A Bayesian Approach to Query Language Identification

Jiří Materna[1,2] and Juraj Hreško[2]

[1]Centre for Natural Language Processing, FI MU Brno

[2]Research department at Seznam.cz, a.s.

December 3, 2011

Motivation

- Search engines
- Query language
  - language sensitive search
- Language of particular words in a query
  - morphological analysis
- Approaches for document language detection are insufficient

Existing approaches to language detection

- *n*-gram based approaches
  - compares letter *n*-gram histograms
  - compared using similarity metrics such as the cosine measure
  - Markov models
- dictionary based approaches
  - relative frequencies of words
  - need of thresholds for all languages
- other (based on phoneme transcription, compression rate, etc.)

The Bayesian approach I

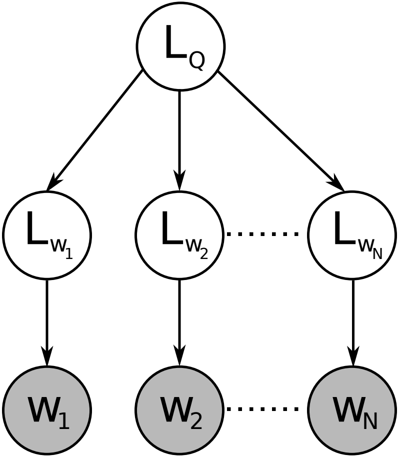


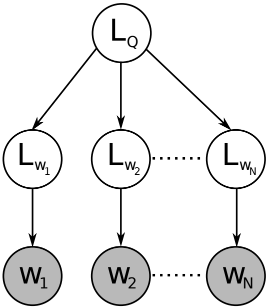
Figure: Graphical model for query language identification.

The Bayesian approach II

$P(L_Q)$ – prior probability of the language

$P(w_i|L_{w_i})$ – smoothed relative frequencies

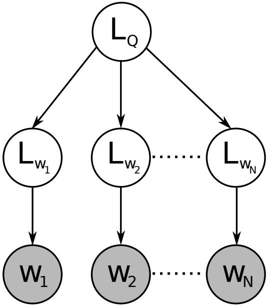
$$P(L_{w_i}|L_Q) = \begin{cases} \frac{9}{10} & \text{if } L_{w_i} = L_Q \\ \frac{1}{10} \times \frac{1}{|L|-1} & \text{else} \end{cases}$$



The inference I

$$P(L_Q|w_1, w_2, \ldots, w_N) = \frac{P(L_Q, w_1, w_2, \ldots, w_N)}{P(w_1, w_2, \ldots, w_N)}$$

$$P(L_{w_i}|w_1, w_2, \ldots, w_N) = \frac{P(L_{w_i}, w_1, w_2, \ldots, w_N)}{P(w_1, w_2, \ldots, w_N)}$$



Very inefficient.

The inference II

$$P(L_Q|w_1, w_2, \ldots, w_N) = \frac{P(L_Q)\prod_{i\in<1\ldots N>} P(w_i|L_Q)}{\sum_{L'_Q} P(L'_Q)\prod_{i\in<1\ldots N>} P(w_i|L'_Q)}$$

$$P(L_{w_i}|w_1, w_2, \ldots, w_N) = \sum_{L_Q} P(L_{w_i}|L_Q, w_i)P(L_Q|w_1, w_2, \ldots, w_N)$$

$$P(w_i|L_Q) = \sum_{L_w} P(w_i|L_w, L_Q)P(L_w|L_Q) = \sum_{L_w} P(w_i|L_w)P(L_w|L_Q)$$

$$P(L_{w_i}|L_Q, w_i) = \frac{P(w_i|L_{w_i})P(L_{w_i}|L_Q)P(L_Q)}{\sum_{L'_{w_i}} P(w_i|L'_{w_i})P(L'_{w_i}|L_Q)P(L_Q)}$$

Evaluation I

Compared against

- an *n*-gram implementation by Josef Toman (MFF UK):

  http://is.cuni.cz/studium/dipl_st/index.php?index.php?doo=detail&did=45800

- and the Google's algorithm:

  http://code.google.com/apis/ajax/playground/#language_detect

Language	cz	en	sk	de	pl	fr
Examples [%]	65.7	18.0	6.0	5.3	2.7	2.3

Table: Language distribution in the query test set (300 examples).

Evaluation II

Set/Method	Bayesian	Google API	*n*-gram
All languages	91.67 %	61.33 %	51.67 %
Czech	91.37 %	50.76 %	46.70 %
English	92.59 %	75.93 %	52.26 %
1 token	79.31 %	36.21 %	39.66 %
2 tokens	95.80 %	61.54 %	47.55 %
3 or more tokens	93.00 %	76.00 %	64.00 %

Table: Language identification accuracy on various test sets.

Conclusions

- Both *n*-gram and Google's approaches significantly outperformed.
- The detection of word languages performs with accuracy of 73.33%.
- Possible extension:
  - learn the word language matrix on some relevant data instead of using just the simple function
  - dependency on previous words in the query (Markov chain)

Thank you for your attention.