Grammatical inference and subregular phonology

Adam Jardine
Rutgers University

December 9, 2019 · Tel Aviv University

Overview

"[V]arious formal and substantive universals are intrinsic properties of the language-acquisition system, these providing a schema that is applied to data and that determines in a highly restricted way the general form and, in part, even the substantive features of the grammar that may emerge upon presentation of appropriate data."

Chomsky, *Aspects*

"[I]f an algorithm performs well on a certain class of problems then it necessarily pays for that with degraded performance on the set of all remaining problems." Wolpert and Macready (1997), *NFL Thms.*

- Phonological patterns are governed by restrictive computational universals

- **Formal language theory** gives us tools to discover and state these universals

- **Grammatical inference** allows us to develop and study learning procedures that derive from these universals

- The result is algorithms...
  - that directly connect linguistic universals with learning
  - whose behavior in the general case is well-understood
  - that make typological and psycholinguistic predictions

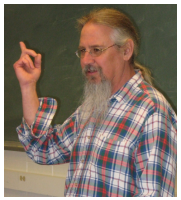
Rough breakdown of course

- **Day 1:** Learning, languages, and grammars
- **Day 2:** Learning strictly local grammars
- **Day 3:** Automata and input strictly local functions
- **Day 4:** Learning functions and stochastic patterns, other open questions

By the end of this course, you should be able to engage with the literature, and start your own research project!

- **Collaborators/Mentors**:


Jeff Heinz
(Stony Brook)


Jim Rogers
(Earlham)


Rémi Eyraud
(Marseilles)


Jane Chandlee
(Haverford)


Kevin McMullin
(Ottowa)

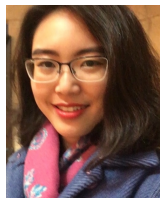
...at Rutgers:

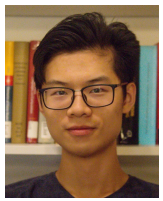

Eileen Blum


Chris Oakden


Nate Koser


Dine Mamadou


Wenyue Hua


Huteng Dai

What is learning?

What is learning?

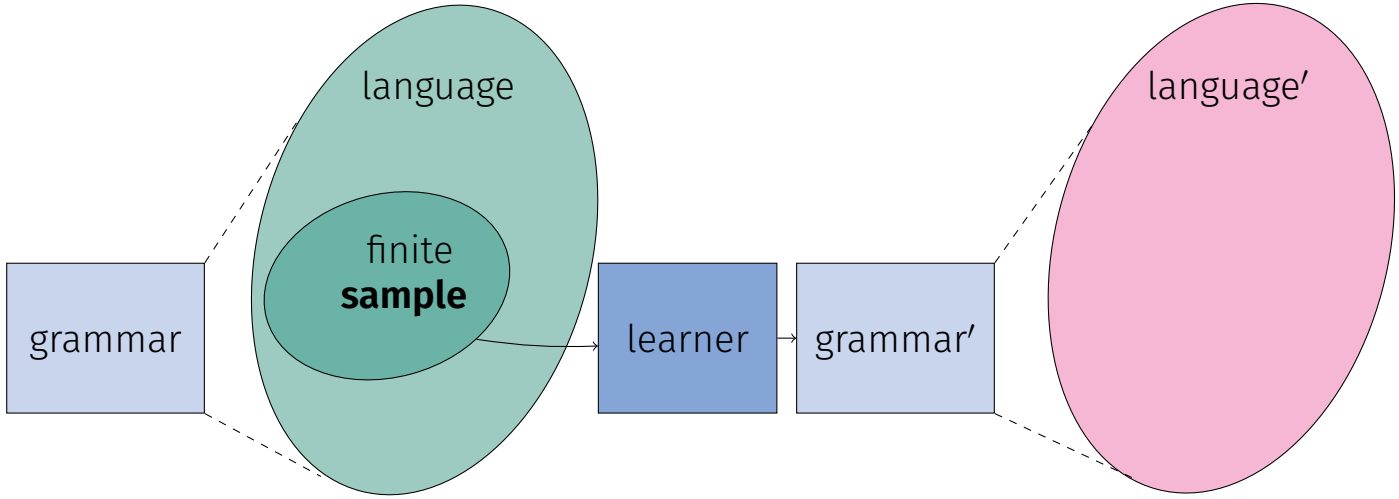
- What do we mean when we say a child/animal/machine has ‘learned’ something?

