# CPSC 503 - Final project
# Part-of-speech filtering in unsupervised learning to improve discourse relation recognition

**James Wright (31120074)**
Department of Computer Science
University of British Columbia
Vancouver, B.C., Canada, V6T 1Z4
jrwright@cs.ubc.ca

## Abstract

I evaluate a technique for improving the accuracy of discourse relation recognition by unsupervised classifiers that involves filtering the input features based upon their parts of speech. I report on experiments on various corpora and training set sizes in which classifiers trained on filtered features are less accurate than equivalent classifiers trained on unfiltered features.

## 1 Introduction

A discourse is a coherent collection of sentences. Discourse relations are relations that operate between clauses or sentences. It is desirable to be able to recognize discourse relations in natural language text. For example, knowledge of the discourse-level structure of a document can aid summarization (Blair-Goldensohn and McKeown, 2006).

Discourse relations are often signalled using *cue phrases* such as "because", "although", "for example", etc. Discourse relations signalled in this manner can be detected based on these cue phrases, as in Marcu (2000). However, discourse relations also frequently exist between two sentences or clauses without being explicitly signalled. Given the importance of discourse structure to natural language understanding, it is desirable to develop techniques for recognizing these relations without relying on cue phrases.

In this project I evaluated a part-of-speech based filtering technique for training classifiers to recognize discourse relationships in text. In Section 2, I describe some background concepts, including a sketch of the filtering technique that I evaluated and the corpora that I used. In Section 3, I survey some related work. In Section 4, I summarize the contributions of this project. In Section 5, I give a high-level description of the implementation. In Section 6, I summarize my results, and finally in Section 7, I evaluate the project, discuss some lessons learned, and suggest some avenues for future work.

## 2 Background

Marcu and Echihabi (2002) describe a system for training a naïve Bayesian classifier to recognize 4 coarse-grained discourse relations: CONTRAST, EXPLANATION-EVIDENCE, CONDITION, and ELABORATION. In this approach, training examples are mined from a large, unannotated corpus based on relatively unambiguous cue phrases. The cue phrases are then removed in order to simulate an implicate discourse relationship between the two spans. The classifier is trained on these examples *after* the cue phrases have been removed, with the goal of being able to classify discourse relations based on context alone. The features are all possible pairings of one word from each span.

In the same paper, Marcu and Echihabi report that they were able to improve the accuracy of their classifier by only including "most representative words" in the word-pair features, where most representative words are nouns, verbs, and cue phrases (although presumably not the cue phrases that were removed from the examples). The motivating hypothesis is that restricting the training set to these most representative words reduces the noise in the training data.

They report an improvement for classifiers of equivalent numbers of training examples, but only for small numbers of training examples. In other words, their EXPLANATION vs. ELABORATION classifier has a better performance when trained on $100,000$ filtered examples than when trained on $100,000$ unfiltered examples, but the classifier trained on $1,000,000$ unfiltered examples has a better performance than either, and no results are reported for a classifier trained on $1,000,000$ filtered examples.

In this project, I aimed to investigate whether

the reported improvement due to filtering held for large quantities of training data as well as for small quantities. In other words, is a classifier trained on $1,000,000$ filtered examples better than a classifier trained on $1,000,000$ unfiltered training examples? Or do the accuracies of filtered and unfiltered classifiers converge?

### 2.1 Corpora

Marcu and Echihabi (2002) used three corpora. The first was the "Raw" corpus, a corpus of approximately 1 billion words that they created by concatenating several unspecified corpora from the Linguistic Data Consortium that they had available. Marcu and Echihabi trained their unfiltered classifiers on examples extracted from this corpus.

The second was the BLLIP corpus (Charniak et al., 2000), a collection of approximately 30 million words of Wall Street Journal articles. The BLLIP corpus includes mechanically-generated parse trees for each article. Marcu and Echihabi trained their filtered classifiers from this corpus, using the supplied parse trees to determine the part of speech for each word.

