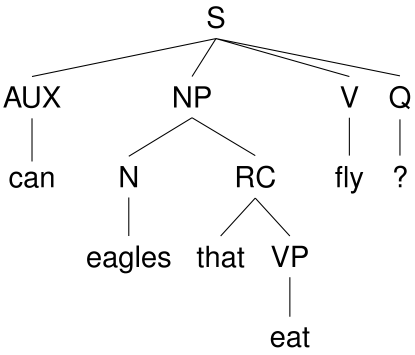
43 concepts and 10 primes

[Berwick et al., 2011]

Label	Strings	Context
OAUX	can, λ	eagles □ eat.
AUX	can	□ eagles eat?
N	eagles	□ that eat fly.
NP	eagles that eat	□ can eat.
RC	that can eat, λ, ...	eagles □can fly.
VI	eat, fly	eagles that eat can □.
VP	can eat, died	eagles that □ can fly.
C	that	eagles □ can fly fly.
STOP	.	eagles fly □
Q	?	can eagles fly □

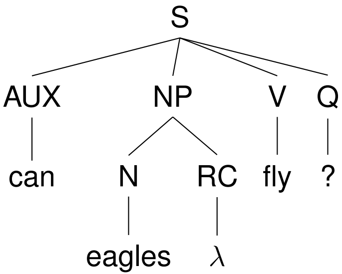
Example

[Berwick et al., 2011]



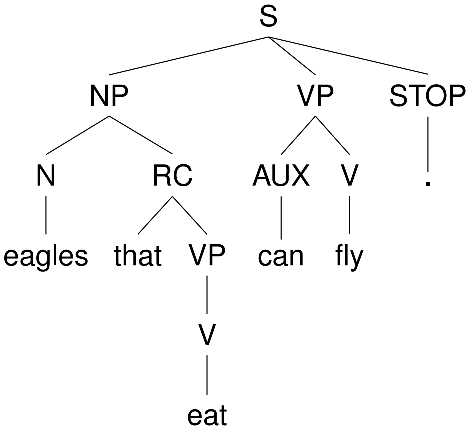
Example

[Berwick et al., 2011]



Example

[Berwick et al., 2011]



Question

Can we extend this to MCFGs?

Discussion

- The standard view is that the strings are not informative enough to recover the syntactic structures.
- This seems false:
  - The lexical categories can be recovered
  - The structure too (subject to the successful extension of these results)

Discussion

- The standard view is that the strings are not informative enough to recover the syntactic structures.
- This seems false:
  - The lexical categories can be recovered
  - The structure too (subject to the successful extension of these results)

Key idea

- There is a Galois connection between derivation contexts and yields for any formalism (in a larger class)
- There is a canonical grammar which is based on the irreducible elements of this structure.

Learning from string meaning pairs

Decomposition of strings

- john likes mary
- john
- likes mary

Decomposition of lambda term

- (**likes**(**mary**))**john**
- (**likes**(**mary**))
- **john**

Decomposition of strings

- every student hates exams
- every student
- hates exams

Decomposition of lambda terms

- An expression of type t:

$$\forall x_e(\mathbf{student}_{e\to t}(x) \to (\mathbf{hates}(\mathbf{exams}))(x))$$

- An expression of type $(e \to t) \to t$:

$$\lambda P_{e\to t}.\forall x_e(\mathbf{student}_{e\to t}(x) \to (P)(x))$$

- An expression of type $e \to t$: $(\mathbf{hates}(\mathbf{exams}))$

Some preliminary results using ACGs:
[Yoshinaka and Kanazawa, 2011]

Copying is ubiquitous in semantics

Decomposing strings

- Jennie likes Keiko and Chihiro
- Jennie
- likes Keiko and Chihiro

Decomposition of lambda terms

- $\mathbf{likes}(\mathbf{jennie}, \mathbf{keiko}) \wedge \mathbf{likes}(\mathbf{jennie}, \mathbf{chihiro})$
- $\mathbf{jennie}$
- $\lambda x.(\mathbf{likes}(\mathbf{x}, \mathbf{keiko}) \wedge \mathbf{likes}(\mathbf{x}, \mathbf{chihiro}))$

Framing the problem of language acquisition

Weak learning

- We want to find a learning algorithm A
- That can learn a class of grammars $\mathcal{G}$
- Such that $\mathcal{G}$ can describe all attested natural languages.
- Under a reasonable set of assumptions (!?)