- What do we mean when we say a child has learned their language?

What is learning?

- What do we mean when we say a child/animal/machine has ‘learned’ something?

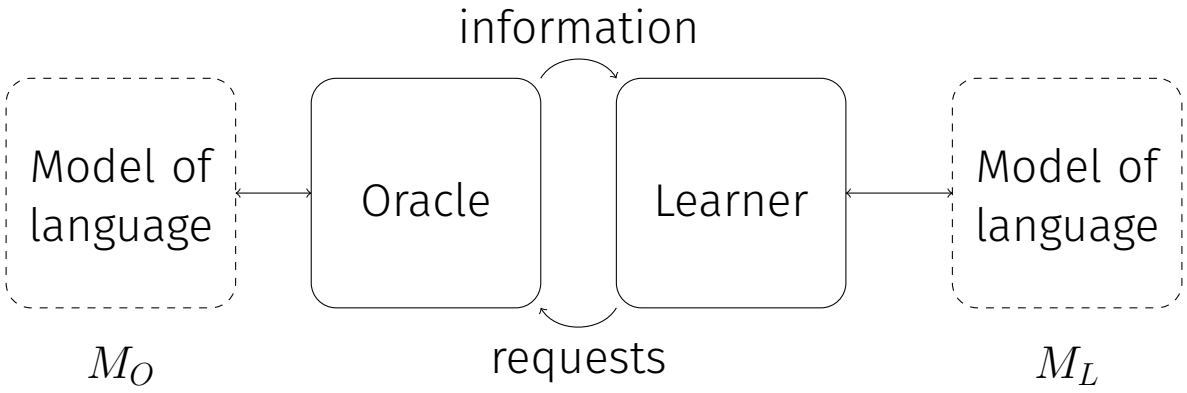
- What do we mean when we say a child has learned their language?



What is learning?

- What is the nature of the sample?
- When is learning successful?

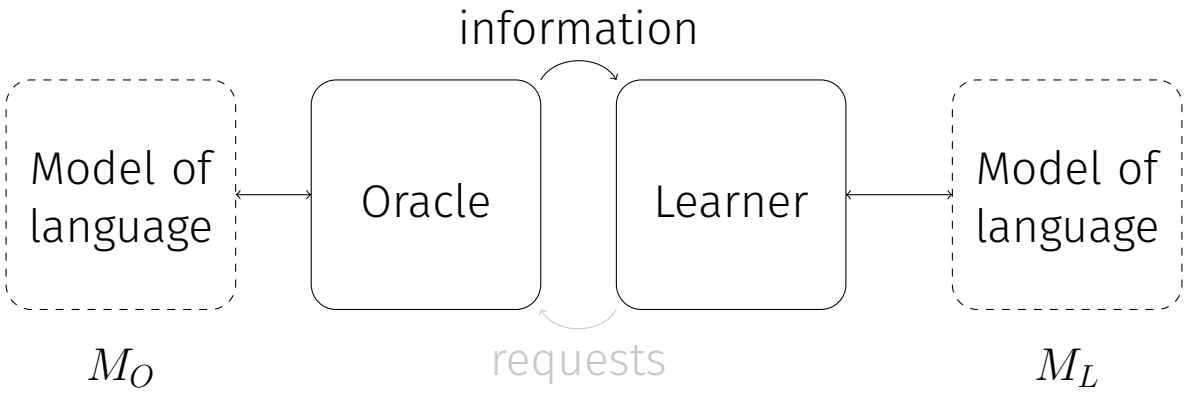
Grammatical inference



(from Heinz et al., 2016)

- **Formal** GI studies **solutions** to specific **learning problems**

Grammatical inference



(from Heinz et al., 2016)

Problem

Given a **positive** sample of a language, return a grammar that describes that language **exactly**

Languages and grammars

What is a pattern?

