Learning OT grammars (introduction)

24.964—Fall 2004
Modeling phonological learning

Class 6 (14 Oct, 2004)
Agenda for today

• Discussion of Bailey & Hahn (from last time)

• Wrap-up of statistical approaches

• Intro to learning phonotactics with OT
(see Week 5 Overheads)
Statistical approaches to phonotactics

How would we characterize statistical approaches?

- Open/closed domain?
- Clean/noisy data?
- Hypothesis space?
- Batch/incremental learning?
- Supervised/unsupervised?
- Makes use of negative evidence?
Statistical approaches to phonotactics

Thinking back to the AI model:

(Where does most of the action lie in a statistical approach?)
A very different approach: OT

Tesar & Smolensky 1996 [2000]: Recursive Constraint Demotion

Goal: outline a learning algorithm which ranks constraints in such a way that they correctly derive the input data

- Target grammar should be able to derive observed surface forms from their corresponding UR's

- Grammar should also exclude forms that are not permissible in the target language

- (Not part of the goal: differentiate possible but unlikely forms from possible and likely forms)
Tesar & Smolensky (1996)

The OT architecture: (familiar parts)

- Lexicon
- GEN
- CON
- EVAL

(What is the learner’s task?)
The OT architecture: (familiar parts)

- **Lexicon** *(language particular, must be learned)*
- **GEN** *(universal, and very generic)*
- **CON** *(universal, but must have particular form)*
- **EVAL** *(language particular ranking, procedure is universal)*
Some assumptions:

- The constraint set is fixed by UG (p. 4)
- Constraints are *total functions* from candidates
- Set of constraint rankings (dominance hierarchy) also total
- Competition between candidates consists of determining their harmonic ordering (winner is most harmonic)
- Learning = finding a ranking under which all desired winners are more harmonic than their respective losers
  - The relative order of losers does not matter
Another important distinction:

- **Full structural descriptions**: outputs of GEN, "including overt structure and input"

- **Overt structure**: the part of a description directly accessible to the learner

Example: /VCVC/ $\rightarrow \langle V \rangle .CV.\langle C \rangle \rightarrow [CV]$
The broader picture:

- **Grammar**
  - well-formedness conditions on structural descriptions

- **Full structural descriptions**
  - Overt structure
  - ‘Hidden’ structure

- **Robust Interpretive Parsing**

- **Overt Structure**

---

Tesar & Smolensky (1996)
Tesar & Smolensky (1996)

The learning scenario:

- Learner hears overt structure: [CVC]

- Infer full structural description most likely to be associated with it, under the current grammar; e.g. .CVC.
  - Robust interpretive parsing

- Then flip around the problem: assume underlying form like overt form

- Attempt to learn grammar that derives correct structural description from assumed UR
Tesar & Smolensky (1996)

One other issue that arises:

- OT assumes that constraint rankings are *total*

- Yet there is often no evidence for ranking between particular pairs, because they do not conflict for the data at hand

- To avoid making unmotivated (and possibly wrong) commitments, the ranking algorithm produces *partial orderings* (*strata* of constraints), consistent with numerous total rankings
Tesar & Smolensky (1996)

The ranking strategy:

1. Construct *mark-data pairs*
   - For each loser/winner pair, collect all violations
   - If both violate same constraint $C$ an equal number of times, these marks cancel each other out
   - Identify $C$ that assess uncancelled marks

2. Start with all $C$ in a single stratum

3. Look for $C$ that assign uncancelled marks to winners (that is, all constraints with L). Demote any such $C$, unless it is already dominated by another constraint $C'$ that has uncancelled *loser* marks (that is, a higher $W$)
4. Continue, creating subsequent strata, unless there are no uncancelled winner marks without higher-ranked uncancelled loser marks
**Tesar & Smolensky (1996)**

Example (6), p. 5

(6) Constraint Tableau for $L_1$

<table>
<thead>
<tr>
<th>Candidates</th>
<th>ONSET</th>
<th>NOCODA</th>
<th>$\text{FILL}^{\text{Nuc}}$</th>
<th>PARSE</th>
<th>$\text{FILL}^{\text{Ons}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>/VCVC/ →</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d. □V.CV.⟨C⟩</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>b. ⟨V⟩.CV.⟨C⟩</td>
<td></td>
<td></td>
<td>*</td>
<td>**</td>
<td></td>
</tr>
<tr>
<td>c. ⟨V⟩.CV.C□.</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>a. .V.CVC.</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Tesar & Smolensky (1996)**

In *comparative tableau* form (Prince 2000, 2002)

<table>
<thead>
<tr>
<th>/VCVC/ → ¥V.CV.⟨C⟩</th>
<th>ONS</th>
<th>*CODA</th>
<th>DEP(V)</th>
<th>MAX</th>
<th>DEP(C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>d. ~ a. ⟨V⟩.CV.⟨C⟩</td>
<td></td>
<td></td>
<td></td>
<td>W</td>
<td>L</td>
</tr>
<tr>
<td>d. ~ b. ⟨V⟩.CV.C□</td>
<td></td>
<td></td>
<td>W</td>
<td></td>
<td>L</td>
</tr>
<tr>
<td>d. ~ c. .V.CVC.</td>
<td>W</td>
<td>W</td>
<td></td>
<td>L</td>
<td>L</td>
</tr>
</tbody>
</table>
Tesar & Smolensky (1996)

General principles:

- Constraints are ranked in as high a stratum as possible
- Constraints with L’s can’t be in the top stratum; they are placed immediately below the top stratum with a corresponding W
- Constraints are always demoted, never promoted
Tesar & Smolensky (1996)

Characterizing the RCD approach:

- Open/closed domain?
- Clean/noisy data?
- Hypothesis space?
- Batch/incremental learning?
- Supervised/unsupervised?
- Makes use of negative evidence?
Prince & Tesar (1999)

(Student-led discussion)
For next week

• Download this week’s perlscripts file from the website, and “read” RCD.pl to understand how it implements Tesar & Smolensky 1996

• Try running it on the accompanying text files, to make sure it yields the “right” results for each (that is, understand why it yields what it yields)

• Modify RCD.pl to do ONE of the following:
  1. Incorporate the non-persistent “initial state” approach described by T&S, §4.4 ($\mathcal{M} \gg \mathcal{F}$)
  2. Calculate the r-measure of the final grammar

    ○ HINT: you will need to modify the format of the input file to tell the learner which constraints are $\mathcal{M}$ vs. $\mathcal{F}$; there is no way for it to infer this

• Reading: Hayes (1999) Phonological Acquisition in OT