# Part-of-speech Tagging

Felix Ruzzoli

Albert-Ludwigs-Universität Freiburg

14. November 2011

# Part-of-speech

**Part-of-speech**,
also **word class** or **lexical category**

# Part-of-speech

Simplified:

- nouns
- verbs
- adjectives
- adverbs

# Part-of-speech

More complex and suited for computer analysis:

| Tag | Meaning | Example |
|---|---|---|
| DT | Determiner | the |
| VBZ | Verb, 3rd person singular present | takes |
| JJ | Adjective | green |
| NN | Noun, singular or mass | table |
| IN | Preposition or subordinating conjunction | in, of |
| NNP | Proper noun, singular | John |

# The Brown Corpus

**The Brown Corpus**

- developed at Brown University
- by Henry Kucera and Nelson Francis
- in the mid-1960s
- pioneered the field of corpus linguistics

# The Brown Corpus

- 500 Text samples
- about 1 million words
- compiled from works published in the U.S.

Today the Penn Treebank is most widely used.

# Ambiguities

**Trust nobody!**

vs.

**Invest in the family trust.**

# Ambiguities

First case: *Trust* is a verb.
Second case: *trust* is a noun.

**How to distuingish between the two?**

# Real Ambiguities

The Duchess was entertaining last night.

**No way to find the POS of 'entertaining'.**
**Even for a human.**

# Word vs. Token

- don't
- vice versa
- John's

# Simple Tagging

The baseline approach:

- Count how often a certain word is tagged with a certain category
- Apply the category to the token which occurs most often
- Tag unknown words as nouns

Already achieves about 90% accuracy.

# Rule based Taggers

- Assign to each token all the possible tags
- Apply rules which remove tags on the tokens
- One tag per token

The classic Brill Tagger works like this.

# Stochastic Taggers

Assign tags using probability theory:

- Hidden markov models
- Decision trees
- Maximum entropy probability distribution

TreeTagger & Stanford Tagger are stochastic.

# Machine Learning

These approaches can be combined with machine learning:

- Learn the rules to improve on errors
- Train perceptrons to categorize
- Learn the probability distributions

# Detailed look at the maximum entropy model

Used by Tsuruoka's and the stanford tagger.
Terminology:

- Token or Word ($w_i$)
- Tag ($t_i$)
- History (a set, $h_i$)
  - Token and Tag context
  - e.g. $h_i = \{w_i, w_{i+1}, w_{i+2}, w_{i-1}, w_{i-2}, t_{i-1}, t_{i-2}\}$
- Feature (a function, $f_j(h_i, t_i)$)
  - Input: Tag and History
  - Output: $\in \{0, 1\}$
  - e.g. $f_1(h_i, t_i) = 1$ if suffix($w_i$) = „ing“ and $t_i$ = VBG

# Detailed look at the maximum entropy model

The algorithm:

1. Generate feature space from training data
   - Compile a history for every token
   - Look at (history, tag) pairs and extract positive (helpful) features
   - Compile a list of positive features

$\Rightarrow$ Mapping from (history, tag) to a list of features (feature map)

# Detailed look at the maximum entropy model

The algorithm continued:

3. Generate tag dictionary (*optional*)
   - Contains for every token the tags found in the training data
4. Iterate input data word by word per sentence
5. Search a tag sequence which most likely resembles the word sequence
   - Assign a tag to each word
   - Compute probability of sequence using the feature map
   - Iterate

# Detailed look at the maximum entropy model

For more insights read:

- Berger, A. L., Pietra, V. J. D., & Pietra, S. A. D. (1996). **A maximum entropy approach to natural language processing**. Computational linguistics, 22(1), 39-71. Morristown, NJ, USA: MIT Press.
- Ratnaparkhi, A. (1996). **A Maximum Entropy Model for Part-Of-Speech Tagging**. Computational Linguistics, 1, 133-142.

# Practical speed

Test was conducted on stromboli using a textfile with 220k words

| Tagger | Running time |
|---|---|
| Tsuruoka's C++ Tagger | 33s |
| Stanford Tagger | 5s |
| TreeTagger | 2s |

**(Demo)**

# Profiling

Running profilers on Tsuruoka's C++ Tagger & Stanford Tagger.

| Tagger | Most time spent in | % of execution time |
|---|---|---|
| Tsuruoka's C++ | conditional_probability() | 79 |
| Stanford | MaxentTagger.getNum() | 16 |

Most expensive: HashMap lookups of the features.

# End of presentation

Thanks for listening.

# Sources


- Brill Tagger
  Brill, E. (1992).
  **A simple rule-based part of speech tagger.** Proceedings of the third conference on Applied natural language processing (pp. 112-116). Morristown, NJ, USA: Association for Computational Linguistics.
- TreeTagger
  Schmid, H. (1994).
  **Probabilistic part-of-speech tagging using decision trees.** Proceedings of International Conference on New Methods in Language Processing, 12, 44-49. doi:10.1.1.28.1139

# Sources


- Tsuruoka's Tagger
  Tsuruoka, Y., & Tsujii, J. (2005).
  **Bidirectional inference with the easiest-first strategy for tagging sequence data.** Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing - HLT '05 (pp. 467-474). Morristown, NJ, USA: Association for Computational Linguistics.