Phonotactic learning and the conjunction of Tier-based Strictly Local languages

Kevin McMullin and Blake Allen
Department of Linguistics
University of British Columbia

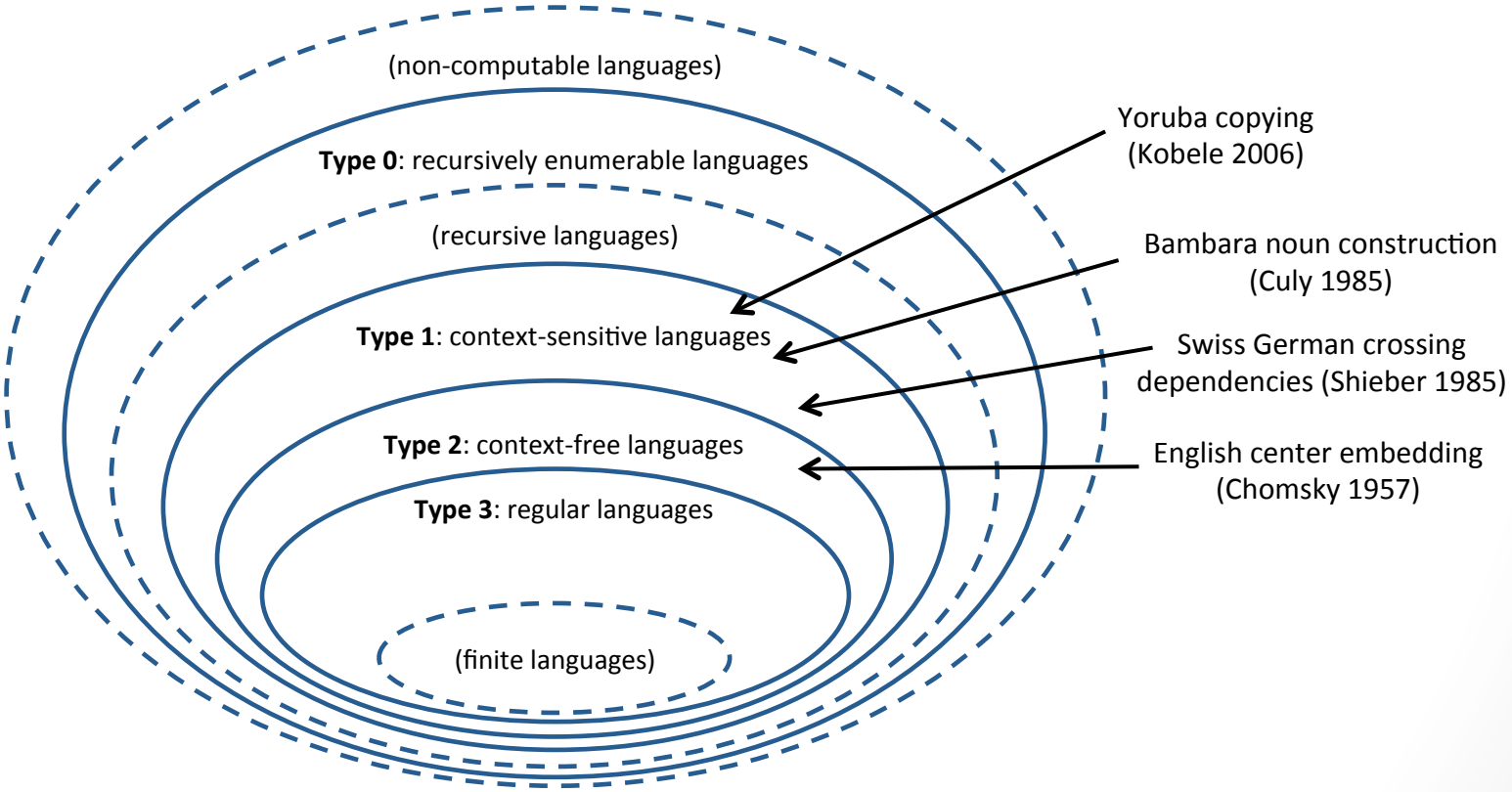
89th Annual Meeting of the Linguistic Society of America
January 8-11, 2015, Portland, OR

