The typology of phonological generalizations: A computational perspective

Jeffrey Heinz
heinz@udel.edu

University of Delaware

Sommerville College, Oxford
August 12, 2013

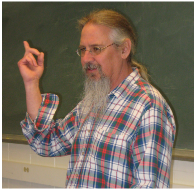
*This research has received support from NSF awards CPS#1035577 and LING#1123692.

In this talk...

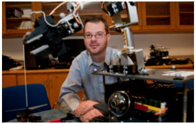
1. Explain why computational characterizations of language patterns matter, especially for typology.
   - They help distinguish language patterns, which provides insight into the nature of those patterns
2. Explain the subregular computational classes that phonological generalizations appear to belong to.
3. Time permitting, provide some pyscholinguistic evidence indicating certain computational boundaries are psychologically real.

Collaborators

- Prof. Jim Rogers (Earlham College)
- Prof. Herbert G. Tanner (UD)
- Prof. Bill Idsardi (UMCP)
- Dr. Regine YeeKing Lai, PhD 2012
- Cesar Koirala (PhD exp. 2013)
- Jane Chandlee (PhD exp. 2014)
- Adam Jardine (PhD exp. 2016)
- Amanda Payne (PhD exp. 2016)
- Huan Luo (PhD exp. 2017)
- Brian Gainor (LDC)
- Sean Wibel (U. Washington)



Jim



Bert

Regine

Cesar

Adam

Amanda

Unpictured
Bill, Jane, Huan, Brian, Sean

Typological questions

1. How are languages the same?
2. How are languages different?
3. Why? (suggest learning – analytic bias)

Position

The phonologies of different languages are measurably different.

- Nonetheless, there are clear *strong, abstract, computational* properties shared by almost all patterns studied so far.
- The exceptions are exciting and interesting for what they reveal.
- For instance suprasegmental patterns allow *more complex* patterns than segmental ones.

Phonological generalizations in this talk

We look at *individual* generalizations:

1. Phonotactics
2. Phonological processes; i.e. *mappings*

Generalizations regarding *contrast* (e.g. Kiparsky this morning, Dresher 2009) are not part of this talk.

Phonotactics - Knowledge of word well-formedness

ptak thole hlad plast sram mgla vlas flitch dnom rtut

Halle, M. 1978. In *Linguistic Theory and Pyschological Reality*. MIT Press.

Phonotactics - Knowledge of word well-formedness

possible English words	impossible English words
thole	ptak
plast	hlad
flitch	sram
	mgla
	vlas
	dnom
	rtut

Phonotactics - Knowledge of word well-formedness

possible English words	impossible English words
thole	ptak
plast	hlad
flitch	sram
	mgla
	vlas
	dnom
	rtut

This knowledge can be modeled as a stringset

Example

All possible English words are in the set; all logically possible, impossible words are out of the set.

This knowledge can be modeled as a stringset

Example

All possible English words are in the set; all logically possible, impossible words are out of the set.

$$mgl$$

This knowledge can be modeled as a stringset

Example

All possible English words are in the set; all logically possible, impossible words are out of the set.

$$mgl \cdot \Sigma^*$$

This knowledge can be modeled as a stringset

Example

All possible English words are in the set; all logically possible, impossible words are out of the set.

$$\overline{mgl \cdot \Sigma^*}$$

This knowledge can be modeled as a stringset

Example

All possible English words are in the set; all logically possible, impossible words are out of the set.

$$\overline{mgl \cdot \Sigma^*} \cap \overline{pt \cdot \Sigma^*} \cap \ldots$$

This knowledge can be modeled as a stringset

Example

Any markedness constraint in Optimality Theory.

All surface forms with zero violations are in the set; all surface forms with nonzero violations are out of the set.

Phonological processes can be modeled as sets of pairs (relations)

Word-final obstruent devoicing

[-sonorant] ⟶ [-voice] / __#

*[+VOICE,-SONORANT]#, MAX-C >> ID(VOICE)

(rat, rat)	(sap, sap)
(rad, rat)	(sab, sap)
...	(sag, sat)
(flugenrat, flugenrat)	...
(flugenrad, flugenrat)	
...	

How can we compare the phonologies of different languages?

How can we compare the phonologies of different languages?

Use size as a proxy for complexity.

Inventories

We can measure the size of the phonemic inventory. It's finite. Larger inventories are more complex. Many more sophisticated methods: TPD, etc.
(Maddieson 1984, 1992, et seq. ... Atkinson 2011)

How can we compare the phonologies of different languages?

Use size as a proxy for complexity.

But what about phonological processes or constraints?

Constraints and processes describe sets of strings and mappings from one set to another. These objects are of *infinite* size so counting doesn't help!

How can we compare the phonologies of different languages?

Use size as a proxy for complexity.

SPE grammars

We can measure the size of a SPE-style grammar by measuring the size of each rule (feature counting). They're finite. Larger grammars are more complex. (Chomsky and Halle 1968)

How can we compare the phonologies of different languages?

Use size as a proxy for complexity.

Principles and Parameters

Count the number of parameters needed to be set.

- For example in some metrical theories, QI stress patterns require fewer parameters to be set than QS patterns because QS patterns need to set parameters for which syllables count as heavy, etc.

How can we compare the phonologies of different languages?

Optimality Theory

In OT, phonologies only differ in their ranking. So all are of equal size.

- Counting the number of “active” constraints may be one way to go, but even understanding the effects of simple constraints interacting can be complicated and difficult.
- Perhaps the most concrete approach in this area is T-orders (Antilla 2008)

How can we compare the phonologies of different languages?

Computational complexity.

There exist independently-motivated, converging mathematical criteria for ordering the complexity of these infinite objects.

- These ideas have been around since the early 1970s (McNaughton and Papert 1971), but were not applied to phonological processes or constraints (until recently).
- These criteria have been argued to be important cognitively (Rogers and Pullum 2011, Rogers et al. 2012, Heinz and Idsardi 2013).

Classifying Sets of Strings

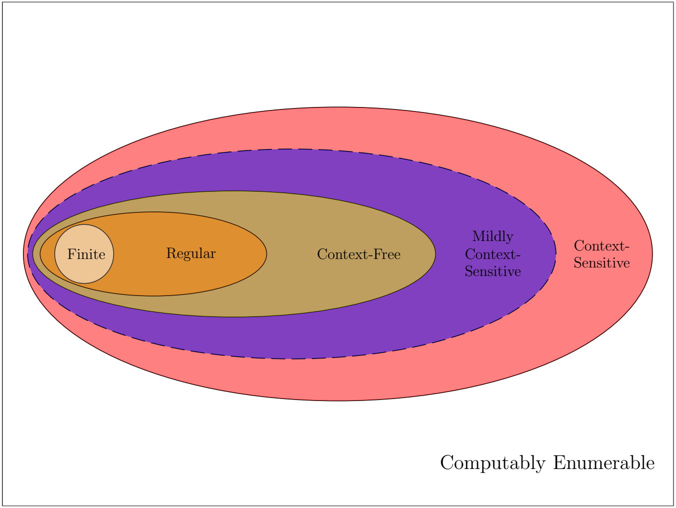
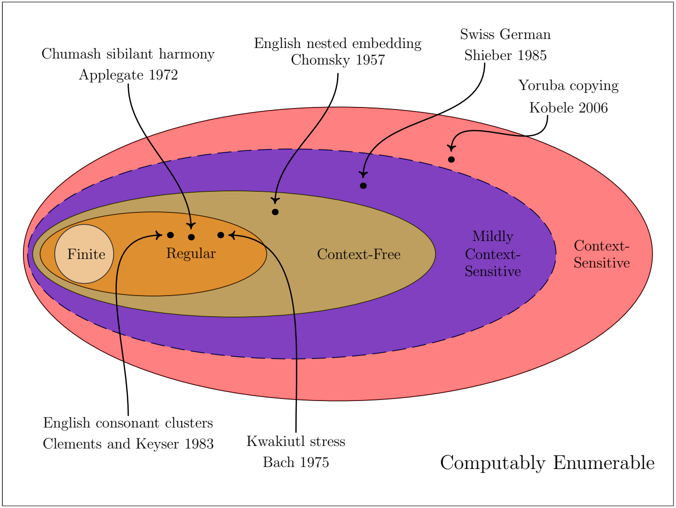


