Words and Morphology

April 27, 2023

1 Linguistic Morphology

Linguistic morphology is the study of the structure of words and morphemes, which are the smallest units of meaning in a language.

1.1 Morphology

Morphology deals with the internal structure of words and how morphemes combine to form words. Morphemes can be divided into different kinds such as:

- Roots: The central morphemes in words, which carry the main meaning
- Affixes
  - Prefixes
  - Suffixes
  - Infixes
    - inserted inside the base
  - Circumfixes
    - added both to the end and to the front
- Reduplication
  - sulat susulat
  - anak anak
- Apophony
- Transfixiation (root and pattern/ templatic morphology)
  - ktb kitab "book"

1.2 Functional Differences in Morphology

- Inflectional morphology
  - Adds information to a word consistent with its context within a sentence
- Derivational morphology
  - Creates new words with new meanings (and often with new parts of speech)
  - s/es suffix
  - -ing suffix: progressiveness of a verb

1.2.1 Get rid of inflectional morphemes

- Lemmatization: return the dictionaries form/
- Stemming: "Chop off"

1.3 Morphological Typology

- **Isolating** (or **analytic**) languages like Chinese or English have very little inflectional morphology and are also not rich in derivation. Most words consist of a single morpheme.
- **Agglutinative** languages like Turkish or Telugu have many affixes and can stack them one after another like beads on a string
- **Fusional** (or **flexional**) languages like Spanish or German pack many inflectional meanings into single affixes, so that they are morphologically rich without "stacking" prefixes or suffixes
- **Templatic** languages like Arabic or Amharic are a special kind of **fusional** languages that perform much of their morphological work by changes internal to the root.

1.3.1 Problem of Morphology

The problem with morphology is that inflectional morphology, especially, makes instances of the same word appear to be different words, which can increase **sparsity** and be problematic in information extraction and retrieval. However, morphology, both derivational and inflectional, encodes information that can be useful (or even essential) in NLP tasks.

1.3.2 Level of Analysis

Level	hugging	panicked	foxes
Lexical form	hug +V +Prog	panic +V +Past	fox +N +Pl fox +V +Sg
Morphemic form (intermediate form)	hug^ing#	panic^ed#	fox^s#
Orthographic form (surface form)	hugging	panicked	foxes

2 Finite State Technologies

Finite state technologies define regular relations between strings.

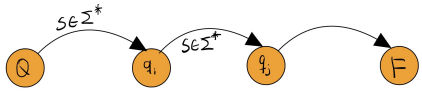
- Example:
  - foxes $\rightarrow$ fox+N+Pl
  - foxes $\rightarrow$ fox+V+3p+Pres

2.1 Finite-State Automaton (FSAs)

A finite-state automaton (FSA) is a mathematical model used to recognize regular languages. It consists of a finite set of states, a start state, a set of final states, and a set of transitions that connect the states.

- Q: a finite set of states
- $q_0 \in Q$: a special start state
- $F \subseteq Q$: a set of final styate
- Σ: a finite alphabet

Encodes a **set** of strings that can be recognized by following paths from q_0 to some state in F.



2.1.1 FSA and Regular Expressions

The set of languages that can be characterized by FSAs are called "**regular**" as in **"regular expression"**

2.1.2 Formal Languages

A **formal language** is a set of strings, typically one that can be generated/recognized by an automaton. However, a lot of NLP and CL involves treating natural languages like formal languages The set of languages that can be recognized by FSAs are called regular languages. Conveniently, (most) natural language morphologies.

2.2 Finite State Transducers

FSTs have an input tape and an output tape. Labels on transitions have to "sides" (Read/Write)

$$\langle Q, Q_0, F, \Sigma, \Delta, \delta \rangle$$

- A finite set of states Q
- A special start state $q_0 \in Q$
- A set of final states $F \subseteq Q$
- A finite input alphabet Σ
- A finite output alphabet Δ
- A state transition function δ

This mapping can be **many-to-many**.If the mapping is one-to-many or many-to-many, the FST is **non-deterministic.**

2.2.1 Properties

- FSTs are **invertable**. Inversion switches the input and output symbols/strings on each transition.
- FSTs are **Composable**.When two FSTs are composed, they behave as if the output of the first was fed as input to the second.
- FSTs are closed under union
  - FSTs are sets of pairs (mappings).Taking the union of two FSTs gives you the union of these mappings

2.2.2 Implementing Lemmatization with FST

1. Write and compile a morphotactic FST (with guesser and lexicon) that generates inflected forms from lemmas
2. Write and compile FSTs for allomorphic rules (spelling rules) that change morphemic forms into surface forms
3. Combine allomorphic FSTs (using composition, intersection, etc.) to form allomorphic FST
4. Compose morphotactic FST with allomorphic FST
5. Invert resulting FST

2.3 Misslaneous

As interest in non-English NLP grows, more tools are resources for morphology become available:

- Data resources (Unimorph, UD)
- Stemmers
- Analyzers

Stemming—reducing wordform to substrings such that each inflected form of a word yields the same string. Good for many quick-and-dirty NLP and information retrieval application