

Computational Linguistics II

— Grammars, Algorithms, Statistics —

Dan Flickinger

Oslo, Stanford, and Saarland Universities

`danf@csli.stanford.edu`

Tore Langholm

Universitetet i Oslo

`tore.langholm@iln.uio.no`

Stephan Oepen

Oslo and Stanford Universities

`oe@csli.stanford.edu`

Modification: Dimensions of variation

- Determine what can modify what
 - PPs modify either nominal phrases or verbal phrases
 - Adjective modify nominal phrases
 - Adverbs modify verbal phrases



Modification: Dimensions of variation

- Determine what can modify what
 - PPs modify either nominal phrases or verbal phrases
 - Adjective modify nominal phrases
 - Adverbs modify verbal phrases
- Account for order of modifiers and heads
 - PPs follow head
 - Adjectives precede head (simple ones, anyway)
 - Adverbs either precede or follow



Modification: Dimensions of variation

- Determine what can modify what
 - PPs modify either nominal phrases or verbal phrases
 - Adjective modify nominal phrases
 - Adverbs modify verbal phrases
- Account for order of modifiers and heads
 - PPs follow head
 - Adjectives precede head (simple ones, anyway)
 - Adverbs either precede or follow
- Avoid spurious ambiguity for phrases with multiple modifiers



Modification: Dimensions of variation

- Determine what can modify what
 - PPs modify either nominal phrases or verbal phrases
 - Adjective modify nominal phrases
 - Adverbs modify verbal phrases
- Account for order of modifiers and heads
 - PPs follow head
 - Adjectives precede head (simple ones, anyway)
 - Adverbs either precede or follow
- Avoid spurious ambiguity for phrases with multiple modifiers
- Distinguish modifier PPs from complement PPs
 - The dog barked **near the cat***
 - Kim gave the cat **to Sandy***



What modifies what, and who decides?

- Decide whether head or modifier does selection

If head selects,
then must allow for multiple modifier types,
and determine order of each – redundancy for e.g. PPs

If modifier selects,
then can use abstract *pos* type *modable* as supertype of *verb* and *noun*



What modifies what, and who decides?

- Decide whether head or modifier does selection
 - If head selects,
then must allow for multiple modifier types,
and determine order of each – redundancy for e.g. PPs
 - If modifier selects,
then can use abstract *pos type modable* as supertype of *verb* and *noun*
- Decide whether head is SPR-saturated
 - Do PPs attach to S and NP, or to VP and N-bar?
(and similarly for adjectives and adverbs)



Order of modifier and head

- Have two modifier rules (unavoidable)
- Constrain HEAD value for each modifier daughter
 - Requires additional abstract *pos* types: *premodifier*, *postmodifier*
 - Requires multiple inheritance for the type *adv*



Candidate grammar: VP/N-bar attachment

Positive

- Correct grammaticality results

The cat chased that fierce dog.

**The cat chased fierce that dog.*



Candidate grammar: VP/N-bar attachment

Positive

- Correct grammaticality results

The cat chased that fierce dog.

**The cat chased fierce that dog.*

Negative

- Spurious ambiguity

The fierce dog near the cat barked.

[[fierce dog] [near the cat]]

[fierce [dog [near the cat]]]



Spurious ambiguity

- Consider restricting pre-head modifiers to modify only word, not phrase
Undergenerates: *the fierce fierce dog*



Spurious ambiguity

- Consider restricting pre-head modifiers to modify only word, not phrase
Undergenerates: *the fierce fierce dog*
- Consider attachment of post-head modifiers to NP/S, not Nbar/VP



NP/S attachment for (post-head) modifiers

Positive

- Still correct grammaticality for these data sets
- No spurious ambiguity



NP/S attachment for (post-head) modifiers

Positive

- Still correct grammaticality for these data sets
- No spurious ambiguity

Potential Negatives

- Asymmetry for adverb attachment
 - The dogs left quickly.* (attaches to S)
 - The dogs quickly left.* (attaches to VP)
- Asymmetry for adjectival modifiers (eventually)
 - The dogs angry at the cats bark.* (attaches to NP)
 - The angry dogs bark.* (attaches to N)
- Difficult semantics: scope of negation
 - No dogs near the cat bark.*



Other alternatives, using VP/N-bar attachment

- Add boolean feature --PM ('Post-Modified')
 1. Modifier-head-rule says head-dtr must be [--PM -], but mother is unmarked (enabling *fierce fierce dog*)
 2. Head-modifier-rule says mother is [--PM +] so a post-modified phrase cannot be head-dtr in modifier-head rule
 3. Other rules preserve the --PM feature from head-dtr to mother



