Linguistics unfolded

Ralph Debusmann

Programming Systems Lab
Saarland University, Saarbrücken
Grammar formalisms

- should be able to modularize the different dimensions of linguistic description, i.e. to *unfold* a linguistic description, ...
- ... and at the same time, be able to treat the linguistic description in an integrative way.
- existing grammar formalisms are unable to do both
- we will introduce Extensible Dependency Grammar (XDG), which *can* do both
Why modularize?

- consider *Jeder Mann liebt eine Frau*.
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects
Topicalization

- consider *Eine Frau liebt jeder Mann*.
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Mittelfeld, *eine Frau* is in the Vorfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects
Topicalization

- consider *Eine Frau liebt jeder Mann*.
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Mittelfeld, *eine Frau* is in the Vorfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects
Topicalization

- consider *Eine Frau liebt jeder Mann.*
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Mittelfeld, *eine Frau* is in the Vorfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects = Vorfeld
  - surface objects = Mittelfeld
  - surface subjects = deep subjects
  - surface objects = deep objects
Passive

- consider *Von jedem Mann wird eine Frau geliebt.*
- *von jedem Mann* is the surface object, *eine Frau* is the surface subject
- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jedem Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects
Passive

- consider *Von jedem Mann wird eine Frau geliebt.*

  - *von jedem Mann* is the surface object, *eine Frau* is the surface subject
  
  - *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld

  - *jedem Mann* is the deep subject, *eine Frau* is the deep object

- generalizations:
  
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects
Passive

- consider *Von jedem Mann wird eine Frau geliebt.*
- *von jedem Mann* is the surface object, *eine Frau* is the surface subject
- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jedem Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects = Vorfeld
  - surface objects = Mittelfeld
  - surface subjects = deep subjects
  - surface objects = deep objects
Modularize!

- simple generalizations will backfire when confronted with more complicated constructions
- so don’t make them
- instead, properly modularize the different dimensions of linguistic description
Why integrate?

• consider *Peter sieht die Frau mit dem Teleskop*
• PP can either attach to *sieht* or to *Frau*
• additional information needed for disambiguation
• information may come from any source, i.e. also from semantics (e.g. a database of world knowledge)
• a non-integrated system cannot make use of information from semantics early on:

Peter sieht die Frau mit dem Teleskop
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:

```
Peter sieht die Frau mit dem Teleskop
```

- Syntactic reading 1
- Syntactic reading 2
a non-integrated system cannot make use of information from semantics early on:

Peter sieht die Frau mit dem Teleskop

Syntactic reading 1 → Semantic reading 1

Syntactic reading 2 → Semantic reading 2
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:

```
Peter sieht die Frau mit dem Teleskop
```

Syntactic reading 1 → Semantic reading 1
Syntactic reading 2 → Semantic reading 2
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:
Non-integrated

- a non-integrated system cannot make use of information from semantics early on:

Peter sieht die Frau mit dem Teleskop

Syntactic reading 2 → Semantic reading 2

semantic information
in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:

Peter sieht die Frau mit dem Teleskop
in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:
in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:
Integrate!

- in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:

Peter sieht die Frau mit dem Teleskop

Syntactic reading 2
Semantic reading 2

semantic information
The situation

grammar formalisms
The situation

modularity → grammar formalisms
The situation

modularity → grammar formalisms ← integration

Linguistics unfolded – p.10
The situation

modularity → grammar formalisms ← integration

Linguistics unfolded – p.10
The situation

modularity → LFG ▸ HPSG ▸ grammar formalisms → integration
The situation

modularity → LFG XDG → HPSG XDG

grammar formalisms

integration
The situation

modularity → grammar formalisms → integration

LFG XDG  HPSG XDG
The situation

modularity → grammar formalisms → integration

LFG → HPSG

XDG
XDG dimensions

- XDG can be instantiated with *any* number of *dimensions*
- dimensions fold out the linguistic dimensions
- in the following, we show an instantiation with three dimensions: surface syntax, deep syntax, and topological fields
Example surface syntax analysis

- *von jedem Mann* is the surface object (of *geliebt*), *eine Frau* is the surface subject (of *wird*):
Example topological fields analysis

- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld of *wird*:
Example deep syntax analysis

• *jedem Mann* is the deep subject, *eine Frau* is the deep object of *geliebt*:
**XDG principles**

- an XDG grammar makes use of any number of *principles* to restrict the well-formedness conditions of analyses
- principles can be either *one-dimensional* (only restrict one dimension) or *multi-dimensional* (restrict the relation of two or more dimensions)
- principles can be either *lexicalized* or not
- the lexicon can be conveniently built up using abstractions like lexical inheritance and crossings (Candito 1996)
Tree principle (surface syntax)
Tree principle (surface syntax)
Tree principle (topological fields)
Tree principle (topological fields)

Tree
DAG principle (deep syntax)

von jedem Mann wird eine Frau geliebt
DAG principle (deep syntax)

DAG
Valency principle (surface syntax)
Valency principle (surface syntax)

in: {}
out: {subj!, vprt!}
Valency principle (surface syntax)

von jedem Mann wird eine Frau geliebt

in: {} out: {subj!, vprt!}

in: {vprt?} out: {pobj?}
Valency principle (topological fields)
Valency principle (topological fields)

in: {}

out: {vf? mf* rbf?}
Valency principle (topological fields)
Valency principle (deep syntax)
Valency principle (deep syntax)

in: {}
out: {subd!}
Valency principle (deep syntax)

Linguistics unfolded – p.21
Climbing principle
Climbing principle
Climbing principle

The topology is a flattening of the surface syntax.
Syntactic linking

von jedem Mann wird eine Frau geliebt
von jedem Mann wird eine Frau geliebt

synlink:
{ subj: { (subd,objd) } }
Syntactic linking

The subject of the passive auxiliary becomes the deep object of a subordinated verb.

von jedem Mann wird eine Frau geliebt

The subject of the passive auxiliary becomes the deep object of a subordinated verb.