Figure: The Chomsky hierarchy

computably enumerable
|
context-sensitive
|
mildly context-sensitive
|
context-free
|
regular
|
finite

Classifying Sets of Strings



computably enumerable
|
context-sensitive
|
mildly context-sensitive
|
context-free
|
regular
|
finite

Figure: Natural language patterns in the hierarchy.

Phonological mappings are regular (Johnson 1972, Koskenniemi 1983, Kaplan and Kay 1994)

1. Optional, left-to-right, right-to-left, and simultaneous application of SPE-style rules $A \longrightarrow B$ / $C \_\_ D$ (where A,B,C,D are regular sets) *describe regular relations*, provided the rule cannot reapply to the locus of its structural change.
2. Rule ordering is functional composition (finite-state transducer composition).
3. Regular relations are closed under composition.
4. SPE grammars (finitely many ordered rewrite rules of the above type) can describe virtually all attested phonological patterns.

Phonological mappings are regular (Johnson 1972, Koskenniemi 1983, Kaplan and Kay 1994)

1. Optional, left-to-right, right-to-left, and simultaneous application of SPE-style rules $\text{A} \longrightarrow \text{B} \ / \ \text{C} \ \_\_ \text{D}$ (where A,B,C,D are regular sets) *describe regular relations*, provided the rule cannot reapply to the locus of its structural change.
2. Rule ordering is functional composition (finite-state transducer composition).
3. Regular relations are closed under composition.
4. SPE grammars (finitely many ordered rewrite rules of the above type) can describe virtually all attested phonological patterns.

Phonological mappings are regular (Johnson 1972, Koskenniemi 1983, Kaplan and Kay 1994)

1. Optional, left-to-right, right-to-left, and simultaneous application of SPE-style rules $A \longrightarrow B \ / \ C \_\_ D$ (where A,B,C,D are regular sets) *describe regular relations*, provided the rule cannot reapply to the locus of its structural change.
2. Rule ordering is functional composition (finite-state transducer composition).
3. Regular relations are closed under composition.
4. SPE grammars (finitely many ordered rewrite rules of the above type) can describe virtually all attested phonological patterns.

Phonological mappings are regular (Johnson 1972, Koskenniemi 1983, Kaplan and Kay 1994)

1. Optional, left-to-right, right-to-left, and simultaneous application of SPE-style rules $A \longrightarrow B \ / \ C \ \_\_ D$ (where A,B,C,D are regular sets) *describe regular relations*, provided the rule cannot reapply to the locus of its structural change.
2. Rule ordering is functional composition (finite-state transducer composition).
3. Regular relations are closed under composition.
4. SPE grammars (finitely many ordered rewrite rules of the above type) can describe virtually all attested phonological patterns.

Phonological mappings are regular (Johnson 1972, Koskenniemi 1983, Kaplan and Kay 1994)

1. Optional, left-to-right, right-to-left, and simultaneous application of SPE-style rules $A \longrightarrow B \ / \ C \_\_ D$ (where A,B,C,D are regular sets) *describe regular relations*, provided the rule cannot reapply to the locus of its structural change.
2. Rule ordering is functional composition (finite-state transducer composition).
3. Regular relations are closed under composition.
4. SPE grammars (finitely many ordered rewrite rules of the above type) can describe virtually all attested phonological patterns.

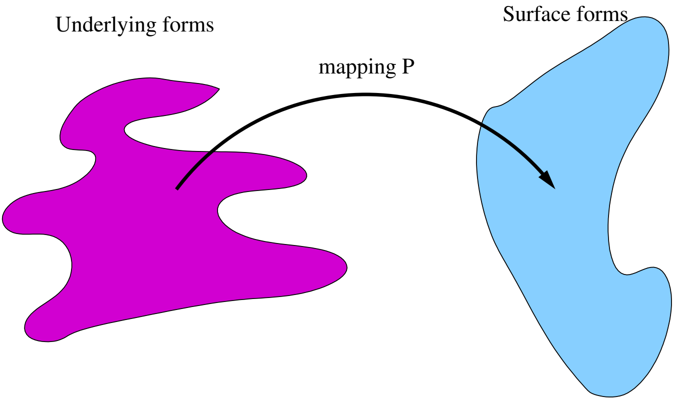
Therefore, phonological mappings are regular relations.

Regardless of whether they are described with SPE, OT, or other formalisms!

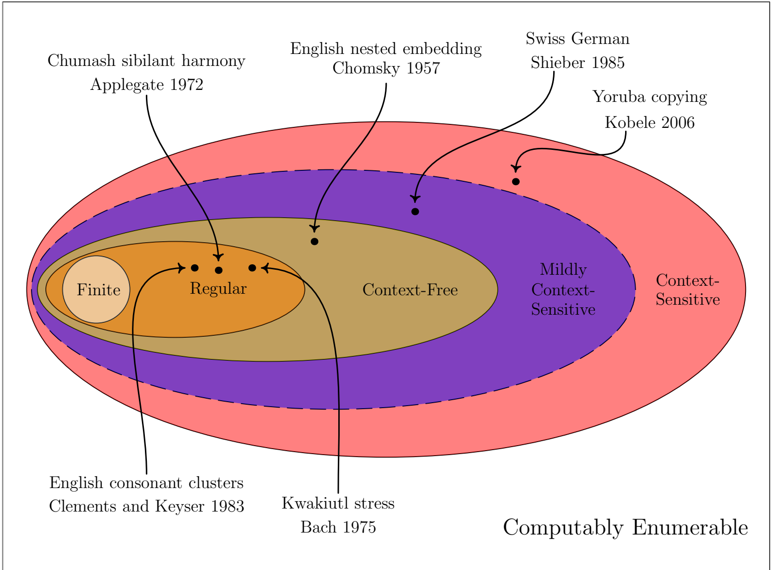
Regular mappings entail regular phonotactics and regular morpheme structure constraints

Theorem (Rabin and Scott 1959)

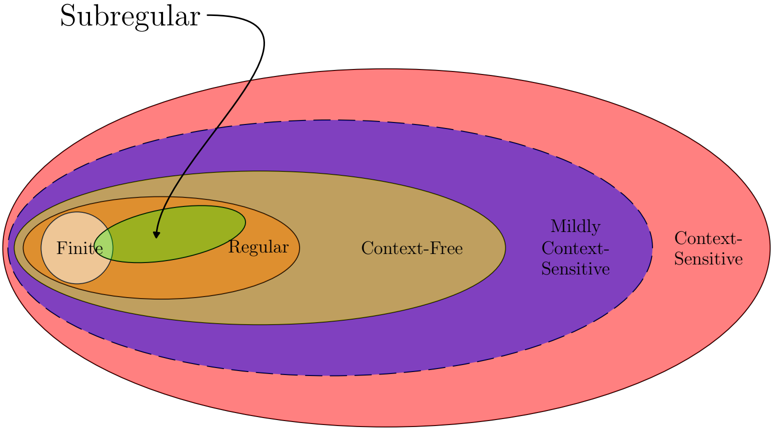
The domain and image of regular relations are regular sets of strings.



