Technologies for Storing and Accessing Information about Syntactic and Prosodic Structures: the Potential Utility of CorpusSearch and the like.

Presented by Yelena Fainleib
University of Massachusetts Amherst
yfainlei@linguist.umass.edu

Issue at hand: storage and access of syntactic and prosodic data.

- Researches in syntax-phonology interface need to address the data on multiple tiers.
- Structural data can be viewed under different perspectives.

Examples:

S V OBJ ADV -> $(S)_{IP}((V\ OBJ)_{ip}\ (ADV)_{ip})_{IP}$

NP V NP AdvP -> $(NP)_{IP}((V\ NP)_{ip}\ (AdvP)_{ip})_{IP}$

NP-S V NP-DO ADV-MNR -> $(NP\text{-}S)_{IP}((V\ NP\text{-}DO)_{ip}(ADV\text{-}MNR)_{ip})_{IP}$

Issue at hand: storage and access of syntactic and prosodic data.

- Structures with differ in syntactic arrangement can be parsed into same prosodic structures.

Examples:

S V OBJ ADV -> $(S)_{IP}(V\ OBJ\ ADV)_{IP}$

S V ADV OBJ -> $(S)_{IP}(V\ ADV\ OBJ)_{IP}$

S V DO -> $(S)\ (V\ DO)_{IP}$

V DO S -> $(V\ DO\ S)_{IP}$

Issue at hand: storage and access of syntactic and prosodic data.

- Prosodic parsing can change as a result of specific lexical items.

  Example:

  Short S V O -> (Short S V)(O)

  Long S V O -> (Long S)(V O)

- Same stimuli are elicited from more several participants.
- End result: large amount of syntactic structure/prosodic structure pairs.

Issue at hand: storage and access of syntactic and prosodic data.

- Database: labeled bracketing records of syntactic and prosodic matching configurations.
- Linear encoding: an utterance is one string.
  Example:
  IP[NP[I] V[saw] NP[him] GER[running]]

- Type of queries: noun phrase followed by a gerund.
- Impossible: V c-commanding NP

Issue at hand: storage and access of syntactic and prosodic data.

- Accounting for depth of embedding?
- Separate storage of labels/codes for additional aspects of the utterance: *eg.* complement of *think/believe verbs.*
- Need to predict in advance the type of structures that would be queried for.
- Types of queries are limited.

Software Overview:

- CorpusSearch and TregEx – software that search for patterns inside parsed corpora.
- Input: files containing parsed sentences: each word is labeled for its part of speech and surrounded by brackets.
- Queries: syntactic relations. *eg*. V c-commands PP; NP is a sister to a CP; VP directly dominates a CP and is preceded by a NP.

Software: Overview

- Output :list of sentences that match the syntactic query criteria.
- Output: can be used as an input to other queries.
- The syntactic parsing storage is not bound to any theory or analysis and is defined by user.
- Names of syntactic categories are user defined. *Eg.* NP, NP-OBJ, NOAM-CHOMSKY.

Software: Overview

- Obligatory: 1:1 pairing of category and the lexical item.

  Examples:

  *[NP the girl]

  [NP [DET the [N girl]]]

- Cost: none.

CorpusSearch: specifics

- Source: http://sourceforge.net/projects/corpussearch/
- Input: parsed corpora in Penn Treebank format.
- Environment: Java under Windows/Mac/Unix/Linux.
- Operation mode: command line.
- Knowledge of Unix/Dos commands is required.
- Queries are written in a command file; the output results are also written into a file.

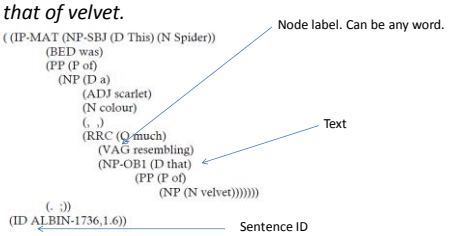
CorpusSearch: execution

- java -classpath "CS.jar" csearch/CorpusSearch query.q input.psd –o output.out
- java -classpath -> invoking Java environment
- "CS.jan" -> name of the file containing the CorpusSearch program
- csearch/CorpusSearch -> invoking the search command
- query.q -> file containing a search criteria command
- input.psd -> file containing parsed corpus sentences
- output.out -> file containing sentences which will match the search criteria
- Query and output file extensions are obligatory.

CorpusSearch: sentence parsing

- Input sentence:

  This spider was of a scarlet colour, much resembling that of velvet.



