Defaults and lexical prototypes

Workshop on defaults in morphological theory
May 21, 2012

Rob Malouf
San Diego State University

Head-Driven Phrase Structure

Representations in HPSG are **typed feature structures**, a class of directed acyclic graphs

An **attribute value matrix** is a description which picks out a sets of these linguistic objects

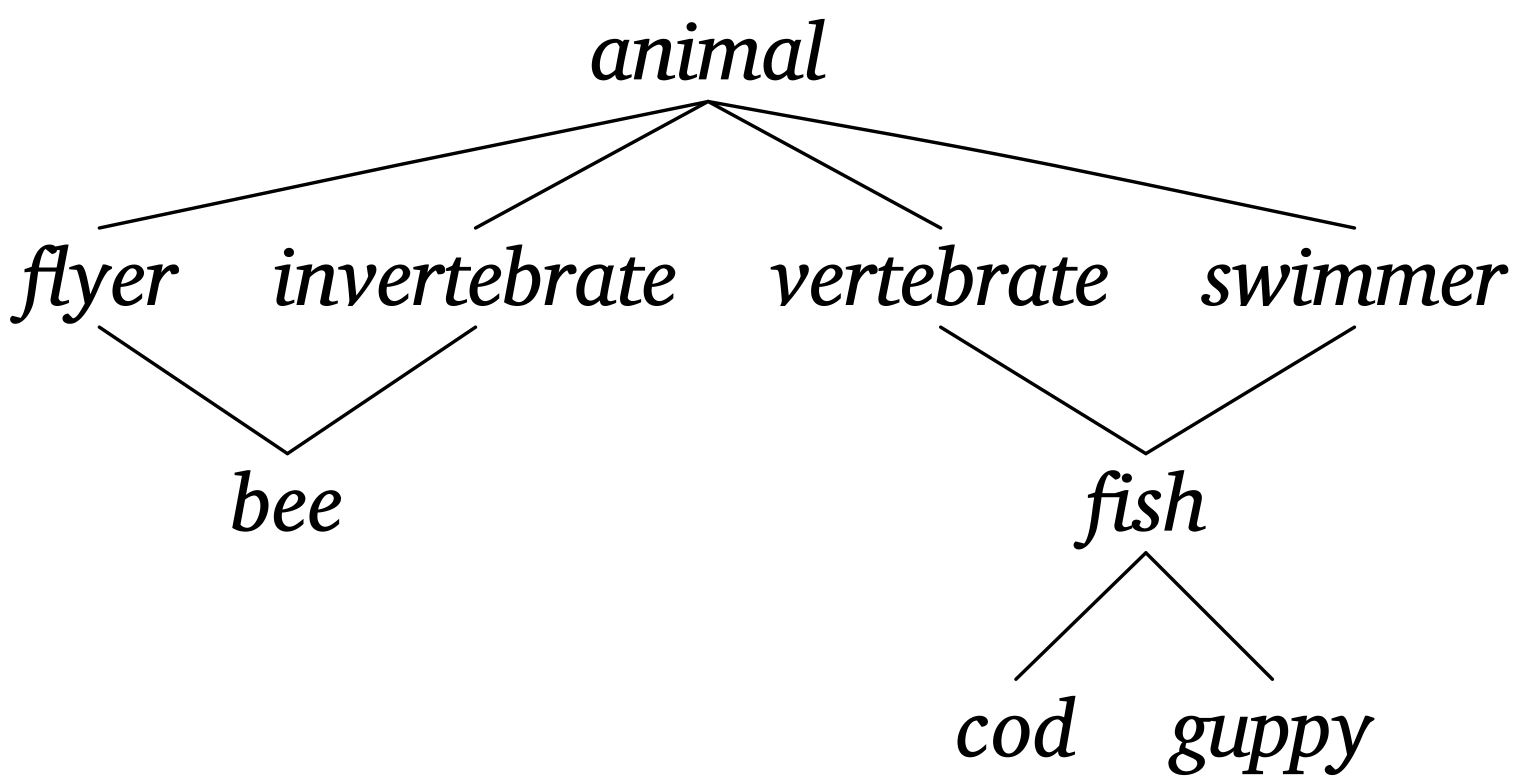
Each feature structure has a **type** associated with it

Types are organized into a **signature** which specifies appropriateness and inheritance relationships

Type hierarchies

Types are organized into an **inheritance hierarchy**, an ontology of object types

The hierarchy is a **bounded complete partial order**: every pair of types have a unique least upper bound and there is a unique most-general-type



Sort hierarchies

The inheritance hierarchy defines an ontology of linguistic objects (**sorts**):

- types and their relations ('is a' and 'has a')
- appropriate features
- appropriate values
- type inference

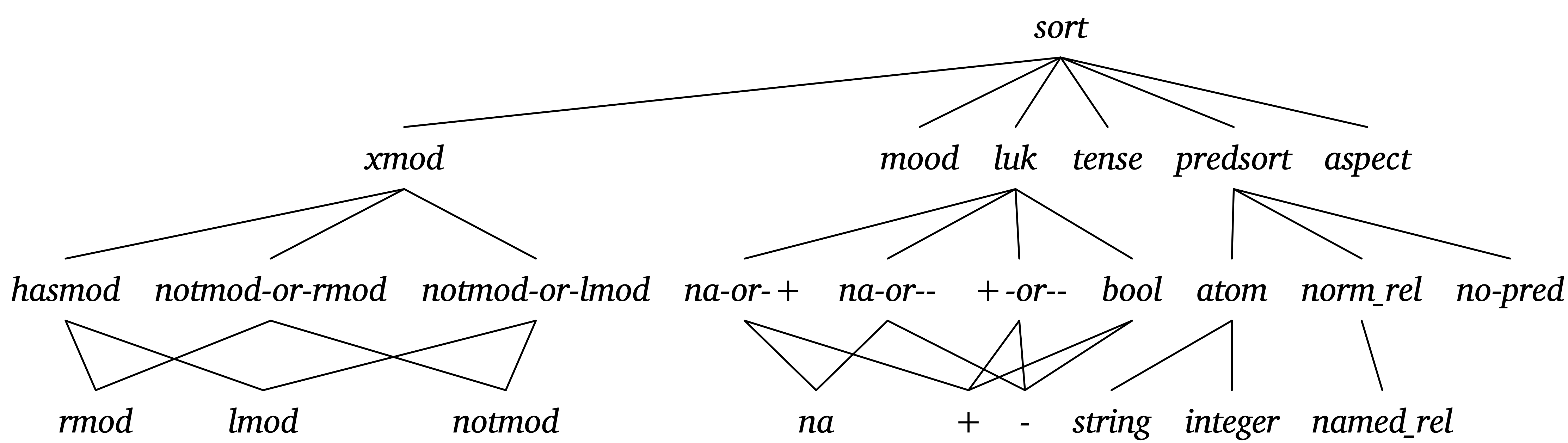
Provides a basis for precise and efficient implementation (Flickinger 2000)

This ontology is (mostly) arbitrary and (mostly) universal

This metalanguage is important but not by itself linguistically very interesting

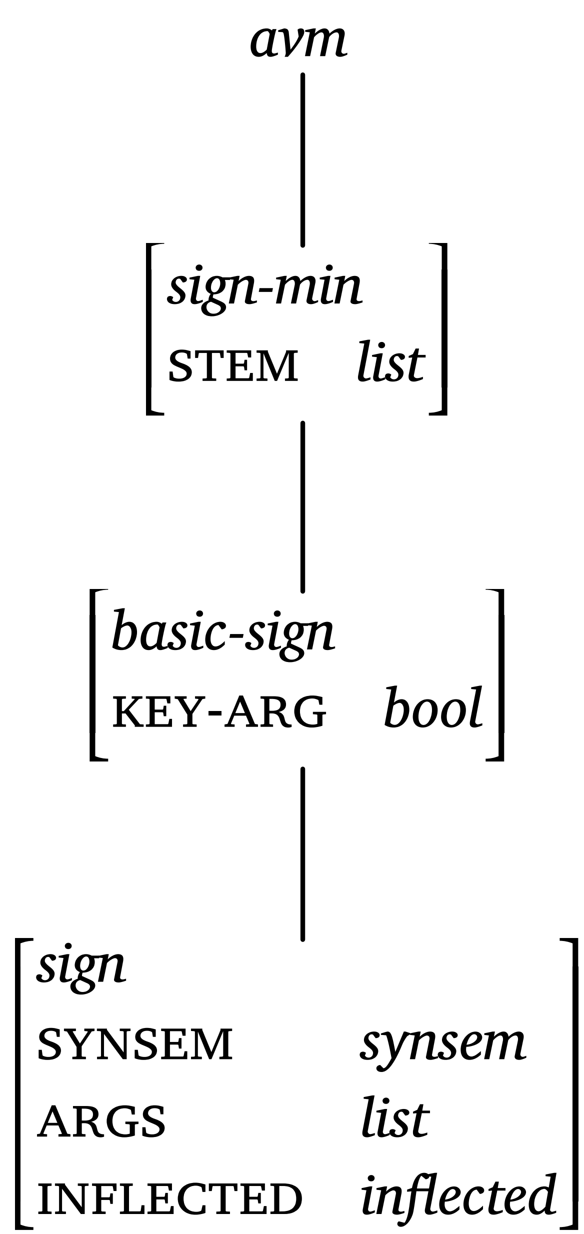
Sort hierarchies

Grammar Matrix (Bender, *et al.* 2010)



Sort hierarchies

Grammar Matrix (Bender, *et al.* 2010)



Lexical hierarchies

The type hierarchy is also used to define constraints on the lexicon and the inventory of constructions