- Two kinds of phonological patterns:
  - **Well-formedness (phonotactics)**

    ex. *NC̥
  - **Transformations (processes)**

    ex. /NC̥/ → [NC̬]

What is a pattern?

- Well-formedness patterns are **sets**

  ex. *N$\underset{\circ}{\text{C}}$

  well-formed: {an, anda, amba, lalalalanda, blık, ffffff, ...}
  ill-formed: {anta, ampa, lalalalaŋka, ...}

What is a pattern?

- Processes are **relations**

  /NC̥/ → [NC̬]

  $\{(\text{an}, \text{an}), (\text{anda}, \text{anda}), (\text{anta}, \text{anda}), (\text{lalalalampa}, \text{lalalalamba}), \ldots\}$

- This is true regardless of how we describe them

$$\text{C} \rightarrow [\text{+voice}] \;/\; \text{N} \_\_ \quad \approx \quad *\text{NC̥} \gg \textsc{Id}[\pm\text{voice}]$$

What is a pattern?

- We're going to first focus on sets as **formal languages**, and then move on to **(functional) relations**.

Formal languages

- An **alphabet** Σ is a finite set of symbols

$$\{0, 1\}$$

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

$$\{\mathrm{a},\ \mathrm{b},\ \mathrm{c},\ ...,\ \text{æ},\ \beta,\ \text{ɔ},\ ...,\ \mathrm{z}\}$$

$$\{\mathrm{N},\ \mathrm{V},\ \textsc{Adj},\ ...,\ \mathrm{C}\}$$

Formal languages

- A **string** w over Σ is some sequence $\sigma_1\sigma_2...\sigma_n$ of symbols in Σ.
- Σ^* is all strings over Σ

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

$$\Sigma^* =$$

Formal languages

- A **string** w over Σ is some sequence $\sigma_1\sigma_2...\sigma_n$ of symbols in Σ.
- Σ^* is all strings over Σ

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

$$\Sigma^* = \{\ \lambda, a, b, c, aa, ab, ac, ba, bb, bc, ca, cb, cc, aaa, aab, aac, ..., abbaaacccbabacb, ...\ \}$$

Formal languages

- A **(formal) language** some subset $L \subseteq \Sigma^*$

- Some formal languages for $\Sigma = \{a, b, c\}$:
  - $\{b\}$
  - $(ab)^n = \{\lambda, ab, abab, ababab, ...\}$
  - $a^n b^n = \{\lambda, ab, aabb, aaabbb, aaaabbbb, ...\}$
  - ...