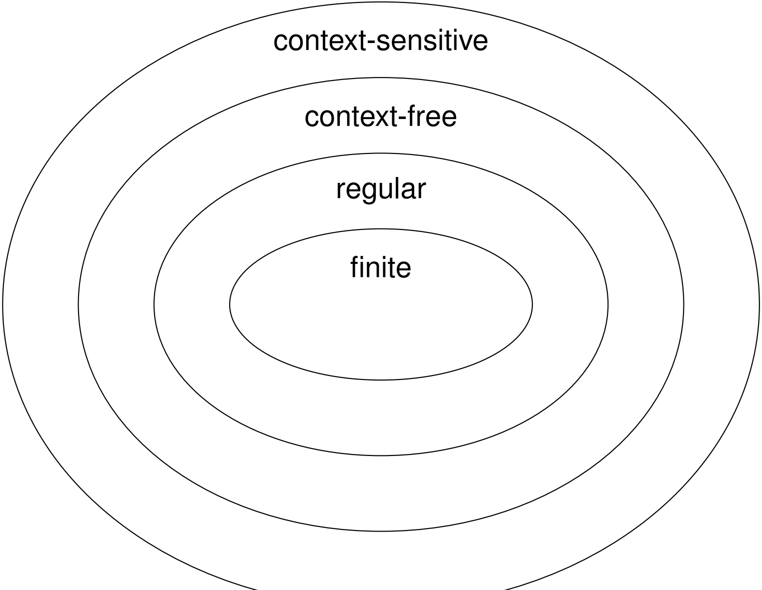
Tension

Chomsky, 1986

To achieve descriptive adequacy it often seems necessary to enrich the system of available devices, whereas to solve our case of Plato's problem we must restrict the system of available devices so that only a few languages or just one are determined by the given data. It is the tension between these two tasks that makes the field an interesting one, in my view.