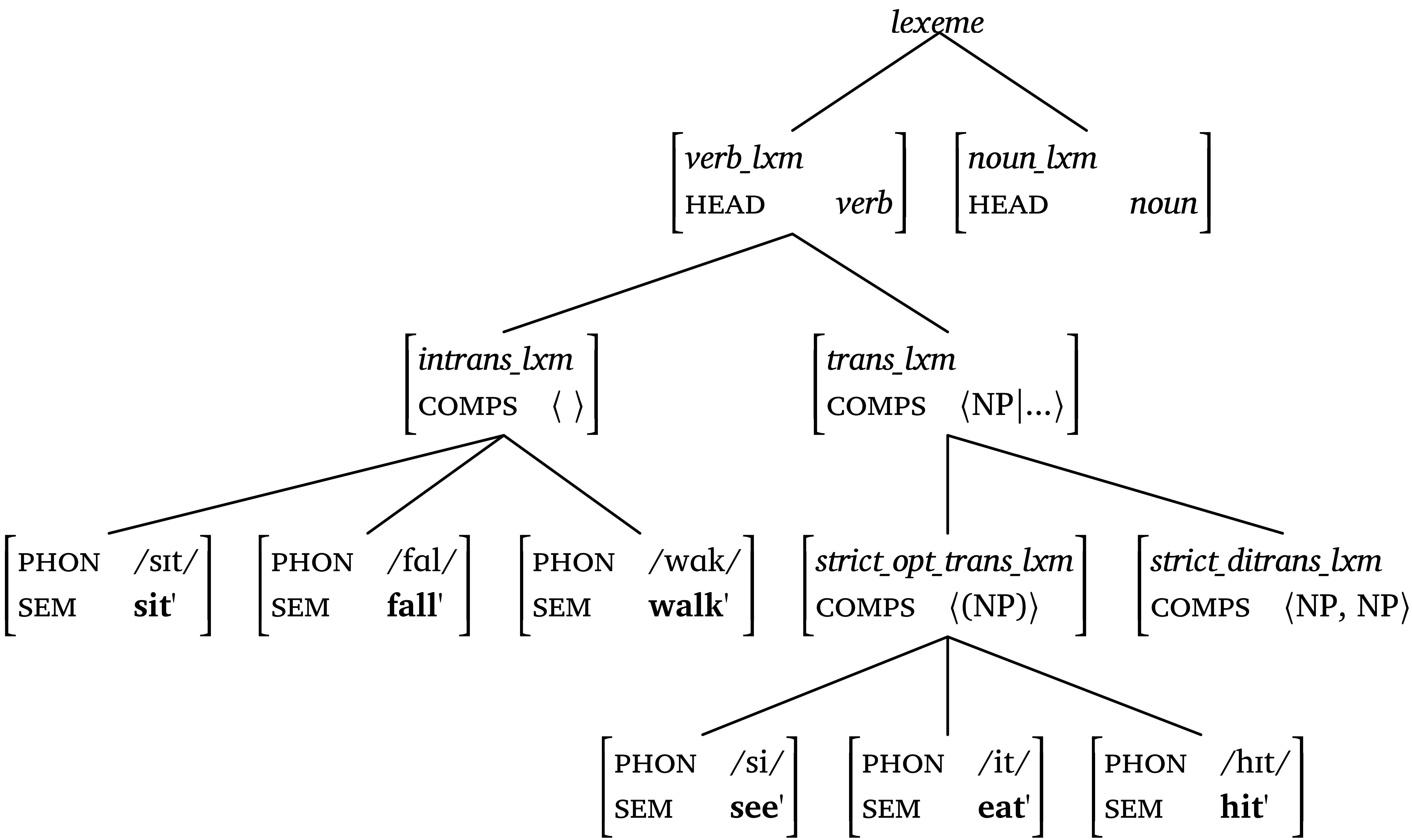
Classes of words can be the same in some ways and different in others

Patterns of **sameness** can be reified as super-types, while **differences** are instantiated on lower types in the hierarchy

Anything that is true of a type is also true of all of any more specific type

Taxonomic approach to linguistic description

Lexical hierarchies

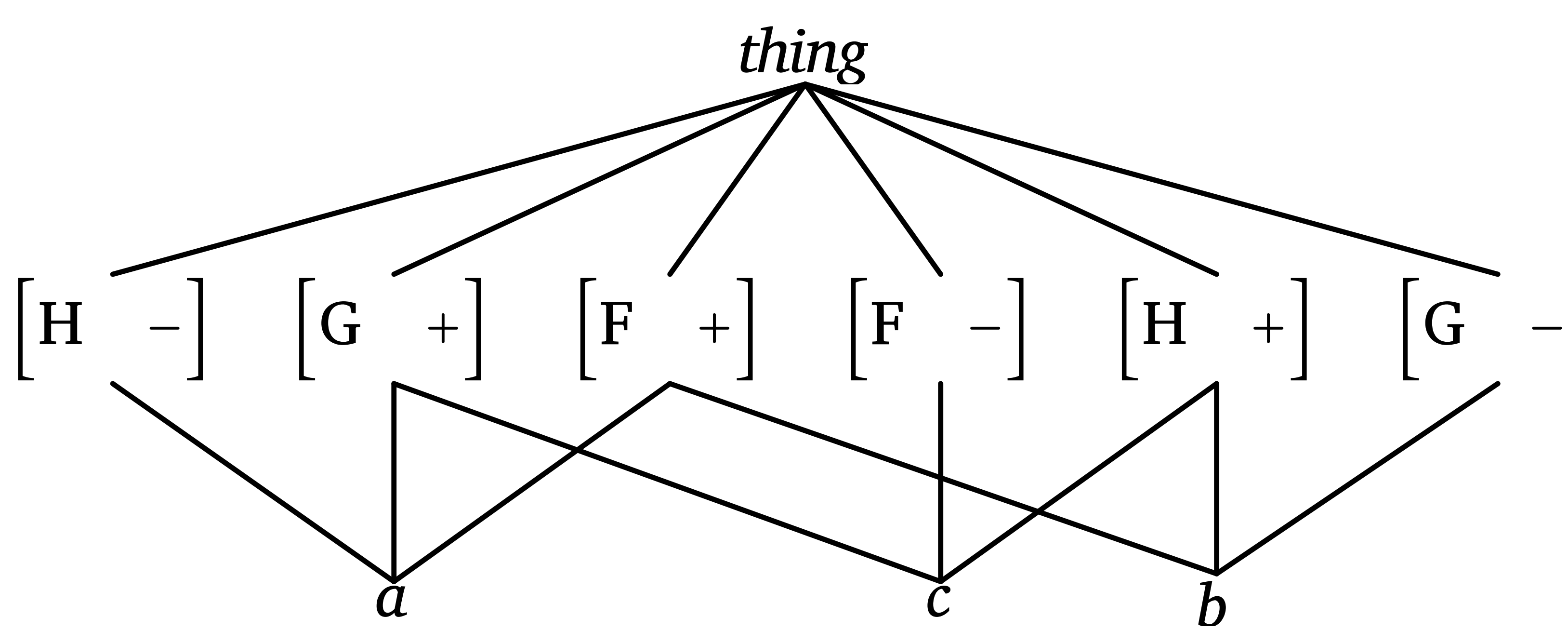


Lexical hierarchies

This style of representation associates patterns of sameness and differentness with particular types

Radial / family resemblance categories (Wittgenstein, Rosch, Lakoff, et al.) pose a problem

	F	G	H
a	+	+	–
b	+	–	+
c	–	+	+



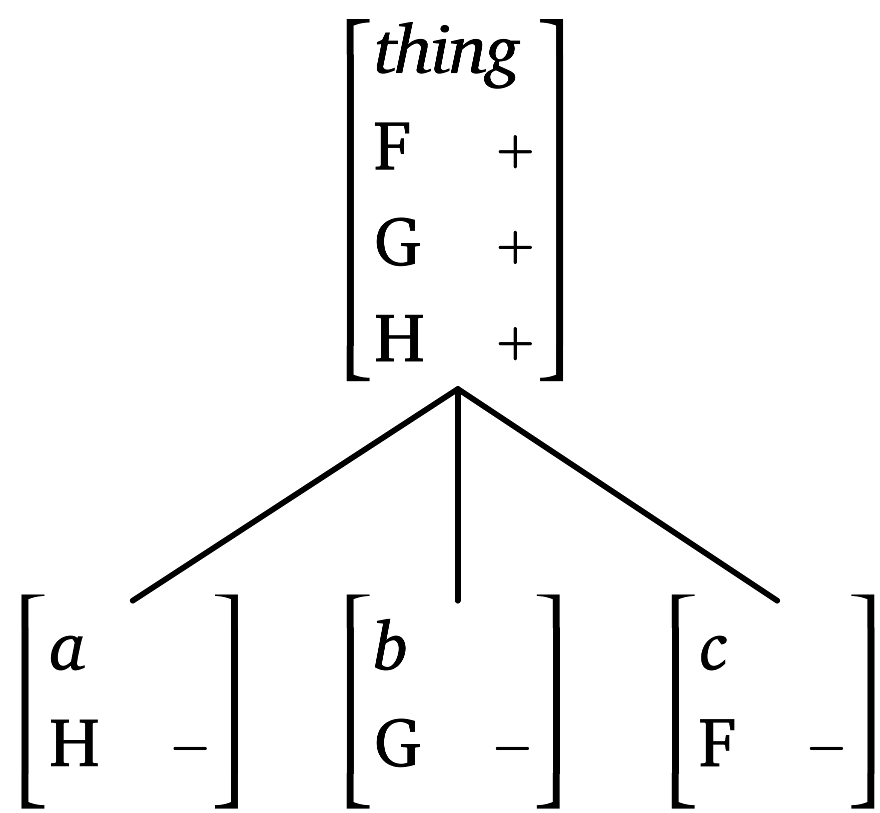
Default inheritance

Default constraints offer a solution to this problem

We can state properties of a type which usually hold, but allow more specific subtypes to override that

Anything that is true of a type is also true of all of any more specific type unless there's a conflict

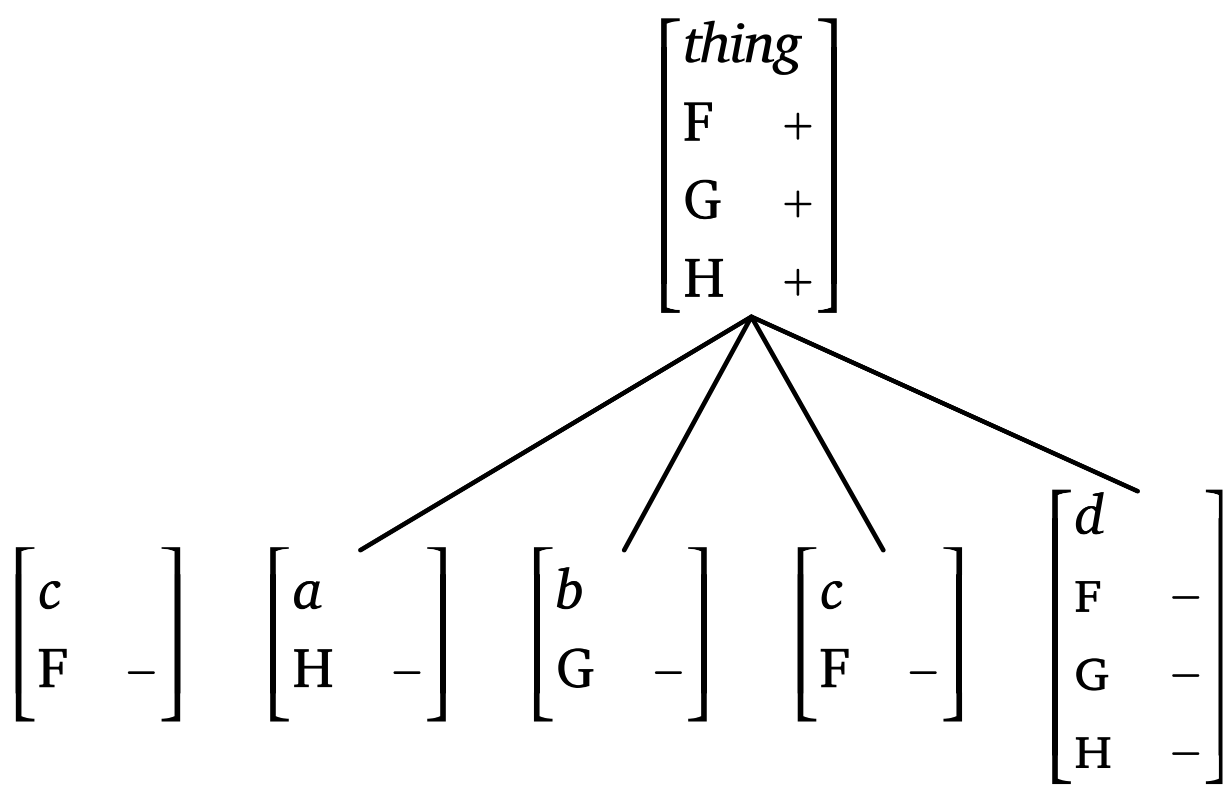
	F	G	H
a	+	+	–
b	+	–	+
c	–	+	+



Default inheritance

Defaults give us a mechanism for representing prototypes

Once we allow overriding, what does it mean to be a member of a category?



