Using Locality to Learn Long-distance Phonological Processes

Jane Chandlee and Jeffrey Heinz

Nemours Biomedical Research  University of Delaware

LSA 2015 | Portland, OR
Main objectives

- Define *Output Strict Locality* (OSL), a restrictive and defining computational property of phonological UR $\rightarrow$ SR maps.
- Analyze various long-distance processes as the composition of OSL maps.
- Demonstrate how the OSL property can be used to learn phonological processes.
Processes as Maps

- Phonological processes are *maps* from an input (UR) to an output (SR) (cf. Tesar 2014).
- We want to identify the best (i.e., most restrictive) characterization of the class of maps that is needed for phonology.
  - Why? Identifying this class helps us understand 1) what is a ‘possible’ phonological process and 2) how processes are learned.
Learnability

For a given class of maps, we want to know

1. Is there an algorithm that we can prove will learn any of the maps in the class?
2. Can it do so *efficiently* (i.e., with a reasonable amount of time and resources)?
3. What is the nature of the data it needs to learn the maps?
Methodology

1. Identify the defining property of the class of maps.
2. Prove that an algorithm that uses this property as an inductive principle will learn *all* and *only* the maps in that class.
3. Implication: no other maps can be learned and will therefore not be attested (i.e., the learner *is* the hypothesis space!).
Phonological maps are Regular Relations

Logically Possible Maps

Regular Relations

A \to B / C \_\_ D

Johnson (1972); Kaplan and Kay (1994); Koskenniemi (1983)
Phonological maps are Regular Relations

- $A \rightarrow B / C \_ D$
- $*\text{CAD} \gg \text{FAITH}(A\rightarrow B)$

Baković (2013)
Phonological maps are Regular Relations

Riggle (2004); Gerdemann & Hulden (2012); Buccola 2013
Phonological maps are Regular Relations

- Not the best characterization.
  - Regular relations can also describe rules that are unattested and phonologically ‘odd’.
- Doesn’t help with learning.
  - The class of regular relations is not learnable from positive data.
Phonological maps are Strictly Local Functions

Chandlee (2014)
Strictly Local Functions

- Input Strictly Local Functions (ISL)
- Output Strictly Local Functions (OSL)
An ISL function is actually ISL-\(k\) for some integer \(k\).

The output for a given input can be determined by only paying attention to \(k\) contiguous segments at a time (i.e., bounded memory).

For a phonological map, \(k = \) the length of the target + the triggering context.
(1) Nasal place assimilation
   a. \( f(\text{iN}+\text{possible}) = \text{impossible} \)

\[ k = |Np| = 2 \]
Using Locality to Learn Long-distance Processes

Jane Chandlee and Jeffrey Heinz
Input Strictly Local Functions
Input Strictly Local Functions
Input Strictly Local Functions

# iN possible #

# imp
Input Strictly Local Functions

# impossible
# impossible
Input Strictly Local Functions
A significant range of phonological processes (local substitution, deletion, epenthesis, synchronic metathesis) can be modeled with ISL functions.

Approximately 95% of the processes in P-Base (v.1.95, Mielke (2008)) meet this criterion (i.e., are ISL).

Chandlee, Jardine, Heinz (2014) also show that the opaque maps discussed in Baković (2007) are ISL.
The class of ISL functions is efficiently identifiable in the limit from positive data (Chandlee, Eyraud, Heinz 2014) in the sense of de la Higuera (1997).
Because ISL functions only pay attention to the input, the trigger of the process must be present underlyingly.
Nasal spreading

(2) Johore Malay (Onn 1980)

a. \([+\text{nasal}][-\text{cons}] \leftrightarrow [+\text{nasal}][+\text{nasal}]\)

b. \(pəŋawasən \leftrightarrow pəŋãwãsən\), ‘supervision’
Output Strictly Local Functions
Output Strictly Local Functions

Jane Chandlee and Jeffrey Heinz
Using Locality to Learn Long-distance Processes
Output Strictly Local Functions

Jane Chandlee and Jeffrey Heinz

Using Locality to Learn Long-distance Processes
Output Strictly Local Functions

# p e n a w a s a n #
# p e n ã #
Output Strictly Local Functions

Jane Chandlee and Jeffrey Heinz

Using Locality to Learn Long-distance Processes
Output Strictly Local Functions

# p e n a w a s a n #
# p e n ã ã ã w ã
Though a target of the assimilation can be arbitrarily far from the trigger, the iterative nature of spreading makes it a local process.