"Being regular" is a start, but it is not sufficient to make the distinctions we want



"Being regular" is a start, but it is not sufficient to make the distinctions we want



There is room at the bottom

Better characterizations of phonological patterns

- Allows us to distinguish phonological patterns according to independent measures of complexity
- Leads to stronger universals
- And thus to new hypotheses regarding what a *humanly* possible phonological pattern is

There is room at the bottom

Payoffs for better understanding *learning*

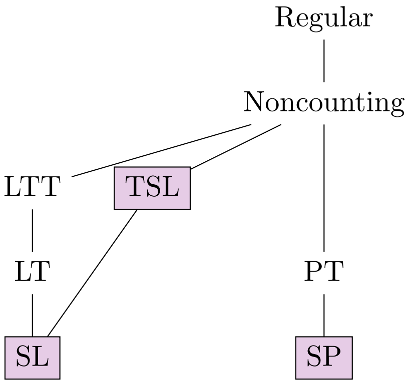
- Are the stronger universals useful for learning (analytic bias)?

There is room at the bottom

Payoffs for natural language processing

- Insights can be incorporated into NLP algorithms
- Factoring and composition may occur with lower complexity

Interesting *subregular* classes of stringsets



(McNaughton and Papert 1971, Rogers et al. 2010, 2012, Heinz et al. 2011)

LTT	Locally Threshold Testable	TSL	Tier-based Strictly Local
LT	Locally Testable	PT	Piecewise Testable
SL	Strictly Local	SP	Strictly Piecewise

Phonotactics - Knowledge of word well-formedness Samala Version

ʃtojonowonowaʃ

stojonowonowaʃ

stojonowonowas

ʃtojonowonowas

pisotonosikiwat

pisotonoʃikiwat

sanisotonosikiwas

ʃanipisotonoʃikiwas

Phonotactics - Knowledge of word well-formedness
Samala Version

possible Samala words	impossible Samala words
ʃtojonowonowaʃ	stojonowonowaʃ
stojonowonowas	ʃtojonowonowas
pistonoskiwat	pisotonoʃikiwat
sanisotonoskiwas	ʃanipisotonoʃikiwas

1. Question: How do Samala speakers know which of these words belong to different columns?
2. By the way, *ʃtoyonowonowaʃ* means 'it stood upright'

(Applegate 1972)

Phonotactics - Knowledge of word well-formedness Samala Version

possible Samala words	impossible Samala words
ʃtojonowonowaʃ	stojonowonowaʃ
stojonowonowas	ʃtojonowonowas
pistonoskiwat	pisotonoʃikiwat
sanisotonoskiwas	ʃanipisotonoʃikiwas

1. Question: How do Samala speakers know which of these words belong to different columns?
2. By the way, *ʃtoyonowonowaʃ* means 'it stood upright'

(Applegate 1972)

Phonotactics - Knowledge of word well-formedness Language X

possible words of Language X	impossible words of Language X
ʃotkoʃ	sotkoʃ
ʃoʃkoʃ	ʃotkos
ʃosokoʃ	ʃoʃkos
soʃokos	soskoʃ
sokosos	
pitkol	
pisol	
piʃol	

Phonotactics - Knowledge of word well-formedness

Language X

possible words of Language X	impossible words of Language X
ʃotkoʃ	sotkoʃ
ʃoʃkoʃ	ʃotkos
ʃosokoʃ	ʃoʃkos
soʃokos	soskoʃ
sokosos	
pitkol	
pisol	
piʃol	

Sibilant sounds which begin and end words must agree (but not ones word medially).

Phonotactics - Knowledge of word well-formedness Language Y

possible words of Language Y	impossible words of Language Y
ʃotkoʃ	ʃoʃkoʃ
sotkoʃ	ʃoskoʃ
ʃotkos	soʃkos
pitkol	ʃoʃkos
soʃkostoʃ	soskoʃ
	soksos
	piskol
	piʃkol

Phonotactics - Knowledge of word well-formedness Language Y

possible words of Language Y	impossible words of Language Y
ʃotkoʃ	ʃoʃkoʃ
sotkoʃ	ʃoskoʃ
ʃotkos	soʃkos
pitkol	ʃoʃkos
soʃkostoʃ	soskoʃ
	soksos
	piskol
	piʃkol

Words must have an *even number* of sibilant sounds.

Typology

Attested Phonotactic Patterns

1. Words don't begin with mgl. (English)
2. Words don't contain both ʃ and s. (Samala)

Unattested Phonotactic Patterns

1. Words don't begin and end with disagreeing sibilants. (Language X = First/Last Harmony)
2. Words don't contain an even number of sibilants. (Language Y = EVEN-Sibilants)

What's the explanation?

Optimality Theory

1. Constraints like *#mgl and *[+strident,α anterior]…[+strident,$-\alpha$ anterior] are part of CON.
2. Constraints like EVEN-Sibilants or *#[+strident,α anterior]…[+strident,$-\alpha$ anterior]# are not.

What's the explanation?

Phonetically-based Phonology (Hayes, Kirchner, Steriade 2004)

1. There are perceptual and/or articulatory reasons for constraints like *#mgl and *[+strident,α anterior]...[+strident,$-\alpha$ anterior].
2. There are no such reasons for constraints like EVEN-Sibilants or #[+strident,α anterior]...[+strident,$-\alpha$ anterior]# .

What's the explanation?

Phonetically-based Phonology (Hayes, Kirchner, Steriade 2004)

1. There are perceptual and/or articulatory reasons for constraints like *#mgl and *[+strident,α anterior]...[+strident,$-\alpha$ anterior].
2. There are no such reasons for constraints like EVEN-Sibilants or #[+strident,α anterior]...[+strident,$-\alpha$ anterior]# .

What are those reasons?

First/Last Harmony

1. Long-distance assimilation is well-attested (Hansson 2001, Rose & Walker 2004)
2. Word edges in phonology are privileged positions (Beckman 1997 Fougeron & Keating 1997, Endress, Nespor & Mehler 2009).

Question

What theory of perception or articulation prevents there from being harmony only in privileged positions?

First/Last Harmony

Are the memory requirements greater?

Given the pattern templates, the answer seems to be no.

	[s]	[ʃ]
[s]	✓	✗
[ʃ]	✗	✓

[. . . ___. . . ___. . .]