Formal languages

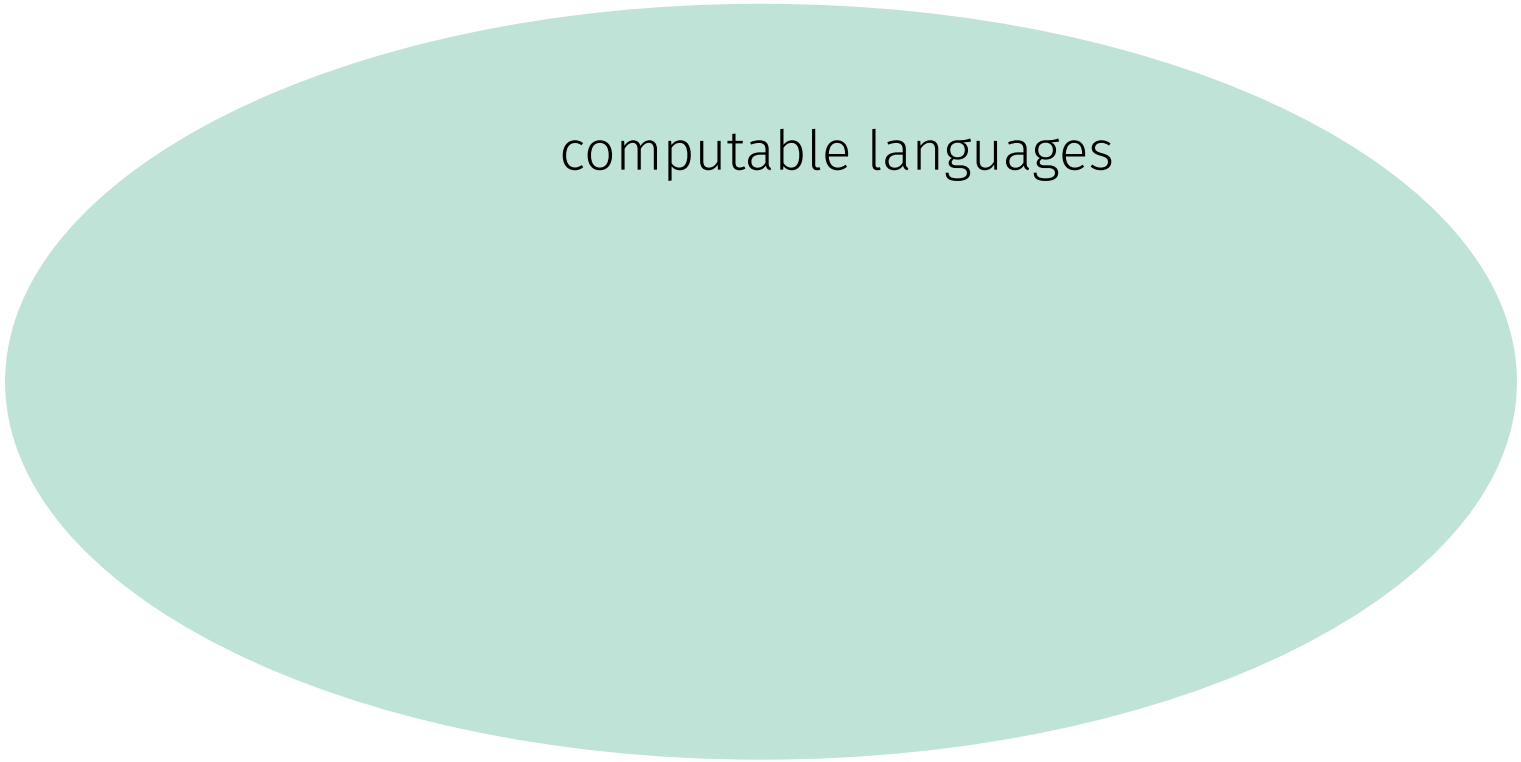
- Equivalently, a formal language maps strings in Σ^* to $\top$ or $\bot$

$(ab)^n$		
λ	$\rightarrow$	$\top$
a	$\rightarrow$	$\bot$
b	$\rightarrow$	$\bot$
aa	$\rightarrow$	$\bot$
ab	$\rightarrow$	$\top$
	$\dots$	
$abaa$	$\rightarrow$	$\bot$
$abab$	$\rightarrow$	$\top$
$abba$	$\rightarrow$	$\bot$
	$\dots$	