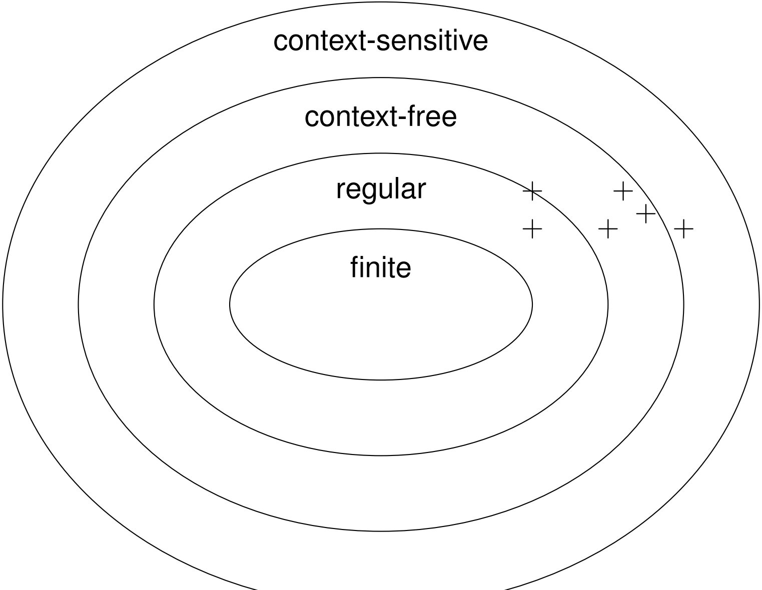
Standard strategy

Restrict hypothesis class



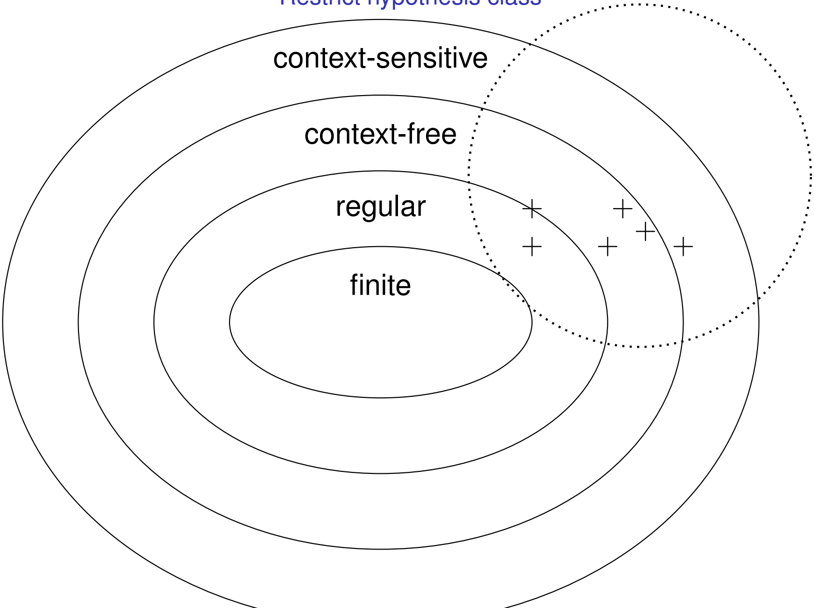
Standard strategy

Restrict hypothesis class



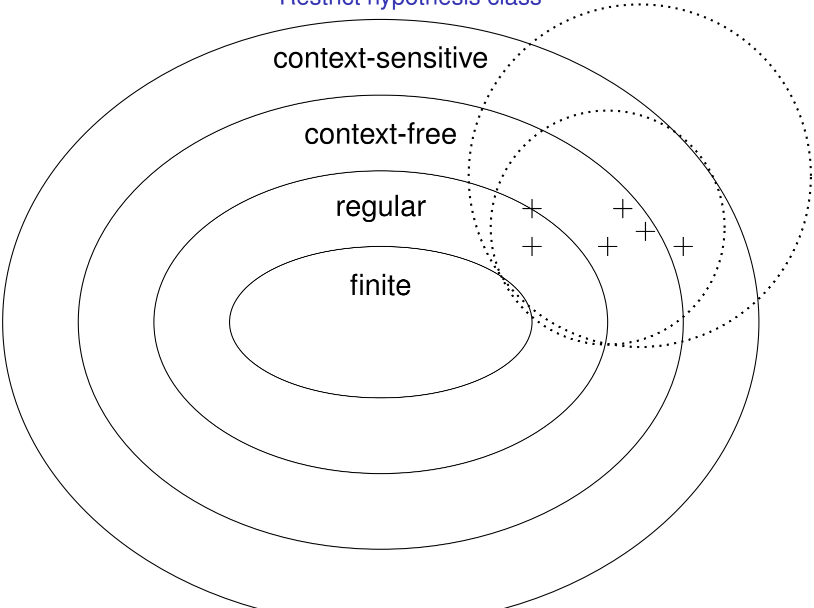
Standard strategy

Restrict hypothesis class



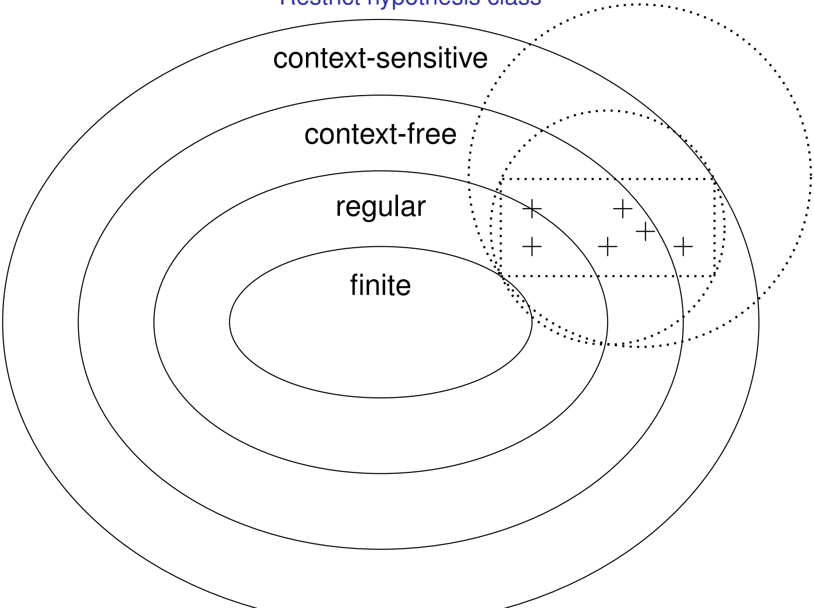
