

Word Sense Disambiguation as Classification Problem

Tanja Gaustad
Alfa-Informatica
University of Groningen
The Netherlands
tanja@let.rug.nl
www.let.rug.nl/~tanja

Overview

- Introduction to Word Sense Disambiguation (WSD)
- WSD as Classification Problem

What problem are we talking about?



“Mijn vader **zagen** we niet meer.”

What problem are we talking about?

Problem: Many words have several meanings

Example: “Mijn vader **zagen** we niet meer.”

zagen ———— past tense of **zien** (to see)
 present tense of **zagen** (to saw)

?

Why do we need Word Sense Disambiguation?

Importance of WSD: Ambiguous words in given context have to be resolved for numerous NLP applications, e.g.:

- Machine Translation
- Information Retrieval
- Parsing
- Language Understanding

Example Case

Alpino: Language understanding system for Dutch; includes

- Hdrug, wide-coverage HPSG grammar for Dutch
- Large-scale lexicon
- Parser
- Disambiguation component

WSD integrated in Alpino: Reduction of ambiguity through

- Selecting probable reading of given word before parsing
- Checking lexical semantics of output parses

WSD in short

Problem: Lexical semantic ambiguity

Goal: Recover correct sense in a given context

Means:

- Collocational information (context words)
- Distributional information (frequency)
- Further related information (morphology, syntax, topic)
- World knowledge

Approach: Combine statistics, corpus and linguistics

Statistics and WSD

Prior probability $p(s_k)$:

probability of sense s_k (relative frequency)

Conditional probability $p(s_k|c)$:

probability of sense s_k given context word c

Joint probability $p(s_k, c)$:

probability of sense s_k occurring together with context word c

N.B. typically never enough occurrences of c to completely specify

$p(s_k, c)$

Statistics and WSD: Example “accident”

crash (Sense 1): Unfortunate or disastrous incident not caused deliberately; a mishap causing injury or damage; in particular, a crash involving road vehicles.

*Fears that fog could cause a serious **accident** on the M40 have united members of the District Council.*

chance (Sense 2): Something that happens without apparent or deliberate cause; a chance event or set of circumstances.

*We planned the first two children, but our third was an **accident**.*

Statistics and WSD: Prior vs. conditional probability

| | prior prob. | cond. prob. | |
|---------------|-------------|---------------|-------------|
| | | contextword c | prob. |
| crash | 0.82 | car | 1 |
| | | happy | 0 |
| | | historical | $0.\bar{3}$ |
| | | great | 0.5 |
| | | sir | 0.5 |
| chance | 0.18 | car | 0 |
| | | happy | 1 |
| | | historical | $0.\bar{6}$ |
| | | great | 0.5 |
| | | sir | 0.5 |

WSD as Classification Problem

Problem restated: Use statistical information about senses and contextwords to build model which correctly predicts word senses
⇒ Classify input (ambiguous words) into correct classes (senses).

Algorithm: e.g.

- Naive Bayes
- Maximum Entropy

Classification Algorithm I: Naive Bayes

Properties: Uses distributional and contextual information

Training: Bayes rule

$$P(s_k|c) = \frac{P(c|s_k)P(s_k)}{P(c)}$$

s_k : sense k of ambiguous word w

c : context words within context window (e.g. ± 3)

Testing: Bayes decision rule

Decide s' if $P(s'|c) > P(s_k|c)$ for $s_k \neq s'$

Classification Algorithm I: Naive Bayes

Input: Corpus

Training: For every ambiguous word, build training file containing:

- prior probability of all senses s_k
- cond. probability of all senses s_k given possible context words c

Testing: For ambiguous word compute sense s_k with highest score

Classification Algorithm I: Naive Bayes

“My mother already had a few **car accidents**.”

| | prior prob. | cond. prob. contextword c | prob. |
|---------------|-------------|------------------------------|-------|
| crash | 0.82 | car | 1 |
| chance | 0.18 | car | 0 |

- score for **crash**: $0.82 + 1 = 1.82$
- score for **chance**: $0.18 + 0 = 0.18$

Classification Algorithm II: Maximum Entropy

Maximum Entropy Principle: In absence of additional information, all events should have equal probability

Entropy: Self-information; measures amount of information contained in random variable

Constraints: Imposed by training data; basically combination of features and corresponding weight

Goal: Search distribution that maximises entropy while satisfying constraints imposed by training data

Classification Algorithm II: Maximum Entropy

$$p(c|x) = \frac{1}{Z} \exp \left(\sum_i \lambda_i f_i(x, c) \right)$$

$f_i(x, c)$ is number of times rule i is used to find class c for event x , and weights λ_i are chosen to maximise entropy of p ;

Properties:

- General technique for estimating probability distributions from data
- allows to integrate heterogeneous information sources

Classification Algorithm II: Maximum Entropy

Training:

- Select set of features, e.g.
lemma, part of speech, syntactic relation(s), topical information
- Compute weight for each feature (feature + weight = constraint)
- Compute model with maximal entropy which satisfies set of constraints

Testing: Classify test data according to model

Classification Algorithm II: Maximum Entropy

“My mother already had a few **car accidents**.”

Feature vector:

| | | | |
|-----------------|--------------|---------------------|--------------|
| accidents | accident | car | crash |
| <i>wordform</i> | <i>lemma</i> | <i>context word</i> | <i>class</i> |

| | |
|------------------------------------|-------------------------------------|
| $f(\text{accidents, crash}) = 0.7$ | $f(\text{accidents, chance}) = 0.2$ |
| $f(\text{accident, crash}) = 1.5$ | $f(\text{accident, chance}) = 0.9$ |
| $f(\text{car, crash}) = 3.5$ | $f(\text{car, chance}) = -2.2$ |

- score for **crash**: $0.7 + 1.5 + 3.5 = 5.7$
- score for **chance**: $0.2 + 0.9 + -2.2 = -1.1$

Conclusion

WSD: Important problem to be solved for successful NLP applications

Classification: Complex statistical models allow to restate WSD as a classification problem

Future Work: Assess use of different sources of information in statistical classification models