Two mechanisms for capturing similarities and differences

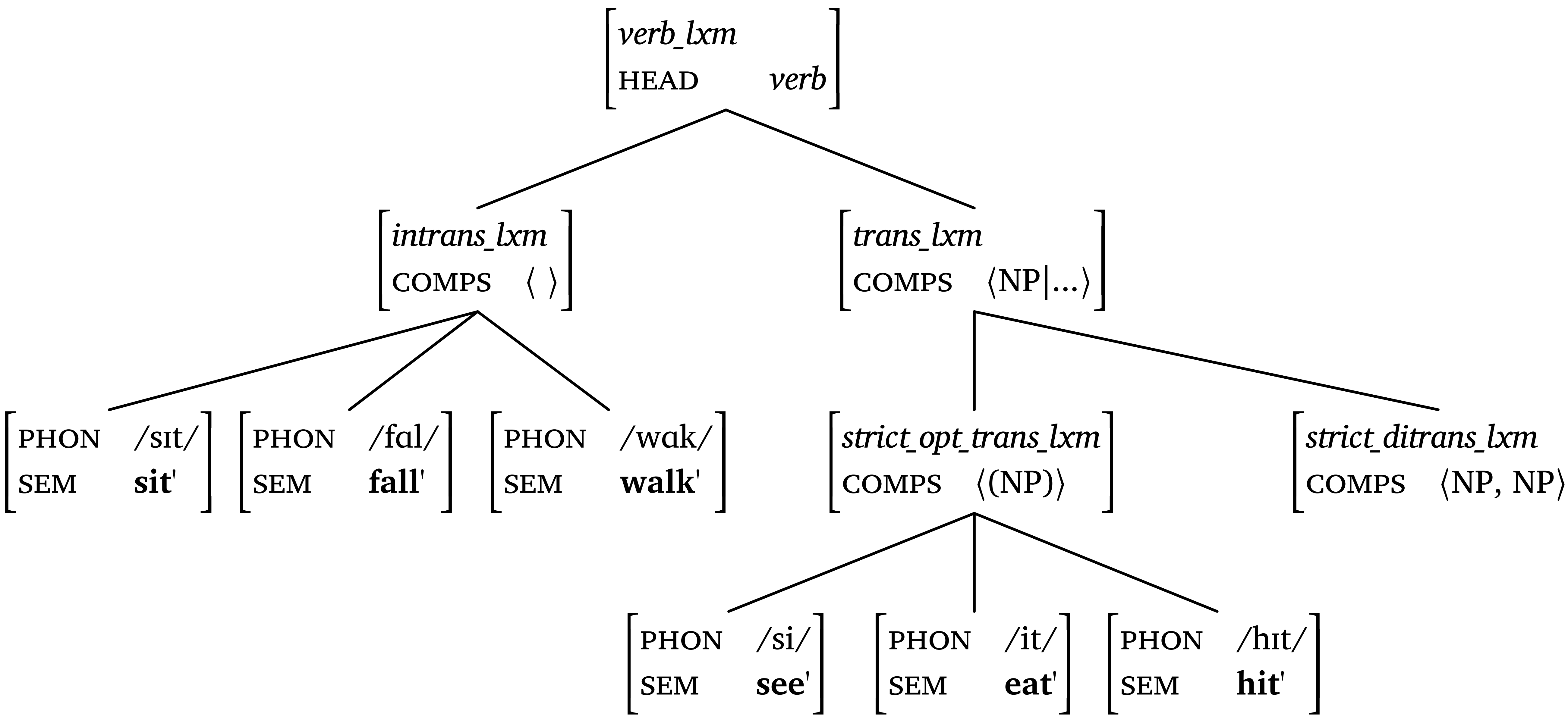
Prototypes

Inheritance hierarchies (with or without overriding) come from the same knowledge representation tradition as **object-oriented programming**

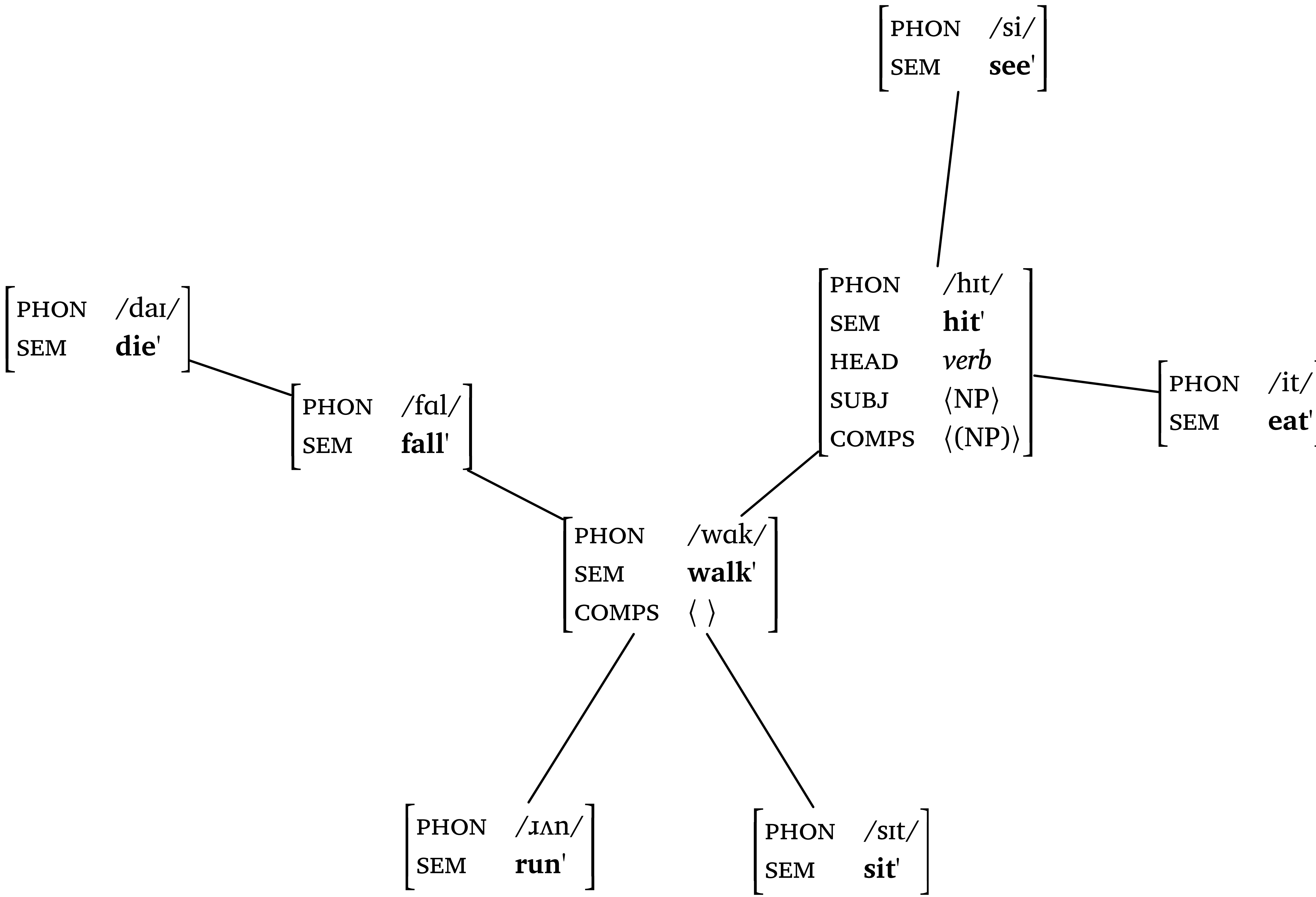
Prototype-based programming is an alternative that has been gaining interest (Borning 1986, Lieberman 1986, Ungar and Smith 1987)

- No abstract classes, only fully specified objects
- All constraints are defaults
- New objects are defined **differentially**
- Objects are related to other objects via **delegation**

Prototypes


verb_lxm
HEAD verb
intrans_lxm
COMPS ⟨ ⟩
trans_lxm
COMPS ⟨NP|...⟩
PHON /sɪt/
SEM sit'
PHON /fɑl/
SEM fall'
PHON /wɑk/
SEM walk'
strict_opt_trans_lxm
COMPS ⟨(NP)⟩
strict_ditrans_lxm
COMPS ⟨NP, NP⟩
PHON /si/
SEM see'
PHON /it/
SEM eat'
PHON /hɪt/
SEM hit'

Prototypes



PHON /si/
SEM **see**'
PHON /daɪ/
SEM **die**'
PHON /hɪt/
SEM **hit**'
HEAD *verb*
SUBJ ⟨NP⟩
COMPS ⟨(NP)⟩
PHON /fɑl/
SEM **fall**'
PHON /it/
SEM **eat**'
PHON /wɑk/
SEM **walk**'
COMPS ⟨ ⟩
PHON /ɹʌn/
SEM **run**'
PHON /sɪt/
SEM **sit**'

Prototypes

Inheritance

reflects an 'is-a' relation: a transitive verb is a kind of verb

default overriding is exceptionality

intensional classes and abstract prototypes

Delegation

reflects and 'is-like' relation: the lexical entry for *walk* is similar to the lexical entry for *hit*

default overriding is difference

extensional classes and concrete prototypes

Operationally, the two notions are more or less the same (Lascarides and Copestake 1999)

Prototypes

- Some obvious problems
  - Grammar development
    - Is is possible to construct and maintain differential networks like this?
  - Types as generalization
    - A taxonomic approach to the lexicon encodes the fact that there are many more verbs than there are kinds of verbs
  - Multiple inheritance
    - Words and constructions can be related to each other along multiple orthogonal dimensions

