LEXICON, MORPHOLOGY... AND NON-STANDARD ORTHOGRAPHY

References

- Essential references

- Extended references

“Generative” lexicon

Include in the lexicon any inflected word as independent and «atomic»?

- It will be inefficient:

<table>
<thead>
<tr>
<th></th>
<th>mangi-</th>
<th>sogn-</th>
<th>corr-</th>
<th>puff-</th>
</tr>
</thead>
<tbody>
<tr>
<td>-are/ere</td>
<td>mangi-are</td>
<td>sogn-are</td>
<td>corr-ere</td>
<td>puff-are</td>
</tr>
<tr>
<td>-o</td>
<td>mangi-o</td>
<td>sogn-o</td>
<td>corr-o</td>
<td>puff-o</td>
</tr>
<tr>
<td>-ato</td>
<td>mangi-ato</td>
<td>sogn-ato</td>
<td><em>corr-ato</em> (corso)</td>
<td>puff-ato</td>
</tr>
</tbody>
</table>

- In Turkish (agglutinative language) there would be 600x10^6 entries. In Finnish 10^7

- It will be non-informative:
  - No relation among lexical entries (alphabetical order is not interesting)
  - No processing hints (nouns = verbs?)
A computational lexicon can be conceived as:

- Mental lexicon – the relation among lexical units are psycholinguistically plausible?
- Computational lexicon – is the lexical representation efficient?

Rule of thumb:
- Lexical representation must be explicit and independent (with respect to the application that will use it)
- Global structure of lexical entries is as important as internal structure
- A lexicon must have a sufficient domain coverage (consider nearly 400,000 lexical entries)

“Generative” lexicon

XML tagged sentence:

E.g. XML coding for the "casa" (house) entry:

```xml
<word cat="noun" subcat="common.countable" num="sg" gen="f" sem="c12">
  casa
</word>
```

Remember:
- completeness does not ensure correctness (neither psycholinguistic nor computational)
- psycholinguistic plausibility does not guarantee computational effectiveness (and the way around)

“Generative” lexicon

Computational Lexicon evaluation parameters:
- Coverage (sufficient domain extension, and depth, also in terms of featural information)
- Extensibility (how easy it is enriching the lexicon?)
- Utility (single application benefit)

Single entry structure:

- orthographic/phonetic information
- morphology (inherent features like number, gendered...)
- syntactic (POS and more fine grained features: mass/countable, animacy, selection...)
- semantic (semantic relations, useful information for Machine Translation)

CASA ("house")

```
<word cat="noun" subcat="common.countable" num="sg" gen="f" sem="c12">
  casa
</word>
```

E.g. XML coding for the "casa" (house) entry:

```xml
<word cat="noun" subcat="common.countable" num="sg" gen="f" sem="c12">
  casa
</word>
```
"Generative" lexicon

DTD (Document Type Definition):
<?xml version="1.0" encoding="ISO-8859-1"?>
<!ELEMENT node (node|word)*>
<!ELEMENT word (#PCDATA)>
<!ATTLIST expression id CDATA #REQUIRED>
<!ATTLIST nodeid CDATA #IMPLIED>
<!ATTLIST nodecat CDATA #REQUIRED>
<!ATTLIST nodesubcat CDATA #IMPLIED>
<!ATTLIST noderef CDATA #IMPLIED>
<!ATTLIST noderole CDATA #IMPLIED>
<!ATTLIST nodeagree CDATA #IMPLIED>
<!ATTLIST nodelp CDATA #IMPLIED>
<!ATTLIST word id CDATA #IMPLIED>
<!ATTLIST word cat CDATA #REQUIRED>
<!ATTLIST word subcat CDATA #IMPLIED>
<!ATTLIST word ref CDATA #IMPLIED>
<!ATTLIST word agree CDATA #IMPLIED>
<!ATTLIST word role CDATA #IMPLIED>
<!ATTLIST word lemma CDATA #IMPLIED>
<!ATTLIST word lp CDATA #IMPLIED>
<!ATTLIST word sem CDATA #IMPLIED>

"Generative" lexicon

Global lexicon structure

- Subcategorization (and alternation Levin 1993) might be correlated with semantic class
- We could draw immediate inferences on the basis of the hierarchical organization of the items in an ontology (part_of, member_of...)

