Encoding and Deploying Linguistic Knowledge: The Grammar Matrix and AGGREGATION Projects

Emily M. Bender
University of Washington

Universitetet i Oslo
Forskningsgruppen for språkteknologi
5 februar 2013

DELPH-IN

Acknowledgments

- This talk represents joint work with:

  Joshua Crowgey, Scott Drellishak, Dan Flickinger, Antske Fokkens, Michael Goodman, Joshua Hou, Alex Lascarides, Daniel Mills, Safiyyah Saleem, Sanghoun Song, Stephan Oepen, Kelly O'Hara, Laurie Poulson, David Wax, Fei Xia

- This material is based upon work supported by the National Science Foundation under Grants No. 0644097 & 1160274. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

- We are grateful to additional support from the Utilika Foundation, the Turing Center at the University of Washington, and the Max Planck Institute for Evolutionary Anthropology.

Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

Precision Grammars

- Encode linguistic analyses
- Map surface(-y) strings to semantic representations
- Model grammaticality
- Are more consistent and more scaleable than treebank-trained grammars
- Take sustained effort to develop

Why precision grammars

- Linguistic research
- Scalability/domain portability
- Applications which require rich, precise semantic information
  - NLU
- Applications which require grammatical realizations
  - Grammar checking, anything requiring generation

Normalizing dependencies

- Kim gave Sandy a book.
- Kim gave a book to Sandy.
- A book was given to Sandy by Kim.
- This is the book that Kim gave to Sandy.
- I’m looking for the book given to Sandy by Kim.
- Kim gave Sandy and Pat lent Chris a book.
- Which book do you think that Kim gave to Sandy?
- It’s a book that Kim gave to Sandy.
- This book is difficult to imagine that Kim could give to Sandy.

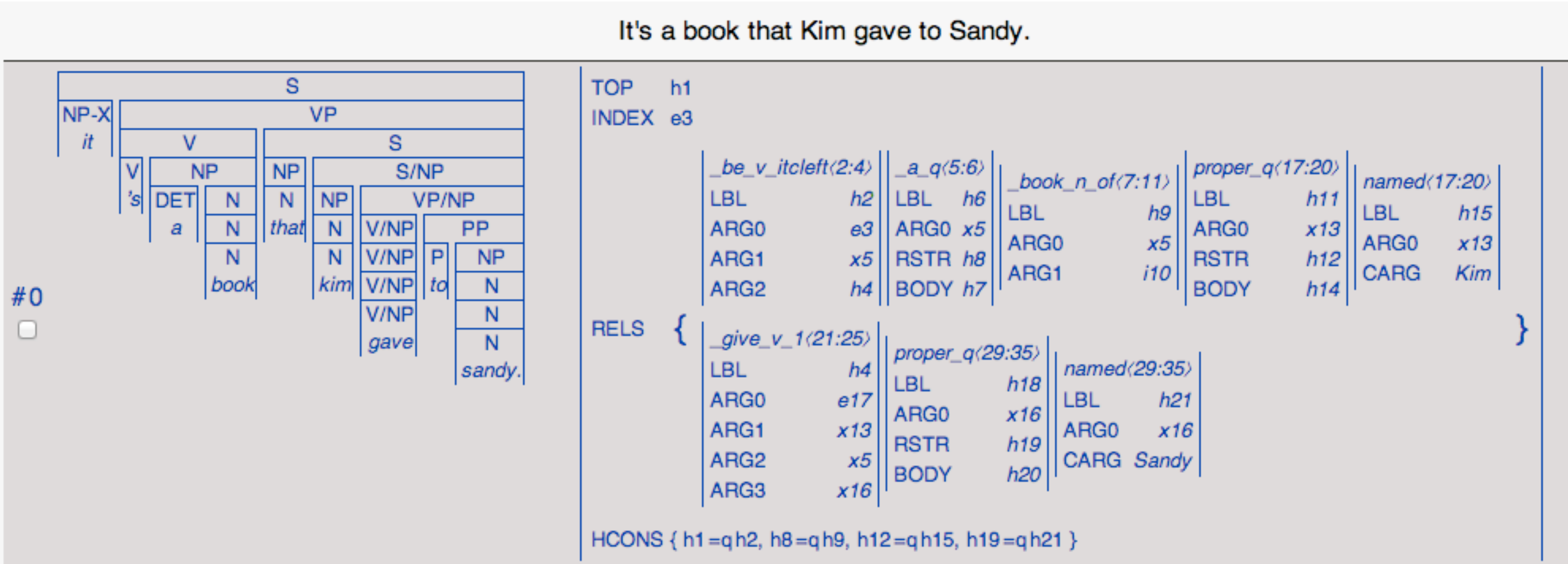
give(Kim,book,Sandy)

Better linguistics

- What computer scientists must imagine syntacticians do
  - We say we study rule systems assigning structure to natural language, and mapping between surface forms and semantic representations
  - The rule systems are formal and the modeling domain is complex
- If we make our analyses machine readable:
  - computers can verify that the systems work as intended
  - and validate against far more data

(cf. Bender 2008)

Example from the English Resource Grammar (Flickinger 2000, 2011)



erg.delph-in.net

ERG complexity (Flickinger 2011)

- As of 2010, the English Resource grammar comprised:
  - 980 lexical types
  - 35,000 manually constructed lexeme entries
  - 70 derivational and inflectional rules
  - 200 syntactic rules
- All of these pieces interact, sometimes in surprising ways
- 20+ person-years of development effort

DELPH-IN

The DELPH-IN ecology www.delph-in.net

- Head-drive Phrase Structure Grammar (Pollard & Sag 1994)
- Joint reference formalism (Copestake 2002a)
- Shared semantic representation formalism (MRS; Copestake et al 2005)
- Grammars: ERG (Flickinger 2000, 2011), Jacy (Siegel & Bender 2002), NorSource (Hellan & Haugereid 2003), ...
- Grammar generator: Grammar Matrix (Bender et al 2002, 2010)
- Parser generators: LKB (Copestake 2002b), PET (Callmeier 2002), ACE (moin.delph-in.net/AceTop), agree (moin.delph-in.net/AgreeTop)

