Grammar Engineering: Implementing Case and dealing with utf-8 Input

Miriam Butt
(University of Konstanz)
and
Martin Forst (NetBase Solutions)

Colombo 2014

Using Different Scripts

- XLE can be used for with all kinds of scripts.
- Current non-Latin scripts:
  - Arabic
  - Urdu
  - Georgian
  - Japanese
  - Chinese
- Limitations:
  - the TCL display of the c-structures and f-structures is not always optimal
  - emacs behavior is not always as it should be

Using Different Scripts

- The XLE documentation provides help on setting up the system for different scripts (see exercise).
- You need to make sure XLE knows the character encoding of the grammar.
- Add the following to the Config(uration) part of your grammar:
  - CHARACTERENCODING utf-8.

Using Different Scripts

- Create a testsuite containing sentences you want to parse.
- This can be done in emacs, but experience has shown that using a different editor is better.
- On Macs: Textedit.
- Generally, use an editor that you know does well with utf-8.

Parse-Testfile

- You can access your testsuite via the parse-testfile command.

```
parse-testfile filename.lfg 1
parse-testfile filename.lfg 1 4
parse-testfile filename.lfg
```

- The first version parses exactly the first sentence from the file `filename.lfg`
- The second version parses the sentences 1 through 4.
- The third version parses all the sentences in the file.

Parsing Different Scripts

Demo: Urdu Grammar

Case Marking

- The Urdu grammar treats case markers as independent lexical items.
- The implementation uses the idea of **Constructive Case** (cf. Nordlinger 1998).
- That is, the case markers determine the grammatical relations/functions.
- Implementationally, this is achieved via **inside-out functional uncertainty (IO-FU)** (e.g., Butt and King 2004, Butt, King and Varghese 2004).

Case Marking via IO-FU

- Each case marker specifies which grammatical function it marks and what (if any) special semantics are associated with it.
- Special marks can be introduced to make OT marks interact with parsing process

```
nE K *   @VOLITION
         @(CASE erg)
         (SUBJ ^)
```

- The (SUBJ ^) says that the f-str the case marker contained in must be a subject.

Case Marking via IO-FU

- The verb, as usual, specifies its subcategorization frame.

```
hit V *   (^PRED)= 'hit<(^SUBJ),(^OBJ)>'
```

- At the clausal level, the GF template on the NP in the S rule allows for all kinds of possibilities.

```
S → NP*:@GF, VC.
GF = { (^SUBJ) = ! |(^OBJ) = ! |
       (^OBL) = ! |(^OBJ2) = ! }.
```

Case Marking via IO-FU

- Which of the grammatical relations are picked thus depends on the interaction of information from the verb's lexical entry, they case marker's lexical entry and the c-structure rule.

Case Marking via IO-FU

Demo: Urdu Grammar

Case Marking

- IO-FU is conceptually very complex and difficult to debug.
- So we will use a less elegant, but simpler way.
  - Each case marker has an independent lexical entry.
  - This can also happen if it is morphological – the tag resulting from the morphological analysis can carry the same information.
  - But the Grammatical Functions (GFs) specify what case markers they are compatible with (and which not).
- This means augmenting the GF Template (see exercise 7/grammar 7).

References

Butt, Miriam and King, Tracy Holloway. 2004. Case Systems: Beyond Structural Distinctions. In E. Brandner and H. Zinsmeister (eds.) *New Perspectives on Case Theory*. Stanford, CA: CSLI Publications, 53-87.

Butt, Miriam, Tracy H. King and Anila Varghese. 2004. Computational Treatment of Differential Case Marking in Malayalam. In Proceedings of the International Conference on Natural Language Processing (ICON) 2004, Hyderabad.