Composition

L545

Dept. of Linguistics, Indiana University
Spring 2013
Finite-state morphology

We have seen how to handle morphology with FSTs. Now, we want to step back & formally characterize morphological operations, focusing on composition:

- Composition handles concatenative morphology cleanly.
- Composition handles:
  - restrictions on the kinds of bases affixes can attach to
  - modifications on the bases affixes attach to

Material is adapted from Roark & Sproat (2007), Computational Approaches to Morphology and Syntax, esp. ch. 2
Example of Latin

Latin *scripserunt* is a combination of:

- stem *scrib-* (‘write’), which becomes *scrip-* before /s/
- perfect stem-forming -s- (for third conjugation verbs)
- (perfect) third person plural suffix *erunt*

Morphological analysis: relate word forms and detect structure of word forms

- structure: *scrib*+s_{\text{perfect}}+erunt_{\text{third,plural,active,indicative}}
  - We will use the function $\mathcal{D}$ to represent this step
- relate to canonical form (lemmatization):
  - *scribo*_{\text{perfect,third,plural,active,indicative}}
    - We can use a function $\mathcal{L}$ to obtain lemma from decomposed form (structure)
    - i.e., $\mathcal{D} \circ \mathcal{L}$
Syntagmatic variation

Simple concatenation

Given a stem $A$ and a suffix $\beta$, we can create a form $\Gamma$ with regular concatenation:

(1) $\Gamma = A \cdot \beta$

But what if instead we have a function $\beta'$ which takes a string as input & outputs a string concatenated with $\beta$

(2) $\beta' = \Sigma^* [\epsilon : \beta]$

- $\Sigma = $ alphabet of symbols
- $\Sigma^*$ is used here to specify a regular relation which maps strings into themselves

Now, we have:

(3) $\Gamma = A \circ \beta'$
Syntagmatic variation
Simple concatenation (2)

What are the advantages of treating concatenation as composition?

▶ especially since composition takes linear time, while concatenation is constant

Affixes often trigger some (phonological, spelling, or morphological) change affecting stem and/or affix

▶ Composition is needed for these cases

▶ Consider English plurals ($\Psi$), with phonological rule (/s/, /z/, /iz/) implemented by transducer $T$

\[(4) \quad \Psi = [S \cdot \sigma] \circ T\]

(5) Re-factor: $\Psi = S \circ [\Sigma^* [\epsilon : \sigma]] \circ T$

(6) Define: $\sigma' = [\Sigma^* [\epsilon : \sigma]] \circ T$

(7) New affix $\sigma'$: $\Psi = S \circ \sigma'$
Syntagmatic variation
Prosodically Governed Concatenation

Some affixes have prosodic conditions, e.g., comparative 
-er and superlative -est in English
  ▶ Generally speaking: only attach to monosyllabic or disyllabic stems
  ▶ The base/stem can be characterized as:

\[
B = C^* V C^* (V C^*)?
\]

▶ and the affix as:

\[
\kappa = B[\epsilon : er[+COMP]]
\]

▶ resulting in:

\[
\Gamma = A \circ \kappa
\]

▶ The only non-null \( \Gamma \) cases will be the ones where the base \( A \) matches \( B \)
Syntagmatic variation
Prosodically Governed Concatenation (2)

This will also capture more complicated templatic morphology, as in Yowlumne

- affix -inay requires the stem to reconfigure to CVC(C)
  
  \[
  (11) \ T_{cvc(c)} = CV[V : \epsilon]^* C[V : \epsilon]^* C?
  \]
  
  \[
  (12) \ \text{caw} \circ T_{cvc(c)} = \text{caw}
  \]
  
  \[
  (13) \ \text{diyi} \circ T_{cvc(c)} = \text{diyi}
  \]
  
  \[
  (14) \ \text{hiiyi} \circ T_{cvc(c)} = \text{hiyi}
  \]

- affix -?aa requires the template CVCVV(C)
  
  - more complicated, as it involves vowel copying

So, the morpheme -inay is represented as:

\[
(15) \ \kappa = T_{cvc(c)}[\epsilon : \text{inay} [+GER]]
\]
Syntagmatic variation

Subsegmental morphology

Subsegmental morphology: morphological alternants can be indicated by a change of a single phonological feature

- e.g., in Irish, genitive forms of nouns palatalize the final consonant
  - \( \text{bád} /d/ \text{ (NOM)} \mapsto \text{báid} /d^y/ \text{ (GEN)} \)
  - This is easily captured by defining a function \( \gamma \) which is a palatalization operation.