DELPH-IN

The DELPH-IN ecology www.delph-in.net

- Parse and realization ranking: (e.g., Toutanova et al 2005, Velldal 2008)
- Robustness measures: (e.g., Zhang & Kordoni 2006, Zhang & Krieger 2011)
- Regression testing: [incr tsdb()] (Oepen 2001)
- Applications: e.g., MT (Oepen et al 2007), QA from structured knowledge sources (Frank et al 2007), Textual entailment (Bergmair 2008), ontology construction (Nichols et al 2006) and grammar checking (Suppes et al 2012)

Multilingual grammar engineering: Other approaches

- The DELPH-IN consortium specializes in large HPSG grammars
- Other broad-coverage precision grammars have been built by/in/with
    - LFG (ParGram: Butt et al 2002)
  - F/XTAG (Doran et al 1994)
  - HPSG: ALE/Controll (Götz & Meurers 1997)
  - SFG (Bateman 1997)
- Proprietary formalisms and Microsoft and Boeing and IBM

Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

LinGO Grammar Matrix: Motivations and early history

- Speed up grammar development
  - Initial context: Project DeepThought
  - Leverage resources from resource-rich language to enhance NLP for resource-poor languages
  - Claim: Some of what was learned in ERG development is not English-specific
- Interoperability: a family of grammars compatible with the same downstream processing tools

Grammar Matrix: Motivations and early history

- With reference to Jacy, strip everything from ERG which looks English-specific
- Resulting “core grammar” doesn’t parse or generate anything, but supports quick start-up for scaleable resources (Bender et al 2002)
- Used in the development of grammars for Norwegian (Hellan & Haugereid 2003), Modern Greek (Kordoni & Neu 2005), Spanish (Marimon et al 2007) and Italian

Linguistic interest as well

Ticket #1 (new task)

UG

Opened 4 years ago
Last modified 4 years ago

Reported by:	ebender	Owned by:	somebody
Priority:	major	Milestone:	
Component:	General	Version:	
Keywords:		Cc:	

Description (last modified by ebender) (diff)

Build comprehensive, implemented, correct Universal Grammar

Sample of hypotheses encoded in Matrix core grammar

- Words and phrases combine to make larger phrases.
- The semantics of a phrase is determined by the words in the phrase and how they are put together.
- Some rules for phrases add semantics (but some don't).
- Most phrases have an identifiable head daughter.
- Heads determine which arguments they require and how they combine semantically with those arguments.
- Modifiers determine which kinds of heads they can modify, and how they combine semantically with those heads.
- No lexical or syntactic rule can remove semantic information.

Questions that can only be answered by building grammars for many languages

- Is Minimal Recursion Semantics and appropriate representation format for the meanings of all languages?
- To what extent can representations designed for particular phenomena in one language (English) be re-used in other languages?
  - Negation
  - Comparatives
  - Nominalization
  - ...

Case study: Sentential negation
(Bender & Lascarides forthcoming)

- Negation (in English) interacts scopally with quantifiers:

Kim didn't read some book.

1. $\exists$x (book(x), $\neg$read(Kim,x))
2. $\neg\exists$x (book(x), read(Kim,x))

Case study: Sentential negation
(Bender & Lascarides forthcoming)

- But its scope is fixed by its position in the sentence:

Kim deliberately didn't read every book.

1. $\forall$x (book(x), deliberately($\neg$read (Kim,x)))
2. deliberately($\forall$x (book(x), $\neg$read (Kim,x)))
3. deliberately($\neg\forall$x (book(x), read(Kim, x)))
4. *$\neg$deliberately ($\forall$x (book(x), read(Kim, x)))
5. *$\forall$x (book(x), $\neg$deliberately(read(Kim,x)))

ERG/MRS Analysis

- Like quantifiers, negation is an elementary predication that takes a scopal argument
- The position of negation with respect to the negated verb is fixed, but quantifiers can "float" in between
- Modeled with "equal modulo quantifiers" (qeq) constraints

$$\left\langle h_1, \left\{ \begin{array}{l} h_3\text{:proper\_q}(x_5,\ h_4,\ h_6), \\ h_7\text{:named}(x_5,\ Kim), \\ h_8\text{:neg}(e_{10},\ h_9), \\ h_{11}\text{:\_read\_v\_1}(e_2,\ x_5,\ x_{12}), \\ h_{13}\text{:\_some\_q\_indiv}(x_{12},\ h_{15},\ h_{14}), \\ h_{16}\text{:\_book\_n\_of}(x_{12},\ i_{17}) \end{array} \right\}, \{h_{15} =_q h_{16}, h_9 =_q h_{11}, h_4 =_q h_7\} \right\rangle$$