Processes in which intervening ‘potential’ targets are skipped pose a challenge (Hansson 2001, Rose & Walker 2004).
Transparent Vowels

(3) Finnish front-back harmony (Goldsmith 1995, van der Hulst & van de Weijer 1995)

a. værttinæ ‘spinning wheel’
b. værttinæ-llæ-ni-hæn ‘with spinning wheel, as you know’
c. palttina ‘linen cloth’
d. palttina-lla-ni-han ‘with linen cloth, as you know’
Assuming harmony is local (Gafos 1996, Ní Chiosáin & Padgett 1997), then transparent vowels create opaque maps:

- Underapplication: high vowels don’t harmonize
- Overapplication: non-high vowels that follow non-harmonizing high vowels do harmonize

(4) palttina-là-ni-han
Transparent Vowels

\[
\begin{align*}
\text{palttina-lla-ni-hAn} & \\
\quad \Downarrow & \quad \checkmark \text{ OSL} \\
\text{palttuna-lla-nu-han} & \\
\quad \Downarrow & \quad \checkmark \text{ OSL} \\
\text{palttina-lla-ni-han}
\end{align*}
\]

(Clements 1977; Vago 1980, see also Baković & Wilson 2000)
Long-distance Assimilation

(5) Kikongo (Rose and Walker 2004)
   a. tu-kun-idi ↦ tu-kun-ini  ‘we planted’
   b. tu-nik-idi ↦ tu-nik-ini  ‘we ground’
Long-distance Assimilation

tu-nik-idi
  ↓
tu-نك-ین
  ↓
tu-nik-ini

✓ OSL

✓ OSL
Interim Summary

\[
\begin{align*}
\text{harmony} & \leftarrow \text{OSL} \circ \text{OSL} \\
\downarrow & \\
\text{spreading} & \leftarrow \text{OSL} \\
\downarrow & \\
\text{adjacency} & \leftarrow \text{ISL}
\end{align*}
\]
Learning OSL

- Modification to the ISLFL Algorithm already proven to learn the class of ISL functions (Chandlee, Eyraud, Heinz 2014).
Learning OSL

- Starts with a finite state representation of the observed data.
- \[\{(p\text{a}_n, p\text{a}_n), (p\text{a}_n, p\text{a}\text{a}_n), (p\text{a}_n p\text{a}_n, p\text{a}\text{a}_n), (p\text{a}_n, p\text{a}_n), (p\text{a}_n p\text{a}_n, p\text{a}_n)\ldots\}\]
Learning OSL
Learning OSL

- Generalize to unobserved inputs by merging states.
- State merging criterion reflects the defining property of the OSL class (i.e., only need to remember the ‘most recent’ output symbols).
Learning OSL

Jane Chandlee and Jeffrey Heinz

Using Locality to Learn Long-distance Processes
Learning OSL

Jane Chandlee and Jeffrey Heinz
Learning OSL

Jane Chandlee and Jeffrey Heinz

Using Locality to Learn Long-distance Processes
Using Locality to Learn Long-distance Processes
Future Work

- What do the intermediate forms look like?
  - How abstract can they be?
  - Are the segments that don’t surface still available within the ‘feature space’?
- How is the composition learned?
Future Work

harmony ← TSL? or SP?
↓
spreading ← OSL
↓
adjacency ← ISL
Proposal for representing and learning long-distance processes as well as an approach to the opacity problem.

Progress on all goals for a theory of the phonological grammar: sufficiently expressive, maximally restrictive, and learnable.


References

Schwa deletion

a. VCəCV ↔ VCCV

(7) tu devenais, ‘you became’
a. ty dəvəνε → ty dνəνε
b. ty dəvəνε → ty dəνενε
c. ty dəvəνε → *ty dνενε
Strict Locality

Post-nasal obstruent voicing: Valid surface strings are a SL-2 language.

TV\textsuperscript{N}DV in the language.
NVNTV in the language.

\textit{TVNTV necessarily} in the language.
Optimization

\textbf{IDENT, DEP} \gg \ast ab \gg \textbf{MAX}

Non-regular relation:
\[ a^n b^m \leftrightarrow a^n, \text{ if } m < n \]
\[ a^n b^m \leftrightarrow b^m, \text{ if } n < m \]

Gerdemann & Hulden (2012)
Transparent Vowels

Targeted Constraints analysis (Baković & Wilson 2000)

(8) Wolof ATR harmony (Archangeli & Pulleyblank 1994)
   a. reer-oon ‘was lost’
   b. ṭɛɛr-ʊʊn ‘had dinner’
   c. tɛɛr-ʊn-ʊʊn ‘welcomed’
Transparent Vowels

Targeted Constraints analysis (Baković & Wilson 2000)

\[ t\varepsilon r-Uw-OOn \rightarrow t\varepsilon r-uw-\text{con} \]

\text{\textsc{agree}}(ATR) prefers the full assimilation candidate \( t\varepsilon r-uw-\text{con} \), but the targeted constraint \( \rightarrow \text{No}(+\text{HI},-\text{ATR}) \) selects the candidate the differs minimally from full assimilation (i.e., the transparency candidate): \( t\varepsilon r-uw-\text{con} \).