Project goals

- Determine the formal complexity of attested long-distance phonotactic patterns
- Model the properties of these patterns with a class of formal languages and accompanying grammars
- Design a learning algorithm that can acquire any language within the hypothesized class

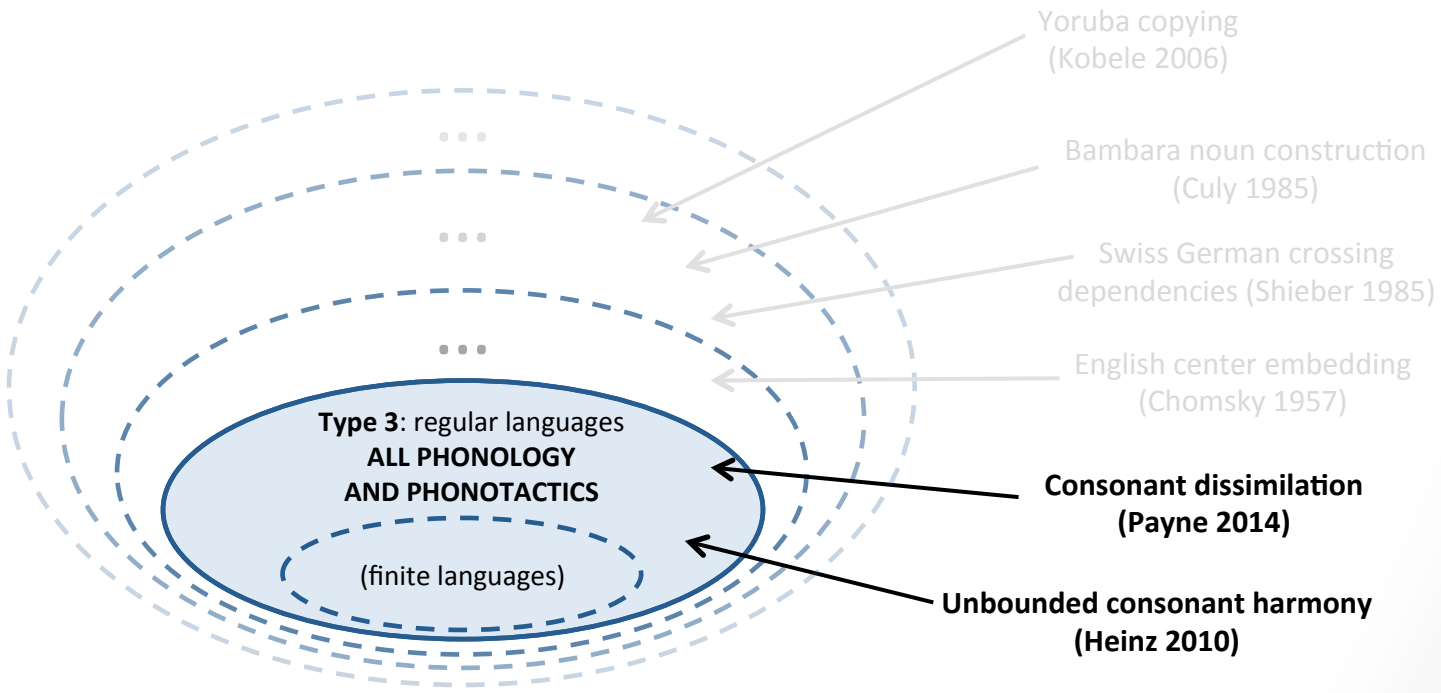
Formal language theory

- Phonotactic patterns: stringsets modeled as miniature languages
- The Chomsky Hierarchy (Chomsky 1956)
  - Classification for the computational complexity of a language based on the type of grammar required to generate it