The third was the RST Discourse Treebank (Carlson et al., 2002). This corpus annotates the text of 385 Wall Street Journal articles from the Penn Treebank (Marcus et al., 1995) with discourse relations. Marcu and Echihabi (2002) used this corpus to investigate the accuracy of their unfiltered classifier against true implicit discourse relations (as distinct from the implicit discourse relations that they simulated by removing cue phrases from explicit discourse relations).

In this project, I used two corpora. The first was the American National Corpus (Reppen et al., 2005), a corpus of approximately 18 million words. The ANC contains a variety of fiction and non-fiction text, and includes annotations that specify the parts of speech, noun phrases, verb phrases, and sentence boundaries contained in each file of the corpus. I used this corpus as an approximate equivalent to the BLLIP corpus.

The second corpus was a collection of approximately 250 million words' worth of news articles downloaded from the web. I built this corpus by automatically downloading articles found through searches of the Yahoo! News search engine (Yahoo!, 2007) (see Section 5.1). I used this corpus as an approximate equivalent to Marcu and Echihabi's Raw corpus.

## 3 Related work

Several other researchers have explored techniques similar to those of Marcu and Echihabi (2002). Blair-Goldensohn and McKeown (2006) reproduce Marcu and Echihabi's unfiltered discourse relationship classifier in order to improve the behaviour of their summarizer. Sporleder and Lascarides (2005) extend Marcu and Echihabi's unfiltered classifier by using a much richer set of lexical features (rather than just word-pairs), as well as a more sophisticated classifier. Similarly, Blair-Goldensohn (2007) uses a richer set of syntactic features extracted from parse trees to improve the accuracy of the classifier.

Finally, Sporleder and Lascarides (in press) evaluate the whole technique of using automatically-extracted examples to train classifiers for recognizing rhetorical relations. They suggest that this technique may not be valid at all, since its central assumption (that explicit discourse relations that have had their cue-phrases stripped are an adequate simulation of true implicit discourse relations) may not be valid.

## 4 Contribution

The main contribution of this project is to critically evaluate the "most representative words" filtering technique proposed by Marcu and Echihabi (2002) as an extension to their discourse relationship classifier. I compare the performance of filtered and unfiltering classifiers on two different corpora and with different quantities of training data.

Marcu and Echihabi (2002) use automatically-generated parse trees to determine the parts of speech that they use for filtering. Since parsing is considerably slower than statistical part-of-speech tagging,[1] I used a simple bigram based part of speech tagger to determine parts of speech rather than parse trees for this investigation. To confirm that this modified filtering method provides a valid basis for comparison, I also evaluated whether the simple part of speech tagger provides equivalently accurate part of speech tags for the filtering application as parse-tree based parts of speech.

## 5 Implementation

The project was implemented in three main implementation phases of the project. The first phase was

[1] In a proof-of-concept test, I was able to tag 7275 words from the New York Times section of the American National Corpus (Reppen et al., 2005) in 5.8 seconds using a bigram tagger (Coburn, 2005). The parser that was used to produce the syntax trees provided with the BLLIP corpus (Charniak, 2000) took 173.7 seconds to process the same data.

to collect the webnews corpus. The second phase was to extract the training examples from both the webnews corpus and the ANC. The final phase was to train and test several naïve Bayesian classifiers.

### 5.1 Webnews corpus

I used a Perl script to automatically issue searches to Yahoo! News (Yahoo!, 2007) using the Web-based API. The script then downloaded the article from each result, discarding any duplicates.

Each article was then "cleaned" using the `PotaModule` component from BootCaT (Baroni and Bernardini, 2004), which extracts the block of text with the lowest HTML tag density from a web page. This block is returned after having its HTML tags stripped out. The hypothesis is that the actual rich text portion of an article will likely have a lower tag density than the "boilerplate" navigation and advertising portions. Empirically this seems to have been a very workable assumption. The stripped articles that I have checked have always contained the text of the article itself.

See Table 1 for a complete list of the search terms used to drive the web search. The search terms were based on words appearing in my cue-phrase patterns (see Table 2).