"Generative" lexicon

Simple text, corpus-extracted, lexicon (Tab-Separated Values, TSV):

<table>
<thead>
<tr>
<th>Token</th>
<th>type/lemma</th>
<th>cat</th>
<th>agree</th>
</tr>
</thead>
<tbody>
<tr>
<td>il</td>
<td>il</td>
<td>D.art.def</td>
<td>m.s</td>
</tr>
<tr>
<td>presidente</td>
<td>presidente</td>
<td>N.comm.count</td>
<td>m.s</td>
</tr>
<tr>
<td>non</td>
<td>non</td>
<td>ADV.neg</td>
<td></td>
</tr>
<tr>
<td>ha</td>
<td>avere</td>
<td>V.aux.ind.pres</td>
<td>3.s</td>
</tr>
<tr>
<td>commentare</td>
<td>commentare</td>
<td>V.part.past</td>
<td>m.s</td>
</tr>
</tbody>
</table>

"Generative" lexicon"

Global structure of the lexicon: semantic networks

- Wordnet (Miller 90)
  - Example of semantic network (goal: organizing the lexicon on the basis of the meaning rather than orthography) based on the following principles:
    - Relation among nouns (hierarchy and inheritance), verb (implicatures), adjectival and adverbial (oppositions) (but no functional words are included).
    - Every lexical concept (synset) can be represented using their synonyms (other synsets).
    - Kind of relations used:
      - Hyponymy (relation between a more concept general (bird) and a more specific one «robin»; «robin» is an hyponym of “bird”)
      - Hypernymy (inverse of hyponymy)
      - Meronymy (part_of)...
    - Representation of the «semantic ambiguity» problem: polysemy (cane = animal and cane = part of a gun, are two distinct synsets in wordnet).
Global structure of the lexicon: semantic networks
- Semantic Relations example (Miller 1993)

Morphology – the theoretical model
- Goal: recognizing a well-formed string and decompose it in morphemes
- Theoretical model:
  Surface form: \[
  \begin{array}{c}
  \# \ c \ a \ s \ a + e \ # \\
  \end{array}
  \]
  Lexical form: \[
  \begin{array}{c}
  \# \ c \ a \ s \ a + e \ # \\
  \end{array}
  \]

Morphological analysis with FSA
- An FSA can be used for recognizing or generating a lexical item, but also for representing the lexicon.
- FSA recognizing casa and its plural:
  \[
  q_0 \rightarrow c \rightarrow q_1 \rightarrow a \rightarrow q_2 \rightarrow s \rightarrow q_3 \rightarrow a \rightarrow q_4
  \]
  \[Q = \{q_0, q_1, q_2, q_3, q_4\},\]
  \[\Sigma = \{c, a, s, e, \#\},\]
  \[Q_0 = \{q_0\},\]
  \[F = \{q_4\},\]
  \[\delta = \]

Morphological analysis FSA and two-level morphology
- FSA limits: no memory, it is not possible to associate a structural description to an element recognized as belonging to the lexicon, simple FSAs are not sufficient (since it does not exist an external memory, there is no way to keep track of the derivation).
- Koskenniemi (83) two-level morphology: a lexical level and a superficial one that must be put in a specific relation one with the other.
- We use Finite-State Transducers (FST) to do so.

Finite-State Transducers (FST, o Transducers) 
\[ <Q, \Sigma, q_0, F, \delta> \]
- \( \Sigma \) = finite, non-null alphabet, input special (complex) chars of the form \( i:o \) where \( i \) are symbols of the input alphabet \( I \) and \( o \) are symbols of the output alphabet \( O \). \( \varepsilon \) (the null element) can be included both in \( I \) and in \( O \).
- \( \delta \) is defined as \( (q, q', i:o) \) and it represents a transition matrix putting in relation a state \( q \) (start) with a state \( q' \) (arrival) if the \( i:o \) relation is defined. \( \delta \) is then a relation from \( Q \) x \( \Sigma \) to \( Q \).
- FSAs define a formal language (set of strings);
- FSTs define relations among languages.

Example of inflectional morphology approach:
- Problem representation
  - examples: casa > cas; donna > donne; gatto > gatti; ago > agghi; sacco > sacchi ...
- Generalizations/intuitions
  - feminine nouns get as plural inflection «e», «i» for masculine. \( c \) and \( g \) become \( ch \) and \( gh \) respectively.
- formalization
  - regular case: masculine noun > @:C|G|@:CH|GH|@:I
    - feminine noun > @:C|G|@:CH|GH|@:AE
  - irregular case: uomo > @:O:A#:N#:I
- implementation
  - feminine noun > @:C|G|@:CH|GH|@ E:A #:+N #:+PL
  - e.g.: case > casa #:N #:+PL (c=c a:a s:s e:a #:+N #:+PL)

FSTs can be used as recognizers, generators, translators, correlators among sets.
- Some formal property of the FSTs:
  - Inversion (defined as \( T^{-1} \)): input and output labels can be inverted
  - composition, if \( T_1 \) maps \( I_1 \) to \( O_1 \) and \( T_2 \) transduces from \( I_2 \) to \( O_2 \), \( T_1 \circ T_2 \) maps \( I_1 \) into \( O_2 \).

Example of a FST for describing plural morphology in Italian:

Approximation of a FST for describing plural morphology in Italian:
Inadequacy of FSTs (and FSA) to express any morphological phenomena

- There are languages that present morphological derivations more complex than the one described. Such phenomena fall in the class of what we call non-concatenative morphology.