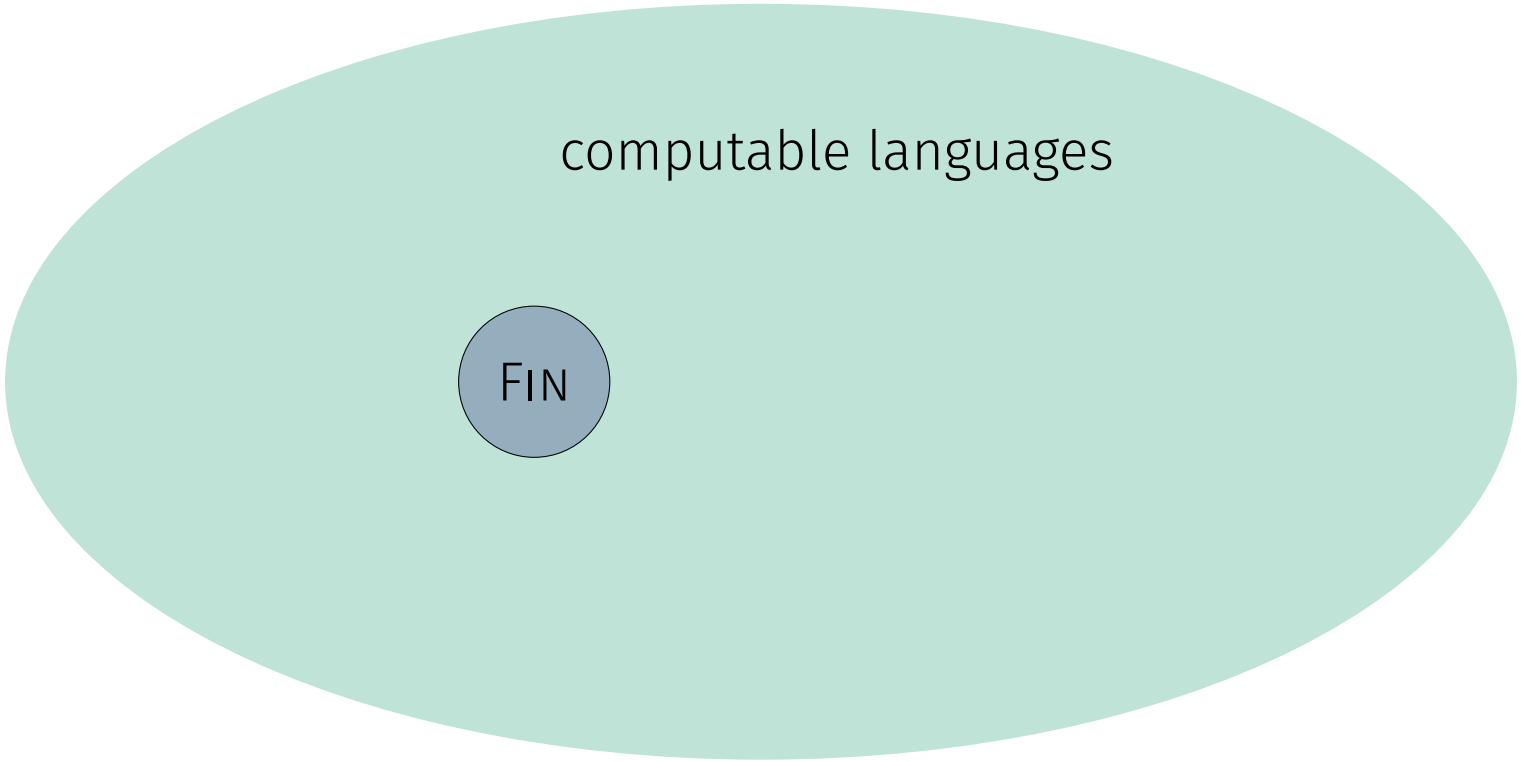
Formal language classes

all possible languages

Formal language classes



Formal language classes



The strictly local languages

The strictly local languages

- How would you compute the *NC̥ language?[1]

$$\{\ \text{an}, \text{anda}, \text{amba}, \text{lalalalanda}, \text{blık}, \text{ffffff}, \ldots\ \}$$

