Identifying Hierarchical Structures

Erik F. Tjong Kim Sang
CNTS - Language Technology Group
University of Antwerp
Belgium
erikt@uia.ua.ac.be

Goal

Finding noun phrases and arbitrary phrases, preferably by using the res
of the base noun phrase and chunking work.

Approach

Bottom-up phrase recognition: identify phrases at one level of the tree w
using the phrases found at lower levels.

Basis

A good method for detecting base phrases.

Algorithm

We have used the memory-based learning algorithm IB1-IG, a nearest-neighbor classifier.

Tokens have been represented by a set of features from a window of surrounding words, part-of-speech tags and chunk tags.

All training data is stored and test data is classified by taking the class of the training data item that is closest to them in the feature space.

Evaluation measures

Phrase detection methods will be evaluated with four rates:

1. **Precision**: percentage of phrases that were found that were correct
2. **Recall**: percentage of correct phrases that were found
3. **F**: a combination of precision and recall:
   F = (2*precision*recall) / (precision+recall)
4. **Crossing rate**: average number of found phrases per sentence that cross correct phrases (only used for full parsing)

Clause Identification

	Precision	Recall	F
Baseline	98.44%	31.48%	47.71
Clause Parser	76.91%	60.61%	67.79
Collins Parser	89.1%	88.3%	88.7

- The baseline scores are produced by a system which puts every sentence in a single clause.
- The clause parser estimates open and close bracket positions and ties these together with heuristic rules.
- State-of-the-art parsers obtain up to F=89 for this task (Collins 1999, model 2, WSJ section 23).

Tasks

Task 1: NP Parsing

Find arbitrary noun phrases (CoNLL-99 shared task). Training data: secti
15-18 of the WSJ part of the Penn Treebank. Test data: WSJ section 2

Task 2: Clause Identification

Find clauses (CoNLL-2001 shared task). Training data: WSJ secti
15-18. Test data: WSJ section 20.

Task 3: Full Parsing

Build complete parse trees. Training data: WSJ sections 02-21. Test da
WSJ section 23.

Full Parsing

	Precision	Recall	F	CB
Baseline	94.15%	33.39%	49.30	0.06
Our Parser	82.34%	78.72%	80.49	1.69
Collins Parser	89.9%	89.6%	89.7	0.87

- The baseline results were obtained by our general chunker.
- The base level phrases are detected by a combination of five MBL systems; other levels use a single MBL system.
- Each test data level was processed with the corresponding training data level only, up to the maximum level of 20.
- State-of-the-art parsers obtain up to F=90 for this task (Collins 2000).

NP Parsing

	Precision	Recall	F
Baseline	93.24%	67.90%	78.58
NP Parser	90.00%	78.38%	83.79
Collins Parser	89.3%	90.4%	89.8

- The baseline scores have been obtained by our best NP chunker.
- The base level NPs are detected by a combination of five MBL syste
other levels use a single MBL system.
- Each test data level was processed with the corresponding training d
level only.
- State-of-the-art parsers obtain up to F=90 for this task (Collins 19
model 2, WSJ section 23).

Concluding remarks

- We have examined bottom-up chunk parsers applied to NP Parsi
Clause Identification and Full Parsing.
- The chunk parsers perform reasonably but worse than state-of-the-
statistical parsers.
- The prime problems of the parsers seems to be their greedy sea
strategy and their inability to use information of different parsing le
at the same time.