- Tagalog (Philippine dialect) uses infixes in the middle of a word:
  - um (marks the agent) + hingi («lend») = h-ulm-ingi («lend to someone»)

- Semitic languages, template morphology:
  - consonant roots (CCC) lmd («learn») + inflection by vocalic schemes (CV/CC) = lamad («learned»)
  - lymad («was learned»)

On inadequacy of FSTs (and FSA)

- Problems:
  - Non-determinism (multiple transition from the same state q might be pursued; ε transitions)
  - Inadequacy (e.g. non-concatenative morphology)
  - Order of application of FSAs

Morphological analysis: some application

- Information extraction
  - (web, unstructured corpora/digital archives)

- Keywords expansion:
  - (hotels in Florence = (hotel AND Florence) OR (hotels AND Florence))

- Stemming
  - retrieving the word root (stem) we can refine queries and make them more tolerant

- Porter Stemming Algorithm
  - simple set of cascade FSTs like:
    - ATIONAL -> ATE (e.g. relational -> relate)
    - ING -> ε (talking -> talk)

- pros e cons:
  - ipergeneralization (Krovetz 93)
    - e.g. organization > organ, generalization > generic,
  - Non captured exception:
    - matrices > matrix or European > Europe.
  - stemming is useful only with expansive research
    - (no standard information retrieval)
How is the mental lexicon structured?

- Full listing hypothesis – runs and run, are two lexical entries in the mental lexicon (no internal morphological structure)
- Minimum redundancy – only morphemes are encoded in the mental lexicon; accessing an inflected lexical item requires accessing distinct morphemes and combination rules

Evidence for a structured lexicon

- Priming Effects (Stanners et al. 79)
  - Irregular inflections: happiness, happily no priming with the root happy Vs. pouring > pour
- Semantic affinity (Marslen-Wilson 94)
  - Government > govern
- Pronunciation errors (Fromkin & Ratner 98)
  - *easy enough vs *easily enough

This suggests that some information of the morphemic structure of the word should be encoded in our mental lexicon.

Different levels, different strategies/resources:

- Lexical
- Syntactic
- Semantic
- Pragmatic

Remember that an error may be a true error, or a system error; e.g. at the lexical level a word could not be present in the lexicon, but present in the language (system error) or the result of a typo or a wrong belief on the orthography (true error)
Kinds of error correction approaches

- **Main idea:** pattern matching against lexical items stored in the repository + some heuristics to find the most suitable candidates for substitution
  - **Symbolic methods** (good representation of the problem)
  - **Sub-symbolic methods** (Machine Learning approach)

Error correction

- **Minimal Distance** (Damerau 64, Wagner 74)
  - the best alternative is the one that minimizes the number of insertions, deletions, substitutions and switching of chars
  - Compare the form with any possible transformation of this form and verify if there are alternatives present in the lexicon
  - This is very inefficient, at worst, we should compare the derived form with any lexical item in the vocabulary!

- **Similarity Key** (SOUNDEX algorithm, Odell e Russel 1918, Davidson 1962)
  - Procedure: extract a key from the wrong form; extract the keys from all lexical items; compare the keys and provide as correction the lexical items sharing the same key
  - **Key** = first letter of the word + sequence of numbers associated to the chars (according to some frequency calculus); «0» items (vowels) and repeated numbers are reported only once
  - Example:
    
    | Vowels | b, f, p, v | Other consonants |
    |--------|-----------|-----------------|
    | 0      | 1         | 2               |

    *casa = c020 > c2; csa = c20 > c2*
### Error correction

- **N-grams** (Kohonen 80; DeHer 82; Angell et al. 83; DeSmedt e VanBerkel 88)
  - word = set of overlapping substrings (n-grams)
  - example:
    - *casa* = \#c + ca + as + sa + a\$
    - *strumento* = \#st str tru num ume men ent nto to\$
  - Vocabulary = indexed n-grams table; every index represent a word in the lexicon.
  - The set of overlapping n-grams indicate an activation.

### Error correction in specific contexts: T9

- Keypad constraints in SMS typing
  - *Silfverberg et al. 1999*

  - Some classic solutions:

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>A</th>
<th>S</th>
<th>A</th>
<th>tot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-press</td>
<td>2-2-2</td>
<td>2</td>
<td>7-7-7</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>Two-key</td>
<td>2-3</td>
<td>2-1</td>
<td>7-4</td>
<td>2-1</td>
<td>8</td>
</tr>
<tr>
<td>T9</td>
<td>2</td>
<td>2</td>
<td>7</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

### Today’s key concepts

- What is a Computational Lexicon
  - Single entry structure (morpho-syntactic features)
  - Global structure (Wordnet)

- How do we deal with morphological analysis
  - Two-level morphology and FST
  - Some application (stemming)
  - The psycholinguistic plausibility of the model

- Input normalization and spell-checking
  - Error classification
  - Standard approach to spell correction (minimal distance, similarity keys, n-grams)
  - The case of T9