Genitive (\( \Gamma \)) is defined as a composition operation of \( \gamma \) applied to the nominative form (\( N \)):

\[
\Gamma = N \circ \gamma
\]
Syntagmatic variation
Extrametrical infixation

Consider infixes like \textit{-um-} in Philipino languages, e.g., Bontoc

- Ignores the onset sound of the word and prefixes to the remainder of the word
  - \textit{antj’\textbackslash ōak} ‘tall’, \textit{umantj’\textbackslash ōak} ‘I am getting taller’
  - \textit{k’\textbackslash āwisat} ‘good’, \textit{kum’\textbackslash āwisat} ‘I am getting better’

- Multiple infixes attach in this same spot, so it makes sense to break this down into 2 parts:
  1. Insert a marker (> for where the infix goes
  2. Convert the marker to the affix (e.g., \textit{-um-})
Syntagmatic variation

Extrametrical infixation (2)

1. Marker transducer $M$: insert $>$ at appropriate spot

\[(17) \quad M = C?[^\varepsilon :>]V\Sigma^*\]

2. Infixation transducer $\iota$: map $>$ to $-um$-

So, now we precompose these 2 steps:

\[(18) \quad \mu = M \circ \iota\]

Meaning that a final word form is:

\[(19) \quad \Gamma = A \circ \mu\]
Syntagmatic variation
Root-and-pattern morphology

Arabic verbs (derivational morphology):

- consonantal roots
- prosodic shape given by a prosodic template
- particular vowels chosen by intended aspect (perfect/imperfect)

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Template</th>
<th>Verb stem</th>
<th>Gloss</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>C₁aC₂aC₃</td>
<td>katab</td>
<td>‘wrote’</td>
</tr>
<tr>
<td>II</td>
<td>C₁aC₂C₂aC₃</td>
<td>kattab</td>
<td>‘caused to write’</td>
</tr>
<tr>
<td>III</td>
<td>C₁aaC₂aC₃</td>
<td>kaatab</td>
<td>‘corresponded’</td>
</tr>
<tr>
<td>VII</td>
<td>nC₁aC₂aC₃</td>
<td>nkatab</td>
<td>‘subscribed’</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Syntagmatic variation
Root-and-pattern morphology (2)

Templates:

(20) $\tau_1 = C_aC_aC$
(21) $\tau_{II} = C_aC_CaC$
(22) $\tau_{III} = C_aaCaC$
(23) $\tau_{VIII} = [\varepsilon: n]CaCaC$

\ldots

To obtain a transducer for all these templates:

(24) $\tau = \bigcup_{p \in \text{patterns}} \tau_p$
Syntagmatic variation
Root-and-pattern morphology (3)

Need a transducer to link the root to the templates:

- Must allow for optional vowels between consonants:
  \[ \lambda_1 = C[\epsilon : V]^* C[\epsilon : V]^* C \]

- Must allow for doubling of center consonant (pattern II) ... need general rewrite rules:
  \[ \lambda_2 : C_i \rightarrow C_i C_i \]
  \[ \lambda = \lambda_1 \circ \lambda_2 \]

We can then derive forms:

\[ \Gamma = P \circ \lambda \circ \tau \]

We can also compile \( \lambda \circ \tau \) into its own “pattern” machine
Paradigmatic variation

A *paradigm* is an array which each cell corresponds to a bundle of features

- characterizes how morphologically complex forms relate to one another
- e.g., Latin nouns, declension 1 (F)

<table>
<thead>
<tr>
<th>Case</th>
<th>Singular</th>
<th>Plural</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominative</td>
<td>femina</td>
<td>feminae</td>
</tr>
<tr>
<td>Genitive</td>
<td>feminae</td>
<td>feminarum</td>
</tr>
<tr>
<td>Dative</td>
<td>feminae</td>
<td>feminis</td>
</tr>
<tr>
<td>Accusative</td>
<td>feminam</td>
<td>feminas</td>
</tr>
<tr>
<td>Ablative</td>
<td>femina</td>
<td>feminis</td>
</tr>
</tbody>
</table>