Over the course of 6 weeks I was able to download approximately 250 million words of news article text.

### 5.2 Example extraction

Once the webnews corpus was prepared, the next step was to extract examples of text spans that were in a discourse relation based on unambiguous cue phrases. The cue phrases were then removed, creating a simulated implicit discourse relation that was used as a training example.

I extracted these examples using cue-phrase based patterns that unambiguously signal discourse relationships with high likelihood. For example, if a sentence begins with "Because" and contains a comma, then the two spans on either side of the comma are very likely to be in an Explanation relationship. Following Marcu and Echihabi (2002), I used four coarse-grained classes of discourse relationship: Condition, Contrast, Elaboration, and Explanation. Also following Marcu and Echihabi (2002), I extracted examples of the No-relationship-same-document relation, which consists of randomly-selected pairs of sentences that come from the same document but are separated by more than 3 sentences. It is assumed that no discourse relationship holds between these pairs of sentences, although this data will be noisy, since long-distance discourse relationships are in fact possible.

| | | |
|---|---|---|
| although | but | because |
| thus | if | for example |
| which | | |

Table 1: Search terms used to download news articles

| |
|---|
| Condition |
| [BOS If ... EOS][then ... EOS] |
| [BOS If ... ,][... EOS] |
| [BOS ...][if ... EOS] |
| Contrast |
| [BOS ... EOS][BOS But ... EOS] |
| [BOS ...][but ... EOS] |
| [BOS ...][although ... EOS] |
| [BOS Although ... ,][... EOS] |
| [BOS ... EOS][BOS However , ... EOS] |
| [BOS ...][however ... EOS] |
| [BOS ... EOS][BOS Nonetheless , ... EOS] |
| [BOS ...][nonetheless ... EOS] |
| [BOS ... EOS][BOS Nevertheless , ... EOS] |
| [BOS ...][nevertheless ... EOS] |
| [BOS ... ,][whereas ... EOS] |
| Elaboration |
| [BOS ... EOS][BOS For example ... EOS] |
| [BOS ...][which ... ,] |
| [BOS ... ][, for example ... EOS] |
| [BOS ... EOS][BOS ... for example EOS] |
| [BOS ... EOS][BOS For instance ... EOS] |
| [BOS ... ][for instance ... EOS] |
| [BOS ... EOS][BOS ... for instance EOS] |
| Explanation |
| [BOS ...][because ... EOS] |
| [BOS Because ... ,][... EOS] |
| [BOS ... EOS][BOS Thus , ... EOS] |
| [BOS ...][, therefore ... EOS] |
| [BOS ...][, hence ... EOS] |
| [BOS ... EOS][BOS Hence , ... EOS] |
| [BOS ... EOS][BOS Therefore , ... EOS] |
| [BOS ... ,][which is why ... EOS] |

Table 2: Cue-phrase patterns used to extract examples of text span pairs in discourse relationships.

I extracted 515,294 Condition examples, 1,389,064 Contrast examples, 234,687 Elaboration examples, 273,616 Explanation examples, and 911,991 No-relation-same-document examples from the webnews corpus. The totals for the ANC were considerably smaller: 38,348 examples of Condition, 95,466 examples of Contrast, 12,798 examples of Elaboration, and 19,087 examples of Explanation.

The cue-phrase patterns that I used are listed in Table 2. They consist of all the cue-phrase patterns listed in Marcu and Echihabi (2002), plus some additional patterns that I constructed based on the extensive list of cue phrases provided in Marcu (1997). I was only able to construct 29 patterns in total that I was confident were sufficiently unambiguous, compared to the roughly 40 patterns per relation that Blair-Goldensohn and McKeown (2006) used. However, Marcu (personal communication) suggested that even just the 14 patterns listed in Marcu and Echihabi (2002) would provide enough examples if used on a sufficiently large corpus.

Each example that was extracted based on a cue-phrase pattern had its cue-phrase removed, in order to convert an explicitly signalled discourse relationship into a simulated implicit relationship.