[1] $\Sigma = \{\text{a}, \text{b}, \text{c}, \ldots, \text{æ}, \beta, \text{ɔ}, \ldots, \text{z}\}$

The strictly local languages

- How would you compute the *NC̥ language?[1]

$$\{\ \text{an}, \text{anda}, \text{amba}, \text{lalalalanda}, \text{blık}, \text{ffffff}, \ldots\ \}$$

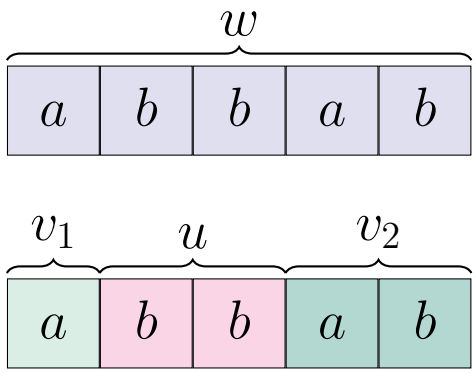
- Make sure the string doesn't contain NC̥ sequences!

$$\{\text{anta}, \text{ampa}, \text{lalalalaŋka}, \ldots\}$$

[1] $\Sigma = \{\text{a}, \text{b}, \text{c}, \ldots, \text{æ}, \beta, \text{ɔ}, \ldots, \text{z}\}$

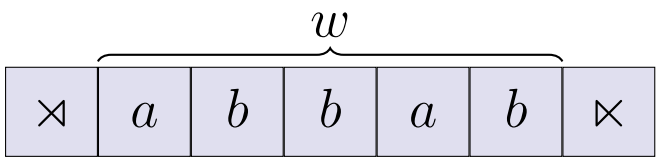
The strictly local languages

- u is a **substring** of w iff $w = v_1 u v_2$

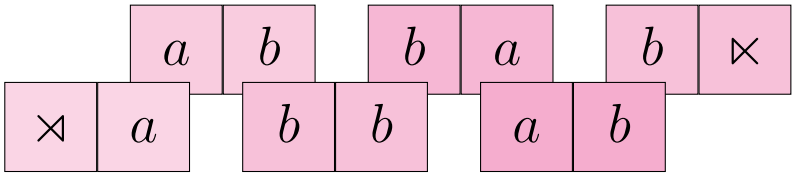


The strictly local languages

- u is a k**-factor** of w iff it is a substring of $\rtimes w \ltimes$ of size k



- $\mathtt{fac}_2(w) =$



The strictly local languages

- A **SL$_k$ grammar** is a set of **forbidden k-factors**

$$G = \{bb, aa\}$$

- $L(G)$ is the set of strings $w \in \Sigma^*$ such that $w \models G$

The strictly local languages

$G = \{bb, aa\}$

w	$w \models G?$
λ	$\top$
a	$\bot$
b	$\bot$
aa	$\bot$
ab	$\top$
aaa	$\bot$
aab	$\bot$
aba	$\bot$