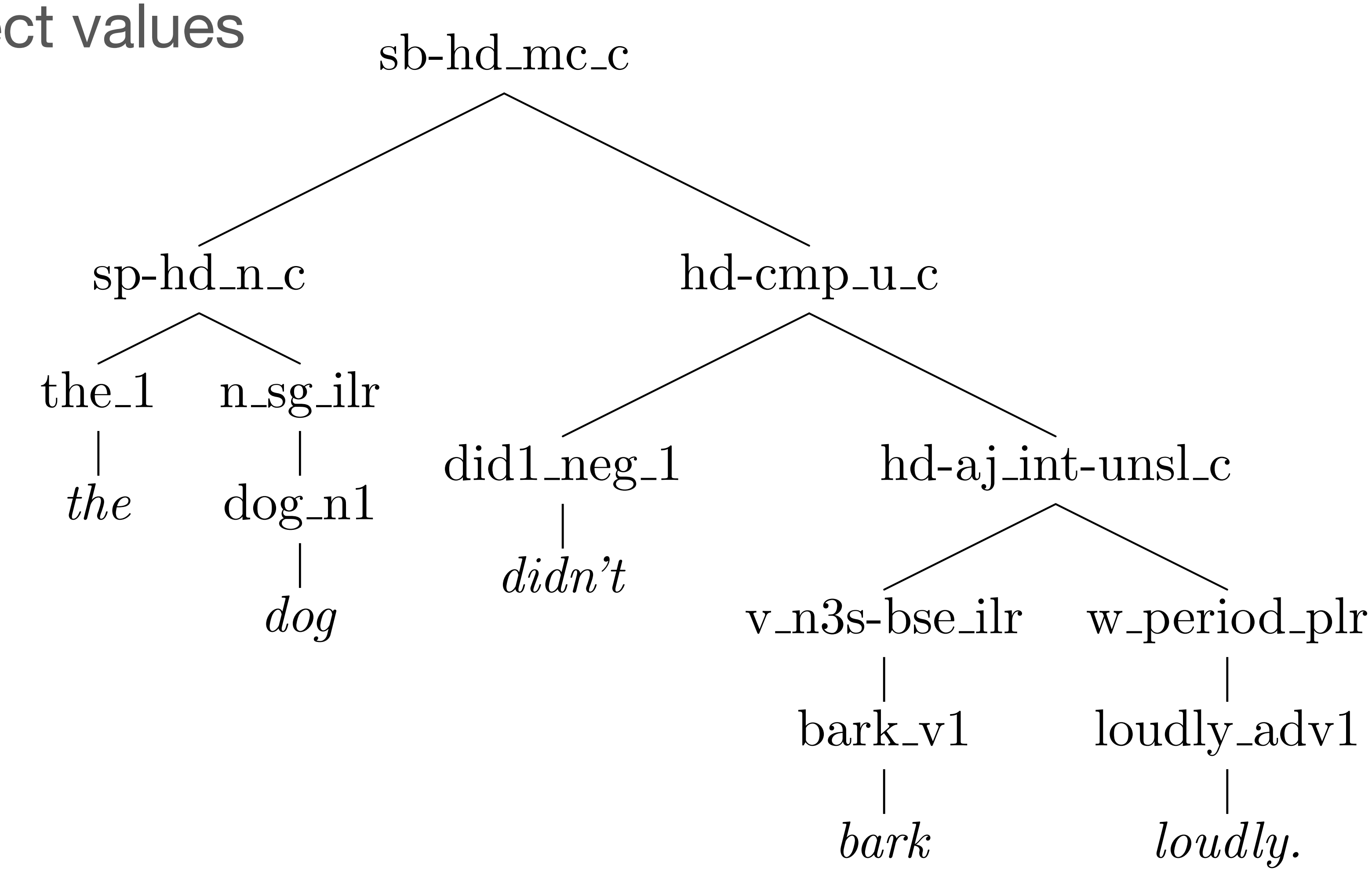
Does this work cross-linguistically?

- Verifying the representations would require semantic fieldwork
  - Do we find the same scope ambiguities in a variety of languages?
  - Do we find the same scope-fixed-by-position effects?
- Verifying the compositional strategy can be approached with typological research and grammar engineering
  - Typology: Dryer (2011) finds that 417 of 1159 languages sampled express negation with inflection on main verbs
  - (Other negation strategies do not appear to be problematic)

Compositional semantics and sentential negation in English (ERG)

- *didn't* “sees” label of *bark* via the COMPS feature

- *didn't* can thus contribute the qeq constraint with the correct values



Compositional semantics and sentential negation in Turkish

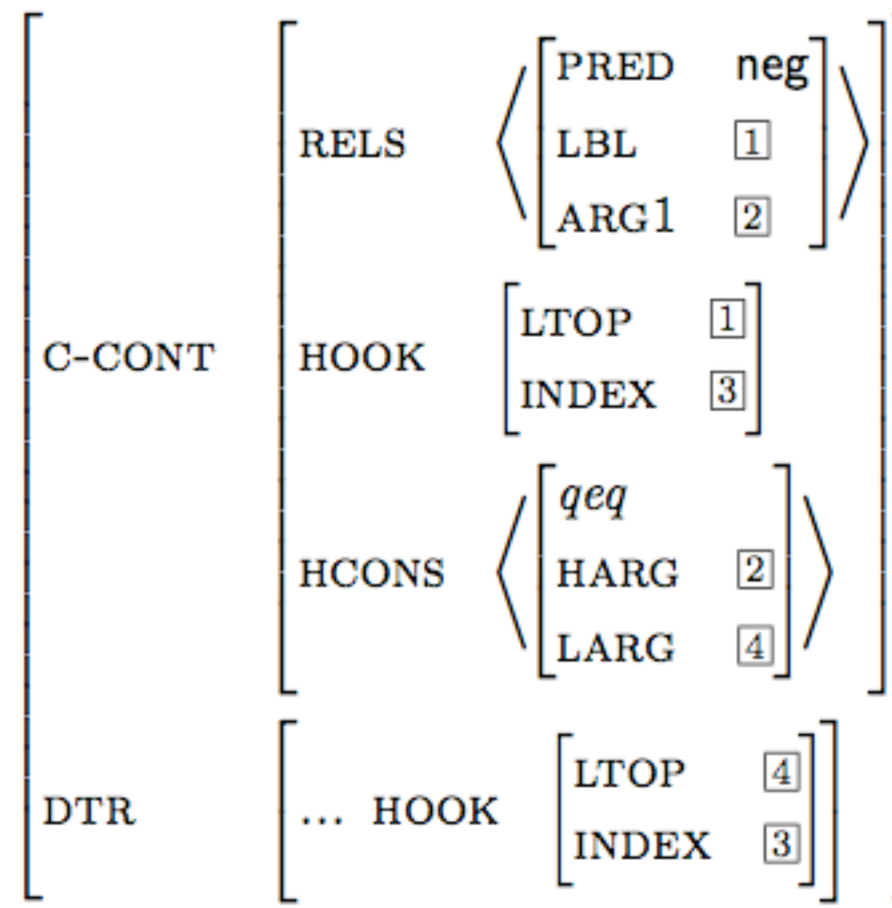
- Sentential negation is expressed via an affix on the verb:

Köpek	yuksek	ses-le	havla-ma-di.
dog	loud	voice-INST	bark-NEG-PST

'The dog didn't bark loudly' [tur]

Compositional semantics and sentential negation in Turkish

- No problem! The lexical rule which adds the negation affix, can also add negative semantics.
- Since its input is the verb stem, it also “sees” the label of *bark*



The problem: Intersective modifiers

- Intersective modifiers are analyzed as sharing the scope (= position in the scope tree) of the heads they modify