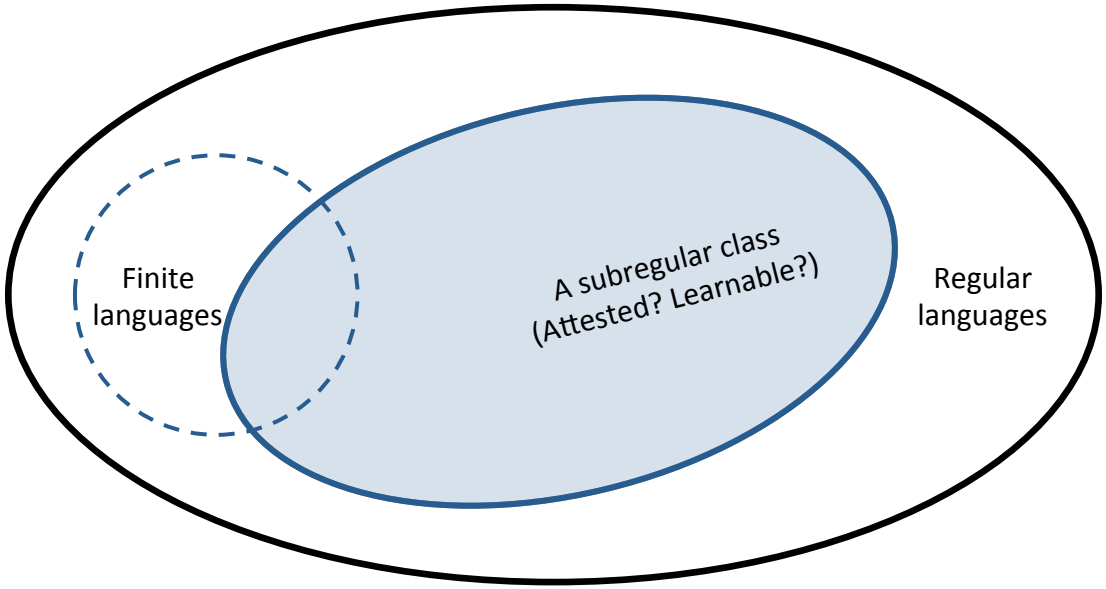
Phonology is regular

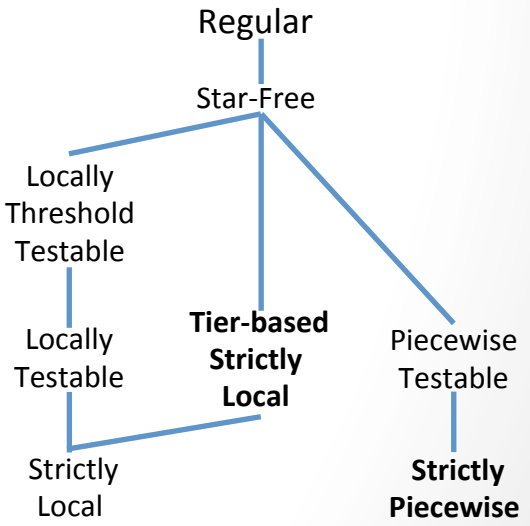
- Virtually all phonological mappings are regular relations (Johnson 1972; Kaplan and Kay 1994)
- Any stringsets generated by these relations are also regular (Rabin and Scott 1959)
  - Surface phonotactics are regular



Subregular hierarchy

- Not all regular languages are attested as phonotactic patterns
- The regular languages are not Gold-learnable
  - Identifiable in the limit from positive data (Gold 1967)
- Instead, consider a proper subset of the regular region
  - We know a lot about the formal properties of some subregular classes (See e.g., Heinz 2010; Heinz, Rawal, and Tanner 2011; Rogers and Pullum 2011)





(Adapted from Heinz et al. 2011)

Outline of this talk

- Long-distance phonotactics in the subregular hierarchy
  - Unbounded patterns as Strictly 2-Piecewise languages
  - Motivating an alternative 'tier-based' approach
- Conjunctions of Tier-based Strictly 2-Local languages
  - Multiple phonotactic dependencies in the same language
- Learning algorithms for languages within this class
  - Brute-force and step-wise guidance through a lattice-structure space
- Ongoing research and further questions

Strictly *k*-Piecewise languages (SP_k)

(Heinz 2010)

- Unbounded co-occurrence restrictions
  - SP_2 patterns prohibit *x…y* subsequences
  - **Unbounded harmony can be described as SP_2** (*l…r, *r…l)
- SP_k languages are Gold-learnable (Heinz 2010)
  - With an algorithm that records all encountered *k*-subsequences
    - ‘abcd’ for *k* = 2 → {a…b, a…c, a…d, b…c, b…d, c…d}
  - The grammar is a set of permitted (prohibited) subsequences
    - {*l…r, *r…l} for unbounded liquid harmony
- One grammar can define multiple Strictly 2-Piecewise patterns
  - For example, a language with both liquid and vowel harmony
    - {*l…r, *r…l, *i…u, *u…i}