English Resource Grammar

Large scale grammar of English (Flickinger & Copstake 2000, Flickinger *et al.* 2000)

Implemented in the LKB

Organized around a large, detailed type hierarchy

Aimed at broad-coverage deep parsing and generation

Version 1111, downloaded from http://lingo.stanford.edu/build/test/erg.tgz

English Resource Grammar

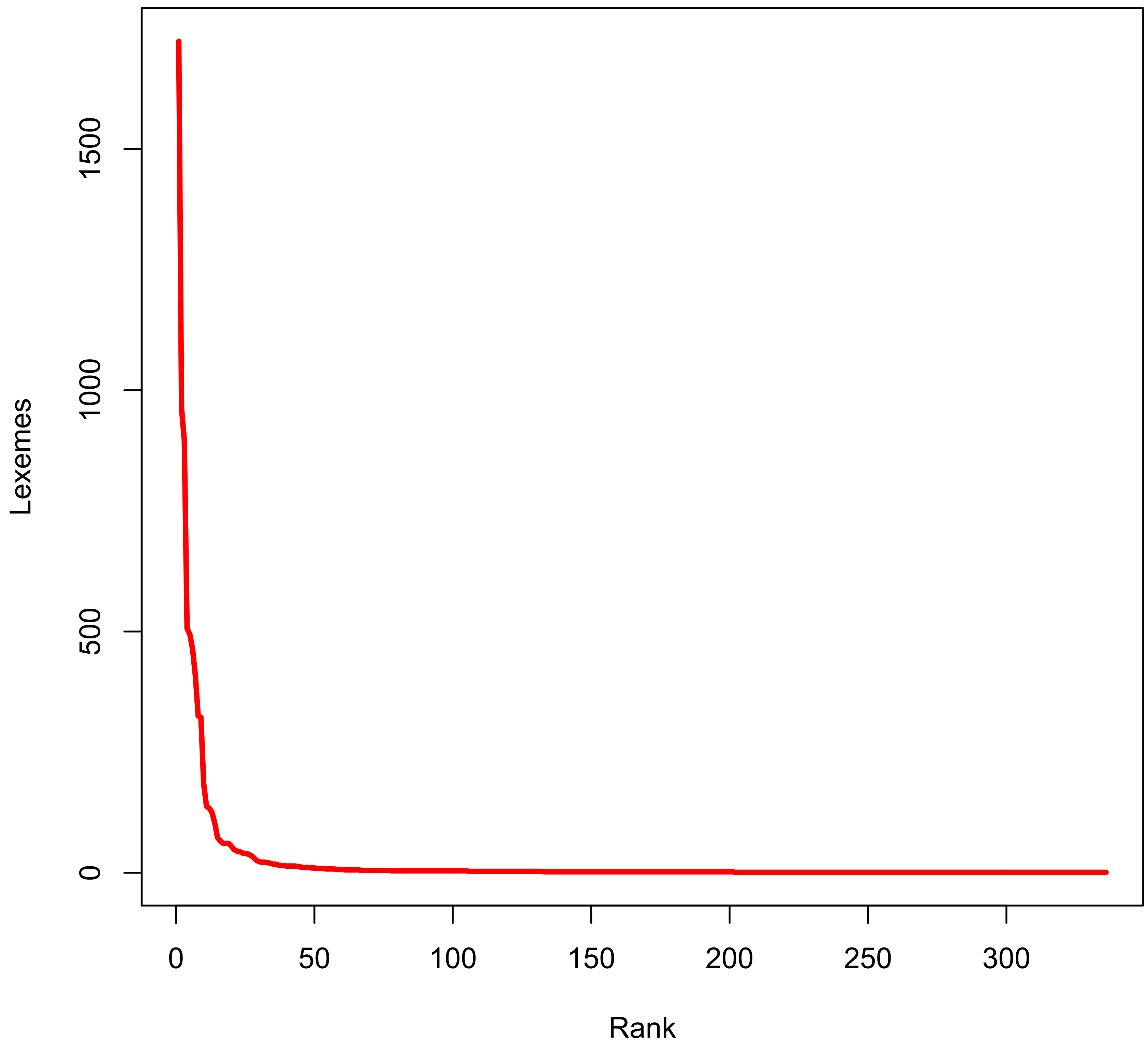
The included lexicon (lexicon.tdl) lists 8,472 verb lexemes representing 336 types

Ten most frequent verb types account for 6,283 lexemes, and 135 verb types have only one member

v_np_le	1,723
v_np*_le	962
v_p-np_le	896
v_p_le	506
v_pp_e_le	494
v_-_le	463
v_np_noger_le	408
v_-_unacc_le	325
v_np-pp_e_le	322
v_pp*_dir_le	184

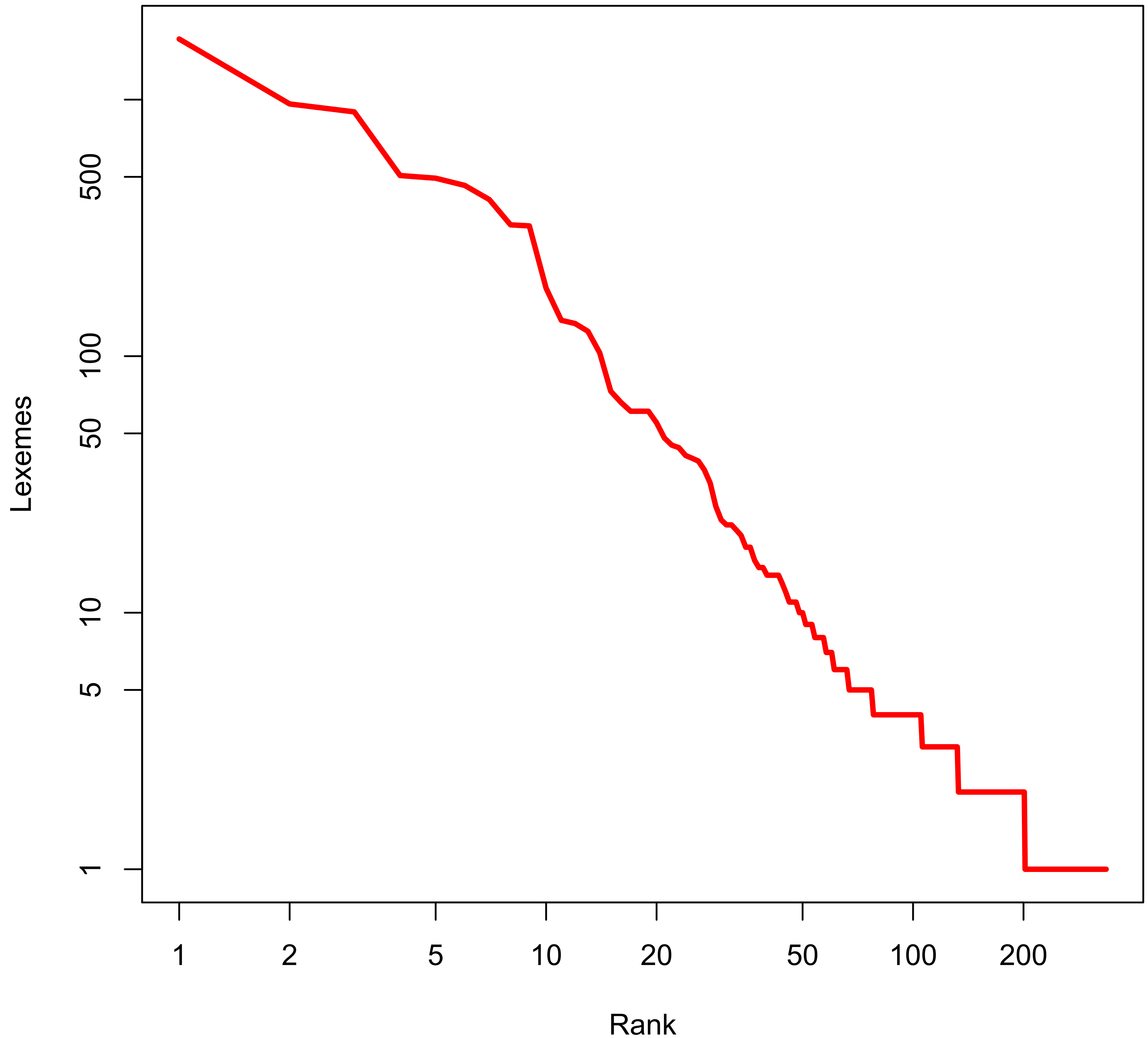
English Resource Grammar

Inverse power-law distribution (Zipf's Law)



