Capturing gradience in long-distance phonology using probabilistic tier-based strictly local grammars

Connor Mayer

UCLA

Society for Computation in Linguistics
February 17th, 2021

Overview

The class of **tier-based strictly local** (TSL) languages has proven useful for modeling long-distance phonotactic phenomena from the perspective of formal language theory.[1]

Long-distance phonology frequently exhibits **gradience** that TSL cannot capture.

This presentation will present **probabilistic tier-based strictly local** (pTSL) grammars, which naturally extend TSL grammars to allow gradience to be represented

[1] **Heinz.etal11**

Subregular phonology

Subregular phonology attempts to find proper subclasses of the regular languages and transductions that are sufficiently powerful to model natural language phenomena.[2]

Let's start by considering two commonly applied subregular classes

[2] **Heinz2016**

Strictly local languages

SL languages are generated by grammars that **prohibit certain substrings**.

- $f_k(s)$ is the set of all length-k substrings of $\rtimes^{k-1} s \ltimes^{k-1}$ for $s \in \Sigma^*$

A SL-k grammar G is a finite set of strings from $(\{\rtimes, \ltimes\} \cup \Sigma)^k$

- $s \in \Sigma^*$ is well-formed with respect to G iff $f_k(s) \cap G = \emptyset$

Simple tier projections

SL grammars have difficulty capturing long-distance restrictions.

Tier-based strictly local grammars provide a solution.

Preliminaries: for $T \subseteq \Sigma$, a *simple tier projection* π_T deletes symbols not in T.

- If $T = \{a, c\}$, $\pi_T(abbc) = ac$

Tier-based strictly local grammars

TSL languages are generated by grammars that **prohibit certain substrings on a tier projection.**

A TSL-k grammar is a tuple (T, G) where

- $T \subseteq \Sigma$
- G is a finite set of strings from $(\{\rtimes, \ltimes\} \cup T)^k$
- $s \in \Sigma^*$ is well formed with respect to a TSL-k grammar (T, G) iff $f_k(\pi_T(s)) \cap G = \emptyset$

Tier-based strictly local grammars

Consider a language where primary stress can occur anywhere, but words must contain exactly one syllable with primary stress. Let $\Sigma = \{\sigma, \acute{\sigma}\}$.

We can't generate this language with any SL-k grammar:

- Any string of the form $\acute{\sigma}\sigma^{k-1}\acute{\sigma}$ violates this stress pattern, but can't be excluded by a SL-k grammar.

Tier-based strictly local grammars

But we can do this with a TSL-2 grammar!

- Let $T := \{\acute{\sigma}\}$ and $G := \{\rtimes\ltimes, \acute{\sigma}\acute{\sigma}\}$.
- $\pi_T(\acute{\sigma}\sigma^{k-1}\acute{\sigma}) = \acute{\sigma}\acute{\sigma}$ for any value of k.
- This projection will be rejected because $f_2(\acute{\sigma}\acute{\sigma}) = \{\rtimes\acute{\sigma}, \underline{\boldsymbol{\acute{\sigma}\acute{\sigma}}}, \acute{\sigma}\ltimes\}$ ✗

Moving to non-categorical outputs

TSL grammars assign categorical membership to input strings.

- An input is either in the language or not.

Sometimes we want to model more gradient properties

- Acceptability ratings[3]
- Production frequencies[4]

[3] **AlbrightHayes2003**; **DalandEtAl2011**
[4] **HayesLonde2006**; **ZurawHayes2017**

Probabilistic tier projection functions

The simple tier projection function π_T can be generalized to a *probabilistic tier projection function* $\pi_P : \Sigma^* \rightarrow (\Sigma^* \rightarrow [0,1])$.

Returns a probability distribution over projections of an input string to a tier.

π_T can be thought of as a special case of π_P.