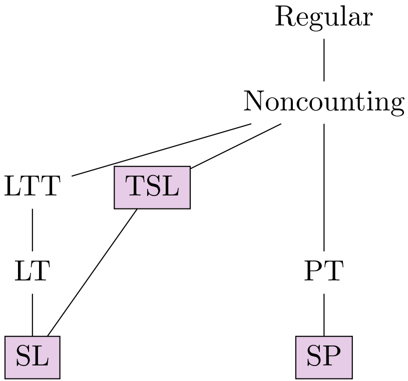
	[s]	[ʃ]
[s]	✓	✗
[ʃ]	✗	✓

[#___. . . ___#]

EVEN-Sibilants

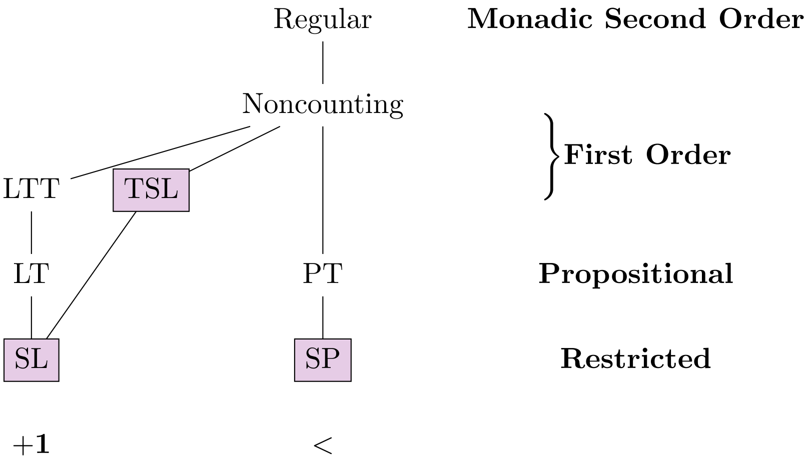
- It's plausible to me at least that perception or articulation should be able to explain the absence of counting mod n patterns in phonology, but I haven't seen any explicit connection.
- Whatever it is, it *should* connect to the computational properties discussed here.

A computational explanation



1. Constraints like *#mgl are Strictly Local.
2. Constraints like *[+strident,α anterior]. . . [+strident,$-\alpha$ anterior] are Strictly Piecewise.
3. Constraints like First Last Harmony are Locally Testable.
4. Constraints like EVEN-Sibilants are Counting (properly regular).

Model Theory and Logical Characterizations



Logical Signatures

The Local Branch (+1)

- (+1) means "successor"
- Literals refer to substrings (contiguous sequences of sounds)

  ex. #mgl, VV, ...

The Piecewise Branch

- (<) means "precedes"
- Literals refer to subsequences (potentially *dis*contiguous sequences of sounds)

  ex. s...s, ʃ...ʃ, a...b ...c...

SL and SP: Restricted Logic

Finitely many conjunctions of negative literals define stringsets.

Strictly Local (+1)

ex. $\neg\#mgl \wedge \neg\#pt \wedge \ldots$
Don't have *#mgl* and don't have *#pt*, …

Strictly Piecewise (<)

ex. $\neg \ldots \text{s} \wedge \neg ʃ \ldots ʃ \wedge \ldots$
Don't have *s…ʃ* and don't have *ʃ…s*, …

LT and PT: Propositional Logic

Well-formed statements of propositional logic with the literals define stringsets.

Locally Testable (+1)

ex. $(\#\text{s} \rightarrow \text{s}\#) \wedge (\#\int \rightarrow \int\#)$
First/Last Harmony

Piecewise Testable (<)

ex. $\text{s}\ldots\text{s} \rightarrow \int\ldots\int$
If a word has a $s\ldots s$ subsequence, it must also have $\int\ldots\int$ subsequence.

LTT and NonCounting: First Order Logic

Well-formed statements of first-order logic with the literals define stringsets. (First order is propositional logic with $\forall, \exists$ quantification over individuals.)

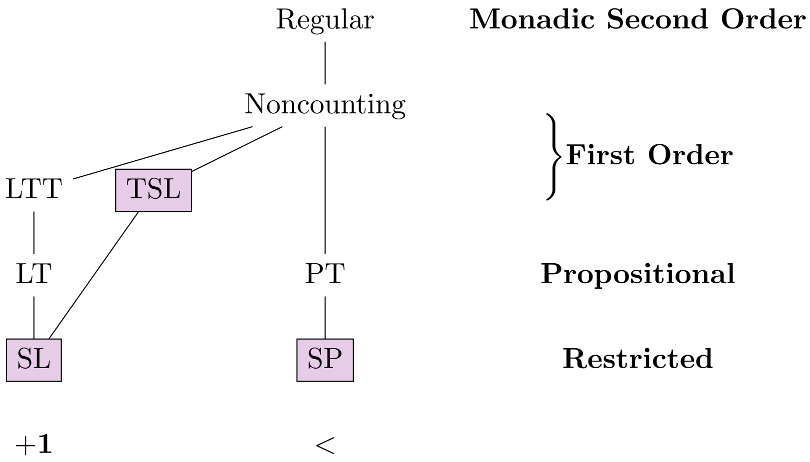
Locally Threshold Testable (+1)

ex. $\exists(x, y, z)[x = p \wedge y = p \wedge z = p \wedge x \neq y \neq z]$
Words must have three [p]s.

Noncounting (<)

ex. $(\forall x)\big[x = s \rightarrow (\exists y)[y = z \wedge y < x]\big]$
If a word has [s] then the [s] must be preceded somewhere by a [z].

LTT and Noncounting



"Successor" is first-order definable from "precedence" but not vice versa, which is why Noncounting properly includes LTT.

Regular: Monadic Second Order Logic

Well-formed statements of monadic second-order logic with literals from either signature (+1) or (<) define stringsets. (Monadic Second Order is propositional logic with $\forall, \exists$ quantification over *sets* of individuals.)

Regular, either (+1) or (<)

ex. Words must have an even number of sibilants.

Tier-based Strictly Local: Ignoring inconsequential events

Finitely many conjunctions of negative literals *over tiers* define stringsets.

Example

Ignoring nonsibilants

tosopiwaʃonikasan
↓
sʃs

Typology of segmental phonotactic patterns

Phonotactic Patterns derived from

- Constraints on consecutive sequences of sounds are SL
- Long-distance consonantal harmony are both SP and TSL
- Long-distance consonantal disharmony are TSL but not SP
- Vowel harmony without neutral vowels are both SP and TSL
- Vowel harmony with opaque vowels are TSL but not SP
- Vowel harmony with transparent vowels are SP and they are TSL *only* if transparent vowels are off the tier

Heinz 2007, 2010, Rogers et al. 2010, Heinz et al. 2011

Typology of (dominant) Stress Patterns

Of the 109 distinct stress patterns studied in Heinz 2009:

- 9 are SL_2.
- 44 are SL_3.
- 24 are SL_4.
- 3 are SL_5. (Asheninca, Bhojpuri, Hindi (Fairbanks))
- 1 is SL_6. (Icua Tupi)
- 28 are not SL_k for any k. (E.g. unbounded patterns)
- 26 of these are either SP+LT or SL+PT.
- 2 are counting (Cairene Arabic and.)

Edlefsen et al. 2009, Rogers et al. 2012, Heinz to appear, Wibel et al. in prep

Learnability

1. SL_k, SP_k, and $\mathrm{TSL}_{T,k}$ are provably identifiable in the limit from positive data by incremental, set-driven, polytime learning algorithms.

   Garcia et al. 1991, Heinz 2007, 2010, Rogers et al. 2010

   Heinz et al. 2011, Heinz et al. 2012

   - k (and T) must be known a priori.
   - k appears to be small for phonology (perhaps ≤ 5).