Standard strategy

Restrict hypothesis class



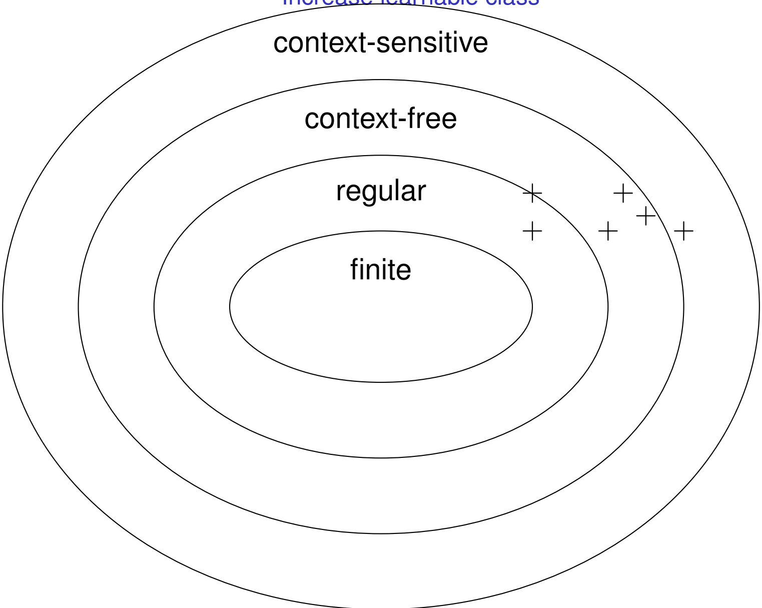
Standard strategy

Restrict hypothesis class



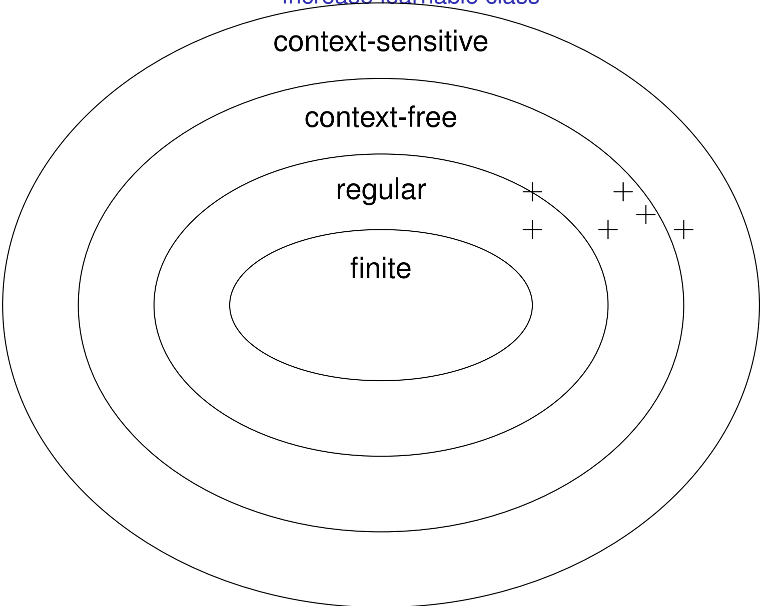
Alternative strategy

Increase learnable class



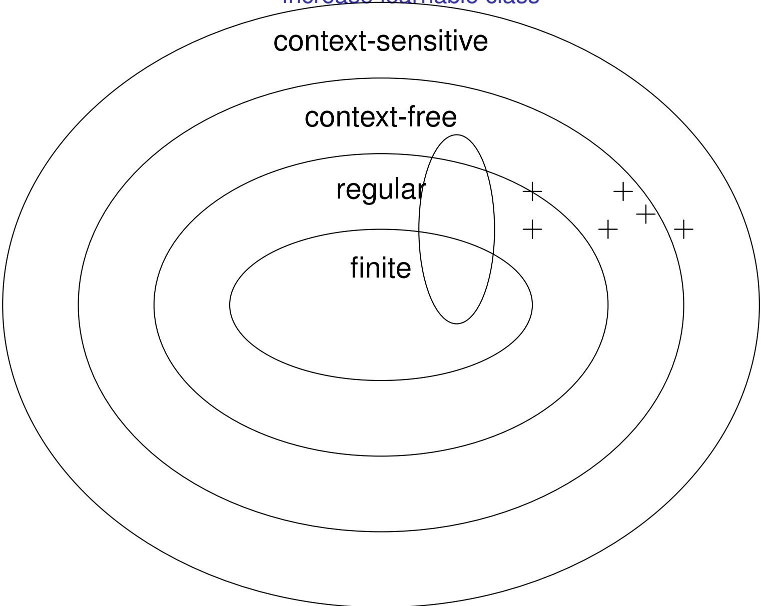
Alternative strategy

