Urdu and the Modular Architecture of ParGram

Tina Bögel, Miriam Butt, Annette Hautli, Sebastian Sulger

Universität Konstanz

CLT 2009, Lahore

1 Introduction

2 Overall Architecture

3 Morphology

4 Syntax

5 Prosody

6 Semantics

7 Conclusion

Introduction

- ParGram: NLP project based on Lexical Functional Grammar (LFG)
  - building large-scale, robust grammars
  - larger grammars: English, French, German, Japanese, Norwegian, Turkish
  - smaller grammars: Arabic, Chinese, Georgian, Malagasy, Urdu, Welsh
- LFG parsing and generation using a modular type of architecture
- this talk: description of the modules used for the grammar; short demos of two of the modules

Introduction

- modules of the grammar: tokenizer, morphological analyzer, syntactic rules, prosodic projection, semantics interface
- modules are connected using development platform XLE
- ParGram architecture design allows for robust, large-scale parsing and generation and satisfactory treatment of language-specific phenomena

Overall Architecture

- tokenizer and morphological analyzer: finite-state machines using the Xerox Finite-State Calculus (xfst)
- morphological analyzer feeds into syntactic rules component
  - morphological tagging interacts with syntactic rules
  - syntactic analyses are informed by theoretical work within LFG
  - c(onstituent)-structure and f(unctional)-structure are produced by the XLE platform
- phonological rules, built in the syntactic module, rephrase prosody (result: p-structure)
- syntactic structures provide basis for semantic analysis

Overall Architecture

tokenizer & morphology (FST)

↓

syntax (f- and c-structure) → prosody (p-structure)

↓

semantics (XFR ordered rewriting)

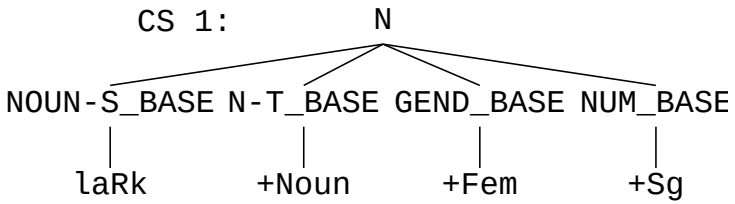
Overall modular architecture of ParGram Urdu grammar

Morphology

- implemented using a finite-state machine
- functions as "black box"
  - usable tag output for XLE - but could be replaced by other morphological resources
  - morphology is a stand-alone resource - may be used for other applications
- connected up to the syntax using a morphology-syntax interface
  - morphological information can easily be extracted from the finite-state machine
  - system allows broad vocabulary coverage
  - system allows description of language-specific morphological phenomena like reduplication, future formation, etc.

Morphology

- sample output of morphology lexicon: MORPHOLOGY
  laRk+Noun+Fem+Sg
- tags are used as input for syntax component: INTERFACE
  +Fem GEND xle @(GEND fem)
  +Sg NUM xle @(NUM sg)
- features are displayed in c- and f-structure: SYNTAX



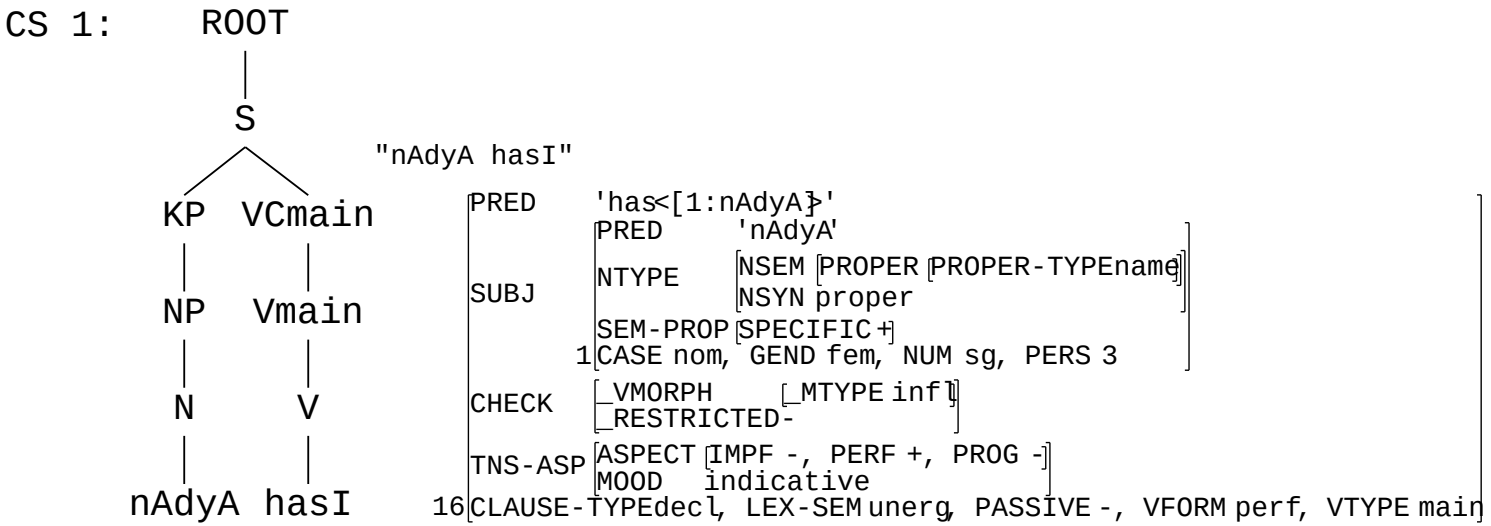
"laRkI"

$$\begin{bmatrix} \text{PRED} & \text{'laRk'} \\ \text{NTYPE} & \begin{bmatrix} \text{NSEM} & [\text{COMMON count}] \\ \text{NSYN} & \text{common} \end{bmatrix} \\ 1 & \text{GEND fem, NUM sg, PERS 3} \end{bmatrix}$$