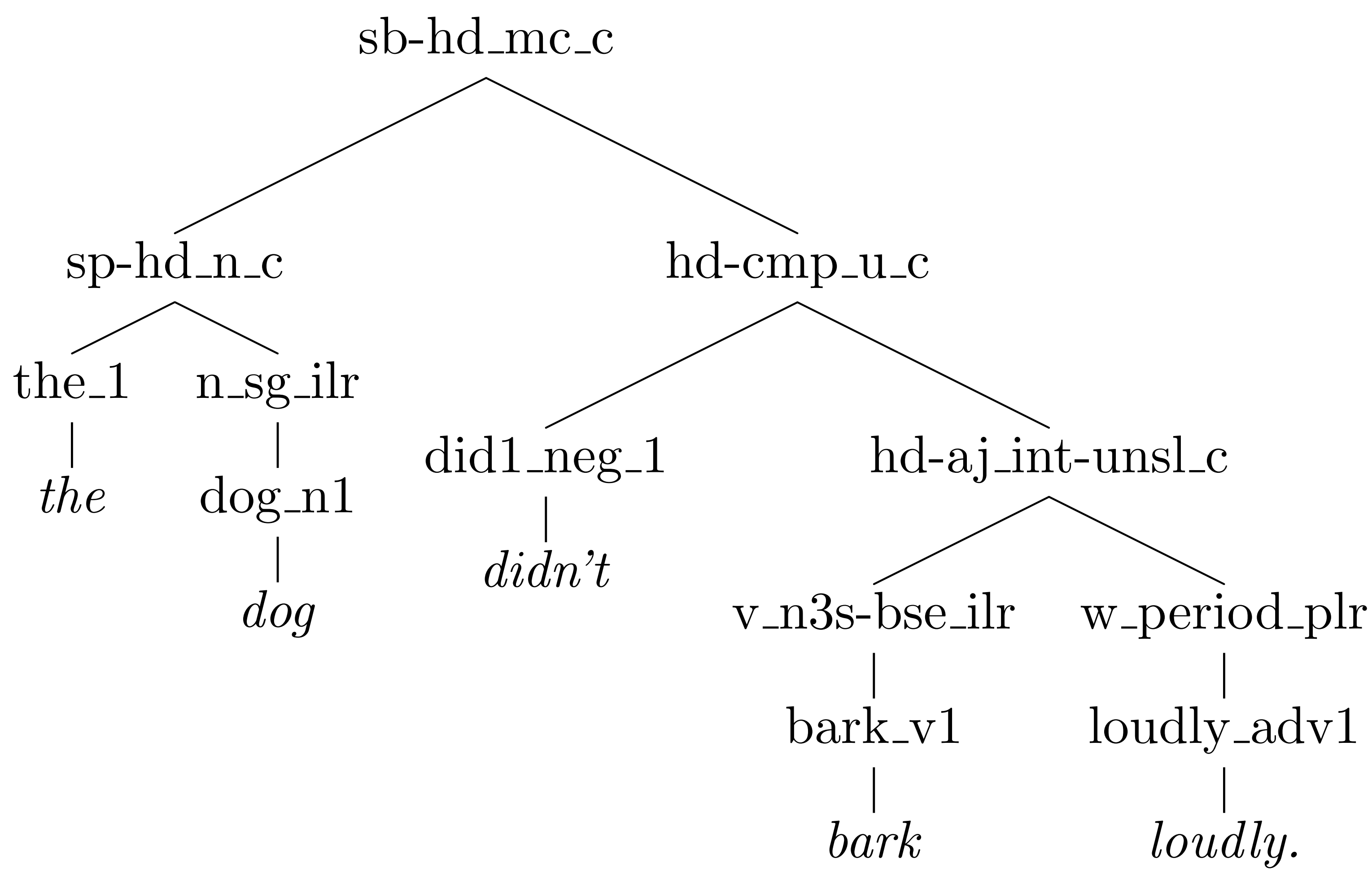
- This seems correct for the interaction with negation in English:

  - *The dog didn't bark loudly* == There was no barking situation in which the barking was loud and the barker was a dog.

  - Apparent "scopings" where only the adverb is involved can be traced to the focus-sensitivity of negation (Beaver and Clark 2008)

Negation and intersective modifiers

- No problem to model this in English; all that is required is sharing of a label between the head and the modifer
- *loudly* can “see” the label of *bark* when it attaches, and do the identification



But what about Turkish?

Köpek	yuksek	ses-le	havla-ma-di.
dog	loud	voice-INST	bark-NEG-PST

'The dog didn't bark loudly' [tur]

- The modifier *yuksek ses-le* attaches to the inflected verb (cf. Lexical Integrity Hypothesis; Bresnan & Mchombo 1995)
- The inflected verb must have the label of negation as its label, not the label of *havla* 'bark'.
- So how can we do the identification?

Two possibilities

- Disassociate morphology from semantics:
  - The lexical rule only introduces the marker and a syntactic feature reflecting its presence
  - A phrase structure rule higher in the tree triggered by that feature introduces the semantics
- Change the way attachment of intersective modifiers is handled in composition
  - Instead of equating the labels of modifier & modifiee, introduce a “less-than or equal to” (leq) constraint (Schlangen 2003, Alahverdzhieva & Lascarides 2011)

Case study: Conclusions

- Bender and Lascarides (forthcoming) argue that the first approach doesn't scale to handle the interaction with other phenomena, such as morphological causatives

- and suggest that leqs are needed, at least for Turkish-type languages

- need to allow for some minor crosslinguistic variation in the underspecified version of the representations

Case study: Take aways

- Level of linguistic detail being encoded in DELPH-IN grammars
- Regarding the question: To what extent can representations designed for particular phenomena in one language (English) be re-used in other languages?
- Importance of
  - cross-linguistic investigation
  - interaction of phenomena
  - computational modeling
- But also: amount of wiggle room available
- Meta: Feedback from Matrix users is critical

Grammar Matrix: Summary

- Precision grammars encode complex linguistic analyses and provide rich information to NLP applications
- Precision grammars are expensive to build
- We can reduce the cost of creating new grammars by reusing what we’ve learned
- And increase interoperability, creating grammars with congruent encodings and output representations
- The resulting core grammar is a linguistically interesting object
    - ... especially to the extent that it is refined in response to feedback from users

Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

Grammar customization: Motivations

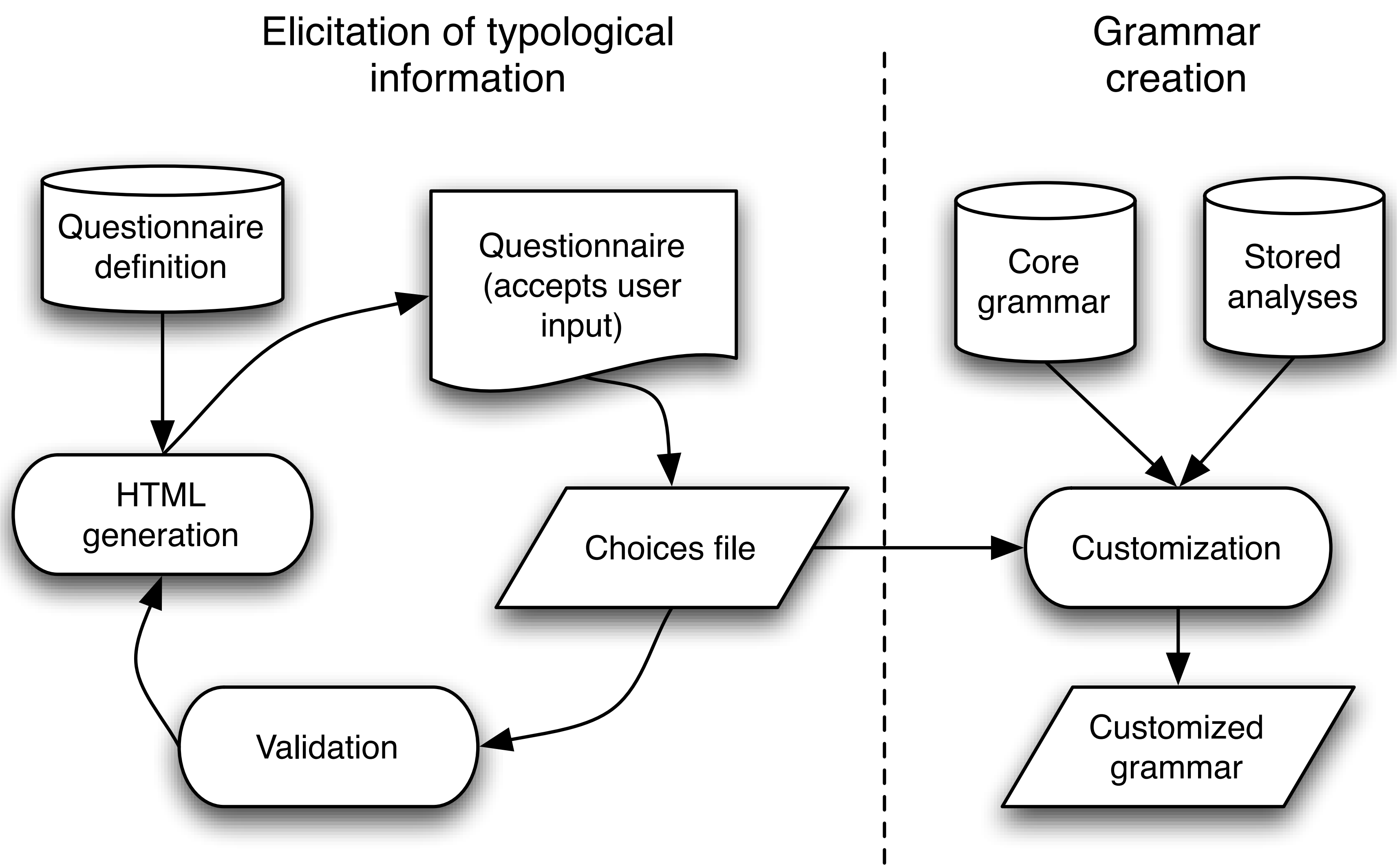
- The Grammar Matrix core grammar is not itself a functioning grammar fragment
  - can’t be directly tested
- Human languages vary along many dimensions, but not infinitely
- Can be seen as solving many of the same problems in different ways
- Many phenomena are “widespread, but not universal” (Drellishak, 2009)
  - we can do more than refining the core
- Also, grammar engineering lab instructions started getting mechanistic

Plus still more linguistic interest

- Can the same analyses of SVO word order, or split ergativity, or “pro-drop” work across all languages which have them?
- Can the interoperability of analyses predict typological patterns of phenomena co-occurrence?

LinGO Grammar Matrix Customization System

(Bender & Flickinger 2005, Bender et al 2010)



http://www.delph-in.net/matrix/customize/matrix.cgi

Current and near-future libraries

- Word order (Bender & Flickinger 2005, Fokkens 2010)
- Morphotactics (O’Hara 2008, Goodman & Bender 2010)
- Case (+ direct-inverse marking) (Drellishak 2009)
- Agreement (person, number, gender) (Drellishak 2009)
- Tense and aspect (Poulson 2011)
- Sentential negation (Bender & Flickinger 2005, Crowgey 2012)
- Coordination (Drellishak & Bender 2005)
- Matrix yes-no questions* (Bender & Flickinger 2005)
- Argument optionality (pro-drop) (Saleem & Bender 2010)
- Information structure (Song forthcoming)

Evaluation: How cross-linguistically adequate are the Matrix libraries?

- Evaluation of cross-linguistic applicability/“language-independence” requires testing on held out *languages*, not just held out *data*
- Ideally, the test languages should be typologically, genealogically, and areally diverse
- Chose seven languages not previously considered in Matrix library development, from different language families and geographic areas

Evaluation: How cross-linguistically adequate are the Matrix libraries?

- Worked from linguists' descriptive grammars
- RA not previously involved with Matrix development created testsuites illustrating the phenomena the Matrix libraries claimed to handle
- Starter grammars developed through the customization system, and iteratively improved
    - with reference to the test suites (testing generalization across languages, not data)
    - collaboratively with Matrix developers (testing potential of the system, not transparency to users)
- Measured coverage, semantic adequacy, and overgeneration

Libraries during evaluation

- Word order (Bender & Flickinger 2005, Fokkens 2010)
- Morphotactics (O'Hara 2008, ~~Goodman & Bender 2010~~)
- Case (+ direct-inverse marking) (Drellishak 2009)
- Agreement (person, number, gender) (Drellishak 2009)
- Tense and aspect (Poulson 2011)
- Sentential negation* (Bender & Flickinger 2005, ~~Crowgey 2012~~)
- Coordination (Drellishak & Bender 2005)
- Matrix yes-no questions* (Bender & Flickinger 2005)
- ~~Argument optionality (pro-drop) (Saleem & Bender 2010)~~
- ~~Information structure (Song forthcoming)~~

Grammar size

Table 5 Definitions of lexical types, items and rules, by language

Language	Lexical types	Stems	Morph. slots	Lexical rules	Total choices
Abkhaz	16	17	16	47	557
Chemehuevi	14	18	12	18	349
Hausa	18	25	4	17	434
Jingulu	14	12	16	29	391
Malayalam	19	22	16	20	337
Nkore-Kiga	6	9	13	30	436
West Greenlandic	9	15	14	33	545