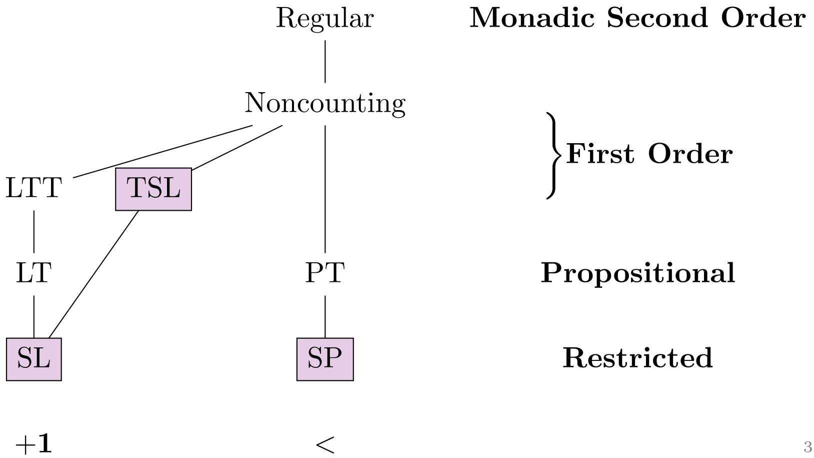
2. Stochastic versions of these algorithms exist which learn probability distributions over stringsets, as well as algorithms incorporating phonological features.

   Jurafsky and Martin 2008, Hayes and Wilson 2008

   Albright 2009, Heinz and Rogers 2010, Heinz and Koirala 2010

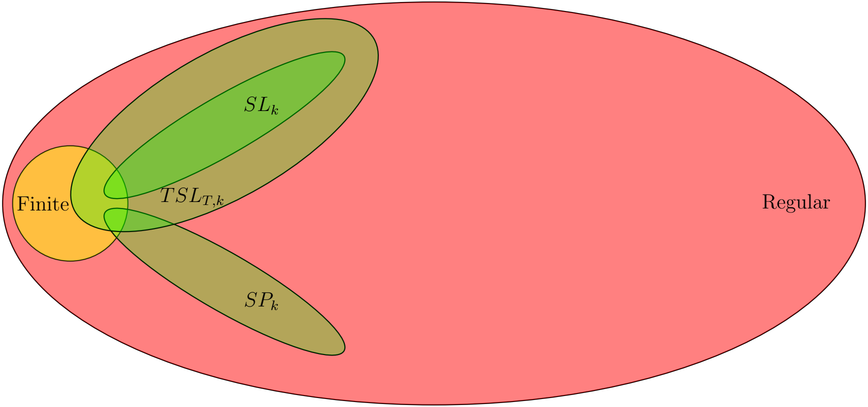
A learning explanation

If people generalize from their phonological experience in the ways suggested by these learning procedures then they can only ever learn SL, SP, or TSL patterns.



A learning explanation

If people generalize from their phonological experience in the ways suggested by these learning procedures then they can only ever learn SL, SP, or TSL patterns.



Phonological processes can be modeled as sets of pairs (relations)

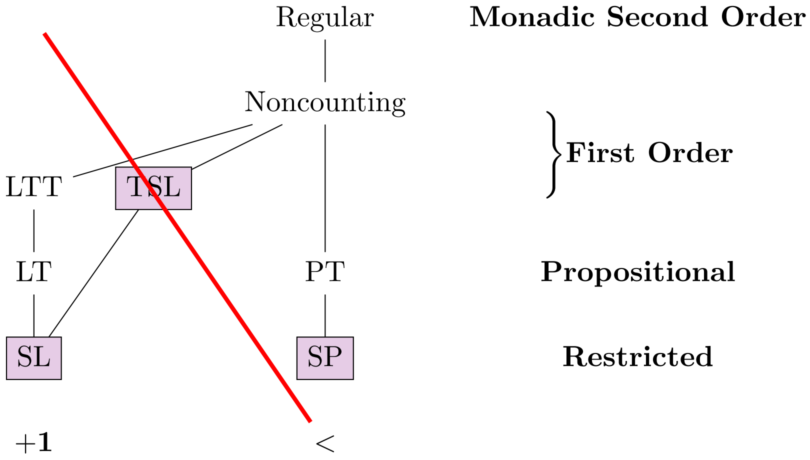
Word-final obstruent devoicing

[-sonorant] $\longrightarrow$ [-voice] / __#

*[+VOICE,-SONORANT]#, MAX-C >> ID(VOICE)

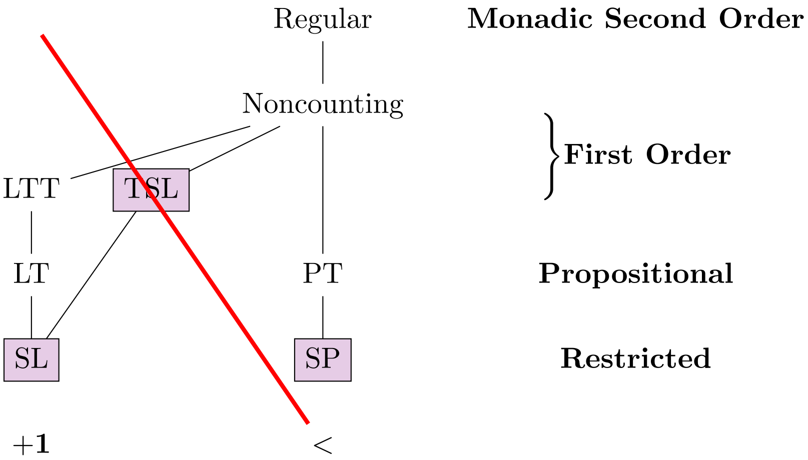
(rat, rat)	(sap, sap)
(rad, rat)	(sab, sap)
...	(sag, sat)
(flugenrat, flugenrat)	...
(flugenrad, flugenrat)	
...	

Regular sets $\neq$ Regular relations



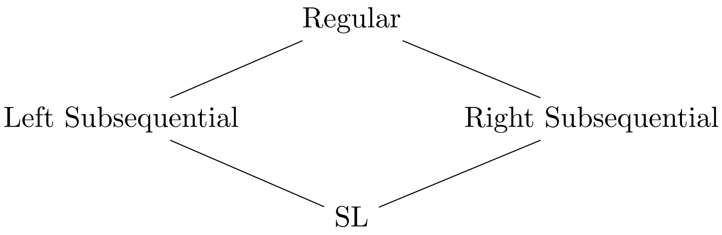
There are no similar subregular hierarchies for relations

Regular sets $\neq$ Regular relations



There are no similar subregular hierarchies for relations (yet)

Subsequential Mappings



- Formally, subsequential functions are those describable with finite-state transducers which process inputs *deterministically* from either left-to-right or right-to-left.
- Informally, left (right) subsequential functions can model mappings where a trigger does not occur arbitrarily far to the right (left) of the target.
- Strictly Local functions are the relational counterpart to SL stringsets (Chandlee in prep)

Survey of results for segmental phonology

1. Mappings describale with SPE-style rules $A \longrightarrow B \,/\, C \_\_ D$, where all strings matching CAD are bounded by length k are SL functions (Chandlee in prep).
2. All the iterative vowel harmony patterns described by Nevins (2010) are left or right subsequential (Gainor et al. 2012, Heinz and Lai 2013).
3. All the synchronically attested metathesis patterns, including long-distance ones, in Beth Hume's NSF-funded metathesis database, are left or right subsequential (Chandlee et al. 2012).
4. The typology of partial reduplication patterns in Riggle (2006) are left or right subsequential (Chandlee and Heinz 2012).
5. The long-distance consonantal dissimilation patterns in Suzuki (1998) and Bennett (2013) are left or right subsequential (Payne, 2012 MS).
6. The long-distance consonantal harmony patterns in Hansson (2001) are left or right subsequential (Luo, 2013 MS) except for Sanskrit n-retroflexion (Schein and Steriade 1985, Graf 2010).