**Because** I hate broccoli**,** I will have peas instead. (1)

For example, sentence (1) would have been extracted based on the pattern [`BOS Because ... ,][... EOS]`. It would then have its cue phrase ("Because" and the comma) stripped, to produce the two spans (2) and (3).

I hate broccoli (2)

I will have peas instead. (3)

After the unfiltered examples had been extracted for each of the discourse relations from both corpora, I filtered them using a Perl script that determined the part of speech of each word using the `Lingua::EN::Tagger` Perl module (Coburn, 2005). The script then removed every word that was not either a verb or a noun.

In addition, I also filtered the unfiltered examples from the American National Corpus using a separate program that removed non-noun, non-verbs based on the part of speech annotations that are included with the corpus. The results of the two filtering scripts were slightly different, indicating that the two filtering methods are not 100% equivalent.

### 5.3 Training the classifiers

Once the training data had been extracted and filtered, I trained a variety of naïve Bayesian classifiers. Marcu and Echihabi (2002) report most of their results for 2-way classifiers that attempt to distinguish between two different discourse relations, so I decided to focus on 2-way classifiers as well, to facilitate comparison.

Ideally I would have trained a classifier for each possible pair of discourse relations, for each corpus, and for each filtering method (based on ANC annotations, based on bigram part of speech tagger output, and unfiltered). This would have resulted in 50 different classifiers, since there are 10 possible pairings, 3 filtering methods for the ANC data, and 2 filtering methods for the webnews corpus. Unfortunately, the classifiers take a very long time to train (some of the unfiltered classifiers took between 9 and 14 hours of runtime), and so due to time constraints I decided to concentrate on the 2-way classifier for the ELABORATION and EXPLANATION relations. I chose those two relations because the ELABORATION versus EXPLANATION classifier is the one that Marcu and Echihabi (2002) report most of their results for when comparing the accuracy of the filtered versus the unfiltered versions of their classifiers.

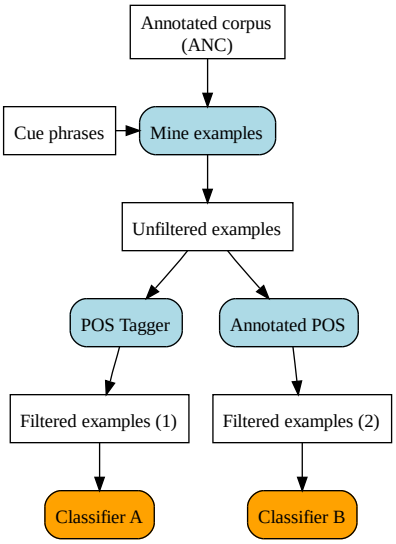


Figure 1: Data flow for training classifiers to compare bigram tagging versus annotated parts of speech

Every classifier was trained against a training set of 5000 examples of each relation that were removed randomly from the set of examples before training, so the baseline for each classifier is 50%. I used a development set of 2,302,011 words from the webnews corpus and 17,373 words from the American National Corpus, which was included in neither the training nor the test sets.

## 6 Results

In this section I describe the results of the project. The first sub-section describes the outcome of a comparison of filtering based on the output of a bigram part of speech tagger versus filtering based on the part of speech annotations of the American National Corpus. The second sub-section describes the out-

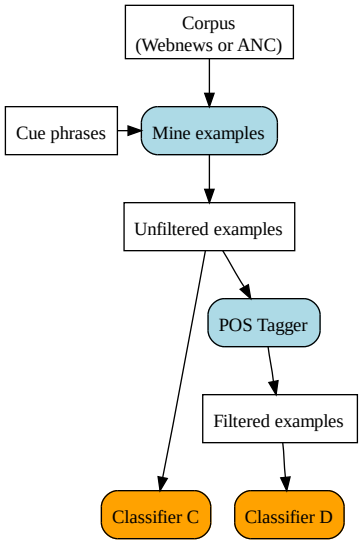


Figure 2: Data flow for training classifiers from filtered versus unfiltered examples

| Type of classifier | ANC | Tagger |
|---|---|---|
| CONDITION v. CONTRAST | 55% | 55% |
| CONDITION v. ELABORATION | 57% | 59% |
| CONDITION v. EXPLANATION | 60% | 60% |
| CONTRAST v. ELABORATION | 52% | 53% |
| CONTRAST v. EXPLANATION | 53% | 55% |
| ELABORATION v. EXPLANATION | 56% | 59% |