Calculating the distribution over projections

Each symbol has a probability of projecting $P_{proj} : \Sigma \rightarrow [0, 1]$

The probability of projecting a subsequence $y = (y_n)_{n \in J}$ from the input $x = (x_n)_{n \in I}$ is

$$\pi_P(x)(y) := \prod_{k \in J} P_{proj}(x_k) \cdot \prod_{k \in I \setminus J} [1 - P_{proj}(x_k)]$$

The probabilities of all possible projections for an input string x sum to one:

$$\sum_{y \in \Sigma^*} \pi_P(x)(y) = 1$$

Probabilistic tier-based strictly local grammars

A *probabilistic tier-based strictly k-local* (pTSL-k) grammar over Σ is a tuple (π_P, G):

- π_P is a probabilistic projection function
- $G \subseteq (\Sigma \cup \{\rtimes, \ltimes\})^k$ is a set of prohibited k-factors

Computing the probability of an input

$val_{(\pi_P, G)}$ computes the value that is assigned to a input string x by the corresponding pTSL-k grammar.

$$val_{(\pi_P, G)}(x) := \sum_{y : f_k(y) \cap G = \emptyset} \pi_P(x)(y)$$

This is the sum of the probabilities of all possible projections that don't contain a prohibited k-factor.

This is a *conditional probability given an input*. In general

$$\sum_{x \in \Sigma^*} val_{(\pi_P, G)}(x) \neq 1$$

An example pTSL grammar

Assume a pTSL-2 grammar defined over the alphabet $\Sigma := \{a, b, c\}$.

π_P is defined using the following projection probabilities:

$$P_{proj}(a) := 1.0$$
$$P_{proj}(b) := 0.5$$
$$P_{proj}(c) := 1.0$$

Let $G := \{ac\}$

An example pTSL grammar

The complete distribution over possible projections of *abbc* is:

$$\pi_P(abbc)(abbc) = 0.25$$
$$\pi_P(abbc)(abc) = 0.5$$
$$\pi_P(abbc)(ac) = 0.25$$

$val_{(\pi_P,G)}(abbc) = 0.75$, because the sum of the probabilities of all projections of *abbc* that do not contain the 2-factor *ac* is $0.25 + 0.5 = 0.75$.

Some properties of pTSL

A stringset $L \subseteq \Sigma^*$ is pTSL-k iff there is some pTSL-k grammar (π_P, G) such that $L = \{w \in \Sigma^* | val_{(\pi_P,G)}(w) > 0\}$.

TSL $\subsetneq$ pTSL

Relating pTSL probabilities to linguistic data

How do we relate the probabilities generated by pTSL to gradient linguistic data?

Word ratings should be *positively correlated* with pTSL probabilities.

For response frequencies between two possible forms y_1 and y_2:

$$\text{freq}(y_1) := \frac{val_{(\pi_P, G)}(y_1)}{val_{(\pi_P, G)}(y_1) + val_{(\pi_P, G)}(y_2)}$$

$$\text{freq}(y_2) := 1 - \text{freq}(y_1)$$

This works quite well in practice!

Hungarian vowel harmony

Suffixes must match the backness of the final front (F) or back (B) vowel in the root.[5]

/i iː eː ɛ/ are *harmonically neutral* (N). N roots generally take front suffixes.

BN^+ roots vary in whether they take front or back suffixes. This is sensitive to:

- **Count effects**: More N → more likely back trigger will be blocked
- **Height effects**: lower vowels more likely to block (/ɛ/ ≫ /eː/ ≫ /i iː/)

Hayes et al. (2009) wug tested 131 native speakers on this variation

- Participants presented with wug words matching several templates (BN, BNN, N)
- Asked to attach dative suffix: we're interested in whether they choose the front or back form (**response frequency**)

[5] **HayesLonde2006**; **HayesEtAl2009**

A pTSL grammar for Hungarian backness harmony

I defined a pTSL-2 grammar to capture wug test responses:

- $\Sigma := \{B, I, \text{e:}, \text{e}, S_f, S_b\}$
- $G := \{BS_f, IS_b, \text{e:}S_b, \text{e}S_b\}$
- P_{proj} fixed to 1 for $\{B, S_f, S_b\}$

The rest of the projection probabilities were fit to response frequencies

$$P_{proj}(\text{I}) = 0.39$$
$$P_{proj}(\text{e:}) = 0.66$$
$$P_{proj}(\text{e}) = 0.82$$

These values allow us to capture both count and height effects!