(Bender et al 2010:61)

Preliminary results (pre-refinement)

Table 6 Results for preliminary choices files

Language	Coverage (%)	Overgeneration (%)	Ambiguous examples (%)
Abkhaz	72.2	11.5	8.3
Chemehuevi	31.0	0	6.9
Hausa	2.6	0	0
Jingulu	57.1	7.7	0
Malayalam	25.6	5.6	0
Nkore-Kiga	0	0	0
West Greenlandic	6.1	0	3.0

(Bender et al 2010:61)

Final results (after refinement)

Table 7 Results for final choices files

Language	Coverage (%)		Overgeneration (%)	Spurious ambiguity (%)	Average readings
	Raw	Treebanked			
Abkhaz	100	94.4	0	2.8	1.08
Chemehuevi	82.8	75.9	0	3.4	1.04
Hausa	42.1	36.8	6.7	5.3	1.31
Jingulu	100	100	0	46.7	2.00
Malayalam	89.7	87.2	2.8	2.8	1.09
Nkore-Kiga	78.6	78.6	11.5	0	1.00
West Greenlandic	93.9	93.9	0	0	1.00

(Bender et al 2010:62)

Phenomenon-by-phenomenon analysis

Table 9 Phenomena evaluation for each language

Phenomenon	abk	hau	jig	kal	mal	nyn	ute
Negation	+	−	+	+	+	+	+/−
Yes–no questions	+	−	+	+	+	+	−
Word order	+/−	+/−	+	+	+	−	−
N/NP coordination	+/−	+/−		−	+/−	+/−	+
S coordination			+	−	−	+	+
V/VP coordination		+/−			−	−	−
Determiners/definiteness	−	−			+		
Tense/aspect	+	+/−	+	+	+	+	+
Auxiliaries		+/−	+			+	
Morphology	+	+	+/−	+	+	+	+/−
Case			+	+	+/−		+
Verb subject agreement	+	+	+	+		+	+
Verb object agreement	+		+	+		+	+
Person	+	+	+	+	+	+	+
Number	+	+	+	+	+	+	+/−
Gender	+	+	+	+	+	+	+

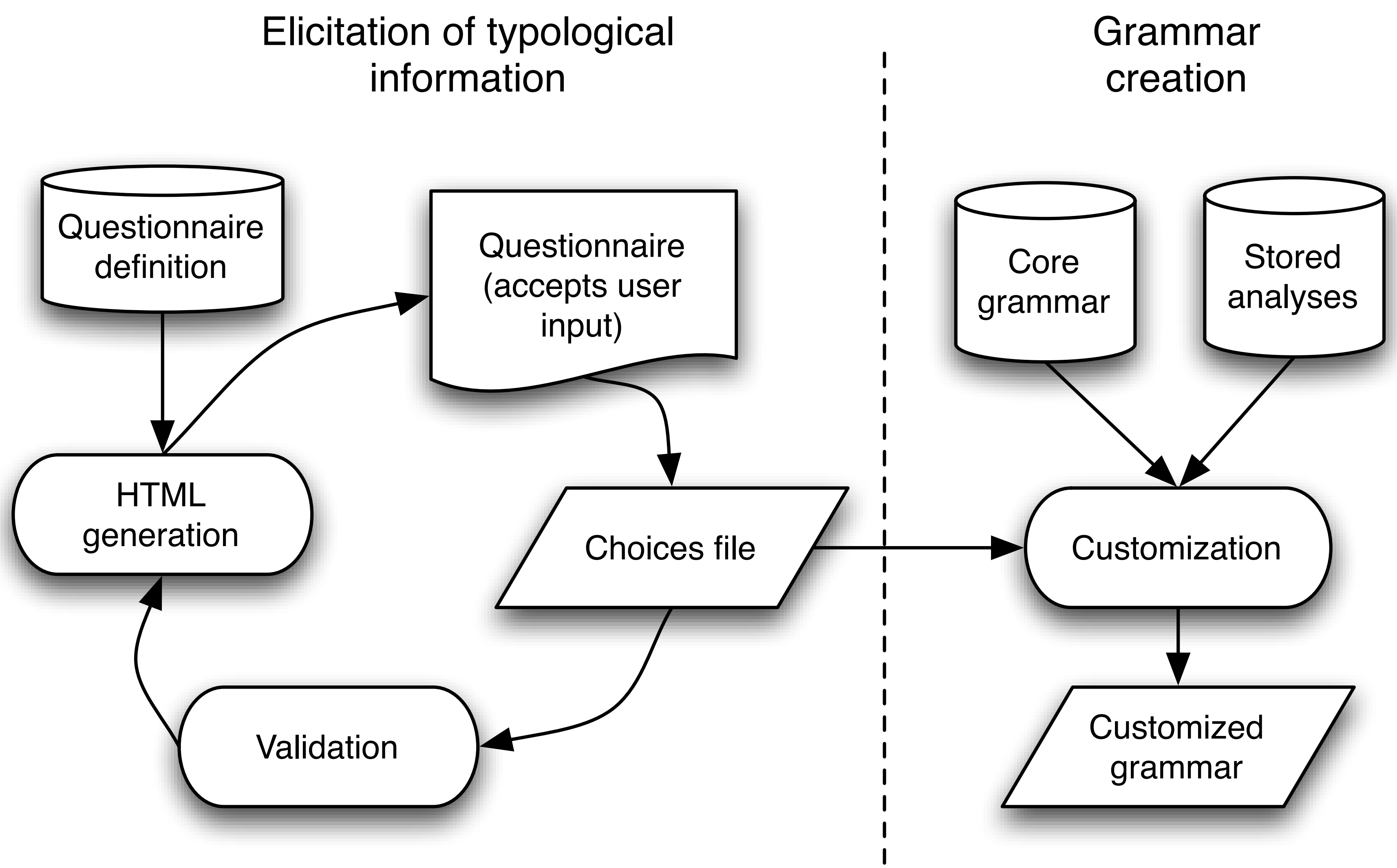
(Bender et al 2010:64)