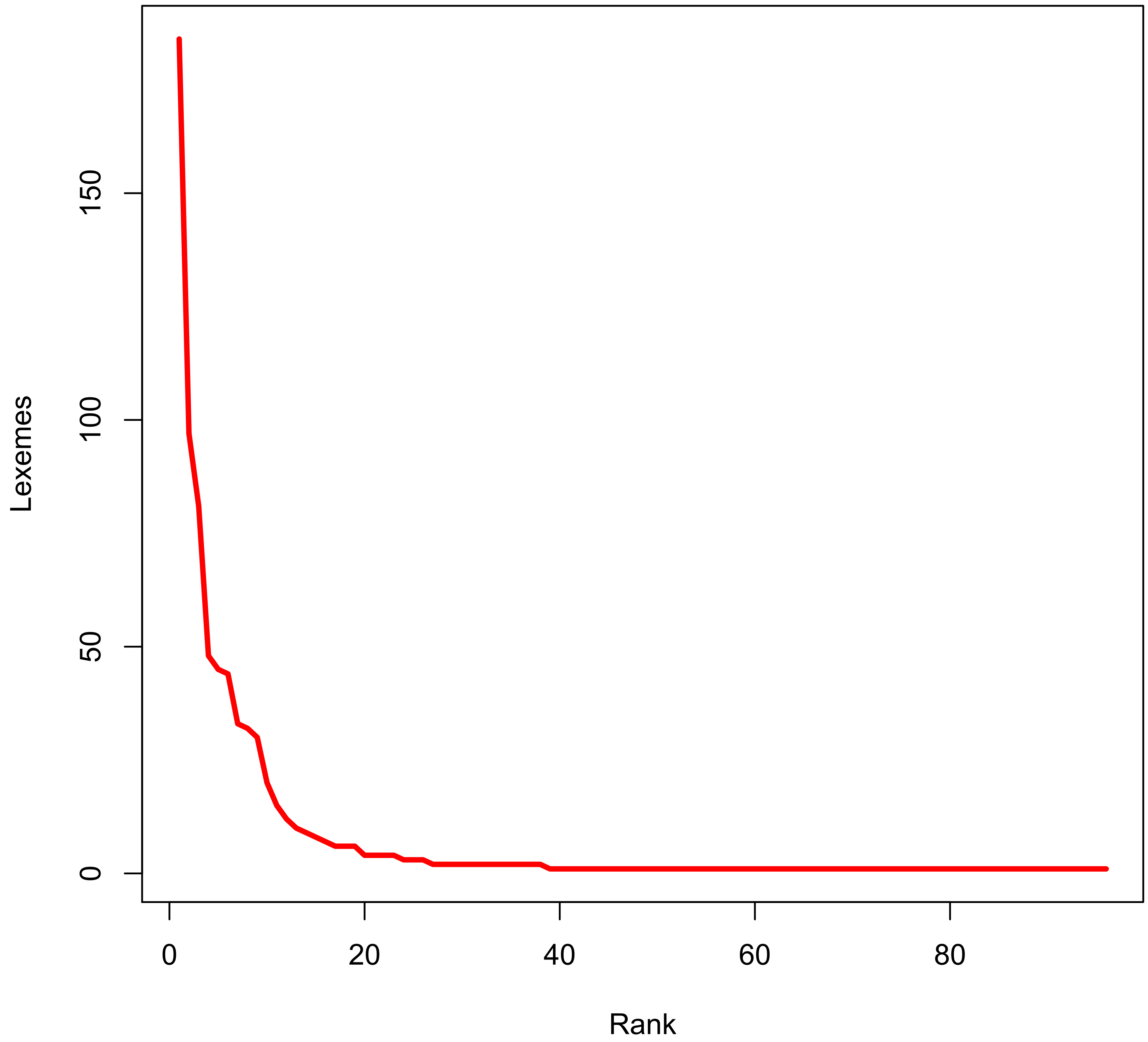
English Resource Grammar

Inverse power-law distribution (Zipf's Law)



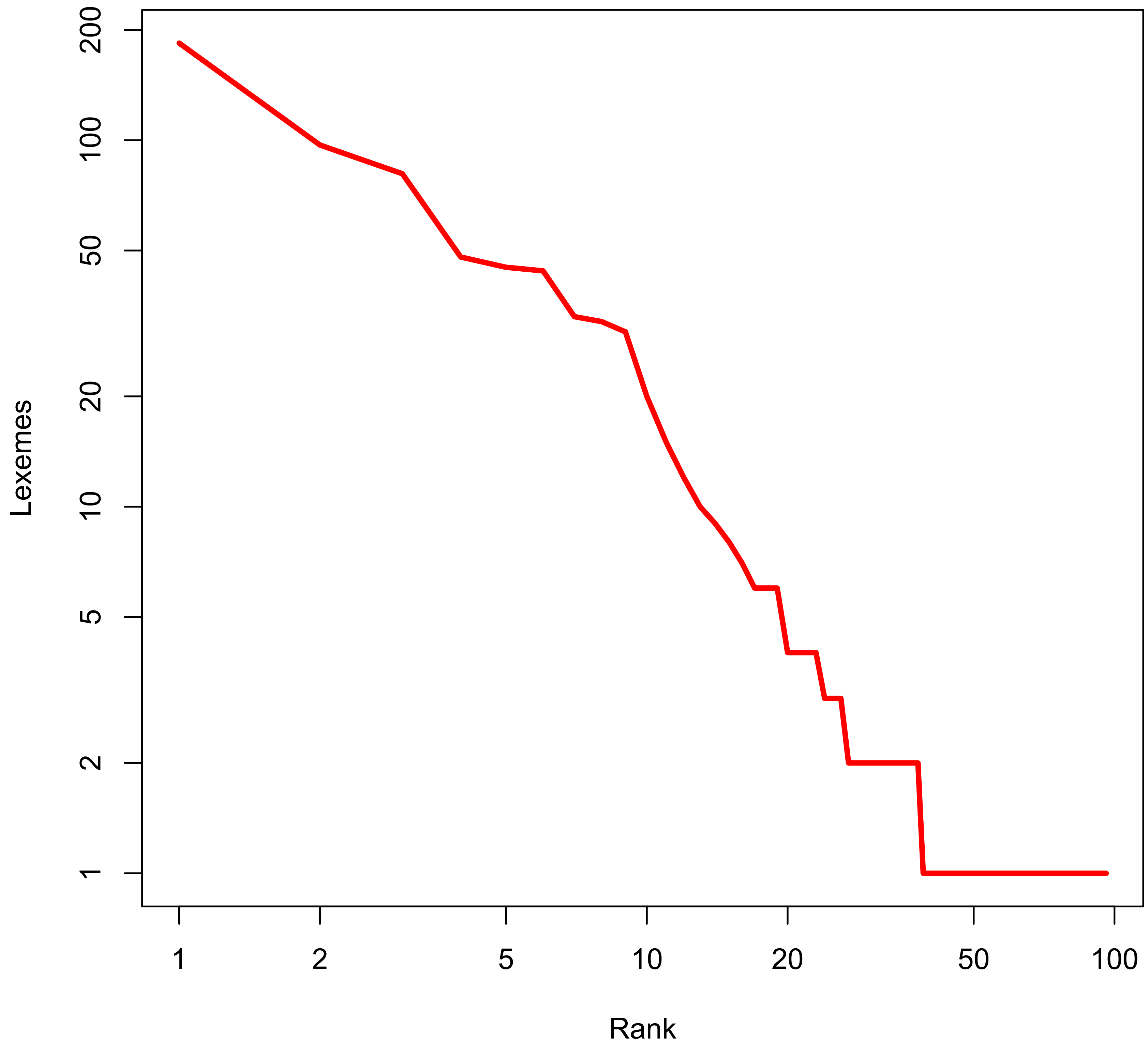
English Resource Grammar

Scale invariance: Sublexicon of 800 randomly selected verbs (96 types)



English Resource Grammar

Scale invariance: Sublexicon of 800 randomly selected verbs (96 types)



British National Corpus

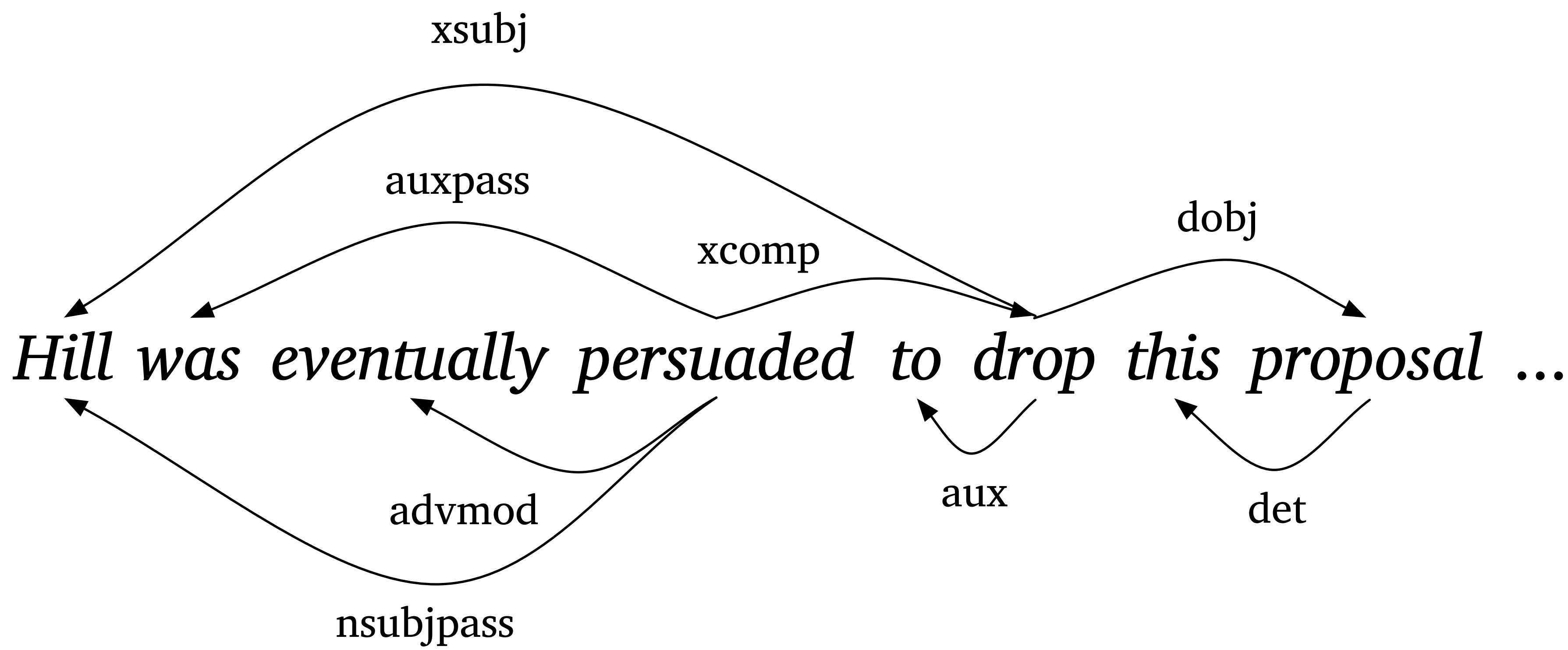
The ERG covers only a small part of the English vocabulary

Even for words that are listed, entries are incomplete (Baldwin, *et al.* 2004)

Suppose we constructed a lexicon with 100% coverage of the BNC . . . How many types would we need?

British National Corpus

Parse each sentence using the Stanford Dependency Parser



A **verb frame** is a bag of relations

persuade	⟨ nsubjpass, advmod, xcomp ⟩
drop	⟨ xsubj, dobj ⟩

British National Corpus

A **verb type** is a collection of frames that a verb occurs in

persuade	xcomp	469	*drop*	nsubj dobj	594
	xsubj xcomp	317		nsubj dobj prep	526
	nsubj xcomp	316		nsubj prep	444
	dobj	254		dobj	383
	dobj xcomp	221		prep	275
	dobj ccomp	144		dobj prep	266
	nsubjpass xcomp	135		nsubj dobj	252
	xsubj dobj	135		nsubj dobj advmod	222
	nsubj dobj	126		nsubj advmod prep	221
	nsubj dobj xcomp	112		nsubj prep prep	186
	. . .			. . .	

British National Corpus

Verb frames with the highest type frequency

nsubj	15,982
dobj	13,611
nsubj dobj	13,574
nsubj ccomp	11,347
prep	9,879
nsubj prep	7,878
dobj prep	6,987
nsubj dobj prep	6,873
nsubj xcomp	5,980
nsubj dobj advmod	5,843

British National Corpus

Applying this method to the BNC, we get

92,612 distinct frames
67,423 verb lexemes
28,778 verb types

For each lexeme, drop frames that occur fewer than 10 times:

4,399 distinct frames
67,423 lexemes
2,554 lexical types

And if we also only consider lexemes that occur at least 500 times:

4,398 distinct frames
1,546 lexemes
1,545 lexical types

British National Corpus

Verbs in the BNC do not appear to be organized into types

Is the lexicon structured at all?