Syntax

- syntax component is at the core of Urdu grammar
- theoretical background: LFG
- well-studied ($\sim$ 30 years) framework with computational usability
- c- and f-structures used for syntactic representation
  - c-structure: basic constituent structure ("tree") and linear precedence ($\sim$ what parts belong together)
  - f-structure: encodes syntactic functions and properties

Syntax



- size: 40 phrase-structure rules, annotated for syntactic function
- coverage: basic clauses with free word order, verbal complex, tense and aspect, causative verbs, complex predicates
- **Demo:** Complex Predicates

Prosody

- LFG architecture allows for additional projections
- prosody implemented as additional layer on top of syntax
- prosodic information important for correct understanding/disambiguation of a sentence
- experimental p(rosodic)-structure in order to model complex phonological properties of clitics, especially Urdu Ezafe

Prosody - Urdu Ezafe

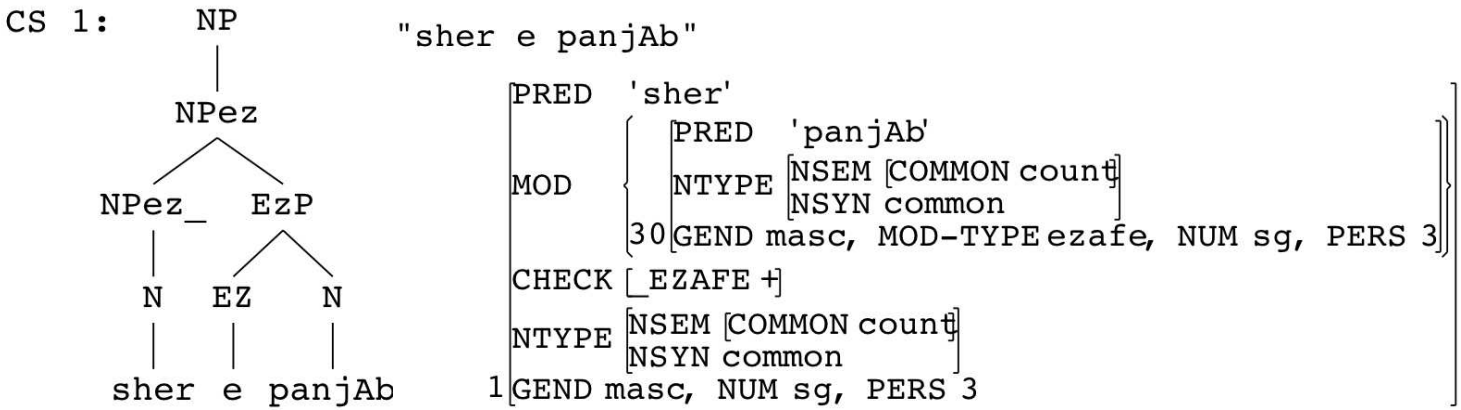
- Urdu Ezafe: loan construction from Persian
- calls for modifier (adjective or noun) to the right of head noun: not in line with usual Urdu head-final pattern
- example:

a. sher=e panjAb
lion=Ez Punjab
'A/The lion of Punjab'

b. sadA=e buland
voice=Ez high
'high voice'

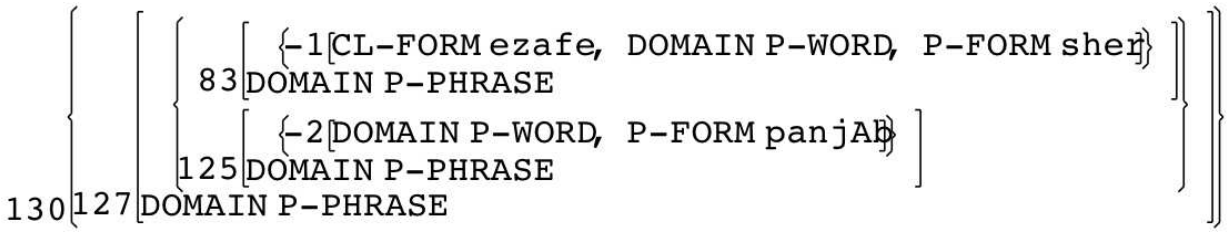
Prosody - Urdu Ezafe

- modifier is licenced by Ezafe
- Ezafe is head of Ezafe phrase constituent
- complement of Ezafe modifies head noun



Prosody - Urdu Ezafe

- Ezafe is clitic: it attaches to the end of a constituent, which is not possible for inflectional morphology
  - example shows clitic status of Ezafe:
    [maal o daulat]=e dunyaa
    material and wealth=Ez world
    'the material and wealth of the world'
- within prosody, the clitic Ezafe is integrated in the prosodic phrase to its left - not modeled at level of syntax
- additional level: p-structure
- Ezafe is coded as part of the head noun within p-structure:



Semantics

- f-structures within XLE are coded in Prolog
- for semantics, we take Prolog code and apply ordered rewrite rules (XFR) on it
  - reasonable approach, as f-structures are equivalent to quasi-logical forms
- input f-structure is consumed step by step by the rewrite rules
- XLE produces output semantic form
- world knowledge may also be included (English ParGram grammar uses WordNet as knowledge base)
- **Demo:** Semantics module

Conclusion

- Urdu ParGram project devoted to developing a large-scale, broad-coverage LFG parsing and generation grammar using XLE
  - pipeline architecture: single components may be used in other contexts
  - informed by well-studied linguistic insights from LFG theory
- currently experimenting with additional annotation using p-structure (prosody) and XFR rewriting (semantics)
- LFG/XLE methodology: powerful, effective, proven and tested