A sample calculation of $val_{(G,\pi_P)}$

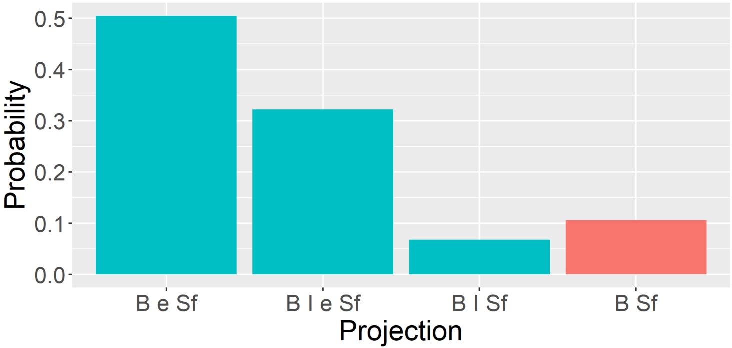

Figure: Probability distribution over projections of $BIeS_f$. $val_{(\pi_P,G)}(BIeS_f) = 0.895$, which is the sum of the probabilities of the grammatical projections.

Fit to Hungarian response frequencies

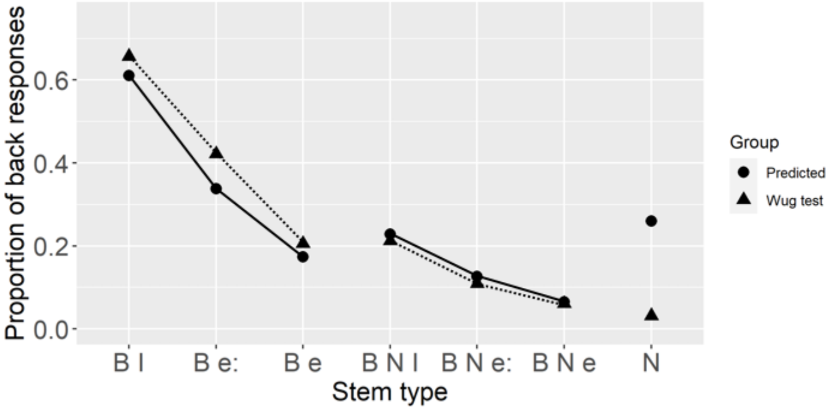

Figure: Observed against predicted proportion of back responses by stem template ($r = 0.83$).

Discussion

pTSL grammars assign (conditional) probabilities that capture gradience in long-distance phonological patterns

The parameters are simple and interpretable

Distance-based decay is captured with no explicit reference to distance

- Decay functions proposed in the literature[6] are equivalent to assigning certain projection probabilities to intervening material[7]

[6] **Kimper2011**; **Zymet2014**
[7] **McMullinND**

Extensions of pTSL

We can extend projection probabilities to be conditioned on context

- Input context (I-TSL/SS-TSL): $P_{proj}(x_i|x_{i-1})$[8]
- Output context (O-TSL): $P_{proj}(x_i|y_{j-1})$[9]
- Both (IO-TSL): $P_{proj}(x_i|x_{i-1}, y_{j-1})$[10]

Conditioning on preceding output can improve the fit to Hungarian N stems.

- More likely to project N when no preceding B

[8] **DeSantoGraf17MOL**
[9] **MayerMajor2018**
[10] **GrafMayer2018**

Acknowledgements

Thanks to all of you!

Thanks also to Bruce Hayes, Tim Hunter and Kevin McMullin for their helpful feedback, Kie Zuraw for helpful feedback and for providing the Hungarian data, Travis Major and Mahire Yakup for collecting the Uyghur data, and Dakotah Lambert and Jonathan Rawski for useful discussions about the relationship between TSL and pTSL. Thanks as well to two anonymous reviewers. This work was supported by the Social Sciences and Humanities Research Council of Canada.

References I