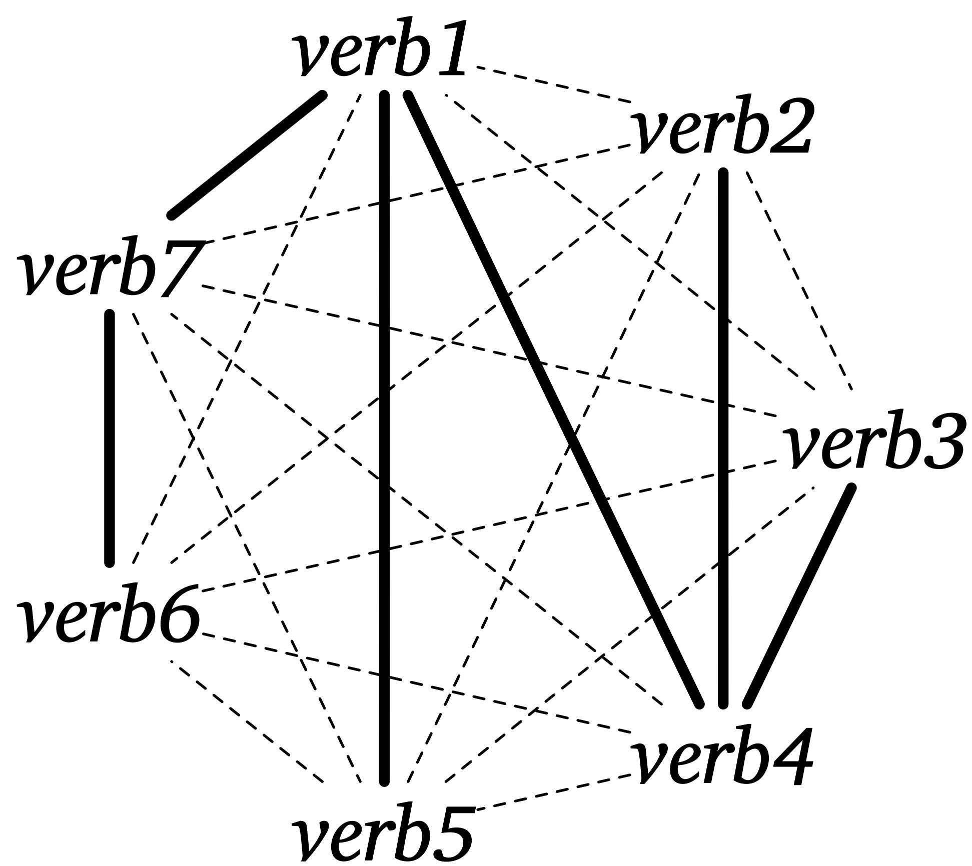
- Verb frames could be interpreted as binary features which define 'natural' classes of verbs

- Or, verbs could be organized into differential network

What evidence is there for internal structure?

Spanning trees

A delegation network is a connected acyclic graph (**spanning tree**) joining all lexical entries



Because lexical constraints are defaults, any network structure will work – but, not all are equivalent

Spanning trees

Evaluate networks on the basis of shared information:

Measure the difference between joined lexical entries by Jaccard distance

$$J_\delta(X, Y) = 1 - \frac{|X \cap Y|}{|X \cup Y|}$$

This captures the degree of default overriding between joined entries

A link between identical lexical entries would have a cost of 0

Find a **minimum** spanning tree – one with the smallest possible sum of edge weights (Kruskal 1956)

Spanning trees

The minimum spanning tree cost for BNC verbs is 597.00

Is that high or low?

- Generate 100 uniform random (not necessarily minimum) spanning trees (Broder 1989, Aldous 1989)
- Average sum of distances is 1227.69
- Min is 1216.90 and max is 1239.14

Conclusion:

- There aren't many more verbs than there are types of verbs
- Verbs also aren't all unique
- A differential network captures at least some of the structure in the verbal lexicon

Constructions

Ginsberg and Sag (2000) present an analysis of a range of English interrogative constructions (and other related phenomena)

Detailed syntactic and semantic model based on HPSG and (more loosely) Situation Semantics

Constructions are organized into a multiple inheritance type hierarchy with a limited degree of default overriding

Location in the hierarchy specifies a constructions syntactic and semantic properties

Sign-Based Construction Grammar (Sag 1997, van Noord *et al.* 1999, Sag 2007, Sag *et al.* 2012)

Constructions

Declarative and interrogative constructions

decl_hd_su_cl	*Kim smiled.*
inv_decl_cl	*doesn't Kim like __*
decl_ns_cl	*to smile*
decl_frag_cl	*Bagels.*
pol_int_cl	*Did Kim leave?*
ns_wh_int_cl	*What did Kim see?*
su_wh_int_cl	*Who left?*
repr_int_cl	*You're leaving?*
dir_is_int_cl	*Kim saw Sandy?*
slu_int_cl	*Who?*

Constructions

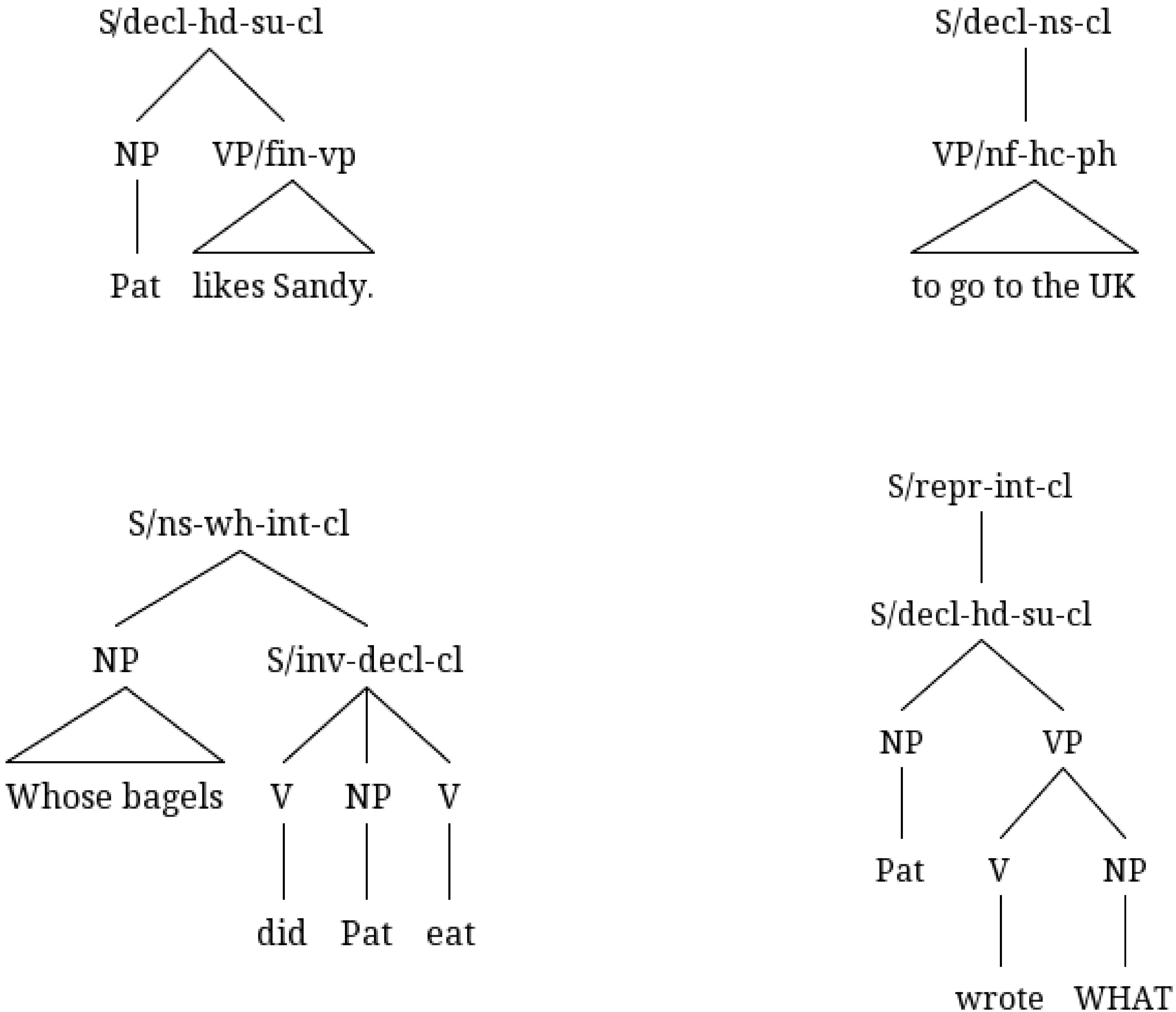
Other clause types

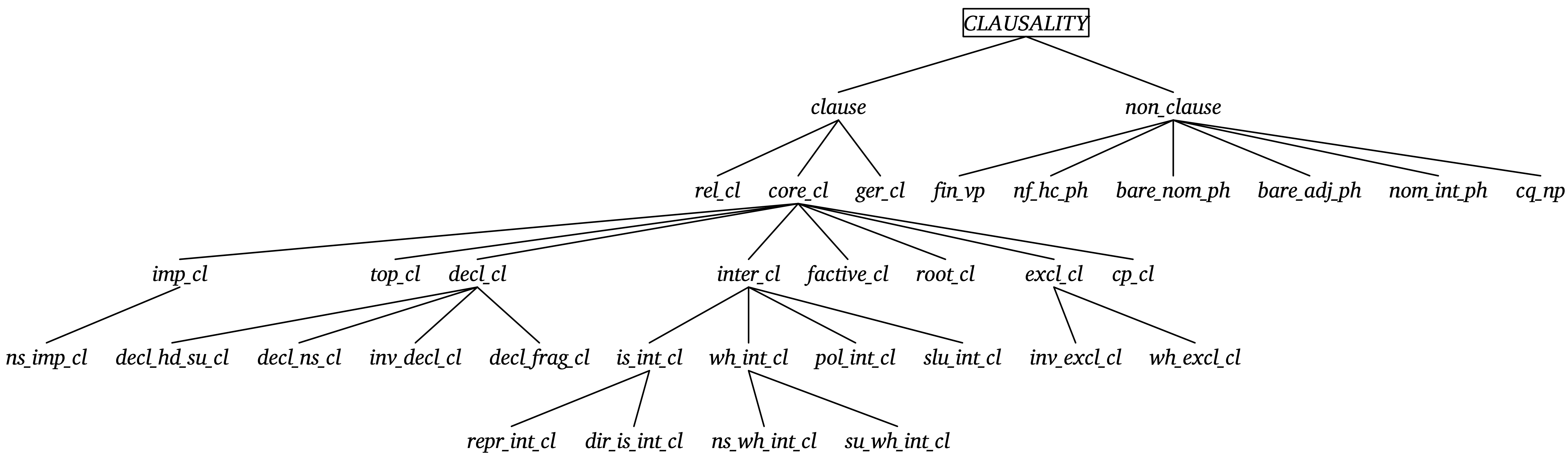
inv_excl_cl	*Am I tired!*
wh_excl_cl	*how odd it is*
ns_imp_cl	*Be quiet!*
top_cl	*The bagels, I like.*
factive_cl	*that Kim left*
root_cl	*Kim left.*
cp_cl	*whether Kim left*

