LSA 125 - Assignment 2 Database Creation

July 21, 2009

Our database swbd.tab contains the following variables (see the appendix for the macro file, the individual pattern files, and the options file):

- Item ID
- the head verb of the extracted VP
- the part-of-speech of the head verb
- the lemma of the head verb
- the sentence containing the extracted pattern
- the first PP
- the length of the first PP
- a categorical variable coding whether or not the first PP contains a pronoun
- a categorical variable coding whether or not the P in the first PP has an NP complement
- the pronoun in the first PP, if there is one
- the NP complement of the P in the first PP, if there is one
- two columns with information content and string length in words of the first PP
- the second PP
- the length of the second PP
- a categorical variable coding whether or not the second PP contains a pronoun
- a categorical variable coding whether or not the P in the second PP has an NP complement
- the pronoun in the second PP, if there is one

- the NP complement of the P in the second PP, if there is one
- two columns with information content and string length in words of the second PP

The database is now ready to be imported into a statistical analysis program like R.

A MACROS.ptn file

The MACROS.ptn file contains the main macro `@PPPP` that extracts all verb phrases containing exactly two PPs with potentially intervening disfluencies, as well as some helper macros:

```
@ WORD       /^'{0,1}[a-zA-Z]+.*/;
@ TERMINAL   * !< *;
@ NP         /^NP/;
@ VP         /^VP/;
@ PP         /^PP/;
@ DISFL      /EDITED|UH|PRN|-UNF/;
@ PPPP       (@VP=VP1 < (@PP=PP1 !<< -NONE- $.. (@PP=PP2 !<< -NONE- !$
             (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1)))));
```

B Pattern files

B.1 CatVar

ID.ptn

```
@PPPP
```

Pronoun1cat.ptn – checks whether PP1 contains a pronoun

```
@VP=VP1 < (@PP=PP1 !<< -NONE- < (@NP < PRP) $.. (@PP=PP2 !<< -NONE- !$
      (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

Pronoun2cat.ptn – checks whether PP2 contains a pronoun

```
@VP=VP1 < (@PP=PP1 !<< -NONE- $.. (@PP=PP2 < (@NP < PRP) !<< -NONE- !$
      (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

PNP1cat.ptn – checks whether the P in PP1 has an NP complement

```
@VP=VP1 < (@PP=PP1 !<< -NONE- <, IN <- (@NP=print !< PRP) $.. (@PP=PP2 !<<
      -NONE- !$ (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

PNP2cat.ptn – checks whether the P in PP2 has an NP complement

```
@VP=VP1 < (@PP=PP1 !<< -NONE- $.. (@PP=PP2 !<< -NONE- <, IN <- (@NP=print !<
      PRP) !$ (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

B.2 POSVar

POS.ptn – gets the part-of-speech of the verb

```
@PPPP < (/^(V|BES)/=pos = /^(V|BES)/=print)
```

B.3 StringVar

FORM.ptn – gets the verb that is the head noun of the extracted VP.

```
@PPPP < (/^(V|BES)/=pos = /^(V|BES)/=print)
```

SENTENCE.ptn – gets the entire sentence.

```
@PPPP >> (*=print !> *)
```

PP1.ptn – gets the first PP.

```
@VP=VP1 < (@PP=print !<< -NONE- $.. (@PP=PP2 !<< -NONE- !$ (@PP != =print) !,,
      (@TERMINAL ,, =print !>> (@DISFL >> =VP1))))
```

PP2.ptn – gets the second PP.

```
@VP=VP1 < (@PP=PP1 !<< -NONE- $.. (@PP=print !<< -NONE- !$ (@PP != =PP1) !,,
      (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

PNP1.ptn – gets the NP complement of the P in PP1, if it exists.

```
@VP=VP1 < (@PP=PP1 !<< -NONE- <, IN <- (@NP=print !< PRP) $.. (@PP=PP2 !<<
      -NONE- !$ (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

PNP2.ptn – gets the NP complement of the P in PP2, if it exists.

```
@VP=VP1 < (@PP=PP1 !<< -NONE- $.. (@PP=PP2 !<< -NONE- <, IN <- (@NP=print !<
      PRP) !$ (@PP != =PP1) !,, (@TERMINAL ,, =PP1 !>> (@DISFL >> =VP1))))
```

Pronoun1.ptn – gets the pronoun in PP1, if there is one.

```
@PPPP < @PP=pp2 < ((@PP < (@NP < PRP=print)) .. =pp2)
```

Pronoun2.ptn – gets the pronoun in PP2, if there is one.

```
@PPPP < @PP=pp1 < ((@PP < (@NP < PRP=print)) ,, =pp1)
```

C options

```
*************************************************************************

data=/home/lsa1/pp_project/data/swbd
results=/home/lsa1/pp_project/results
shellscripts=/home/lsa1/pp_project/shellscripts

corpus=swbd

init ID

add stringvar Verb=FORM
add posvar POS
add lemmavar Verb
add stringvar SENTENCE
add stringvar PP1
add lengthvar LenPP1=PP1
add stringvar Pronoun1
add categoricalvar PronounPP1=pronoun:Pronoun1cat
add stringvar PNP1
add categoricalvar PNP1cat=pnp:PNP1cat
add infodensity PP1
add stringvar PP2
add lengthvar LenPP2=PP2
add stringvar Pronoun2
add categoricalvar PronounPP2=pronoun:Pronoun2cat
add stringvar PNP2
add categoricalvar PNP2cat=pnp:PNP2cat
add infodensity PP2

*************************************************************************
```