How to make a library

1. Delineate a phenomenon
2. Survey the typological literature: How is this phenomenon expressed across the world's languages?
3. Review the syntactic literature for analyses of the phenomenon in its various guises
4. Design target semantic representations
5. Develop HPSG analyses for each variant and implement in tdl
6. Decide what information is required from the user to select the right analysis, and extend questionnaire accordingly
7. Extend customization script to add tdl based on questionnaire answers
8. Add regression tests documenting functionality

LinGO Grammar Matrix Customization System

(Bender & Flickinger 2005, Bender et al 2010)



How to make a library

1. Delineate a phenomenon
2. Survey the typological literature: How is this phenomenon expressed across the world's languages?
3. Review the syntactic literature for analyses of the phenomenon in its various guises
4. Design target semantic representations
5. Develop HPSG analyses for each variant and implement in tdl
6. Decide what information is required from the user to select the right analysis, and extend questionnaire accordingly
7. Extend customization script to add tdl based on questionnaire answers
8. Add regression tests documenting functionality

How to evaluate a library

- Pseudo-languages
- Illustrative languages
- Held-out languages
- Test suites
- Choices files
- Error analysis

How to make a library

1. Delineate a phenomenon
2. Survey the typological literature: How is this phenomenon expressed across the world's languages?
3. Review the syntactic literature for analyses of the phenomenon in its various guises
4. Design target semantic representations
5. Develop HPSG analyses for each variant and implement in tdl
6. Decide what information is required from the user to select the right analysis, and extend questionnaire accordingly
7. Extend customization script to add tdl based on questionnaire answers
8. Add regression tests documenting functionality
9. Add prose documenting how to use

Grammar customization: Summary

- The customization system extends the usefulness of the Grammar Matrix
  - Generate working grammar fragments
  - Allows code re-use for non-universals
- The customization system achieves “language independence” by being linguistically informed (cf. Bender 2011)
- Grammar generation is convenient, especially for lexical rules
- The customization system maps a relatively simple encoding (the choices file) to a complex object (a grammar)

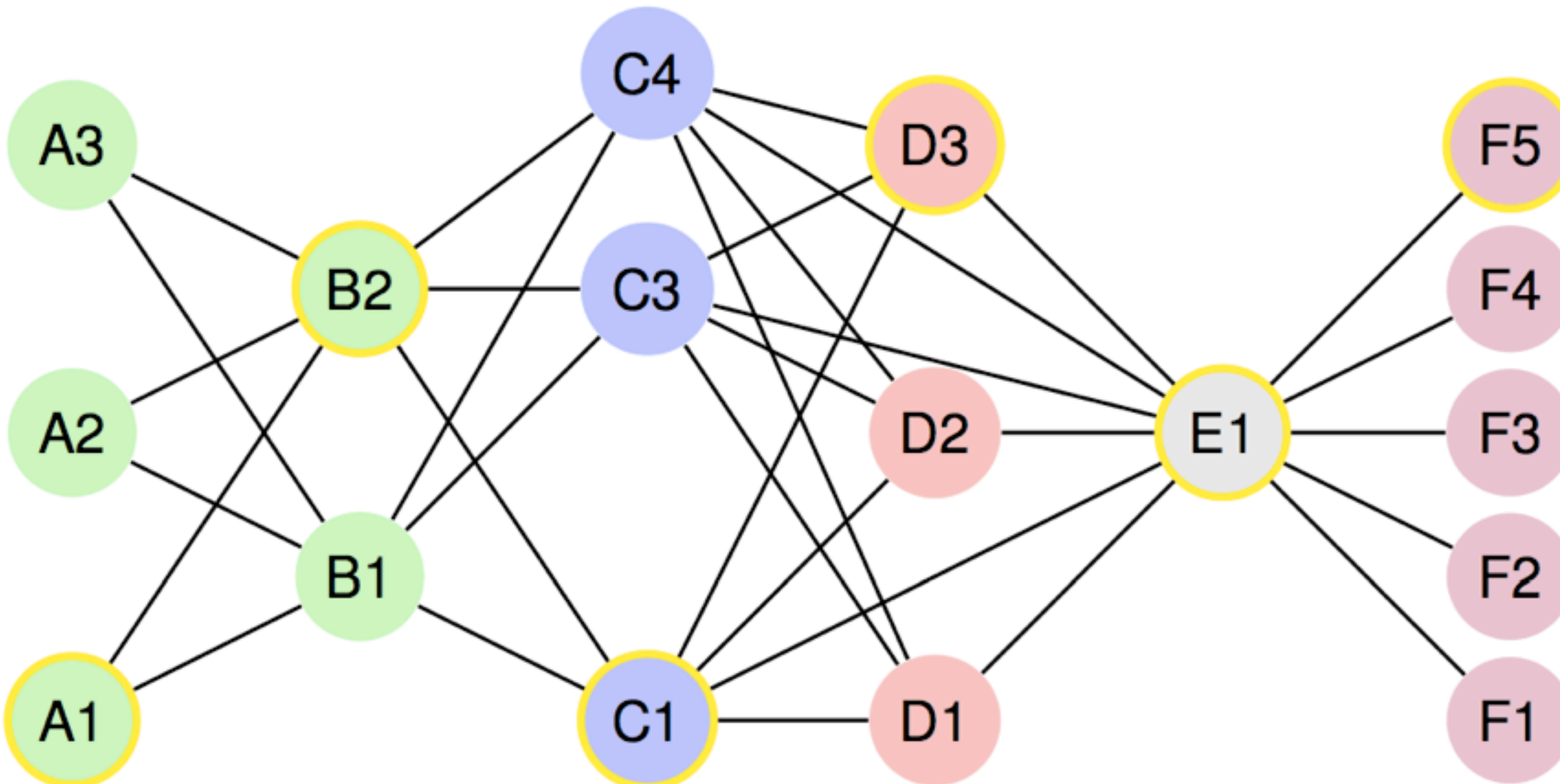
Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

CLIMB (Fokkens 2011): Motivations

- For any given phenomenon, the available data usually underdetermine the analysis
- Analyses of interacting phenomena can mutually constrain each other
    - It is a strength of the grammar engineering approach to syntax that computer implementation makes these interactions discoverable (Bender 2008)
    - ... but a weakness that early analytical decisions inform later ones but not vice versa
- Fokkens (2011) suggests using grammar customization to keep multiple analyses of each phenomenon in play

CLIMB



(Fokkens 2012 - slides)

CLIMB: Methodological characteristics

- The Grammar Matrix customization system focuses on broad typological coverage (at the cost of details of analyses)
- CLIMB work so far on single languages or clusters of closely related languages; explores detailed analyses, with alternatives
- Unlike direct editing of tdl, CLIMB encourages phenomenon-based organization of (meta-)grammar code
- May actually speed up grammar engineering process
- Current developments: tool support for “declarative CLIMB” and “short CLIMB”, adding CLIMB to existing broad-coverage grammars

Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

AGGREGATION: Motivations

- Goal: Combine two rich sources of linguistic information to automatically create precision grammars

- IGT (interlinear glossed text) encodes a lot of information

- Grammar customization maps relatively simple encoding of information (a choices file) to a complex object

- Is the choices file simple enough that it could be automatically generated on the basis of IGT?