Table 3: Accuracy of 2-way classifiers trained on American National Corpus

come of a comparison of classifiers trained on filtered examples versus classifiers trained on unfiltered examples. The third sub-section provides a possible explanation for the difference between my results and those of Marcu and Echihabi (2002).

### 6.1 Bigram part of speech tagger versus annotated part of speech

Since the filtering script based on the bigram tagger produced slightly different output than the filtering program based on the part of speech annotations included with the ANC, I compared the accuracies of classifiers trained on each set of filtered output to determine whether the differences had an impact on accuracy. The results are presented in Table 3.

In no case did the classifiers trained on the output from the bigram-based filtering script have a lower accuracy than the equivalent classifier trained on the output from the ANC-annotation based filtering program. In fact, the classifiers trained from the output of the bigram-based filtering script tended to have a slightly higher accuracy.

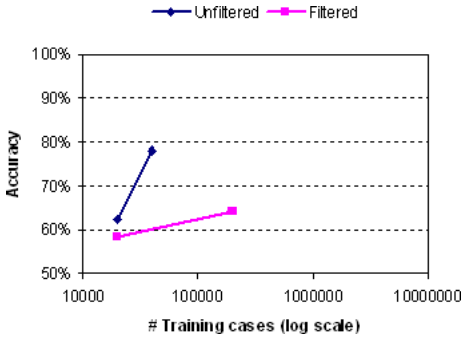


Figure 3: Accuracy of filtered vs. unfiltered EXPLANATION/ELABORATION classifiers

| | ANC | Webnews |
|---|---|---|
| **Filtered** | 59% | 64% |
| **Unfiltered** | 62% | 78% |

Table 4: Accuracy of filtered vs. unfiltered EXPLANATION/ELABORATION classifiers

From these results I concluded that using a bigram based part of speech tagger to determine parts of speech for filtering "most representative words" was equally as valid as using annotated parts of speech based on parse trees. I then moved on to comparing the accuracy of filtered versus unfiltered classifiers.

### 6.2 Part of speech filtering

To determine whether the relative accuracy advantage of filtered classifiers over unfiltered classifiers persisted as the size of the training set grew, I trained four EXPLANATION versus ELABORATION classifiers. The first two were trained on 21,885 examples from the ANC corpus; one was trained against filtered data, and one was trained against unfiltered data. The second two classifiers were trained on examples from the webnews corpus; one was trained against 240,000 filtered examples, and one was trained against 40,000 unfiltered examples.[2]

The results of the filtering comparison are shown in Figure 3 and Table 4. I was not able to reproduce Marcu and Echihabi's (2002) finding that filtering based on part of speech substantially improves the accuracy of equivalent classifiers. On the contrary, the unfiltered classifiers were more accurate. This was true even of the classifiers trained on the webnews corpus, where the filtered classifier was trained

[2]The different numbers of training examples for these two classifiers are again due to time constraints. Unfiltered classifiers are much slower to train than filtered classifiers, since each unfiltered example contains many more word-pair features than the equivalent filtered example.

on 6 times as many examples as the unfiltered classifier!

### 6.3 Explaining the difference in results

Clearly these results were quite strikingly different from Marcu and Echihabi's (2002). I believe that the most likely explanation for this difference has to do with the methodology that Marcu and Echihabi used for training their classifiers.

