Class 2 (Week 1 Thurs.): Deep into SPE, part II

To do for next time
- Prince & Smolensky study questions due Tuesday (get reading from CCLE; get study questions from course web page, to be found at my own web page)
- I’ll post a homework problem (probably Palauan) tonight. Due Fri., Jan 16.

Overview: The less-common expansion conventions from Chomsky & Halle (1968). Making theoretical choices. The dawn of constraints. (Sorry, still only one chunk of real data.)

1. Super- and subscripts
   - $X_n^m$ means from $n$ to $m$ Xs
     - $C_n$: “$n$ or more Cs” (most common is $C_0$)
     - $V^m$: “up to $m$ Vs”
     - $C_n^m$: “anywhere from $n$ to $m$ Cs”

   $$C \rightarrow \emptyset/\_C_0\# = \ldots$$
   $$C \rightarrow \emptyset/\_C_C_C\#$$
   $$C \rightarrow \emptyset/\_C_C\#$$
   $$C \rightarrow \emptyset/\_C\#$$
   $$C \rightarrow \emptyset/\_\#$$

   - The tricky thing is that we apply the longest rule whose structural description matches (disjunctive ordering again).  

   o How would the schema above apply to /tabskt/?

2. Parentheses with star (but see discussion in Week 3 Anderson reading)
   - (…)* means that the material in parentheses can occur zero or more times.

   $$V \rightarrow [+\text{stress}] / #C(VCVC)^* _\_ \_
 expandsto \_
 V \rightarrow [+\text{stress}] / #C_\_$$
   $$V \rightarrow [+\text{stress}] / #CVCVC_\_$$
   $$V \rightarrow [+\text{stress}] / #CVCVCVCVC_\_ \_ \_
 etc.$$  

   - With ( )* , there is yet a third type of ordering: simultaneous—identify targets of any sub-rule and change them simultaneously

   o How would the stress rule above apply to /badupidome/?

   o How would $C \rightarrow \emptyset/\_C^*\#$ apply to /tabskt/?
3. **Angled brackets—we’ll just skim this part, because they’re rarely used**
   - Like parentheses, but when the optional information is in more than one place.
   - A schema with angle brackets expands into two (disjunctively ordered) rules: the rule with the information in the angle brackets and the rule without that information.

   \[ C \to \emptyset / V\langle C\rangle_1 \__ \langle C\rangle_2 V \] (silly example) expands to

   \[
   \begin{align*}
   &C \to \emptyset / VC\_CV \quad \text{both} \\
   &C \to \emptyset / V\_V \quad \text{neither}
   \end{align*}
   \]

   - An exercise for later if you like: expand the following schema and apply it to *putod*, *luged*, and *fesil*.

     \[
     \begin{bmatrix}
     +\text{syl} \\
     [+\text{back}] \\
     \end{bmatrix}
     \to
     \begin{bmatrix}
     -\text{hi} \\
     +\text{back} \\
     -\text{hi}
     \end{bmatrix}
     C >#
     \]

   - You can also subscript angle brackets to show which ones go together:

     \[ C \to \emptyset / V <_1 C>_1 \__ <_2 S>_2 <_1 C>_1 V <_2 h>_2 # \] (even sillier rule) expands to

     \[
     \begin{align*}
     &C \to \emptyset / VC\_sCVh# \quad \text{both the } 1\text{s and the } 2\text{s} \\
     &C \to \emptyset / V\_sVh# \quad \text{just the } 2\text{s} \\
     &C \to \emptyset / VC\_CV# \quad \text{just the } 1\text{s} \\
     &C \to \emptyset / V\_V# \quad \text{neither}
     \end{align*}
     \]

4. **Not an expansion convention, an extension of the theory’s power: Transformational rules**
   - Useful for metathesis, coalescence…anything where more than one segment is affected at once.
   - In SPE, these were given in two parts:

     Structural description:

     \[
     \begin{bmatrix}
     +\text{syl} \\
     +\text{low}
     \end{bmatrix} \\
     \begin{bmatrix}
     +\text{syl} \\
     +\text{hi}
     \end{bmatrix}
     \]

     Structural change: \(1 \ 2 \rightarrow \\
     \begin{bmatrix}
     1 \\
     -\text{lo}
     \end{bmatrix}, \\
     \begin{bmatrix}
     2 \\
     +\text{long}
     \end{bmatrix}
     \]

     - What does this rule do?