Survey of results for segmental phonology

1. Mappings describale with SPE-style rules $A \longrightarrow B \ / \ C \_\_ D$, where all strings matching CAD are bounded by length k are SL functions (Chandlee in prep).
2. All the iterative vowel harmony patterns described by Nevins (2010) are left or right subsequential (Gainor et al. 2012, Heinz and Lai 2013).
3. All the synchronically attested metathesis patterns, including long-distance ones, in Beth Hume's NSF-funded metathesis database, are left or right subsequential (Chandlee et al. 2012).
4. The typology of partial reduplication patterns in Riggle (2006) are left or right subsequential (Chandlee and Heinz 2012).
5. The long-distance consonantal dissimilation patterns in Suzuki (1998) and Bennett (2013) are left or right subsequential (Payne, 2012 MS).
6. The long-distance consonantal harmony patterns in Hansson (2001) are left or right subsequential (Luo, 2013 MS) except for Sanskrit n-retroflexion (Schein and Steriade 1985, Graf 2010).

Survey of results for segmental phonology

1. Mappings describale with SPE-style rules $A \longrightarrow B$ / C __D , where all strings matching CAD are bounded by length k are SL functions (Chandlee in prep).
2. All the iterative vowel harmony patterns described by Nevins (2010) are left or right subsequential (Gainor et al. 2012, Heinz and Lai 2013).
3. All the synchronically attested metathesis patterns, including long-distance ones, in Beth Hume's NSF-funded metathesis database, are left or right subsequential (Chandlee et al. 2012).
4. The typology of partial reduplication patterns in Riggle (2006) are left or right subsequential (Chandlee and Heinz 2012).
5. The long-distance consonantal dissimilation patterns in Suzuki (1998) and Bennett (2013) are left or right subsequential (Payne, 2012 MS).
6. The long-distance consonantal harmony patterns in Hansson (2001) are left or right subsequential (Luo, 2013 MS) except for Sanskrit n-retroflexion (Schein and Steriade 1985, Graf 2010).

Survey of results for segmental phonology

1. Mappings describale with SPE-style rules $A \longrightarrow B \;/\; C \,\_\_ D$, where all strings matching CAD are bounded by length k are SL functions (Chandlee in prep).
2. All the iterative vowel harmony patterns described by Nevins (2010) are left or right subsequential (Gainor et al. 2012, Heinz and Lai 2013).
3. All the synchronically attested metathesis patterns, including long-distance ones, in Beth Hume's NSF-funded metathesis database, are left or right subsequential (Chandlee et al. 2012).
4. The typology of partial reduplication patterns in Riggle (2006) are left or right subsequential (Chandlee and Heinz 2012).
5. The long-distance consonantal dissimilation patterns in Suzuki (1998) and Bennett (2013) are left or right subsequential (Payne, 2012 MS).
6. The long-distance consonantal harmony patterns in Hansson (2001) are left or right subsequential (Luo, 2013 MS) except for Sanskrit n-retroflexion (Schein and Steriade 1985, Graf 2010).

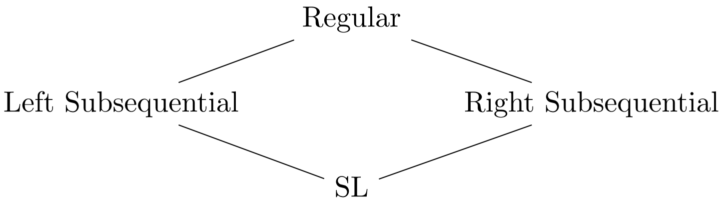
Survey of results for segmental phonology

1. Mappings describale with SPE-style rules $\text{A} \longrightarrow \text{B} \;/\; \text{C} \_\_\text{D}$, where all strings matching CAD are bounded by length k are SL functions (Chandlee in prep).
2. All the iterative vowel harmony patterns described by Nevins (2010) are left or right subsequential (Gainor et al. 2012, Heinz and Lai 2013).
3. All the synchronically attested metathesis patterns, including long-distance ones, in Beth Hume's NSF-funded metathesis database, are left or right subsequential (Chandlee et al. 2012).
4. The typology of partial reduplication patterns in Riggle (2006) are left or right subsequential (Chandlee and Heinz 2012).
5. The long-distance consonantal dissimilation patterns in Suzuki (1998) and Bennett (2013) are left or right subsequential (Payne, 2012 MS).
6. The long-distance consonantal harmony patterns in Hansson (2001) are left or right subsequential (Luo, 2013 MS) except for Sanskrit n-retroflexion (Schein and Steriade 1985, Graf 2010).

Survey of results for segmental phonology

1. Mappings describale with SPE-style rules $A \longrightarrow B \;/\; C \_\_ D$, where all strings matching CAD are bounded by length k are SL functions (Chandlee in prep).
2. All the iterative vowel harmony patterns described by Nevins (2010) are left or right subsequential (Gainor et al. 2012, Heinz and Lai 2013).
3. All the synchronically attested metathesis patterns, including long-distance ones, in Beth Hume's NSF-funded metathesis database, are left or right subsequential (Chandlee et al. 2012).
4. The typology of partial reduplication patterns in Riggle (2006) are left or right subsequential (Chandlee and Heinz 2012).
5. The long-distance consonantal dissimilation patterns in Suzuki (1998) and Bennett (2013) are left or right subsequential (Payne, 2012 MS).
6. The long-distance consonantal harmony patterns in Hansson (2001) are left or right subsequential (Luo, 2013 MS) except for Sanskrit n-retroflexion (Schein and Steriade 1985, Graf 2010).

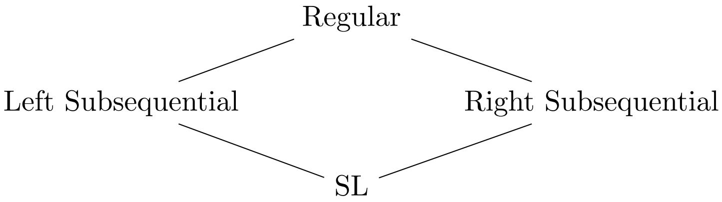
What is not subsequential?



Some ‘pathological patterns’ predicted within certain OT typologies:

7. The “Majority Rules” vowel harmony pattern is not even regular (Riggle 2004).
8. The “Sour Grapes” vowel harmony pattern is neither left nor right subsequential (Heinz and Lai 2013).

What is not subsequential?

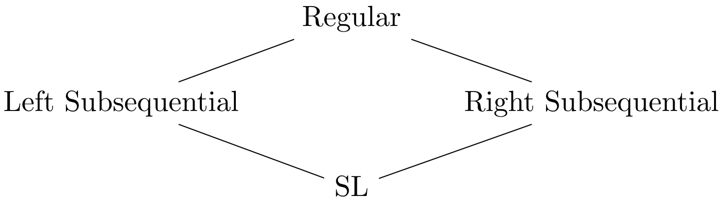


9. Dominant/recessive and stem-control analyses of vowel harmony are also neither left nor right subsequential.

- But they are the composition of a left and right subsequential function without mark-up.
- Majority Rules and Sour Grapes cannot be so described.

(Heinz and Lai 2013)

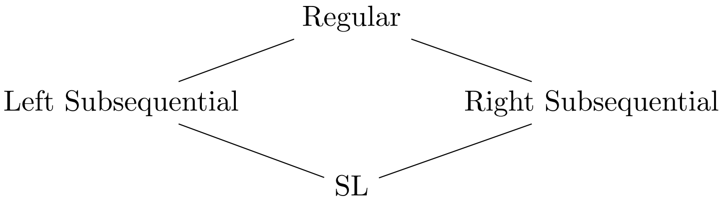
What is not subsequential?