In their reported results, Marcu and Echihabi trained only on *filtered* examples from the BLLIP corpus, and only on *unfiltered* examples from the Raw corpus. There is no comparison between different corpora with the same style of filtering, nor between different styles of filtering for the same corpus. That means that there is no way to tell how much of the difference between the filtered classifiers and the unfiltered classifiers is due to filtering itself, versus how much is due to the differences between the two corpora.

But there are likely to be significant differences between the two corpora. The BLLIP corpus is extremely homogeneous, as it is composed exclusively of Wall Street Journal articles. The Raw corpus, on the other hand, is relatively heterogeneous, being composed of some number of corpora that have nothing particular in common aside from having happened to be available to Marcu and Echihabi.

These confounding corpora issues, in combination with my results in this project, make it seem likely that the improvements that Marcu and Echihabi (2002) report are in fact due to differences between the corpora that they used to train their filtered and unfiltered classifiers, rather than being due to the advantages of filtering itself.

## 7 Concluding remarks

In this section, I discuss some of the lessons that I learned while working on this project, I evaluate the success of the project, and I suggest some possible lines of future work.

### 7.1 Lessons learned

I learned a number of things while working on this project. The biggest lesson was this: When training classifiers from a massive corpus on a deadline, be sure to use any-time techniques! If I had not been able to stop the training and begin the testing of the unfiltered classifiers early, I would not have had any results to report by the time of the final presentation.

I also learned that it is very important to have a plan for how to deal with the experimental results regardless of what they turn out to be, because they will not always be what you expect or hope for.

Finally, I learned that it is vitally important to include progress feedback in long-running processes, to give some way to estimate how long they will require to finish and when to pull the plug.

### 7.2 Evaluation

I believe that this project was a qualified success. I obtained a result that is to my knowledge novel, namely that the improvement due to part of speech filtering that Marcu and Echihabi (2002) report does not appear to stand up to close examination. I also confirmed that using a simple part-of-speech tagger is effectively equivalent to using the parts of speech from a parse tree of the input for this filtering application (although in light of the main result, it is not clear why one would want to use this style of filtering in the first place).

The relatively small number of data points for filtered versus unfiltered classifiers is a definite weakness of this project. The results would be more compelling if they were backed up by a comprehensive set of classifiers for each possible discourse relation pair.

In addition, I was not able to demonstrate everything that I set out to demonstrate. I had originally intended to show that the advantage due to filtering persisted as the size of the training set increased, and then to show by testing against the RST Discourse Treebank that the improvements is accuracy were not spurious (i.e. that they translated into increased accuracy in recognizing true implicit discourse relations as well as in recognizing the implicit relations simulated by removing cue phrases from explicit relations). The actual results of the experiment dictated a different path, however. Arguably that is both a strength and a weakness of the project.

### 7.3 Future work

Although the results of this project strongly suggest that the part of speech based filtering presented by Marcu and Echihabi (2002) is not beneficial, they are not conclusive. This suggests a fairly obvious follow-up line of inquiry. One could perform the following experiment to demonstrate conclusively that part of speech filtering does not improve the accuracy of classifiers in this domain:

1. Train a set of filtering classifiers against the BLLIP corpus, and confirm that they have roughly the same accuracy as those reported by Marcu and Echihabi (2002).

2. Train an equivalent set of unfiltered classifiers against the BLLIP corpus.

3. Compare the accuracies of the classifiers trained in step 1 and step 2.

In light of the results of this project, it seems very likely that the unfiltered classifiers would have higher accuracies than their corresponding filtered classifiers. If that were true, it would conclusively demonstrate that the improvement reported by Marcu and Echihabi (2002) as being due to filtering was actually due to the differences between the corpora that they used.

## References


Marco Baroni and Silvia Bernardini. 2004. BootCaT: Bootstrapping corpora and terms from the web. In *Proceedings of LREC 2004*, pages 1313–1316.

Sasha Blair-Goldensohn and Kathleen McKeown. 2006. Integrating rhetorical-semantic relation models for query-focused summarization. In *Proceedings of 6th Document Understanding Conference (DUC2006)*.