There are regularities which seem to argue for a first-class status of paradigms

- e.g., ablative & dative plurals
Paradigmatic variation
A Computational Characterization

1. Relate morphosyntactic features to abstract morphomic features (transducer $\alpha$)
   - $\text{NEUT NOM } \cup \text{ ACC SG } \rightarrow \text{NEUTNASG}$
   - $\text{NEUT NOM } \cup \text{ ACC PL } \rightarrow \text{NEUTNAPL}$
   - $\text{NOM SG } \rightarrow \text{NOMSG}$
   - $\text{GENDER DAT PL } \rightarrow \text{DATABLPL}$
   - $\text{GENDER ABL PL } \rightarrow \text{DATABLPL}$

2. Relate morphomic forms to particular surface forms (for a particular word class) (transducer $\sigma$)
   - $\Sigma^*$ [I-II DATABLPL : is]
   - $\Sigma^*$ [NEUTNAPL : a]
   - $\Sigma^*$ [I-II NEUTNASG : um]
   - $\Sigma^*$ [III DATABLPL : ibus]
Paradigmatic variation
A Computational Characterization (2)

Given a set of bases annotated with morphosyntactic features, inflected forms:

\[(29) \quad \Gamma = B \circ \alpha \circ \sigma\]

We could also precompile \(\sigma' = \alpha \circ \sigma\), thereby hiding the abstraction
Reduplication
(if we have time ...)

Ruduplication involves potentially unbounded copying
  ▶ Copying not allowed by strict FSTs
  ▶ Bounded copying—however inelegantly—can be handled by FSTs

Gothic past tense of Class VII verbs

<table>
<thead>
<tr>
<th>Infinitive</th>
<th>Gloss</th>
<th>Preterite</th>
</tr>
</thead>
<tbody>
<tr>
<td>haldan</td>
<td>‘hold’</td>
<td>haíhald</td>
</tr>
<tr>
<td>ga-staldan</td>
<td>‘possess’</td>
<td>ga-staístald</td>
</tr>
<tr>
<td>af-áikan</td>
<td>‘deny’</td>
<td>af-aiáik</td>
</tr>
<tr>
<td>slepan</td>
<td>‘sleep’</td>
<td>saíslep</td>
</tr>
</tbody>
</table>
Reduplication (2)

Rule:

- Prefix syllable (A)Caí to the stem
  - C is a consonant position
  - (A) is an optional appendix position
- Copy the onset of the stem to the C position
  - If there is a pre-onset appendix /s/ (i.e., /s/ before /p,t,k/), copy to the (A) position

The transducer for this simply hard-encodes the proper sequences to obtain copying

- e.g., 1) ε:h arc, 2) ε:ái arc, 3) h:h arc
Unbounded Reduplication

Consider Bambara noun reduplication:

(30) \( wulu \ o \ wulu \)
    dog MARKER dog
    ‘whichever dog’

(31) \( wulu-nyinina \ o \ wulu-nyinina \)
    dog searcher MARKER dog searcher
    ‘whichever dog searcher’

(32) \( malo-nyinina-filéla \ o \ malo-nyinina-filéla \)
    rice searcher watcher MARKER
    rice searcher watcher
    ‘whichever rich searcher watcher’

The morpheme \( o \) in principle is unbounded
   * Cannot simply hard-code material before/after \( o \)
Unbounded Reduplication (2)

Think of reduplication as 2 components:

1. Prosodic constraints: e.g., make sure reduplicated material is of form (A)Caí
   ▶ This can be handled with regular finite-state operations

2. Copying component: verify that prefix matches the base
For Gothic, assume transducer $R$, which composes with a base $\beta$ and adds indices to elements in prefix and base

\[(33) \quad \alpha = \beta \circ R = (A_1)C_2 ai\beta'\]

So, the input stem $skáip$ will result in the output $X_1 X_2 aís_1 k_2 áip$

- $X$ ranges over possible segments
- An additional component checks whether $X$ is well-formed, i.e., indices match