10. Unbounded Tone Plateauing is neither left nor right subsequential (Jardine, 2013 MS).

Paraphasing Yip (2001) and Hyman (2011): "Tone can do everything segmental phonology can do and more!"

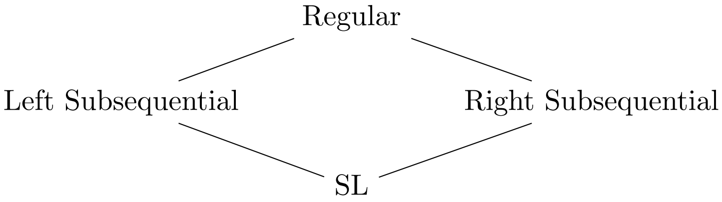
What is not subsequential?



10. Unbounded Tone Plateauing is neither left nor right subsequential (Jardine, 2013 MS).
11. The vowel harmony pattern in Yaka (Hyman 1998), (NOT part of Nevins 2010).

Paraphasing Yip (2001) and Hyman (2011): “Tone can do everything segmental phonology can do and more!”

What is not subsequential?



Generally, processes where there are two triggers on opposite sides of the target and both can be arbitrarily distant from the target are not subsequential.

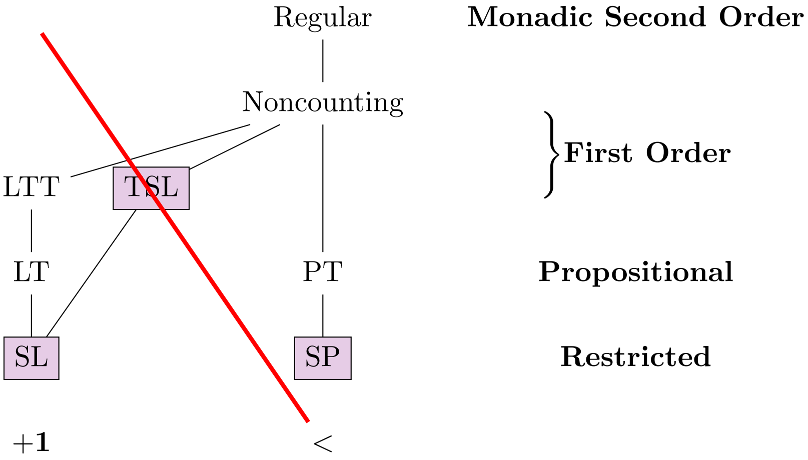
Learning Processes

- Left and Right Subsequential functions are learnable from positive data in theory . . . (Oncina et al. 1993)
- . . . But not in practice for phonological rules (Gildea and Jurafsky 1996)
- Chandlee and Koirala (2013, PLC) show that algorithms for learning SL functions are practical for learning learning locally-triggered phonological processes.
- Chandlee and Jardine are presenting updates on this at the phonology 2013 meeting at UMass.

Open questions/ Future work

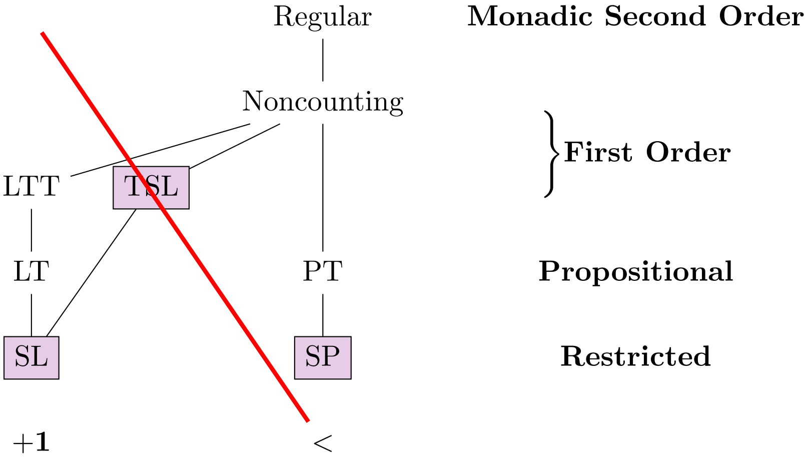
- Can SP *mappings* be defined in a natural way and would they then model the long-distance processes?
- How can such mappings be learned?
- How about the rest of it? LT, PT, LTT, Noncounting, TSL, ...?
- How can *interacting* processes be learned?

Regular sets ≠ Regular relations



There are no similar subregular hierarchies for relations

Regular sets may inform Regular relations



There are no similar subregular hierarchies for relations (yet)

Can college students learn First/Last Harmony?

Artificial language learning experiments

1. Subjects are exposed to training items (exemplars of a pattern).
2. Subjects are tested on novel items in a forced-choice task, some which exemplify the target pattern, and some which don't.
    - "Which word do you think more likely belongs to the language you just heard?"

Comparative artificial language learning experiments (Lai 2012, under review)

		Pattern Type	
		SL/SP/TSL (SH)	non-SL/SP/TSL (FL)
Outcomes	1	Learnable	Learnable
	2	Unlearnable	Unlearnable
	3	Learnable	Unlearnable
	4	Unlearnable	Learnable

- It is not possible to test for the *unlearnability* of some pattern.
- Instead, Lai (2012) tests the *comparative* learnability.

Methodology (Lai 2012, under review)

Subjects

66 adult native English speakers

All Stimuli

Training and test items were $C_1V.C_2V.C_3VC_4$ (tryisyllabic), containing 3 sibilants.

- C_1 & C_4: sibilants
- C_2 & C_3: either sibilant or [k]

Training

40 words × 5 repetitions = 200 words. Subjects listened and repeated each word. 3 Training Conditions:

SH: [s. . . s. . . s], [ʃ. . . ʃ. . . ʃ]

FL: [s. . . s. . . s], [ʃ. . . ʃ. . . ʃ], [s. . . ʃ. . . s], [ʃ. . . s. . . ʃ]

Control: No training

Testing (Lai 2012, under review)

Two alternative forced choice

Words were presented in pairs (minimally different)

E.g. [sakisis] vs. [ʃakisis]

- In the FL and SH conditions, subjects had to answer "Which word do you think belongs to the language you just heard?"
- In the control condition, they were asked "Which word do you prefer?"
- 48 pairs in total

Stimuli (Lai 2012, under review)

Three Stimuli Types

FL/SH	[sokosos]
*FL/*SH	[sokosoʃ]
FL/*SH	[sokoʃos, ʃokosoʃ]

- These 3 types of stimuli were pitted against each other and generated 3 types of pairings.
  (a) FL/*SH vs. *FL/*SH (also includes *FL/*SH vs. FL/*SH)
  (b) FL/SH vs. *FL/*SH (also includes *FL/*SH vs. FL/SH)
  (c) FL/*SH vs. FL/SH (also includes FL/SH vs. FL/*SH)
- The order of presentation was counter-balanced across types

Data Analysis (Lai 2012, under review)

The dependent variable for each pairing is different, so they were analyzed separately