Increase learnable class



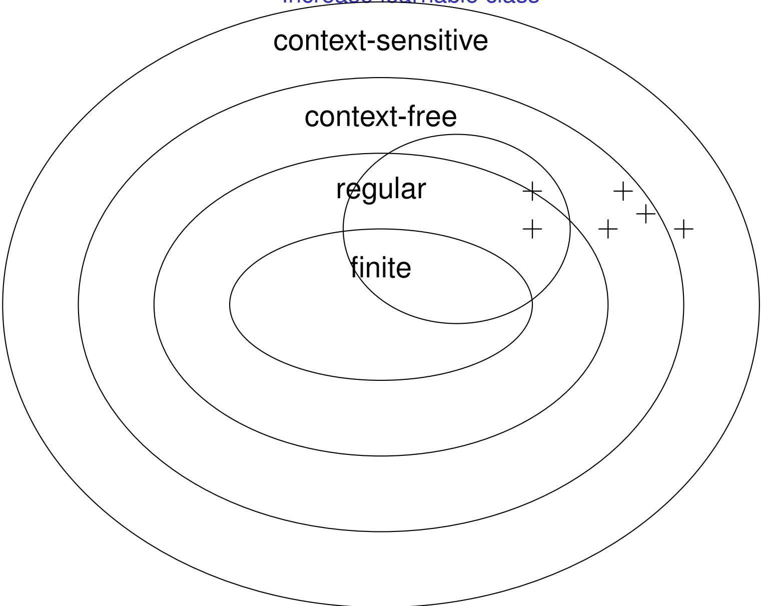
Alternative strategy

Increase learnable class



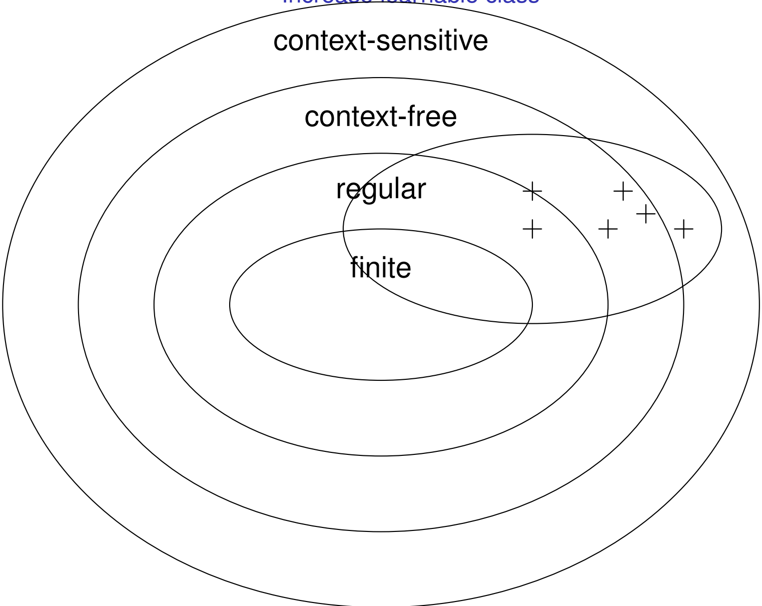
Alternative strategy

Increase learnable class



Alternative strategy

Increase learnable class



Quote

[Chomsky, 1965]

