Ling 555 — Programming for Linguists

Zipf's Law and part of speech tagging

Robert Albert Felty

Speech Research Laboratory
Indiana University

Nov. 24, 2008

Prereqs

While we are waiting

```
svn checkout svn://robfelty.com\
/home/robfelty/svn/l555 myl555
```

Outline

1. Homework questions
   - Homework comments
   - Homework questions
2. Zipf's Law
   - Installation script — setup.py
   - Testing Zipf's law
3. Working with tagged corpora
   - extracting tags
   - Doing a markov analysis with tags

Homework comments

- Output should be written to plain text files (not rich text formatted or word documents) (p.s. try to read a 30 year old plain text file — no problem. A 10 year old word file? Big problem.)
- Use sys.stdout.write instead of print.
  - Print doesn't know about character formatting (won't work with non-English characters)
- print should only be used for debugging

Homework questions

Zipf's Law

Definition

Zipfs law describes a relationship between the ranks and frequencies of words in natural languages . Specifically, it predicts that the frequency, f , of the word with rank r is:

$$f = cr^{-s} \tag{1}$$

where s and c are parameters that depend on the language and the text. If you take the logarithm of both sides of this equation, you get:

$$\log f = \log c - s \log r \tag{2}$$

So if you plot $\log f$ versus $\log r$, you should get a straight line with slope s and intercept $\log c$.

Power laws

- Zipf's Law is an example of a power law.
- The power is in the exponent s.
- Many natural phenomena follow power laws
  - The Internet
  - The world wide web
  - Social networks (e.g. academic citations)
- Complex networks studies these relations

Testing Zipf's law

- We can easily extend our corpus class to give us rank frequency.
- Then we simply print out the frequency and the rank frequency for every word in a text.
- Then we can use a plotting or statistics program to investigate the power law.

extracting tags

Look at `count_tags_orig.py` for an example of how to count the tags from a file in the Penn treebank.

Doing a markov analysis with tags

Let's work on doing a markov analysis using the tags

Doing a markov analysis with tags

Let's work on doing a markov analysis using the tags
Look at `count_tags.py` for an example of how to count the tags from a file in the Penn treebank.