(a) FL/*SH vs. *FL/*SH
Rate of choosing FL/*SH

(b) FL/SH vs. *FL/*SH
Rate of choosing FL/SH

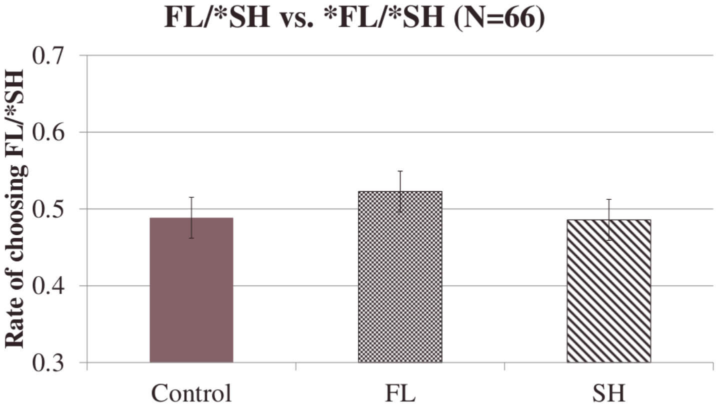
(c) FL/*SH vs. FL/SH
Rate of choosing FL/SH

Predictions (Lai 2012, under review)

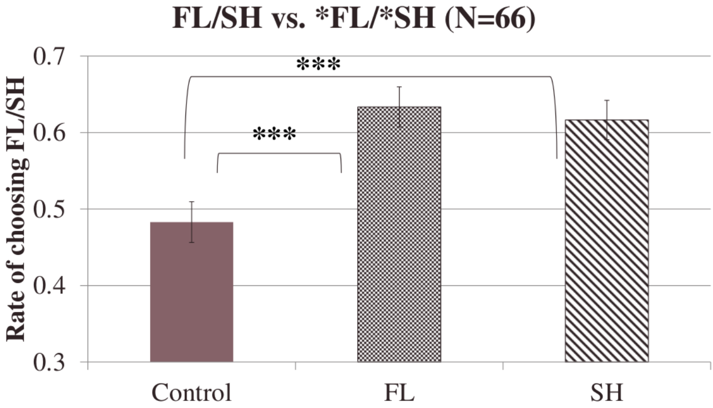
If subjects internalized the pattern they were exposed to during training, they should perform as follows.

	Pairs		
Conditions	FL/*SH vs. *FL/*SH	FL/SH vs. *FL/*SH	FL/SH vs. FL/*SH
SH	No preference	FL/SH	FL/SH
FL	FL/*SH	FL/SH	No preference
Control	No preference	No preference	No preference

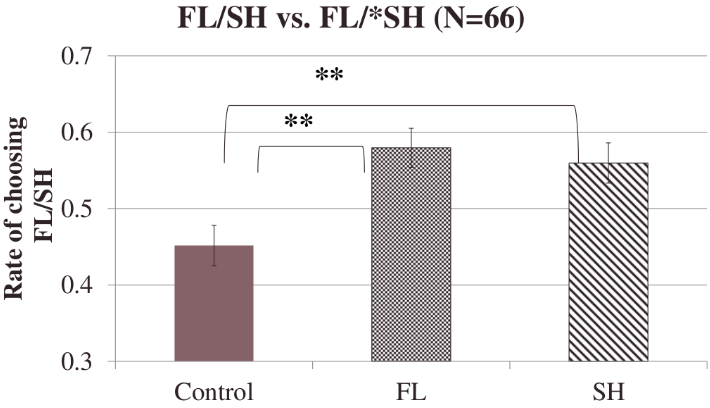
Results (Lai 2012, under review)



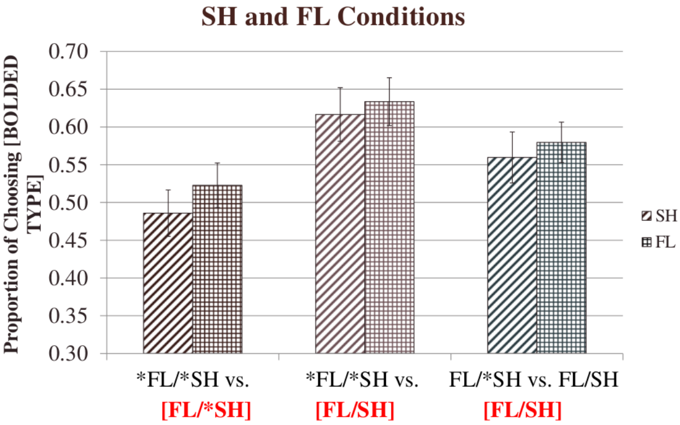
Results (Lai 2012, under review)



Results (Lai 2012, under review)



Results (Lai 2012, under review)



Discussion (Lai 2012, under review)

1. Subjects in the SH condition behaved as if they had internalized the SH pattern.
2. Subjects in the FL condition behaved as if they had internalized the SH pattern, *not* the FL pattern!

Discussion (Lai 2012, under review)

1. Subjects in the SH condition behaved as if they had internalized the SH pattern.
2. Subjects in the FL condition behaved as if they had internalized the SH pattern, *not* the FL pattern!

		Pattern Type	
		SL/SP/TSL (SH)	non-SL/SP/TSL (FL)
Outcomes	1	Learnable	Learnable
	2	Unlearnable	Unlearnable
	3	Learnable	Unlearnable
	4	Unlearnable	Learnable

Discussion (Lai 2012, under review)

1. Subjects in the SH condition behaved as if they had internalized the SH pattern.
2. Subjects in the FL condition behaved as if they had internalized the SH pattern, *not* the FL pattern!

Conclusion

The heavy bias for SH can be understood if only phonotactic patterns which can be modeled as SL, SP, or TSL stringsets are the humanly learnable ones.

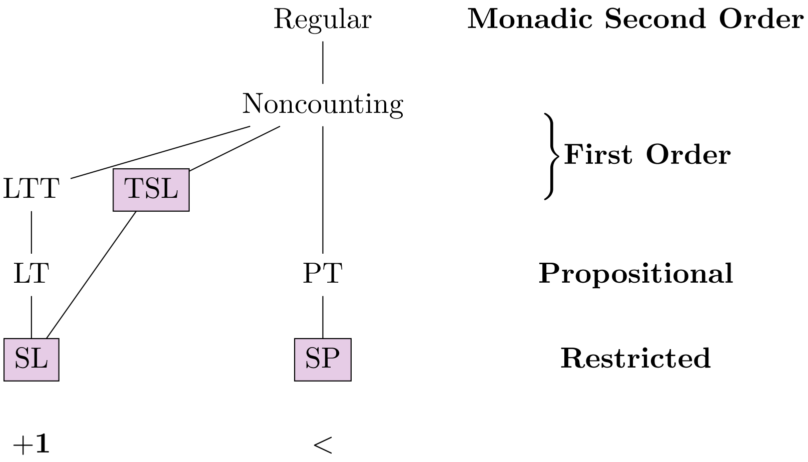
Conclusions (verbose version)

1. Computational analysis of stringsets and string mappings (ongoing) is yielding natural classes of pattern complexity which both identify how phonologies *differ* and how they are the *same*.
2. It provides an "encyclopedia of categories" of logically possible phonologies with which the "encyclopedia of actual phonologies" can be compared.
3. When *individual* phonological generalizations are studied through this lens, strong computational properties are revealed, which:
   3.1 broadly make the right kind of cuts between attested and unattested patterns.
   3.2 broadly draw interesting distinctions between segmental and suprasegmental phenomenon.
   3.3 are strong enough to make learning possible from reasonable amounts of positive evidence.
   3.4 make experimentally testable predictions about humanly possible phonological generalizations.

Conclusions (brief version)

1. Computational properties help determine “the what” in typological analysis, and
2. to the extent to which these properties aid learning help explain “the why.”

THANK YOU


Regular
Monadic Second Order
Noncounting
First Order
LTT
TSL
LT
PT
Propositional
SL
SP
Restricted
+1
<