CorpusSearch: search options

CCommands
Column
Dominates
DomsWords
DomsWords<
DomsWords>
Exists
HasSister
iDominates
iDomsFirst
iDomsLast
iDomsMod
iDomsNumber
iDomsOnly
iDomsTotal
iDomsTotal<
iDomsTotal>
InID
iPrecedes
IsRoot
Precedes
SameIndex

CorpusSearch: search options

- Query: (NP-SBJ* idoms PRO$) AND (PRO$ ccommands NP*)
- Result:

```
(NP-SBJ (PRO$ his)
        (ADVR+Q ouermoch)
        (N fearinge)
        (PP (P of)
            (NP (PRO you))))
```

CorpusSearch: search options

- (NP-SBJ* precedes NP-OB1*)
  AND (NP-SBJ* iDominates ![1]PRO*))
  AND (NP-OB1* iDominates ![2]PRO*))

```
/~*
& +tat schal be a good hors.
(CMHORSES,85.9)
*~/

/*
    1 IP-MAT: 3 NP-SBJ, 7 NP-OB1, 4 D +tat, 10 N hors
*/

(0
   (1 IP-MAT (2 CONJ &)
             (3 NP-SBJ (4 D +tat))
             (5 MD schal)
             (6 BE be)
             (7 NP-OB1 (8 D a) (9 ADJ good) (10 N hors))
             (11 E_S .))
      (ID CMHORSES,85.9))
```

Matched sentence unparsed

Relevant nodes

CorpusSearch: search options

- (C iDominates that|That)

```
/~*
and he shalle do yow remedy, that youre herte shal be pleasyd. '
(CMMALORY,3.47)
*~/

/*
    12 CP-ADV: 13 C that
*/

(
      (12 CP-ADV (13 C that)
                 (14 IP-SUB
                             (15 NP-SBJ (16 PRO$ youre) (17 N herte))
                             (18 MD shal)
                             (19 BE be)
                             (20 VAN pleasyd)))
      (ID CMMALORY,3.47))
```

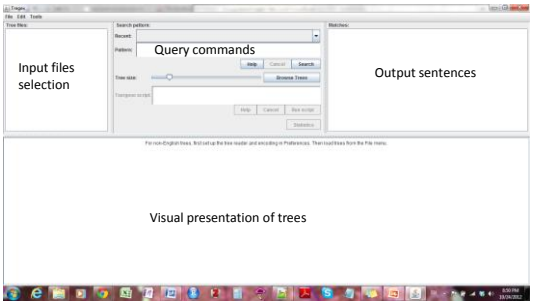
CorpusSearch: summary

- Pros:
  - Intuitive query language
  - Id for each sentence
  - Searches for nodes and labels
- Cons:
  - Inconvenient interface
  - Need to know OS specifics of command line workings
  - Not trivial to install
  - No graphical representation of parsed sentences

TregEx: specifics

- From: http://nlp.stanford.edu/software/tregex.shtml
- Input: parsed corpora in labeled bracketing format.
- Environment: Java under Windows/Mac/.
- Operation mode: graphic interface.
- Source files/Query commands: entered through a form.
- Results: presented visually.

TregEx: specifics



TregEx: sentence parsing

- Input sentence:
  This spider was of a scarlet colour, much resembling that of velvet.

```
(IP-MAT (NP-SBJ (D This) (N Spider))
        (BED was)
        (PP (P of)
          (NP (D a)
              (ADJ scarlet)
              (N colour)
              (RRC (Q much)
                (VAG resembling)
                (NP-OB1 (D that)
                    (PP (P of)
                        (NP (N velvet)))))))
        )
```

TregEx: search options

```
A << B
  A dominates B
A >> B
  A is dominated by B
A < B
  A immediately dominates B
A > B
  A is immediately dominated by B
A $ B
  A is a sister of B (and not equal to B)
A .. B
  A precedes B
A . B
  A immediately precedes B
A ,, B
  A follows B
A , B
  A immediately follows B
A <<, B
  B is a leftmost descendent of A
A <<- B
  B is a rightmost descendent of A
A >>, B
  A is a leftmost descendent of B
A >>- B
  A is a rightmost descendent of B
A <, B
  B is the first child of A
A >, B
  A is the first child of B
A <- B
  B is the last child of A
A >- B
  A is the last child of B
```

TrigEx: summary

- Pros:
  - Easy to install.
  - Convenient graphic interface
  - Graphic presentation of the trees.
  - Easy storage and retrieval of queries.
- Cons:
  - Encoding non lexical data as the leafs will be presented at the tree.
  - No convenient user manual.
  - Query language is not so intuitive.

Software: storage of prosody

- Both tools would allow parsing, storage and accessing of prosodic structures.

Example:

```
(IP (ip (PWd This) (PWd Spider))
        (ip (PWd wasofa)
                (PWd scarlet)
                (PWd colour))
                (ip (PWd much)
                   (PWd resembling)
                   (PWd thatof)
                              (PWd velvet)))
```

Software: storage of prosody

- Issues: matching with syntactic parsing; convenience of side-by-side comparisons.
- CorpusSearch: one could query for the utterances with syntactic relations. Then the ids of each utterance could be queried for their phonological structure.
- TregEx: one could append the phonological parsing to the syntactic one and view them side by side. The upper node would be the id of the particular sentence.

Acknowlegdgments

The Volkswagen Foundation
The Andrew W. Mellon Foundation