

Wh-questions

Ling 567

February 16, 2016

Overview

- Target representation
- The problem
- Solution for English
- Solution for pseudo-English
- Lab 7 overview

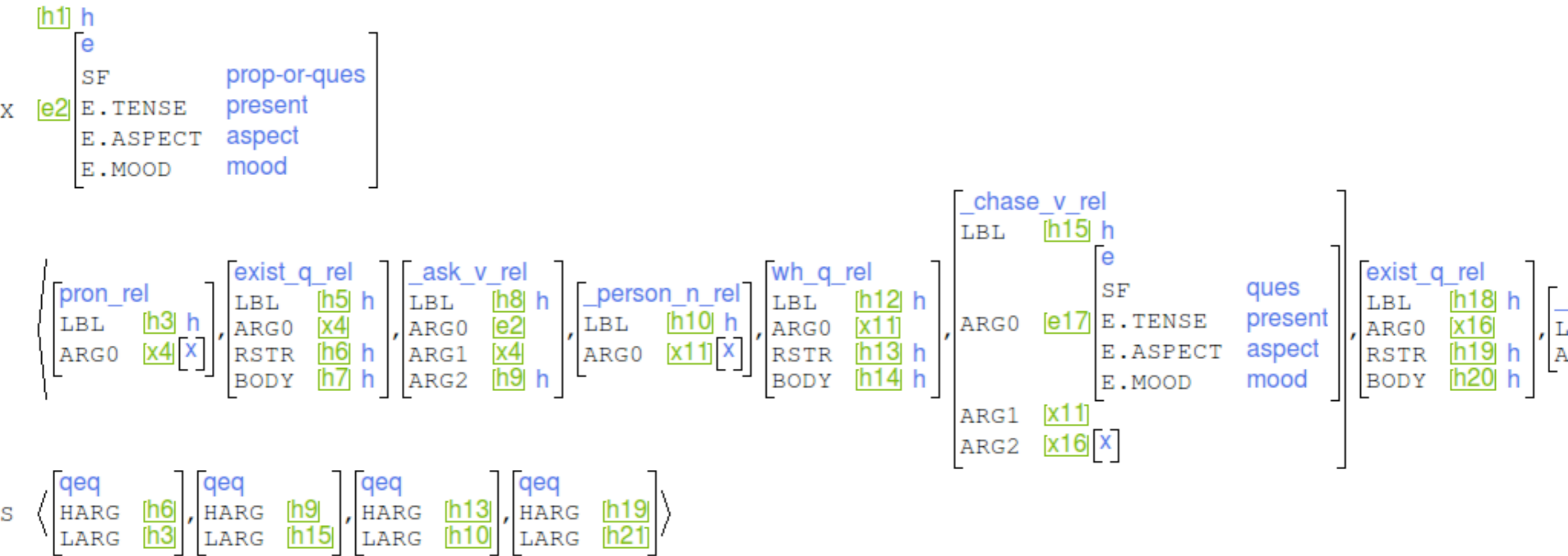
Wh-questions: Target representations

Who chases cars? Simple MRS Display

```
mrs
LTOP [h1] h
INDEX [e2] SF ques
      [e2] E.TENSE present
      E.ASPECT aspect
      E.MOOD mood
RELS  ( [ _person_n_rel ] , [ wh_q_rel ] , [ _chase_v_rel ] , [ _car_n_rel ] , [ exist_q_rel ] )
      ( LBL [h3] h , LBL [h5] h , LBL [h8] h , LBL [h10] h , LBL [h11] h )
      ( ARG0 [x4] [x] , ARG0 [x4] , ARG0 [e2] , ARG0 [x9] , ARG0 [x9] )
      ( RSTR [h6] h , RSTR [h6] h , ARG1 [x4] , RSTR [h12] h , RSTR [h12] h )
      ( BODY [h7] h , BODY [h7] h , ARG2 [x9] [x] , BODY [h13] h , BODY [h13] h )
HCONS ( [ qeq ] , [ qeq ] )
      ( HARG [h6] , HARG [h12] )
      ( LARG [h3] , LARG [h10] )
```


Wh-questions: Target representations

'I ask who chases the dog.' Simple MRS Display



The problem

- Wh questions involve long distance dependencies
 - Connecting sentence-initial wh-words to gaps
 - Detecting the presence of in-situ wh-words, and correlating [SF ques]

Solution for English

- SLASH (\leq GAP) to encode the absence of an element
 - LDD Top: Head-filler rule
 - LDD Bottom: Comp & subj extraction rules (unary phrase structure rules)
 - LDD Middle: Lexical threading/amalgamation of SLASH; constraints on rules other than head-argument
- QUE to encode the wh property
 - Controls eligibility to be filler in wh question rule
- Initial symbol should require SLASH $\langle ! ! \rangle$

Top: Head-Filler Rule

wh-ques-phrase := basic-head-filler-phrase & interrogative-clause &
head-final &
[SYNSEM.LOCAL.CAT [MC bool,
VAL #val,
HEAD verb & [FORM finite]],
HEAD-DTR.SYNSEM.LOCAL.CAT [MC na,
VAL #val & [SUBJ < >,
COMPS < >]],
NON-HEAD-DTR.SYNSEM.NON-LOCAL.QUE <! ref-ind !>].