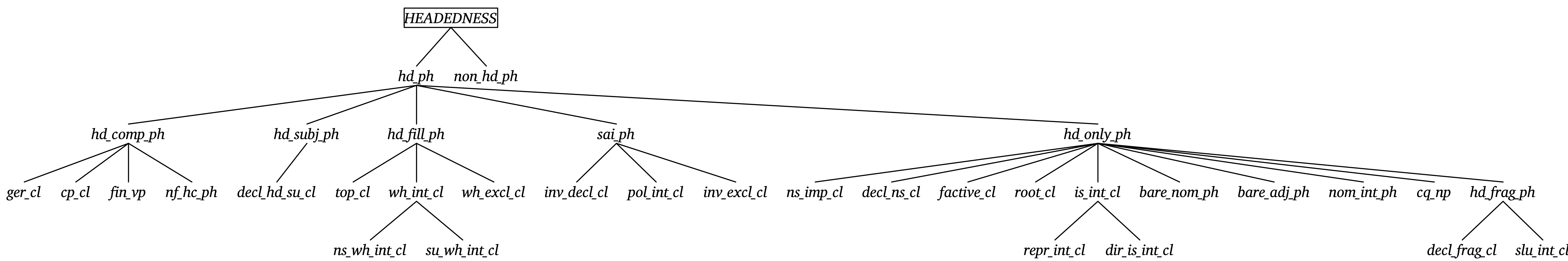
Constructions

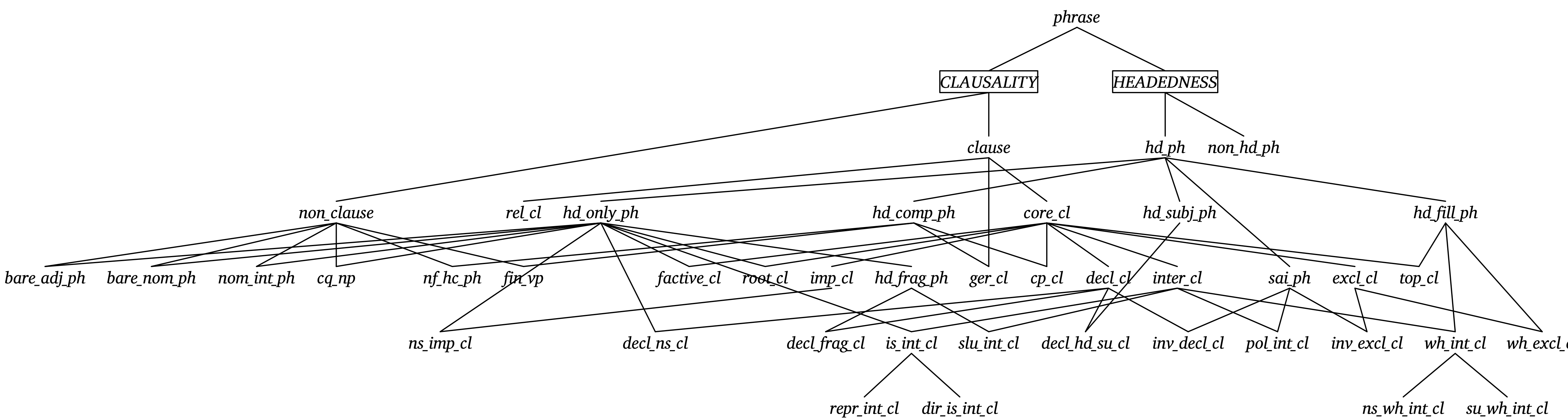
Non-clauses

fin_vp	*went home*
nf_hc_ph	*going home*
bare_nom_ph	*old bagels*
bare_adj_ph	*very sad*
nom_int_ph	*who left*
cq_np	*Your name?*


S/decl-hd-su-cl
NP
VP/fin-vp
Pat
likes Sandy.
S/decl-ns-cl
VP/nf-hc-ph
to go to the UK
S/ns-wh-int-cl
NP
S/inv-decl-cl
Whose bagels
V
NP
V
did
Pat
eat
S/repr-int-cl
S/decl-hd-su-cl
NP
VP
Pat
V
NP
wrote
WHAT

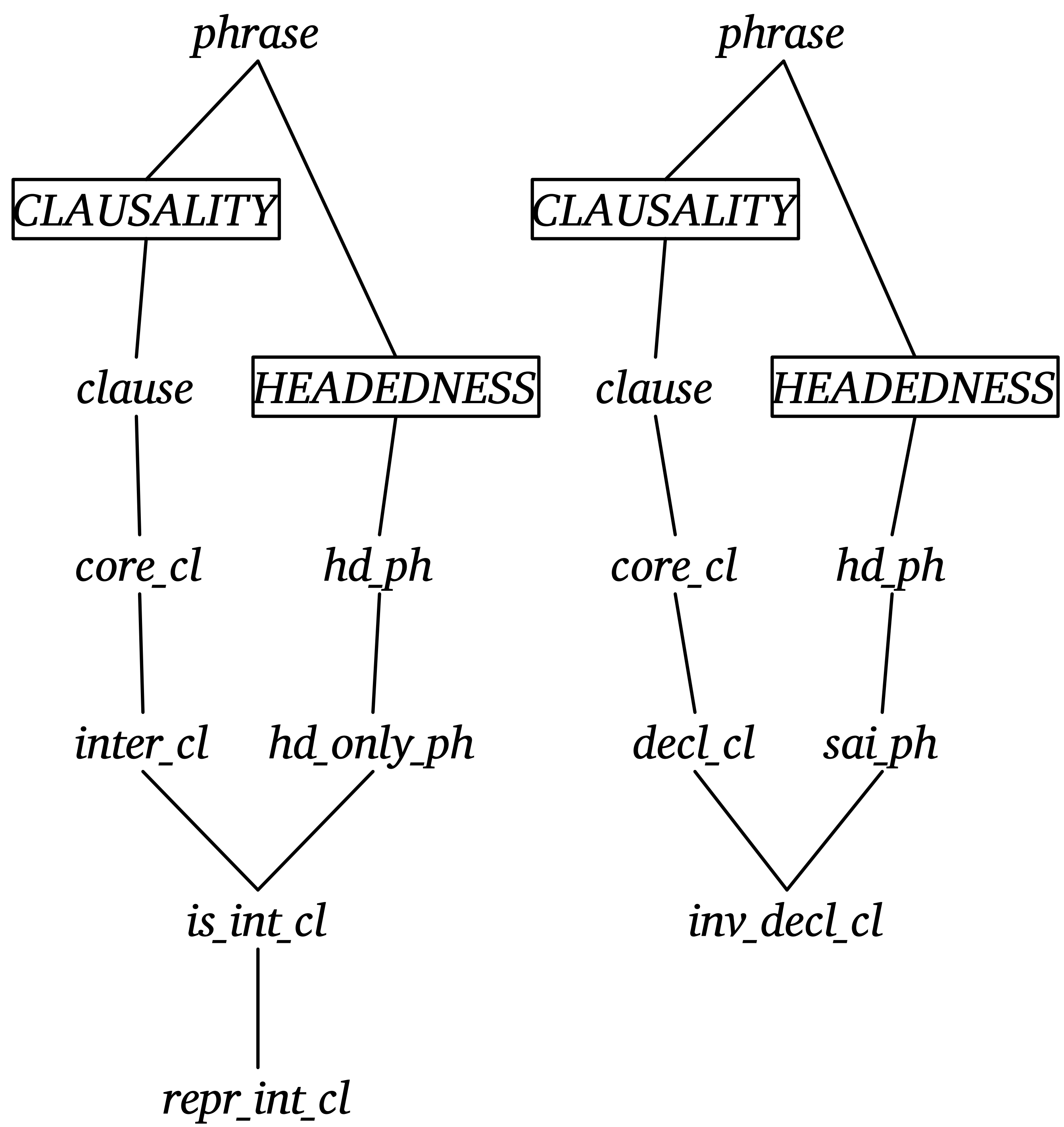

CLAUSALITY
clause
non_clause
rel_cl
core_cl
ger_cl
fin_vp
nf_hc_ph
bare_nom_ph
bare_adj_ph
nom_int_ph
cq_np
imp_cl
top_cl
decl_cl
inter_cl
factive_cl
root_cl
excl_cl
cp_cl
ns_imp_cl
decl_hd_su_cl
decl_ns_cl
inv_decl_cl
decl_frag_cl
is_int_cl
wh_int_cl
pol_int_cl
slu_int_cl
inv_excl_cl
wh_excl_cl
repr_int_cl
dir_is_int_cl
ns_wh_int_cl
su_wh_int_cl


HEADEDNESS
hd_ph
non_hd_ph
hd_comp_ph
hd_subj_ph
hd_fill_ph
sai_ph
hd_only_ph
ger_cl
cp_cl
fin_vp
nf_hc_ph
decl_hd_su_cl
top_cl
wh_int_cl
wh_excl_cl
inv_decl_cl
pol_int_cl
inv_excl_cl
ns_imp_cl
decl_ns_cl
factive_cl
root_cl
is_int_cl
bare_nom_ph
bare_adj_ph
nom_int_ph
cq_np
hd_frag_ph
ns_wh_int_cl
su_wh_int_cl
repr_int_cl
dir_is_int_cl
decl_frag_cl
slu_int_cl