Evidence against the SP_2 account

- Dissimilation with blocking is attested
  - Example: Latin liquid dissimilation (Jensen 1974; Odden 1994)
    - /lun-alis/ → [lun-aris] *l…l is prohibited
    - /flor-alis/ → [flor-alis] *[flor-aris] l…l if [r] intervenes
- **Unbounded dependencies with blocking are not SP_2**
  - The prohibited subsequence will be observed when a blocker intervenes
- Consonant harmony with blocking is also attested
  - Kinyarwanda (Walker and Mpiranya 2005), some Berber dialects (Hansson 2010b), and Slovenian (Jurgec 2011)

Tier-based Strictly *k*-Local languages (TSL_k)

(Heinz, Rawal, and Tanner 2011)

- Latin liquid dissimilation can be described as a restriction on [ll] or [rr] sequences when considering only liquids
  - /lun-alis/ → [lun-aris] is now: /ll/ → [lr]
  - /flor-alis/ → [flor-alis] is now: /lrl/ → [lrl]
- Co-occurrence restrictions with blocking are Tier-based Strictly 2-Local (Heinz et al. 2011)
  - The grammar prohibits {*ll} on the liquid tier
  - [lrl] does not violate these restrictions, since [lr], [rl] are permitted
- TSL_2 grammars are defined as a 2-tuple G=(T, S)
  - The tier T is the relevant subset of the segment inventory
  - S is the set of permitted 2-factors ($\overline{S}$ for prohibited)
- e.g., T = {l,r} and $\overline{S}$ = {*ll, *rr}

Advantages of TSL_2 approach

- McMullin and Hansson (2014) argue that the typology of long-distance consonant interactions is closely approximated by TSL_2
  - Agreement and disagreement at varying levels of locality, with or without blocking
  - TSL_2 languages avoid pathologies of other approaches
- TSL_2 languages are Gold-learnable
  - Given a finite segment inventory, there are finitely many possible TSL_2 grammars
  - A learner that picks randomly would eventually be correct
- Jardine (2014) provides an algorithm that learns Tier-based Strictly 2-Local languages efficiently
  - Latin liquid dissimilation, Finnish vowel harmony

One TSL_2 grammar is not sufficient

- The grammar of a Tier-based Strictly 2-Local language defines co-occurrence restrictions for a particular subset of the inventory
- Phonotactic dependencies defined on multiple tiers cannot by represented by one TSL_2 grammar
  - A language with both liquid harmony and vowel harmony
    - G_1: T = {l,r} and $\overline{S}$ = {*lr, *rl}
    - G_2: T = {i,a} and $\overline{S}$ = {*iu, *ui}
  - Liquid harmony in a language with CV syllable structure
    - G_1: T = {l,r} and $\overline{S}$ = {*lr, *rl}
    - G_2: T = {all segments} and $\overline{S}$ = {*CC, *VV}

Conjunctions of TSL_2 languages

- We propose that a phonotactic pattern can be defined as the conjunction of multiple Tier-based Strictly 2-Local languages
- A word is grammatical if and only if it is grammatical on every tier
  - A language with both liquid harmony and vowel harmony
    - G: T_1 = {l,r}, $\overline{S}_1$ = {*lr, *rl} and T_2 = {i,a}, $\overline{S}_2$ = {*iu, *ui}
  - Liquid harmony in a language with CV syllable structure
    - G: T_1 = {l,r}, $\overline{S}_1$ = {*lr, *rl} and T_2 = {all segments}, $\overline{S}_2$ = {*CC, *VV}
- A grammar for a conjunction of TSL_2 languages can include restrictions for all possible tiers
  - Avoids the problem of tier discovery
- For a finite segment inventory, there are finitely many possible such grammars
  - The class of languages defined by conjunctions of TSL_2 languages is Gold-learnable