\{ subj: \{ (subd, objd) \} \}
Representing semantics

- *Von jedem Mann wird eine Frau geliebt.*
- semantics (weak reading):
  \[ \forall x. \text{man}(x) \rightarrow \exists y. \text{woman}(y) \land \text{love}(x, y) \]
- two dimensions: predicate-argument structure and scope
- in XDG, we can use the same ideas also to represent semantics
∀ x man(x) ⇒ ∃ y woman(y) ∧ love(x, y)
\[ \forall x \text{ man}(x) \Rightarrow \exists y \text{ woman}(y) \land \text{ love}(x,y) \]

\[ \forall x \text{ man}(x) \quad \exists y \text{ woman}(y) \quad \text{ love}(x,y) \]
Predicate-argument dimension

\[ \forall x \text{ man}(x) \Rightarrow \exists y \text{ woman}(y) \land \text{ love}(x, y) \]

\[ \forall x \text{ man}(x) \quad \exists y \text{ woman}(y) \land \text{ love}(x, y) \]
\[ \forall x \ \text{man}(x) \Rightarrow \exists y \ \text{woman}(y) \land \text{love}(x,y) \]
\( \forall x \, \text{man}(x) \Rightarrow \exists y \, \text{woman}(y) \land \text{love}(x,y) \)
∀ x man(x) ⇒ ∃ y woman(y) ∧ love(x,y)
\[ \forall x \ \text{man}(x) \Rightarrow \exists y \ \text{woman}(y) \land \text{love}(x,y) \]
Scope dimension

∀x man(x) ⇒ ∃y woman(y) ∧ love(x,y)
Scope dimension

\[ \forall x \ \text{man}(x) \Rightarrow \exists y \ \text{woman}(y) \land \text{love}(x,y) \]

\[ \forall x \bullet \Rightarrow \bullet \ \text{man}(x) \exists y \bullet \land \bullet \ \text{woman}(y) \ \text{love}(x,y) \]
Scope dimension

∀ x man(x) ⇒ ∃ y woman(y) ∧ love(x,y)
\[ \forall x \; \text{man}(x) \Rightarrow \exists y \; \text{woman}(y) \land \text{love}(x,y) \]
Scope dimension

\[ \forall x \text{ man}(x) \Rightarrow \exists y \text{ woman}(y) \land \text{ love}(x,y) \]
### Scope dimension

\[ \forall x \text{ man}(x) \implies \exists y \text{ woman}(y) \land \text{ love}(x,y) \]
\[ \forall x \; \text{man}(x) \implies \exists y \; \text{woman}(y) \land \text{love}(x,y) \]
∀ x\, man(x) \implies \exists y\, woman(y) \land love(x,y)
DAG principle (predicate-argument)
DAG principle (predicate-argument)

DAG
Tree principle (scope)

von jedem Mann wird eine Frau geliebt
Tree principle (scope)

von jedem Mann wird eine Frau geliebt
Valency principle (predicate-argument)

Example: von jedem Mann wird eine Frau geliebt

Diagram:

```
von  jedem  Mann  wird  eine  Frau  geliebt
```
Valency principle (predicate-argument)

in: \{\text{arg}, \text{arg1}, \text{arg2}\}
out: \{\}

\text{von jedem Mann wird eine Frau geliebt}
Valency principle (predicate-argument)
Valency principle (predicate-argument)

von jedem Mann wird eine Frau geliebt
Valency principle (scope)

von jedem Mann wird eine Frau geliebt
**Valency principle (scope)**

```
von jedem Mann wird eine Frau geliebt
```

in: \{r? s?\}
out: \{r! s!\}
Valency principle (scope)
Valency principle (scope)

Linguistics unfolded – p.30
Semantic linking

semlink:

{ arg1: { (subj, detd) }
arg2: { (objd, detd) } }

Linguistics unfolded – p.31
The first argument is realized by the determiner under the deep subject.
The second argument is realized by the determiner under the deep object.
Contra-dominance

von jedem Mann wird eine Frau geliebt.
Contra-dominance

Linguistics unfolded – p.32
Contra-dominance

The first argument takes scope over the verb.
The second argument takes scope over the verb.
Modularity and integration

- XDG is modular *and* integrated
- modularity by unfolding the dimensions of linguistic description, one-dimensional principles
- integration by multi-dimensional principles
- integration: inferences can flow from any dimension to any other
Inferences from syntax to semantics
Inferences from syntax to semantics
Inferences from syntax to semantics
Inferences from semantics to syntax
Inferences from semantics to syntax
Inferences from semantics to syntax

Peter sieht die Frau mit dem Teleskop.
Conclusions

- it is important to distinguish the different dimensions of linguistic description,
- ... and at the same time to integrate them.
- existing grammar formalisms cannot do both
- XDG can
Related work


- interface to Information structure (Kruijff/Duchier EACL 2003)


Future work

- obtain large-scale grammars
- find out how to parse them efficiently with XDG
- many many more issues to solve (e.g. coordination, incremental parsing . . . )
- but before all that: start writing the dissertation ;-)
Thank you!