One-level phonology
Linguistics 696b (Spring 2005)

A. Overview

(1) The basic idea:
Can phonology be handled without an “input” representation? Can such a theory be “computationally parsimonious”?

(2) Organization:
a. General ideas;
b. Finite state machinery;
c. English examples;
d. One-level phonology;
e. Working it out;
f. Previous proposals.

B. General ideas

(3) How do we do without an “input” (Hammond 2000; Russell 1995)?

(4) There should be only a single level of grammatical description.

(5) That level of description can be compiled from separate generalizations.

(6) How do we make a “computationally parsimonious” theory? We use finite-state machinery to implement the model.

C. Finite state machinery

(7) What is a finite state device (Hopcroft & Ullman 1979)?

(8) A quintuple: \((S, \Sigma, s_0, F, \delta)\):
\(S\): a set of states
\(\Sigma\): a finite alphabet
\(s_0\): a start state
\(F\): a set of final states
\(\delta\): a mapping from \(S \times \Sigma\) to \(S\)

(9)

(10) Finite state automata (FSAs) can only accept “regular languages”.

(11) What is a regular expression or language?
Union (“or”), concatenation, Kleene star (“repeat”)

(12) The FSA above generates: \(b^*a(a|b)^*\).
Regular languages are “closed” under:

- Union;
- Concatenation;
- Kleene star;
- Complementation;
- Intersection.

Why are FSAs a good thing?

- mathematically/computationally restrictive
- easy to use
- efficient to process

D. English examples

Inventory: English speakers know [kʰ] is a sound of English, but [ü] is not.

Distribution: [ð] can occur word-initially in English, but [n] cannot.

Alternations: flapping occurs between vowels when the second vowel is stressless, e.g. spot vs. spotter [spətər], feed vs. feeder [fɛdər], pet vs. petting [pɛtɪŋ], ride vs. riding [raɪdɪŋ], hat vs. hatted [hætɪd], wed vs. wedded [wɛdɪd], etc.

Interactions: vowel nasalization, nasal consonant deletion

Vowel nasalization

Nasal vowels only occur before nasal consonants: before [m] as in tam [tʰæm], [n] as in tan [tʰæn], and [ŋ] as in tang [tʰæŋ].

Nasal consonant assimilation

Nasal vowels can optionally also occur before voiceless consonants, but this only occurs if it is also possible to pronounce a nasal consonant after the nasal vowel, e.g.: lamp [læmp] ∼ [læp], cant [kʰænt] ∼ [kʰæt], and tank [tʰæŋk] ∼ [tʰæk].

<table>
<thead>
<tr>
<th>V-nas</th>
<th>input</th>
<th>can</th>
<th>canned</th>
<th>cant</th>
<th>cant</th>
<th>cat</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>/kæn/</td>
<td>/kænd/</td>
<td>/kænt/</td>
<td>/kænt/</td>
<td>/kæt/</td>
</tr>
<tr>
<td>N-del</td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>output</th>
<th>/kæn/</th>
<th>/kænd/</th>
<th>/kænt/</th>
<th>/kænt/</th>
<th>/kæt/</th>
</tr>
</thead>
</table>

E. One-level phonology

Inventory: define Σ appropriately.

In English, Σ does not include [ü].

Distribution: define regular expressions for different generalizations and intersect them.

Excluding word-initial [ŋ]: Σ *

Alternations: define regular expressions for different alternations and intersect them with lexical/syntactic strings.
\( /t/ = (Vr\overline{V}|t) \)  
\( /rayt/ = ray(r\overline{V}|t\overline{V}) \)

F. Working it out

(28) What isn’t done yet:  
   a. Rule interactions;  
   b. Probabilistic generalizations (intersecting HMMs; Pereira & Riley 1997);  
   c. Learning generalizations, or ‘deintersection’.

G. Previous proposals

(29) There are a number of previous proposals involving implementing phonology in finite-state terms, e.g. Bird & Ellison 1994, Bird 1995, Kaplan & Kay 1994, Karttunen 1983, Karttunen 1998, etc.

(30) The most similar to the current proposal is Bird & Ellison 1994.

References