     - It may seem arbitrary to say that 1 changes and 2 deletes rather than the reverse. Try writing the rule the other way too.
• You’ll often see a simplified notation instead that collapses the structural description and structural change:

\[
\begin{array}{c}
+\text{syl} \quad +\text{syl} \\
+\text{low} \quad +\text{hi} \quad \alpha \text{round} \\
\end{array}
\rightarrow
\begin{array}{c}
1 \\
-\text{lo} \\
+\text{long} \\
\alpha \text{round} \\
\alpha \text{back} \\
\end{array}
\]

1 2

○ What’s wrong with just saying this (same rule without the numerical indices):

\[
\begin{array}{c}
+\text{syl} \quad +\text{syl} \\
+\text{low} \quad +\text{hi} \quad \alpha \text{round} \\
\end{array}
\rightarrow
\begin{array}{c}
-\text{lo} \\
+\text{long} \\
\alpha \text{round} \\
\alpha \text{back} \\
\end{array}
\]

○ Say you want to write a metathesis rule that changes s-stop into stop-s. What’s wrong with writing \( s[\text{–sonorant} \quad \text{–continuant}] \rightarrow [\text{–sonorant} \quad \text{–continuant}] s \)?

○ Re-write the defective rule with transformational notation.

5. **How does the learner choose a grammar?**

• SPE proposed that if more than one grammar can generate the observed linguistic data, the learner must have some **evaluation metric** for choosing one.

• The evaluation metric tentatively proposed in SPE is brevity: learner chooses the grammar with the fewest symbols. (What about ties?)

• If that’s right, and if we’ve got the notation right too, then you can tell which grammar, out of some set of candidate grammars, the learner would choose.

• More plausibly, we want to find independent evidence as to which grammar is right, and then make sure our theory explains how/why the learner chose that one—this is a lot harder!
6. **Example: French elision/liaison (SPE p. 353 ff.)**

- By the logic above, a theoretical innovation is held, in SPE, to be a good one if it allows more-concise descriptions of attested/common phenomena than of unattested/uncommon phenomena.

<table>
<thead>
<tr>
<th>obstruent-nasal-initial</th>
<th>liquid-initial</th>
<th>vowel-initial</th>
<th>glide-initial</th>
</tr>
</thead>
<tbody>
<tr>
<td>/potit/ ‘small’</td>
<td>poti garsõ</td>
<td>poti livr</td>
<td>potit ăfă</td>
</tr>
<tr>
<td>/ʃɛr/ ‘dear’</td>
<td>ʃɛr garsõ</td>
<td>ʃɛr livr</td>
<td>ʃɛr ăfă</td>
</tr>
<tr>
<td>/la/ ‘the’</td>
<td>la garsõ</td>
<td>la livr</td>
<td>l_ ăfă</td>
</tr>
<tr>
<td>/parej/ ‘similar’</td>
<td>parej garsõ</td>
<td>parej livr</td>
<td>parej ăfă</td>
</tr>
</tbody>
</table>

For the sake of reconstructing the argument, use the archaic feature \[vocalic\] and the still-current feature \[consonantal\]:

<table>
<thead>
<tr>
<th></th>
<th>vocalic</th>
<th>consonantal</th>
</tr>
</thead>
<tbody>
<tr>
<td>obstruents and nasals</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td>liquids</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>glides</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>vowels</td>
<td>+</td>
<td>–</td>
</tr>
</tbody>
</table>

- Propose rules to account for the C- and V- deletions, without using Greek-letter variables.

- Combine the rules into a schema, using Greek-letter variables

- Greek-letter variables don’t allow us to compress these two rules:

  \[ [+voc] \rightarrow \emptyset / _ # [+voc] \] “low vowels delete before a vowel or glide”

  \[ [+voc] [+cons] \rightarrow \emptyset / _ # [+cons] \] “liquids delete before a non-glide consonant”

  With that in mind, how should the typology guide us in deciding whether to allow the same Greek-letter variable to apply to different features within a rule?
7. **(skip if no time) Reasoning above relies on assumptions about linguistic typology:**

- Assume a rule is cross-linguistically common only if it’s favored by learners—i.e., learners tend to mislearn, in the direction of a more-favored grammar.
- Assume that learners favor short/simple/whatever rules.
- Therefore, rules that are cross-linguistically common should tend to be short.
- Therefore, our theory of rules, which determines what type of notation length is calculated on, should make common rules shorter than uncommon ones.
- Therefore, a theoretical innovation is good if it makes common rules shorter than uncommon ones.

