Composition

L445 / L545

Dept. of Linguistics, Indiana University

Spring 2017

Finite-state morphology

We have seen how to handle morphology with FSTs

We will step back & formally characterize morphological operations, focusing on *composition*

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

Material is adapated from Roark & Sproat (2007), *Computational Approaches to Morphology and Syntax*, esp. ch. 2

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

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: i) detect structure of word forms, and ii) relate word forms

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

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Syntagmatic variation
Simple concatenation

Given a stem A and a suffix β, we can create a form Γ with regular concatenation:

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

What if instead we have a function β' which takes a string as input & outputs a string concatenated with β?

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

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

Now, we have:

(3) $\Gamma = A \circ \beta'$

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

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 (Π), with phonological rule (/s/, /z/, /iz/) implemented by transducer T

(4) $\Pi = [S \cdot \sigma] \circ T$

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

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

(7) New affix σ': $\Pi = S \circ \sigma'$

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Syntagmatic variation
Prosodically Governed Concatenation

Some affixes have prosodic conditions, e.g., comparative *-er* and superlative *-est* in English

- Generally speaking: ony attach to monosyllabic or disyllabic stems
- The base/stem can be characterized as:

  (8) $B = C^*VC^*(VC^*)?$

- and the affix as:

  (9) $\kappa = B[\epsilon : er[+COMP]]$

- resulting in:

  (10) $\Gamma = A \circ \kappa$

  - The only non-null Γ cases will be the ones where the base of A matches B
  - i.e., instead of Σ^* as the base, B is the base

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

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) caw $\circ T_{cvc(c)} =$ caw

(13) diiyl $\circ T_{cvc(c)} =$ diyl

(14) hiwiit $\circ T_{cvc(c)} =$ hiwt

Morpheme *-inay* is represented as:

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

Syntagmatic variation
Subsegmental morphology

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

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
  - *bád* /d/ (NOM) ↦ *báid* /d^y/ (GEN)
- This is easily captured by defining a function γ which is a palatalization operation.

Genitive (Γ) is defined as a composition operation of γ applied to the nominative form (N):

(16) $\Gamma = N \circ \gamma$

Syntagmatic variation
Extrametrical infixation

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Consider infixes like *-um-* in Philipino languages, e.g., Bontoc

- Ignores the onset sound of the word and prefixes to the remainder of the word
  - *antj'ǒak* 'tall': *umantj'ǒak* 'I am getting taller'
  - *k'ǎwǐsat* 'good': *kum'ǎwǐsat* '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., *-um-*)

Syntagmatic variation
Extrametrical infixation (2)

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Two-step insertion process:

1. Marker transducer M: insert > at appropriate spot

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

2. Infixation transducer ι: map > to *-um-*

Precompose these two steps:

(18) $\mu = M \circ \iota$

Meaning that a final word form is:

(19) $\Gamma = A \circ \mu$

Syntagmatic variation
Root-and-pattern morphology

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Arabic verbs (derivational morphology):

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

Pattern	Template	Verb stem	Gloss
I	$C_1aC_2aC_3$	*katab*	'wrote'
II	$C_1aC_2C_2aC_3$	*kattab*	'caused to write'
III	$C_1aaC_2aC_3$	*kaatab*	'corresponded'
VII	$nC_1aC_2aC_3$	*nkatab*	'subscribed'
...	...	...	...

Syntagmatic variation
Root-and-pattern morphology (2)

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Templates:

(20) $\tau_I = CaCaC$

(21) $\tau_{II} = CaCCaC$

(22) $\tau_{III} = CaaCaC$

(23) $\tau_{VIII} = [\epsilon : n]CaCaC$

...

To obtain a transducer for all these templates:

(24) $\tau = \bigcup_{p \in patterns} \tau_p$

Syntagmatic variation

Root-and-pattern morphology (3)

Need a transducer to link the root to the templates:

- Allow for optional vowels between consonants:

  (25) $\lambda_1 = C[\epsilon : V]^* C[\epsilon : V]^* C$

- Allow for doubling of center consonant (pattern II)
  ... need general rewrite rules:

  (26) $\lambda_2 : C_i \rightarrow C_i C_i$

  (27) $\lambda = \lambda_1 \circ \lambda_2$

Derive forms:

(28) $\Gamma = P \circ \lambda \circ \tau$

One can also compile $\lambda \circ \tau$ into its own "pattern" machine

Paradigmatic variation

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

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

	Singular	Plural
Nominative	femina	feminae
Genitive	feminae	feminarum
Dative	feminae	feminis
Accusative	feminam	feminas
Ablative	femina	feminis

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 α)
   - NEUT NOM ∪ ACC SG → NEUTNASG
   - NEUT NOM ∪ ACC PL → NEUTNAPL
   - NOM SG → NOMSG
   - GENDER DAT PL → DATABLPL
   - GENDER ABL PL → DATABLPL

   ...
2. Relate morphomic forms to particular surface forms (for a particular word class) (transducer σ)
   - Σ^* [I-II DATABLPL : is]
   - Σ^* [NEUTNAPL : a]
   - Σ^* [I-II NEUTNASG : um]
   - Σ^* [III DATABLPL : ibus]

   ...

Paradigmatic variation

A Computational Characterization (2)

Given a set of bases annotated with morphosyntactic features, inflected forms are derived as so:

(29) $\Gamma = B \circ \alpha \circ \sigma$

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

Infinitive	Gloss	Preterite
haldan	'hold'	haíhald
ga-staldan	''possess'	ga-staístald
af-áikan	'deny'	af-aiáik
slepan	'sleep'	saíslep

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-codes the proper sequences to obtain copying

- e.g., 1) ϵ:h arc, 2) ϵ:aí 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*
rice searcher watcher MARKER
malo-nyinina-filéla
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)

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

Think of reduplication as two 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 the prefix matches the base

Unbounded Reduplication (3)

Composition
Finite-state morphology
Syntagmatic variation
Simple concatenation
Prosodically Governed Concatenation
Subsegmental morphology
Extrametrical infixation
Root-and-pattern morphology
Paradigmatic variation
Reduplication

For Gothic, assume transducer *R*, which composes with a base β and adds indices to elements in prefix and base

(33) $\alpha = \beta \circ R = (A_1)C_2 a\acute{i}\beta'$

Input stem *skáip* will result in the output $X_1 X_2 a\acute{\imath}s_1 k_2 \acute{a}ip$

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