Other alternatives, using VP/N-bar attachment

- Add boolean feature --PM ('Post-Modified')
 1. Modifier-head-rule says head-dtr must be [--PM -], but mother is unmarked (enabling *fierce fierce dog*)
 2. Head-modifier-rule says mother is [--PM +] so a post-modified phrase cannot be head-dtr in modifier-head rule
 3. Other rules preserve the --PM feature from head-dtr to mother
- Use additional types to distinguish “nuclear” vs. “extended” phrases
Leads to additional complexity in the phrase type hierarchy



Prepositional phrases: Modifiers or complements?

Modifiers

- Iteration within a phrase

The dog barked near the cat on Monday near the office.

- Transparent semantics



Prepositional phrases: Modifiers or complements?

Modifiers

- Iteration within a phrase

The dog barked near the cat on Monday near the office.

- Transparent semantics

Complements

- Obligatory co-occurrence

**The dog gave the cat.*

- Idiosyncratic selection of preposition

**The dog gave the cat at that aardvark.*

- Opaque (unpredictable) semantics

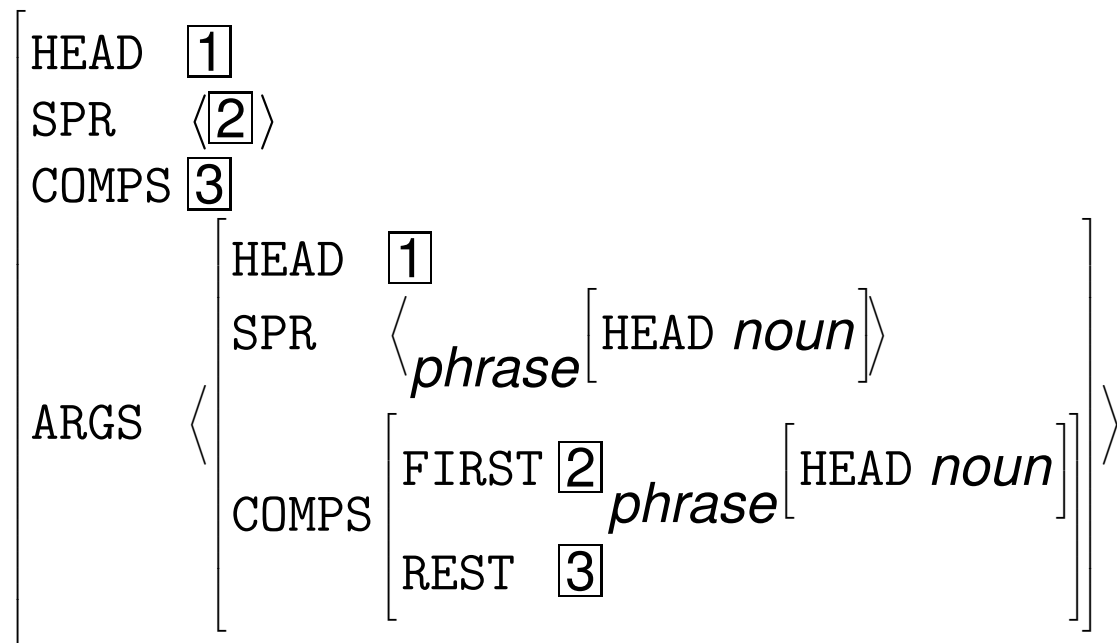
The dog belongs to the boy.



Lexical Variation: Lexical Rules

- Dative shift, passivization, et al. are systematic processes in the lexicon;
- *lexical rules* are unary grammar rules operating 'within the lexicon';
- take as input a lexical sign (*syn-struct*) and output a derived lexical sign.

Rough Approximation of Passive Lexical Rule



Orthographic Variation: 'Inflectional' Rules

```
%(letter-set (!s abcdefghijklmnopqrtuvwxyz))
```

```
noun-non-3sing_irule :=
```

```
%suffix (!s !ss) (!ss !ssses) (ss sses)
```

```
word &
```

```
[ HEAD [ AGR non-3sing ],
```

```
  ARGS < noun-lxm > ].
```

```
noun-3sing_irule :=
```

```
word &
```

```
[ ORTH #1,
```

```
  HEAD [ AGR 3sing ],
```

```
  ARGS < noun-lxm & [ ORTH #1 ] > ].
```

dog

|

dogs

bus

|

busses

pass

|

passes