The only proposals that are explicit enough to support serious study are those that have been developed within taxonomic linguistics. It seems to have been demonstrated beyond reasonable doubt that quite apart from any questions of feasibility, methods of the sort that have been studied in taxonomic linguistics are intrinsically incapable of yielding the systems of grammatical knowledge that must be attributed to the speaker of a language.

" intrinsically incapable"?

We now know this is false.

- The systems of grammatical knowledge can be quite limited: (Minimalist Grammars (MGs), Tree Adjoining Grammars, etc.)
- Distributional learning methods can learn large classes of MCFGs which are weakly and strongly equivalent to MGs.

How does language acquisition take place?

One proposal:

- The learner looks at the relation between contexts and yields for some suitable class of grammars.

Questions

- Is this the whole story? Or are there additional more language specific processes at work?
- What is the explanation of Greenberg surface universals?
- What about deep universals like subjacency?
- How are rare phenomena (parasitic gaps etc.) learned?
- What are the alternative explanations at the moment?

Bibliography I

Berwick, R., Pietroski, P., Yankama, B., and Chomsky, N. (2011).
Poverty of the stimulus revisited.
Cognitive Science, 35:1207–1242.

Chomsky, N. (1965).
Aspects of the Theory of Syntax.
MIT.

Chomsky, N. (1966).
Explanatory models in linguistics.
In Ernest Nagel, P. S. and Tarski, A., editors, *Logic, Methodology and Philosophy of Science Proceeding of the 1960 International Congress*, volume 44 of *Studies in Logic and the Foundations of Mathematics*, pages 528 – 550.
Elsevier.

Bibliography II

Chomsky, N. and Schützenberger, M. P. (1959).
The algebraic theory of context-free languages.
Studies in Logic and the Foundations of Mathematics, 26:118–161.

Clark, A. (2015).
Canonical context-free grammars and strong learning: Two approaches.
In *Proceedings of the 14th Meeting on the Mathematics of Language (MOL 2015)*.

Crain, S. and Pietroski, P. (2001).
Nature, nurture and universal grammar.
Linguistics and Philosophy, 24:139–186.

Bibliography III

Kirby, S. (2002).
Learning, bottlenecks and the evolution of recursive syntax.

In Briscoe, T., editor, *Linguistic evolution through language acquisition: Formal and computational models*, chapter 6, pages 173–203. Cambridge University Press, Cambridge, UK.

Paull, M. C. and Unger, S. H. (1968).
Structural equivalence of context-free grammars.
Journal of Computer and System Sciences, 2(4):427 – 463.

Pinker, S. (1995).
Language acquisition.
In Osherson, D., Gleitman, L. R., and Liberman, M., editors, *An invitation to cognitive science: Volume 1 Language*, chapter 6, pages 135–182. MIT Press, 2nd edition.

Bibliography IV

Steedman, M. (1996).
Phrasal intonation and the acquisition of syntax.
In Morgan, J. and Demuth, K., editors, *Signal to syntax: Bootstrapping from speech to grammar in early acquisition*, pages 331–342. Lawrence Erlbaum Associates.

Wexler, K. and Culicover, P. W. (1980).
Formal Principles of Language Acquisition.
MIT Press, Cambridge, MA.

Yoshinaka, R. and Kanazawa, M. (2011).
Distributional learning of abstract categorial grammars.
In Pogodalla, S. and Prost, J.-P., editors, *Logical Aspects of Computational Linguistics*, pages 251–266. Springer Berlin Heidelberg.