NLP Project: Yarowsky Word Sense Disambiguation Algorithm

Juho Kallio and Otto Wallenius

7. joulukuuta 2014

1 Project topic

As a mini-project for the Natural Language Processing class we implemented Yarowsky's unsupervised word sense disambiguation algorithm. The program takes as input an ambiguous word with more than one possible meanings — we call this word a *pattern* — and tries to solve all disambiguities regarding the pattern in a given corpus (the program works only for the 1988 AP news article corpus). Additionally, the program takes as input "seed words", one for each meaning of the pattern. These are words that the user thinks that occur often together with the pattern in cases where the pattern is used in a specific sense. In the following sections we present some test results and thoughts about them.

2 Results

Below are some results from test runs. Accuracy is the percentage of cases where the algorithm agreed with a human on the meaning of an occurrence of the tested pattern. For each pattern the results were calculated using 100 occurrences.

Pattern	**Seeds**	**k**	**Threshold**	**Epsilon**	**Accuracy**
plant	growth, car	19	4.5	0.0001	94%
light	wind, bright	19	7.5	0.0001	60%
space	shuttle, office	10	7	0.2	89%
tank	army, gallons	19	10	0.001	88%
rock	music, stone	19	10	0.001	70%

3 Conclusions

The algorithm did not perform very well, worse than we expected. With low values of k, we experienced problems with functional words. Common words like *since* that generally do not correlate with a specific meaning of a pattern were chosen by the algorithm as good indicators for the majority sense. We believe this happened because of the smallish data set. When the lexicon is too big compared to the corpus, some functional words appear only with the more common sense of the original word. With a bigger value of k this effect diminishes. Also, the different meanings of patterns were in many cases quite unevenly distributed in the corpus. For example the meaning 'factory' for the pattern *plant* is far more frequent in the AP corpus than 'an organic plant'. This seemed to make the problem worse.

We got best results using a very small smoothing parameter (around $\epsilon = 0.001$) for the Laplace smoothing. This is the With the small data set the rules for the less common sense gained only few hits, and with a big epsilon these fell quickly out.

4 Instructions to run the program

To run the program, download the sources from

- `https://github.com/juhokallio/YarowskyWSD`

and the AP corpus from the department file system

- `/fs/home/tkt_plus/nlp/Corpus/ap-1988`.

Save the AP corpus files to a directory named `data` in the document root directory. Them extract the AP corpus files (with command `gzip -d *` in the data directory). After that, in the project root directory run the program with command `python YarowskyWSD pattern seed1 seed2 ...`

The parameters k, ϵ and classification threshold are fixed. They are set in `YarowskyWSD.py`.