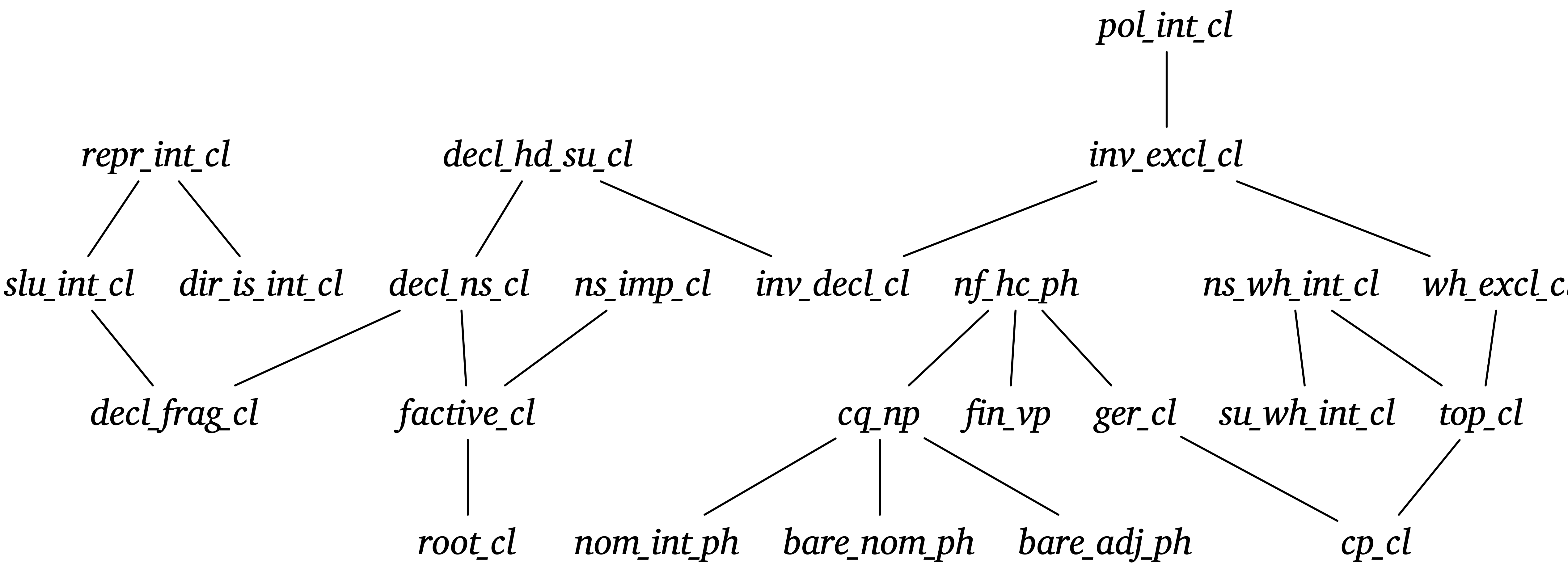

phrase
CLAUSALITY
HEADEDNESS
clause
hd_ph
non_hd_ph
non_clause
rel_cl
hd_only_ph
hd_comp_ph
core_cl
hd_subj_ph
hd_fill_ph
bare_adj_ph
bare_nom_ph
nom_int_ph
cq_np
nf_hc_ph
fin_vp
factive_cl
root_cl
imp_cl
hd_frag_ph
ger_cl
cp_cl
decl_cl
inter_cl
sai_ph
excl_cl
top_cl
ns_imp_cl
decl_ns_cl
decl_frag_cl
is_int_cl
slu_int_cl
decl_hd_su_cl
inv_decl_cl
pol_int_cl
inv_excl_cl
wh_int_cl
wh_excl_cl
repr_int_cl
dir_is_int_cl
ns_wh_int_cl
su_wh_int_cl

Constructions

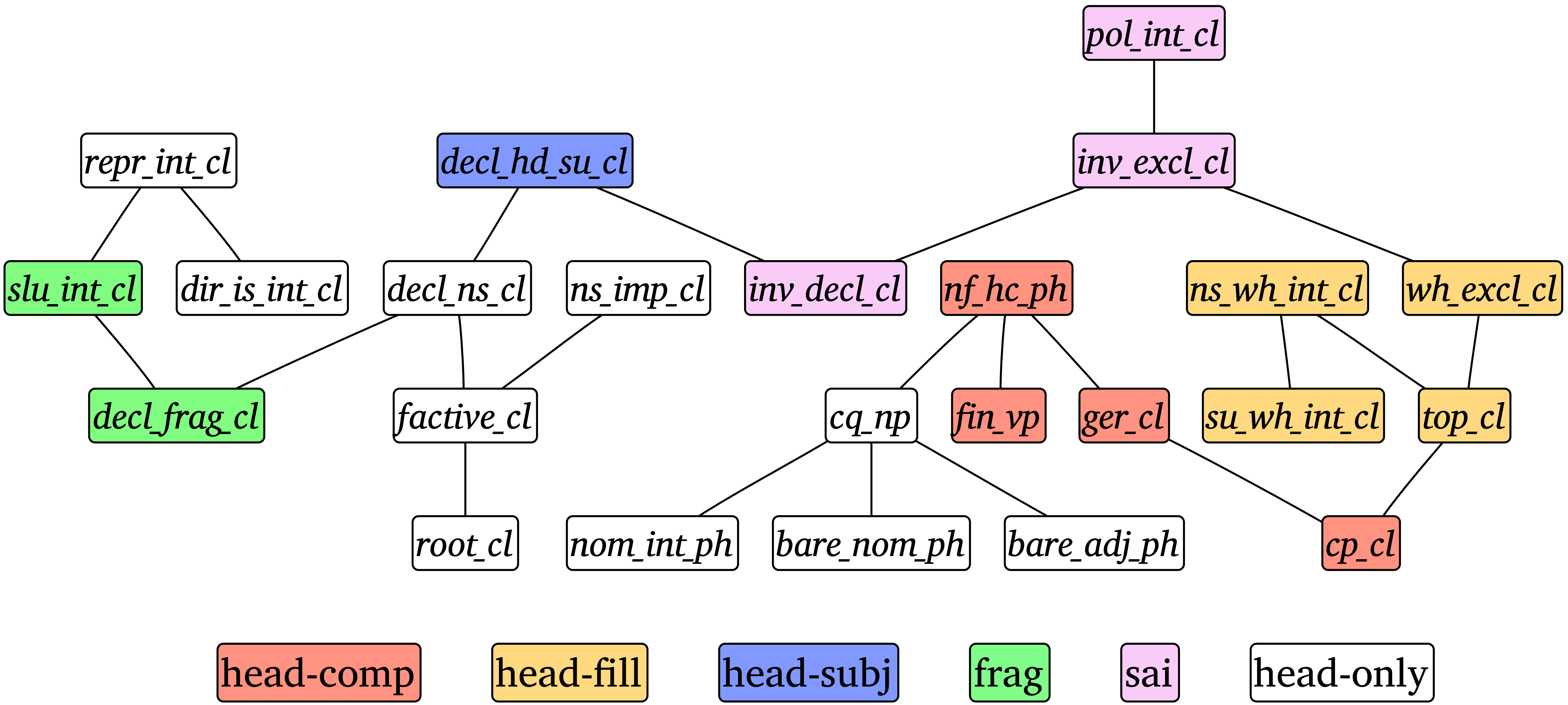
Jaccard distance



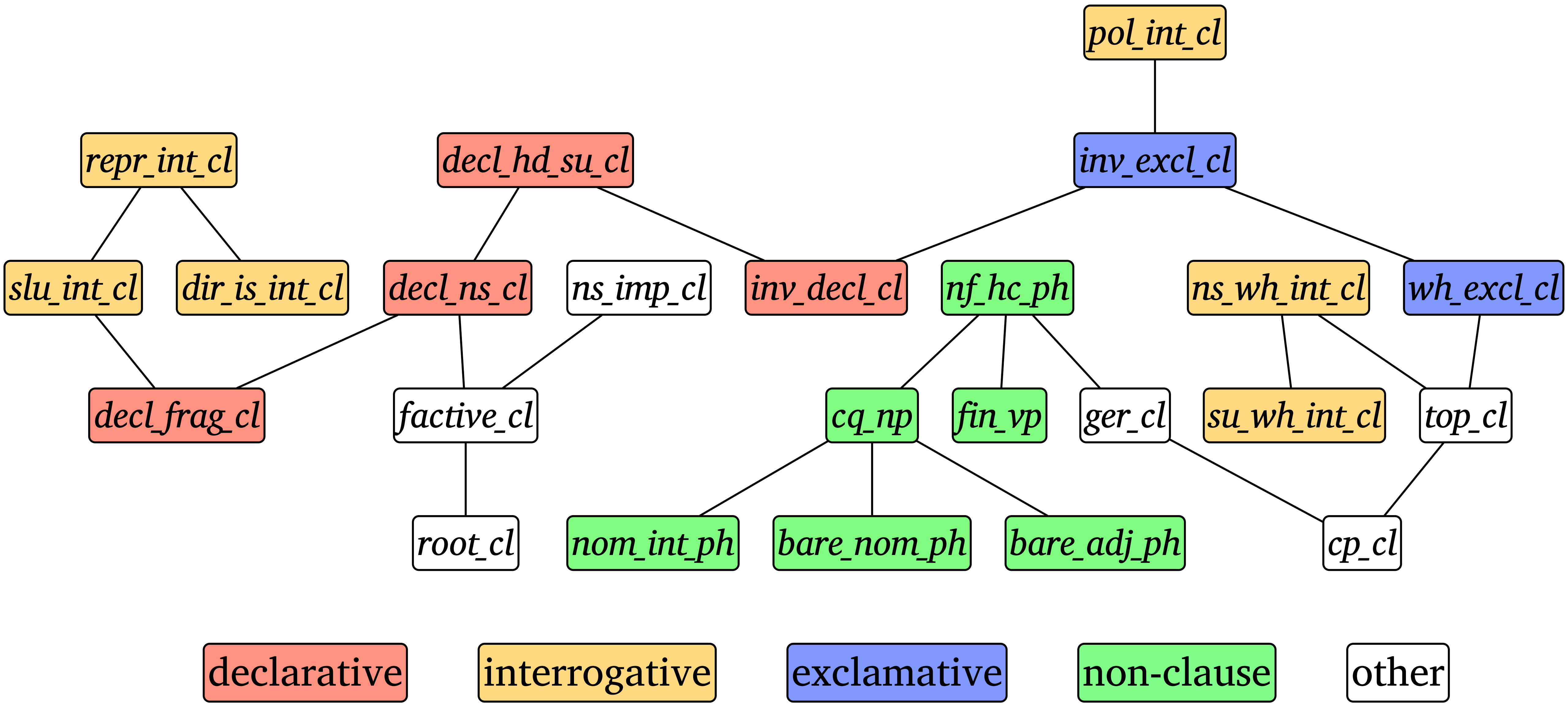
Constructions



Constructions



Constructions



Constructions

The diversity is among constructions is lower than would be expected if HEADEDNESS and CLAUSALITY really were orthogonal dimensions

A flat differential network captures most (all?) of the generalizations that G&S's complex multiple inheritance hierarchy does

Differential and hierarchical analyses aren't mutually exclusive options (cf. **traits**)

Approaching the problem of organizing constructions quantitatively may reveal patterns that aren't otherwise obvious

Prospects

Differential networks are a viable alternative to taxonomic representations

How far can they be extended?

- Richer datasets
- Other lexicalist frameworks (Network Morphology, Word Grammar)

How can they be refined?

- **Families** as a step towards types (Astudillo and Schilling 1993)
- No reason to limit focus to spanning trees (Ackerman and Bonami)
- Types, tokens, exemplars (Abbot-Smith and Tomasello 2006, Baayen *et al.* 2007)