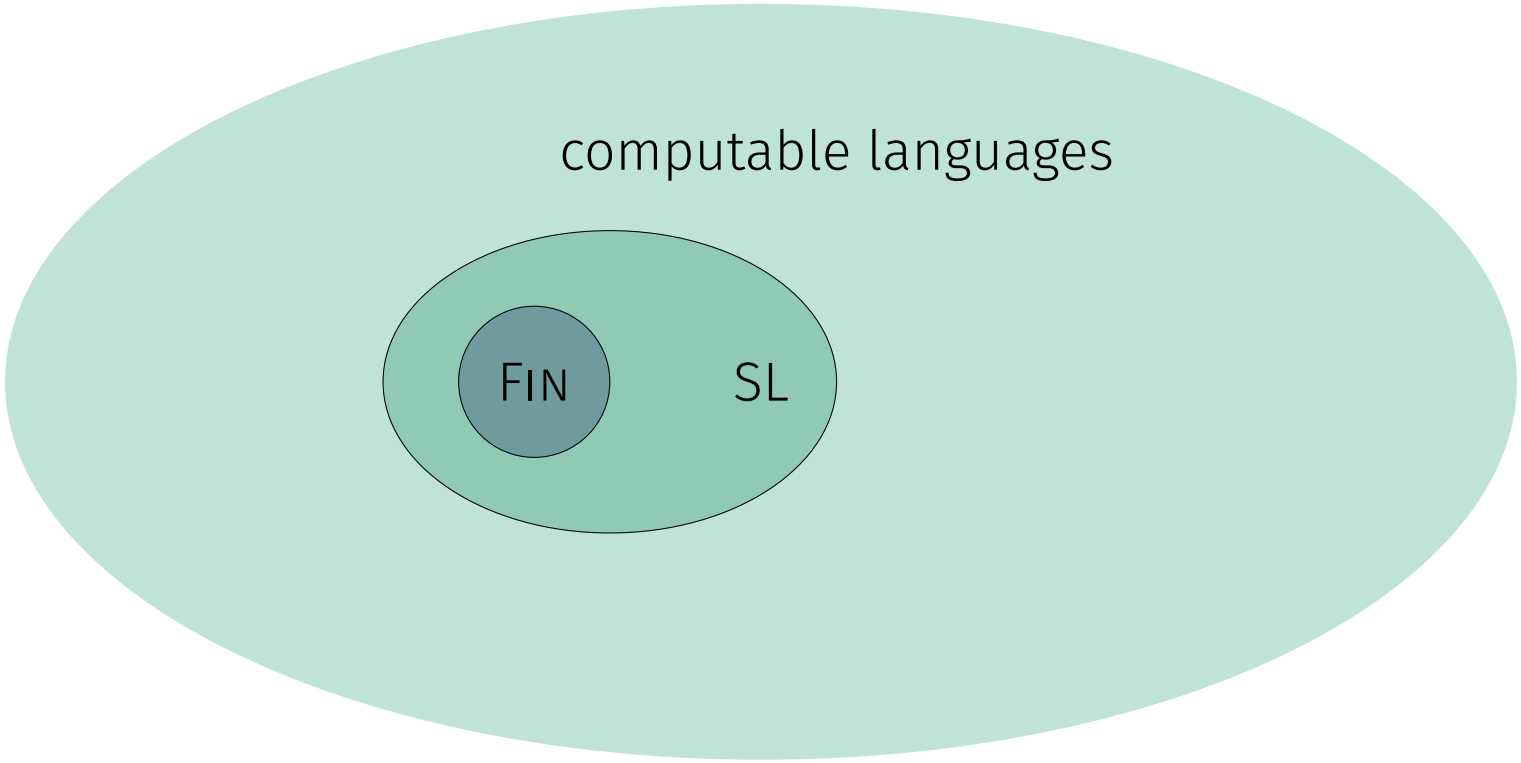
w	$w \models G?$
abb	$\bot$
baa	$\bot$
$aaaa$	$\bot$
...	
$abab$	$\top$
$abba$	$\bot$
$baba$	$\top$
...	

The strictly local languages

- A language is **strictly local** iff it can be described by a SL_k grammar for some k

- Let's do some examples...

The strictly local languages



The strictly local languages

- A good many (but not all!) phonotactics are SL (Heinz, 2010)
- Long-distance phonotactics can be captured with two similar classes:
  - **Strictly piecewise (SP)** languages (Heinz, 2010)
  - **Tier-based strictly local (TSL)** languages (Heinz et al., 2011; McMullin, 2016)
- For a general, formal review see Rogers et al. (2013)

Review

Problem

Given a **positive** sample of a language, return a grammar that describes that language **exactly**

- We're going to learn how SL languages have a solution to this problem
- We're going to learn other language classes that have a similar solution