

Name:

Incremental Statistical Parsing

Tutorial 7

The Parser Files

- The files and programs you will need are in:
/proj/courses/comppsy/Tutorial9/
Copy the entire directory to your account.

For this tutorial we will again experiment with the top-down incremental probabilistic parser developed by Brian Roark. While not explicitly developed as a cognitive model, it allows us to examine the kinds of behavior the parser exhibits for some of the lexical, structural, and combined ambiguities we've discussed in the lecture.

1. Incremental parsing

First, create a symbolic link to the parser:

```
ln -s /proj/contrib/tdp.distrib/bin/tdparse tdp
```

The parser will output the best k incremental parsers computed at each word if the `-z` switch is used. In this way, we can inspect the parsers preferred ranking of parses as it moves through the sentence. Look at the two sentences in `rels.txt`, and then parse them with the `-z` switch:

```
./tdp -v -k 2 -z -F rels.output  
/proj/contrib/tdp.distrib/parse.model.slc.p05 rels.txt
```

Note: if you want to use the tree viewer (`stanford-tregex-2012-07-09.jar`), you will need to add the closing brackets to the partial parse trees. The script `parenth_completion.py` will do this automatically: `python parenth_completion.py inputFile outputFile`
Alternatively, you can use the `-e` flag to make the parses easier to read in the output.

What behavior do you observe? Is this what you would expect? Briefly explain.

2. Relative clauses & transitivity

Now do the above for the three sentences in `relatives.txt`, which vary the transitivity of the verb.

3. NP/S Complement Ambiguity

The file `nps.txt` contains examples of the NP/S ambiguity discussed in class. Does the parser initially prefer the direct object parse, or the embedded subject. Explain why it behaves as it does.