Acquisition of inflectional paradigms with minimal supervision

Radoslav Klíč

MFF UK

Outline

1. Introduction
2. Paramor
3. Seeding
4. Allomorphy
5. Edit distance
6. Results

Introduction

- The assignment: Acquisition of inflectional paradigms with minimal supervision
- The approach: Modification and extension of an unsupervised paradigm learner

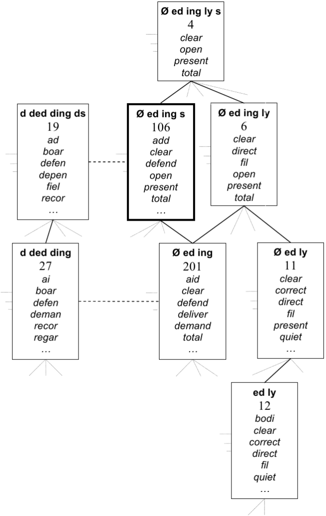
Work done in the thesis:

- Modification of Paramor, an unsupervised morphology learner by Monson (2009), to:
  - accept manually provided inflections with marked morpheme boundary.
  - handle allomorphy.
- A framework for hierarchical clustering using modified edit distance and other string distance metrics.

Schemes

- In Paramor, partial paradigms are modelled by *schemes*.
- A scheme is defined by a set of its suffixes e.g., (*0, ed, ing, s*).
- The scheme's stem set is obtained deterministically by selecting all the candidate stems which form a word (present in the corpus) with all the schemes suffixes.
- Thus, adding a suffix can decrease the number of scheme's adherent stems (and cannot increase it). (More stems combine with (*0, ed, ing, s*) than with (*0, ed, ing, s, ly*))

Scheme lattice



Paramor algorithm

- Bottom-up search. Starts with single-suffix schemes and ascends the lattice. Stops when the c-stem ratio drops below 0.25.
- Scheme clustering. Similar schemes are joined into scheme clusters. Similarity is defined as similarity of produced <stem, suffix> pair sets. For example, schemes (*0*, *ly*, *ness*) and (*0*, *ly*, *er*, *est*) can be merged, as they share a lot of stem-suffix pairs like *deep* + *0*, *deep* + *ly*.
- Scheme cluster pruning.

Seeding

- Seed example: *matk/matc/matek* + *a*, *u*, *y* / *e* / *0*
- Usage:
  - Add two-suffix schemes to the initial scheme set for bottom-up search. The suffix pairs are taken from the manual seed. (I use pairs because schemes with larger subsets need not be present in the corpus)
  - Protect some scheme clusters from discarding.
  - Induction of allomorphy rules.

Allomorphy

- Paramor does not recognise allomorphic stems. As a result, suffixes triggering phonological changes are often not selected in the bottom-up search, because they form words with different surface stems.
- I induce rules from the manual seed which allow Paramor to join two or more surface stems into one.
- For example, from a seed entry

  politik/politic + a, u, ovi, em, y, ů, ům / i, ích

  the following rule is generated:

  $*k \leftrightarrow *c$ / {*a, u, ovi, em, y, ů, ům*}, {*i, ích*}

Edit distance

In my clustering framework, I have experimented with modified Levenshtein distance, for which:

- the cost of operations linearly decreases with the position in the string where it occurs. (cost of *walk* → *talk* higher than *talked* → *talker*)
- the costs for diacritics adding/removing and vowel changes are lower than for other operations. (*hranici* → *hranicí*, *žena* → *ženy*)

Evaluation

- The subjects of evaluation are clusters of words which are compared to lexemes in a lemmatised corpus. (Lexeme – set of all forms of one lemma.)
- The evaluation method is pair-wise. For each pair of words, I check whether they belong to the same lemma and whether they belong to the same cluster created by the algorithm. I count true/false positives and true/false negatives and from them I get precision and recall to compute the F-score.

Results

Corpus	no seed	seed	edit	noseed + edit
cz	69.63	72.74	61.89	72.26
si	74.83	75.61	69.28	77.96
de	60.12	60.13	64.87	63.28
cat	62.74	65.95	56.50	63.91

Discussion

Problems with German:

- Stem-internal changes (*Mutter*/*Mütter*)
- Compounds – creation of schemes as (*0*, *organisation*) or (*0*, *gruppe*)

Future work

- Rules for stem-internal vowel change (*Mutter*/*Mütter*)
- More information sources (context, semantics, . . .)