Language Models for Spelling Correction

presented by Dustin Boswell

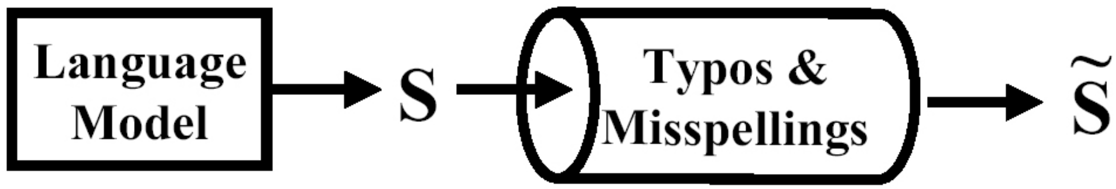
June 14, 2004

Noisy Channel Model of Spelling Correction

- S is a sequence of words that the writer intended
- $\widetilde{S}$ is the produced sequence, after typos and misspellings

- We are given $\widetilde{S}$, we have to figure out S.

Noisy Channel Model of Spelling Correction



- S^* is the best estimate of S given $\widetilde{S}$:

$$\begin{aligned} S^* &= arg \max_S P(S|\widetilde{S}) \\ &= arg \max_S \frac{P(\widetilde{S}|S) \cdot P(S)}{P(\widetilde{S})} \\ &= arg \max_S P(\widetilde{S}|S) \cdot P(S) \end{aligned}$$

- $P(S)$ is the language model
- $P(\widetilde{S}|S)$ is the error model

Fixing Single-Word Errors in Context

Motivating Example:

``of his was much --geater-- than we had thought''

Fixing Single-Word Errors in Context

<u>Motivating Example:</u>

``of his was much --geater-- than we had thought''

- Some Candidate Corrections:

  { greater, getter, heater, smaller }

Fixing Single-Word Errors in Context

Motivating Example:

``of his was much --geater-- than we had thought''

- Some Candidate Corrections:

  { greater, getter, heater, smaller }

- Language Model: getter, heater have very low probability

- Error Model: smaller $\longrightarrow$ geater is an unlikely error

- greater should maximize $P(\widetilde{S}|S) \cdot P(S)$

Project Outline

- Implement the following Language Models
  - Before-Bigram
  - After-Bigram
  - Spaced-Bigrams
  - Bag Of Bigrams
  - Hybrid Bigrams

- Compare their performance

Before-Bigrams

P(``of his was much greater than we had thought'') =

P(``much greater'') =

$$\frac{\text{count(``much greater'')}}{N}$$

N is the size of the training corpus

After-Bigrams

P(``of his was much greater than we had thought'') =

P(``greater than'') =

$$\frac{\text{count}(\text{``greater than''})}{N}$$

N is the size of the training corpus

Spaced-Bigrams

P(``of his was much greater than we had thought'')

$$= \mathrm{P}(\text{``greater''}) * \prod \mathrm{P}( \quad \bullet \quad | \text{``greater''})$$

$$= \frac{\mathrm{C}(\text{``greater''})}{N} *$$

$$\left(\frac{\mathrm{C}(\text{of, 4, greater})}{\mathrm{C}(\text{greater})} * \frac{\mathrm{C}(\text{his, 3, greater})}{\mathrm{C}(\text{greater})} \cdots \frac{\mathrm{C}(\text{greater,4,thought})}{\mathrm{C}(\text{greater})}\right)$$

- Takes a lot of memory! (And requires a lot of training data.)

Bag-of-Bigrams

P(``of his was much greater than we had thought'')

$= \text{P}(\text{``greater''}) * \prod \text{P}( \;\bullet\; \mid \text{``greater''})$

$= \frac{\text{c(``greater'')}}{N} *$

$\left(\frac{\text{c(of, X, greater)}}{\text{c(greater)}} * \frac{\text{c(his, X, greater)}}{\text{c(greater)}} \cdots \frac{\text{c(greater,X,thought)}}{\text{c(greater)}}\right)$

- discard all location information (just a bag of words)
- c(w_1, X, w_2) counts how many times
  w_1 and w_2 appeared in the same window.

Experiment Process

- Take a random window of 9 words. Eg:
  ``‘‘of his was much --greater-- than we had thought’’``
- Simulate a random 1-character typo: `greater` $\longrightarrow$ `geater`

Experiment Process

- Take a random window of 9 words. Eg:

  ``of his was much --greater-- than we had thought''

- Simulate a random 1-character typo: greater → geater

- Consider all words that are 1 typo away from geater:

  greater heater beater eater seater getter neater
  grater gefter gater geter gealer weater glater teater

  - Compute P(.. much greater than ..)
  - Compute P(.. much heater than ..)
  - Compute ...

- Choose candidate with highest probability

Experiment Results

The procedure was done on 100,000 windows for each language model.

Language Model Used	Accuracy
Null	12.9 %
Unigram	74.4 %
Before-Bigram	83.7 %
After-Bigram	84.6 %
Spaced-Bigrams	86.6 %
Bag Of Bigrams	79.2 %
Hybrid Bigrams	85.5 %

The end

Questions?