

# Erik - Katja



## Lexical Patterns or Dependency Patterns: Which is better?

Katja Hofmann and Erik Tjong Kim Sang  
University of Amsterdam

CLIN 2007

07 December 2007

### Debate: shallow parsing vs full parsing

- **Erik (engineer)**

Shallow parsing information (part-of-speech tags) is sufficient for performing NLP tasks. It is robust and easy to obtain. Parsers make many errors and they are slow!

- **Katja (scientist)**

Actually parsers are getting better and faster all the time. From parse trees you can obtain generalized information and retrieve non-local dependencies. Dependency parsing is the way to go!

Erik - Katja 0-0

1

CLIN 2007

07 December 2007

### Previous work

- Lexical patterns: P=0.48; R=0.45; F=0.46; T=45,979

Tjong Kim Sang and Hofmann, Automatic Extraction of Dutch Hypernym-Hyponym Pairs. In: *Proceedings of CLIN-2006*.

- Dependency parsing: P=0.22; R=0.30; F=0.25; T=5,115

Hofmann and Tjong Kim Sang, Automatic Extension of Non-English WordNets. In: *Proceedings of SIGIR'07*.

But papers use different evaluation methods and the systems have access to different amounts of training data.

Erik - Katja 0-0

2

CLIN 2007

07 December 2007

### Task: hypernym extraction

A hypernym of a term X is another term Y which both covers the meaning of X and is broader.

Examples:

meubel is a hypernym of tafel  
dier is a hypernym of slak  
iets is a hypernym of alles

Hyponym information can be found in lexical resources like WordNet and EuroWordNet but these resources are incomplete.

Erik - Katja 0-0

3

CLIN 2007

07 December 2007

### Identifying hypernyms with lexical patterns

- Hearst (1992) identified hypernym pairs from lexical patterns

- Example pattern: *H such as A, B and C*

- From the pattern we conclude: *H* is a hypernym of *A, B and C*

- The patterns make use of part-of-speech tags and lemma information

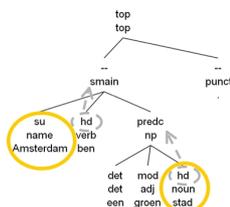
Erik - Katja 0-0

4

CLIN 2007

07 December 2007

### Finding hypernyms with dependency patterns



Erik - Katja 0-0

5

CLIN 2007

07 December 2007

### Predicting hypernymy from learned patterns

- Collect and store all contexts of noun pairs in sentences.

- Use frequent contexts (patterns) as features representing noun pairs.

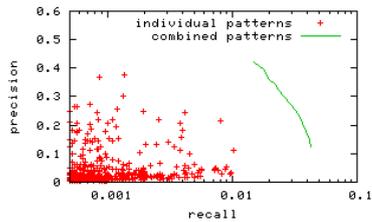
- Present a collection of positive (hypernym-hyponym) pairs and negative pairs to a machine learner as training data

- Let the system build a model for predicting whether two nouns are related according to hypernymy based on their contexts in text

Erik - Katja 0-0

6

### Graph of the approach



### Evaluation

- We measure precision (P), recall (R) and F-rates ( $F_{\beta=1}$ ) according to the performance on the test data
- Since it is difficult to evaluate both systems on exactly the same test set, we also record the target number of positive pairs (T)

### Results for the Twente Nieuws Corpus

News texts; 23 million sentences; 300 million words

	Lexical approach	Dependency approach
Targets	35,441	43,267
Precision	0.051	0.085
Recall	0.121	0.160
$F_{\beta=1}$	0.072	0.111

The dependency approach performs better despite many efforts (data representation optimizations) to improve the lexical results.

### Results for Wikipedia

Encyclopedic texts: 5 million sentences; 174 million words

	Lexical approach	Dependency approach
Targets	15,623	???
Precision	0.321	???
Recall	0.137	???
$F_{\beta=1}$	0.192	???

The lexical approach needs one day for processing Dutch Wikipedia. The dependency parsing approach requires several months!

### Concluding remarks

For the current task, automatically predicting noun hypernyms, dependency patterns outperform lexical patterns on **ALL** tested evaluation measures.

However, the dependency parsing approach needs considerably more computational resources than the lexical approach.

**Tip:** if your corpus is small or if you have access to a computer cluster, use the dependency parsing approach. Otherwise use the lexical approach.

**THE END**

Erik - Katja  
0 4