Sasha Blair-Goldensohn. 2007. *Long-Answer Question Answering and Rhetorical-Semantic Relations*. Ph.D. thesis, Columbia University.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2002. RST discourse treebank. Linguistic Data Consortium, Philadelphia.

Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, John Hale, and Mark Johnson. 2000. BLLIP 1987-89 WSJ corpus release 1. Linguistic Data Consortium, Philadelphia.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the first conference on North American chapter of the Association for Computational Linguistics*, pages 132–139.

Aaron Coburn. 2005. Lingua::EN::Tagger - Part-of-speech tagger for English natural language processing. Available from `http://search.cpan.org/dist/Lingua-EN-Tagger/Tagger.pm`.

Daniel Marcu and Abdessamad Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *ACL '02: Proc. 40th Annual Meeting on Assoc. for Comp. Ling.*, pages 368–375.

Daniel Marcu. 1997. *The Rhetorical Parsing, Summarization and Generation of Natural Language Texts*. Ph.D. thesis, University of Toronto, Department of Computer Science.

Daniel Marcu. 2000. The rhetorical parsing of unrestricted texts: a surface-based approach. *Computational Linguistics*, 26(3):395–448.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1995. Treebank-2. Linguistic Data Consortium, Philadelphia.

Randi Reppen, Nancy Ide, and Keith Suderman. 2005. American national corpus (ANC) second release. Linguistic Data Consortium, Philadelphia.

Caroline Sporleder and Alex Lascarides. 2005. Exploiting linguistic cues to classify rhetorical relations. In *Proceedings of Recent Advances in Natural Language Processing (RANLP-05)*, pages 532–539.

Caroline Sporleder and Alex Lascarides. in press. Using automatically labelled examples to classify rhetorical relations: A critical assessment. (to appear in *Natural Language Engineering*).

Yahoo! 2007. Yahoo! news. `http://news.yahoo.com`.


## A Source code

The source code is available upon request as a `.zip` file. The zipfile contains the following files:

`all_terms.sh` I ran this script file as a `cron` job every 5 hours for roughly 6 weeks to collect the webnews corpus.

`collect_news_data.pl` This is the Perl script that `all_terms.sh` called to do the actual work of downloading the articles for the corpus.

`condition.patterns` List of patterns that indicate the CONDITION discourse relation with high likelihood.

`contrast.patterns` List of patterns that indicate the CONTRAST discourse relation with high likelihood.

`elaboration.patterns` List of patterns that indicate the ELABORATION discourse relation with high likelihood.

`explanation.patterns` List of patterns that indicate the EXPLANATION discourse relation with high likelihood.

`cpsc503-project.asd` ASDF project file for the Lisp portion of the source.

`cue-phrases.txt` List of cue phrases extracted from (Marcu, 1997).

`db-bayes.lisp` Implementation of the DB-backed naïve Bayesian classifier.

`filter-examples.pl` Script that uses the `Lingua::EN::Tagger` module to filter "most representative words" from example files.

`lingua-en-sentence.lisp` Lisp port of the `Lingua::EN::Sentence` Perl module for regexp-based sentence-splitting.

`mine-examples.lisp` Contains most of the code for mining example from corpus files based on patterns.

`mine-examples.pl` Initial Perl implementation of mine-examples. I wound up using a Lisp version instead of this one because the Perl version was too slow.

`naive-bayes.lisp` Implementation of an in-memory naïve Bayesian classifier.

`packages.lisp` Defines all the symbol packages used by the Lisp code.

`relation-classifiers.lisp` Code for training and testing the classifiers.

`rt.lisp` Unit testing library.

`stats.txt` Some statistics about different classifier types.

## B Corpora

The American National Corpus is available from the Linguistic Data Consortium.

The webnews corpus can be downloaded from `http://http://chumsley.org/webnews_corpus/`. Each `webnews-?.tar.gz` file contains several files named `txt-*`, containing one article per line, and `url-*`, containing one URL per line (indicating the source of the article on the corresponding line of the corresponding `txt-*` file).