Bottom: Subject & complement extraction

extracted-comp-phrase := basic-extracted-comp-phrase &
[SYNSEM.LOCAL.CAT.HEAD verb,
HEAD-DTR.SYNSEM.LOCAL.CAT.VAL.SUBJ cons].

extracted-subj-phrase := basic-extracted-subj-phrase &
[SYNSEM.LOCAL.CAT.HEAD verb,
HEAD-DTR.SYNSEM.LOCAL.CAT.VAL.COMPS < >].

Bottom: Subject & complement extraction

```
basic-extracted-comp-phrase := basic-extracted-arg-phrase & head-compositional &
[ SYNSEM canonical-synsem &
  [ LOCAL.CAT [ VAL [ SUBJ #subj,
                    SPR #spr,
                    COMPS #comps ],
                MC #mc ] ],
HEAD-DTR [ SYNSEM
  [ LOCAL.CAT [ VAL [ SUBJ #subj,
                    SPR #spr,
                    COMPS < gap &
                        [ NON-LOCAL.SLASH #slash ]
                        . #comps > ],
                MC #mc ],
  NON-LOCAL.SLASH #slash ] ],
C-CONT [ RELS <! !>,
        HCONS <! !>,
        ICONS <! !> ] ].
```

Bottom: Subject & complement extraction

gap := expressed-non-canonical &
[LOCAL #local,
NON-LOCAL [REL 0-dlist,
QUE 0-dlist,
SLASH 1-dlist &
[LIST < #local >]]].

Middle: Lexical threading of SLASH values (cf: GAP Principle, Ling 566)

- Inspired by Bouma, Malouf & Sag 2001
- SLASH on mother comes from head daughter (except in head-adj rules)
- Elements with non-empty ARG-ST build their own SLASH value by appending SLASH of their arguments (basic-one-arg et al)
- Unexpressed arguments are stamped with the type unexpressed:

```
unexpressed := synsem-min &  
  [ NON-LOCAL [ SLASH 0-dlist,  
                REL 0-dlist,  
                QUE 0-dlist ] ].
```

Middle: head-mod phrases

```
basic-head-mod-phrase-simple := head-mod-phrase & binary-headed-phrase &
[ SYNSEM [ LOCAL.CAT.MKG #mkg,
          NON-LOCAL [ SLASH [ LIST #first,
                          LAST #last ],
                      REL 0-dlist ] ],
HEAD-DTR.SYNSEM
[ ... ,
  NON-LOCAL #nonloc &
  [ SLASH [ LIST #middle,
          LAST #last ] ],
NON-HEAD-DTR.SYNSEM
[ ... ,
  NON-LOCAL [ SLASH [ LIST #first,
                    LAST #middle ],
            QUE 0-dlist & [ LIST null ] ] ],
C-CONT [ RELS <! !>, ICONS <! !> ] ].
```

The wh words

- english.tdl

```
wh-pronoun-noun-lex := norm-hook-lex-item & basic-icons-lex-item &
  [ SYNSEM [ LOCAL [ CAT [ HEAD noun,
    VAL [ SPR < >,
      SUBJ < >,
      COMPS < >,
      SPEC < > ] ],
    CONT [ RELS <! [ LBL #larg,
      ARG0 #ind & ref-ind ],
      [ PRED "wh_q_rel",
        ARG0 #ind,
        RSTR #harg ] !>,
      HCONS <! [ HARG #harg,
        LARG #larg ] !> ] ],
    NON-LOCAL.QUE <! #ind !> ] ].
```

- lexicon.tdl

The wh words

- english.tdl

```
wh-pronoun-noun-lex := norm-hook-lex-item & basic-icons-lex-item &
  [ SYNSEM [ LOCAL [ CAT [ HEAD noun,
                        VAL [ SPR < >,
                              SUBJ < >,
                              COMPS < >,
                              SPEC < > ] ],
            CONT [ RELS <! [ LBL #larg,
                            ARG0 #ind & ref-ind ],
                  [ PRED "wh_q_rel",
                    ARG0 #ind,
                    RSTR #harg ] !>,
                  HCONS <! [ HARG #harg,
                              LARG #larg ] !> ] ],
            NON-LOCAL.QUE <! #ind !> ] ].
```

- lexicon.tdl

```
what := wh-pronoun-noun-lex &
  [ STEM < "what" >,
    SYNSEM.LKEYS.KEYREL.PRED "_thing_n_rel" ].
```

```
who := wh-pronoun-noun-lex &
  [ STEM < "who" >,
    SYNSEM.LKEYS.KEYREL.PRED "_person_n_rel" ].
```

