Counting Words:
Pre-Processing and Non-Randomness

Marco Baroni & Stefan Evert

Málaga, 11 August 2006

Pre-processing

▶ IT IS IMPORTANT!!! (Evert and Lüdeling 2001)
▶ Automated pre-processing often necessary (13,850 types begin with re- in BNC, 103,941 types begin with ri- in itWaC)
▶ We can rely on:
  ▶ POS tagging
  ▶ Lemmatization
  ▶ Pattern matching heuristics (e.g., candidate prefixed form must be analyzable as PRE+VERB, with VERB independently attested in corpus)
▶ However . . .

The problem with low frequency words

▶ Correct analysis of low frequency words is fundamental to measure productivity, estimate LNRE models
▶ Automated tools will tend to have lowest performance on low frequency forms:
  ▶ Statistical tools will suffer from lack of relevant training data
  ▶ Manually-crafted tools will probably lack the relevant resources
▶ Problems in both directions (under- and overestimation of hapax counts)
▶ Part of the more general “95% performance” problem
Underestimation of hapaxes

- The Italian TreeTagger lemmatizer is lexicon-based; out-of-lexicon words (e.g., productively formed words containing a prefix) are lemmatized as UNKNOWN
- No prefixed word with dash (ri-cadere) is in lexicon
- Writers are more likely to use dash to mark transparent morphological structure

Overestimation of hapaxes

- “Noise” generates hapax legomena
- The Italian TreeTagger thinks that dashed expressions containing pronoun-like strings are pronouns
- Dashed strings can be anything, including full sentences
- This creates a lot of pseudo-pronoun hapaxes: tu-tu, parapaponzi-ponzi-pò, altri-da-lui-simili-a-lui

Productivity of ri-
with and without an extended lexicon

Productivity of the pronoun class
before and after cleaning
**P** (and **V**) with/without correct post-processing

<table>
<thead>
<tr>
<th>class</th>
<th>V</th>
<th>V₁</th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>ri</em>-</td>
<td>1098</td>
<td>346</td>
<td>1,399,898</td>
<td>0.00025</td>
</tr>
<tr>
<td>pronouns</td>
<td>72</td>
<td>0</td>
<td>4,313,123</td>
<td>0</td>
</tr>
</tbody>
</table>

**With:**

**Without:**

<table>
<thead>
<tr>
<th>class</th>
<th>V</th>
<th>V₁</th>
<th>N</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>ri</em>-</td>
<td>318</td>
<td>8</td>
<td>1,268,244</td>
<td>0.000006</td>
</tr>
<tr>
<td>pronouns</td>
<td>348</td>
<td>206</td>
<td>4,314,381</td>
<td>0.000048</td>
</tr>
</tbody>
</table>

A final word on pre-processing

- IT IS IMPORTANT
- Often, major roadblock of lexical statistics investigations

Non-randomness

- LNRE modeling based on assumption that our corpora/datasets are *random* samples from the population
- This is obviously not the case
- Can we pretend that a corpus is random?
- What are the consequences of non-randomness?
A Brown-sized random sample from a ZM population estimated with Brown

Where does non-randomness come from?

- Syntax?
- *the the* should be most frequent English bigram
- If the problem is due to syntax, randomizing by sentence will not get rid of it (Baayen 2001, ch. 5)

The real Brown

The Brown randomized by sentence
Where does non-randomness come from?

- Not syntax (syntax has short span effect; *the* counts for 10k intervals are OK)
- **Underdispersion** of content-rich words
  - The chance of two Noriegas is closer to $\pi/2$ than $\pi^2$ (Church 2000)
  - *diethylstilbestrol* occurs 3 times in Brown, all in same document (recommendations on feed additives)
- Underdispersion will lead to serious **underestimation** of rare type count
- fZM estimated on Brown predicts $S = 115,539$ in English

Assessing extrapolation quality

- We have no way to assess goodness of fit of extrapolation from corpus to larger sample from same population
- However, we can estimate models on subset of available data, and extrapolate to full corpus size (Evert and Baroni 2006)
- I.e., use corpus as our population, sample from it
Goodness of fit to spectrum elements
Based on multivariate chi-squared statistic

<table>
<thead>
<tr>
<th>model</th>
<th>estimation size</th>
<th>max extrapolation size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>X2</td>
<td>df</td>
</tr>
<tr>
<td>ZM</td>
<td>7,856</td>
<td>14</td>
</tr>
<tr>
<td>fZM</td>
<td>539</td>
<td>13</td>
</tr>
<tr>
<td>GIGP</td>
<td>597</td>
<td>13</td>
</tr>
</tbody>
</table>

Compare to $V$ fit:

The corpus as a (non-)random sample

- In our experiment, we had access to full population (the Brown) and could take random sample from it.
- In real life, full corpus is our sample from the population (e.g., “English”, an author’s mental lexicon, all words generated by a wfp).
- If it is not random, there is nothing we can do about it (randomizing the sample will not help!).
What can we do?

- Abandon lexical statistics
- Live with it
- Re-define the population
- Try to account for underdispersion when computing the models (will get mathematically very complicated, but see Baayen 2001, ch. 5)

Outline

Motivation: studying distribution and $V$ growth rate of type-rich populations (sample captures only small proportion of types in population)

LNRE modeling:
- Population model with limited number of parameters (e.g., ZM), expressed in terms of type density function
- Equations to calculate expected $V$ and frequency spectrum in random samples of arbitrary size using population model
- Estimation of population parameters via fit of expected elements to observed frequency spectrum

zipfR package to apply LNRE modeling

Problems

Not always that bad

Our Mutual Friend
What we (and perhaps some of you?) would like to do next

▶ Study (and deal with) non-randomness
▶ Better parameter estimation
▶ Improve zipfR (any feature request?)
▶ Use LNRE modeling in applications, e.g.:
  ▶ Good-Turing-style estimation
  ▶ Productivity beyond morphology
  ▶ Better features for machine learning
  ▶ Mixture models

That's All, Folks!