Brute-force algorithm

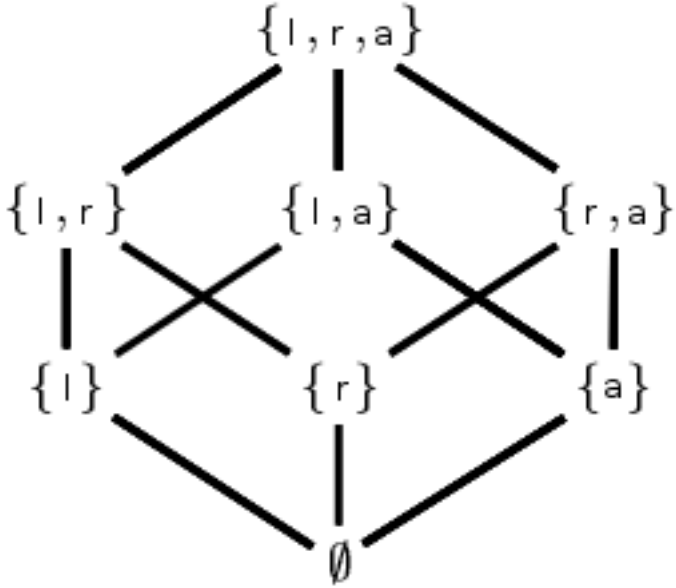
- Generate a list of all possible tiers
  - For a segment inventory Σ, there are $2^{|\Sigma|}$ possible tiers
  - If Σ={C,V}, there are four tiers: T_1={C,V} T_2={C} T_3={V} T_4={}
- For each tier:
  - Reduce the training data to only those segments
    - A word 'CVCCV' is reduced to: CVCCV, CCC, VV, and Ɛ respectively
  - Record all *k*-factors attested in the reduced data
- This set of tier-indexed 2-factors constitutes the final grammar
  - We report the complement of this set, only for tiers that do not allow all possible 2-factors
  - i.e. the grammar is the set of prohibited sequences on each tier

Disadvantages of brute-force algorithm

- The brute-force algorithm generates highly redundant grammars
  - For unbounded liquid harmony, the restriction *lr is included on:
    - *T* = {l,r}
    - *T* = {l,r,p}
    - *T* = {l,r,t}
    - *T* = {l,r,p,t}
    - …
  - *lr on tier *T* implies *lr on all supersets of *T*
- These redundant grammars are accurate, but:
  - They are poor reflections of the grammatical pressures shaping the lexicon
  - They are difficult for humans to (visually) parse

Alternative algorithm: lattice search

- Goal: learn a conjunction of TSL_2 grammars without redundancy
- Solution: for every possible bigram of Σ, search a lattice of the power set of Σ and return the minimal subset on which that bigram is prohibited



- This algorithm's generality lacks a formal proof, but it succeeds on all test cases we have provided, such as…

Example language 1

- Segment inventory: {b,m,n,k,i,a,u}
- Labial place dissimilation (*[lab]...[lab])
  - banaka, kamaka
  - *babaka, *mabana
- Front/back harmony among high vowels, transparent [a]
  - nikini, nikani, nakunu
  - *nikanu, *nukuni

- The lattice search algorithm learns:
  - On *T* = {i,u}: *ui, *iu
  - On *T* = {b,m}: *bb, *bm, *mb, *mm

Example language 2

- Segment inventory: {b,m,n,k,i,a,u}
- Labial place dissimilation blocked by [n]
- Front/back harmony among high vowels, transparent [a]
- CV syllable structure

- The lattice search algorithm learns:
  - On *T* = {b,m,n}: *bb, *bm, *mb, *mm
  - On *T* = {i,u}: *ui, *iu
  - *CC and *VV for all other CC and VV on appropriate tiers

Ongoing research and further questions

- Our model currently makes two assumptions that are unreasonable for natural language data:
  - phonotactic patterns lack gradience
  - complete language data available (unattested = ungrammatical)
- This algorithm is capable of learning any number of dependencies on multiple, overlapping tiers
  - For an inventory with 26 segments, there are >67 million possible tiers
- How can we reduce the learner's hypothesis space?
  - Restrict the set of tiers that the learner must consider by defining them with natural classes or phonetic distance
- Can we use multiple constraints defined by individual TSL_2 grammars to generate the same typological predictions?

Acknowledgements

- Gunnar Ólafur Hansson (University of British Columbia)
- Jeff Heinz and Adam Jardine (University of Delaware)
- Audiences at the Conference on Agreement by Correspondence (Berkeley, CA) and at the Annual Meeting of Phonology 2014 (MIT)