Solution for English

- SLASH (\leq GAP) to encode the absence of an element
 - LDD Top: Head-filler rule
 - LDD Bottom: Comp & subj extraction rules (unary phrase structure rules)
 - LDD Middle: Lexical threading/amalgamation of SLASH; constraints on rules other than head-argument
- QUE to encode the wh property
 - Controls eligibility to be filler in wh question rule
- Initial symbol should require SLASH $\langle ! ! \rangle$

Solution for pseudo-English

- Psuedo-English mimics a wh-in-situ language; the wh words stay put
- Still need to be able to tell that they're there
- Don't need SLASH, will use QUE to detect presence of the wh words
 - => Still need the lexical threading stuff working
 - => Same entries for wh pronouns
- Special rule at the top of the tree to license wh interrogatives
- Initial symbol requires QUE <! !>, to force the wh interrogative rule to apply

wh-int-cl

wh-int-cl := clause & head-compositional & head-only &
[SYNSEM [LOCAL.CAT [VAL #val,
 MC bool],
 NON-LOCAL non-local-none],
C-CONT [RELS <! !>,
 HCONS <! !>,
 HOOK.INDEX.SF ques],
HEAD-DTR.SYNSEM [LOCAL.CAT [HEAD verb & [FORM finite],
 VAL #val &
 [SUBJ < >,
 COMPS < >]],
 NON-LOCAL [SLASH <! !>,
 REL <! !>,
 QUE <! ref-ind !>]]].

Ancillary changes

- Constrained the value of NON-LOCAL on the root symbol in roots.tdl to non-local-none.
- Added norm-zero-arg as supertype for adverb-lex.
- Modified bare-np-phrase to inherit from head-valence-phrase as an additional supertype (this copies up the NON-LOCAL value).
- Modified bare-np-phrase to constrain the element of the daughter's non-empty SPR list to be unexpressed

Ancillary changes

- Modified subj-raise-aux to constrain the element of the non-empty SUBJ list of its complement to be unexpressed
- Made the following modifications to the analysis of coordination (as type addenda):

topormid-coord-phrase :+ [SYNSEM.NON-LOCAL #nl,
LCOORD-DTR.SYNSEM.NON-LOCAL #nl,
RCOORD-DTR.SYNSEM.NON-LOCAL #nl].

bottom-coord-phrase :+ [SYNSEM.NON-LOCAL #nl,
NONCONJ-DTR.SYNSEM.NON-LOCAL #nl].

Wh-questions further thoughts

- English does different things for matrix v. embedded, subject v. non-subject questions:
 - Who sleeps?
 - Who did they think sleeps?
 - What does the dog chase?
 - I ask who sleeps.
 - I ask who the dog chases.

Wh-questions further thoughts

- I think our semantic representations aren't quite capturing all the info yet:
 - Who did you ask if Sandy saw?
 - => Two clauses with [SF ques], but which one does 'who' belong to as a question?

Lab 7 overview

- Wh-questions:
 - How do they work in your language? How do we need to modify the solutions posted to work with your grammar/the facts of your language?
- Negation: Working properly? If not, post to GoPost & we'll fix things
 - What is the pattern you are trying to model?
 - What tdl do you have so far?
 - What is going wrong?

Lab 7 overview: Corpus sentence

- Create a profile from your test corpus skeleton, and run a baseline.
- Use Browse | Results to see if anything is parsing.
- Look for some plausible candidate sentences. These should be relatively short and ideally have minimal additional grammatical phenomena beyond what we have already covered.
- Examine the lexical items required for your target sentence(s). Add any that should belong to lexical types you have already created.
- Try parsing the test corpus again (or just your target sentence from it).

Lab 7 overview: Corpus sentence

- If your target sentence parses, check the MRS to see if it is reasonable.
- If your target sentence doesn't parse, check to see whether you still have lexical coverage errors. Fixing these may require adapting existing lexical rules, adding lexical rules, and/or adding lexical types. Post to GoPost for assistance.

Lab 7 overview: Corpus sentence

- If your target sentence doesn't parse but your grammar does find analyses for each lexical item, then examine the parse chart to identify the smallest expected constituent that the grammar is not finding, and debug from there. Do you have the phrase structure rule that should be creating the constituent? If so, try interactive unification to see why the expected daughters aren't forming a phrase with that rule. Do you need to add a phrase structure rule? Again, post to GoPost for assistance.
- Iterate until either the sentence parses or you at least have a clear understanding of what you would need to add to get it parsing.
- Run your full test suite after any changes you make to your grammar to make sure you aren't breaking previous coverage/introducing spurious ambiguity.

Overview

- Target representation
- The problem
- Solution for English
- Solution for pseudo-English
- Lab 7 overview