=> We’re not really using “short” (or “simple”) in any fixed sense. Rather, we’re tailoring the notation to make the rules that we think learners favor appear short. [And of course, that first assumption is questionable...]

This leads us into slippery territory in deciding whether shortness is the right criterion:

- Are learners innately endowed with a certain notation, which they use to calculate grammar length? (i.e., shortness really is the evaluation criterion)
- Or is it the case that learners employ some other evaluation metric entirely, but we’ve created a system of notation that makes goodness according to the real evaluation metric translate into shortness in our notation?

Something for you to think about, though no answers will be forthcoming: We’ve seen how to evaluate a particular description or even a theoretical innovation, given a framework like SPE.

- But how do you evaluate the framework itself—in particular, how can we evaluate a principle such as “if more than one grammar can generate the observed linguistic data, the learner chooses the grammar with the fewest symbols”?

8. **Shortening the (previously) unshortenable: constraints**

- Kisseberth (1970) introduced the following problem, using Yawelmani Yokuts as a case study.
- These rules can’t be collapsed into a schema:

  \[
  \emptyset \rightarrow V / C \_ CC \\
  C \rightarrow \emptyset / CC + \_ 
  \]

- And yet, they seem to have something in common—can you guess what? It will help to invent a form that each rule can apply to and see what that rule does.

- Cases like this became known as *conspiracies*, and their widespread existence as the *conspiracy problem*. 
9. Constraints as rule blockers

- Kisseberth proposes using a constraint to make the rules of Yawelmani Yokuts simpler:

Instead of \( V \rightarrow \emptyset / V C \quad \text{[\text{\text{\text{–\text{long}}}]}]} \)

use \( V \rightarrow \emptyset / C \quad \text{subject to the constraint *CCC (or *\{C,#\}C\{C,#\}) [\text{\text{\text{–\text{long}}}]}]} \)

- The constraint can *block* the rule: the rule applies *unless* the result violates the constraint.

- Let’s try to lay out, step by step, what an algorithm would have to do to implement the rule and its blocking constraint.

10. Constraints as rule triggers

- Kisseberth also proposes that constraints can *trigger* rules: a rule applies *only if* it gets rid of a constraint violation.

- What happens if the rule \( \emptyset \rightarrow i \) (context-free) applies only when triggered by the constraint *CC*? Let’s try to break this down into simple steps too.
11. Why is this good?
- In a system without constraints, these two grammars have equal length and should be equally plausible:

   **Yokuts**
   - \( C \rightarrow \emptyset / CC + \_ \)
   - \( \emptyset \rightarrow i / C \_ CC \)
   - \( V \rightarrow \emptyset / V C \_\_\_ C \_ V \)
   - \( V \rightarrow \emptyset / V C \_\_\_ C C \)  
   
   **imaginary and implausible**
   - \( C \rightarrow \emptyset / CV + \_ \)
   - \( \emptyset \rightarrow i / V \_ CC \)
   - \( V \rightarrow \emptyset / V C \_\_\_ C C \)  

- But in Kisseberth’s system the Yokuts grammar is shorter than the “implausible” grammar

   **Yokuts**
   - \( C \rightarrow \emptyset / + \_ \)
   - \( \emptyset \rightarrow i \)
   - \( V \rightarrow \emptyset / C \_\_\_ C \)
   - \( *\{C,#\}C\{C,#\} \)  

   **imaginary and implausible**
   - \( C \rightarrow \emptyset / CV + \_ \)
   - \( \emptyset \rightarrow i / V \_ CC \)
   - \( V \rightarrow \emptyset / V C \_\_\_ C C \)  

- If we’re right that the language on the right is less plausible than Yokuts, Kisseberth’s theory is better because it captures that difference.

12. Rule+constraint theories
- Many more conspiracies were identified, giving rise to more constraints.
- People liked constraints, because they solved the conspiracy problem and also gave theoretical status to the idea of “markedness”, which had been floating around.
  - Everyone knew languages don’t “like” CCC sequences (they are “marked”), but this was not directly encoded in grammars until constraints like *CCC (or a syllable-based equivalent) came along.
- On the other hand, using constraints introduces some problems into the theory—I invite you to invent cases!
  - What if there’s more than one rule (or more than one way of applying a single rule) that could fix a constraint violation?
  - What if one rules makes a constraint violation worse, but feeds another rule that makes it better?
  - What if one constraint wants to trigger a rule, but a different constraint wants to block it?

**Next time:** Deep into OT and how it deals with the above problems.

References