IGT: Interlinear glossed text

- Three-line format:
  - Source language (possibly with morpheme segmentations)
  - Morpheme-by-morpheme gloss (target language lemmas + grams)
  - Free translation into target language
- Collected by ODIN from pdfs on the web (Lewis 2006, Lewis & Xia 2010)
- Produced by field linguists doing documentary and descriptive linguistics
  - This is hard work!

IGT as a source of information

- What can you tell about Turkish from this example?

  Ebeveynler çocuklarına meyve yedirtmediler.

- What can you tell about Turkish from this example? (ex from Bender & Lascarides, forthcoming)

  Ebeveyn-ler çocuk-lar-ına meyve yedir-t-me-di-ler.
  Parent-PL child-PL-DAT fruit eat-CAUSE-NEG-PST-3PL

  1. 'The parents did not make (or force) the kids to eat the fruit.'
  2. 'The parents made the kids not eat the fruit.' [tur]

- What if we had 100 or 1000 or 10000 such examples?

Xia & Lewis 2007: Projecting information from translation to source

- Human (linguist) readers recover structure implicit in the translation line and map it onto the source line using the gloss line.

- Could a computer do the same?

  1. Align source line to gloss line (easy: one-to-one, if IGT is clean)

  2. Align gloss line to translation line (easish: lemmas match)

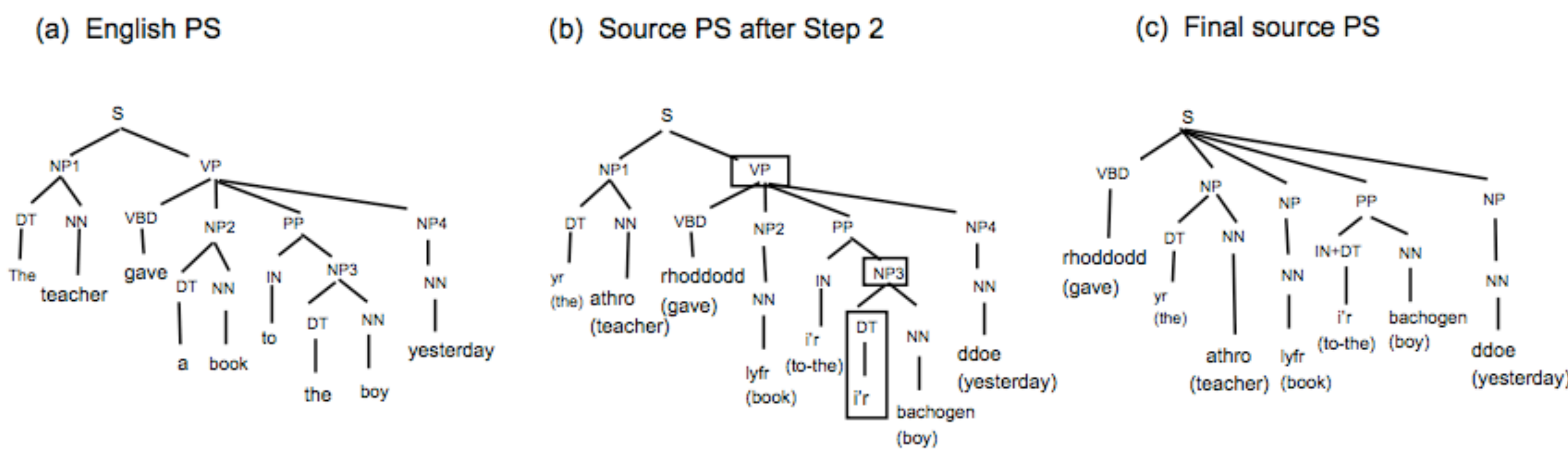
  3. Parse translation line (English, resource rich)

  4. Project parse onto source line

  5. Extract information from parsed structure (and possibly aggregate)

Projection (Xia & Lewis 2007)

Rhoddodd yr athro lyfr i'r bachgen ddoe
gave-3sg the teacher book to-the boy yesterday
"The teacher gave a book to the boy yesterday" [cym]
(Bailyn, 2001)



(Xia & Lewis 2007:454)

Learning word order (Lewis & Xia 2008)

- Project structure, adding function tags (NP-SBJ, NP-OBJ)
- Extract CFGs
- Look at predominant order of NP-SBJ + VP (or V) and NP-OBJ + VP (or V)

Table 5: Word Order Accuracy for 97 languages

# of IGT instances	Average Accuracy
100+	100%
40-99	99%
10-39	79%
5-9	65%
3-4	44%
1-2	14%

(Lewis & Xia 2008: 689)

Could we fill out the whole Grammar Matrix questionnaire?

- Constituent ordering and presence/absence of constituents
  - Extend methodology of Xia & Lewis 2007 to handle notions like flexible word order, detection and placement of auxiliaries
- Morphosyntactic features: what's marked morphologically in a language?
  - Interpret grams based with reference to GOLD (Farrar & Langendoen 2003)
- Lexical classes and their instances:
  - Cluster words based on information in IGT enriched with projection
  - Import directly from field linguists' lexicons (Bender et al 2012b)

Could we fill out the whole Grammar Matrix questionnaire?

- Lexical rules: Affix form, morphosyntactic/morphosemantic features, ordering, and co-occurrence restrictions
  - Alignment between source & gloss lines, unsupervised morphological segmentation, and gram interpretation (again, with GOLD) (Bender et al 2012a)
- Morphosyntatic systems:
  - Reasoning over grams and structural information, example: case system

Extracting morphological rules: Case study of one French verb (Bender et al 2012a)

- All 15,685 forms of the French verb *faire* (Olivier Bonami, pc)

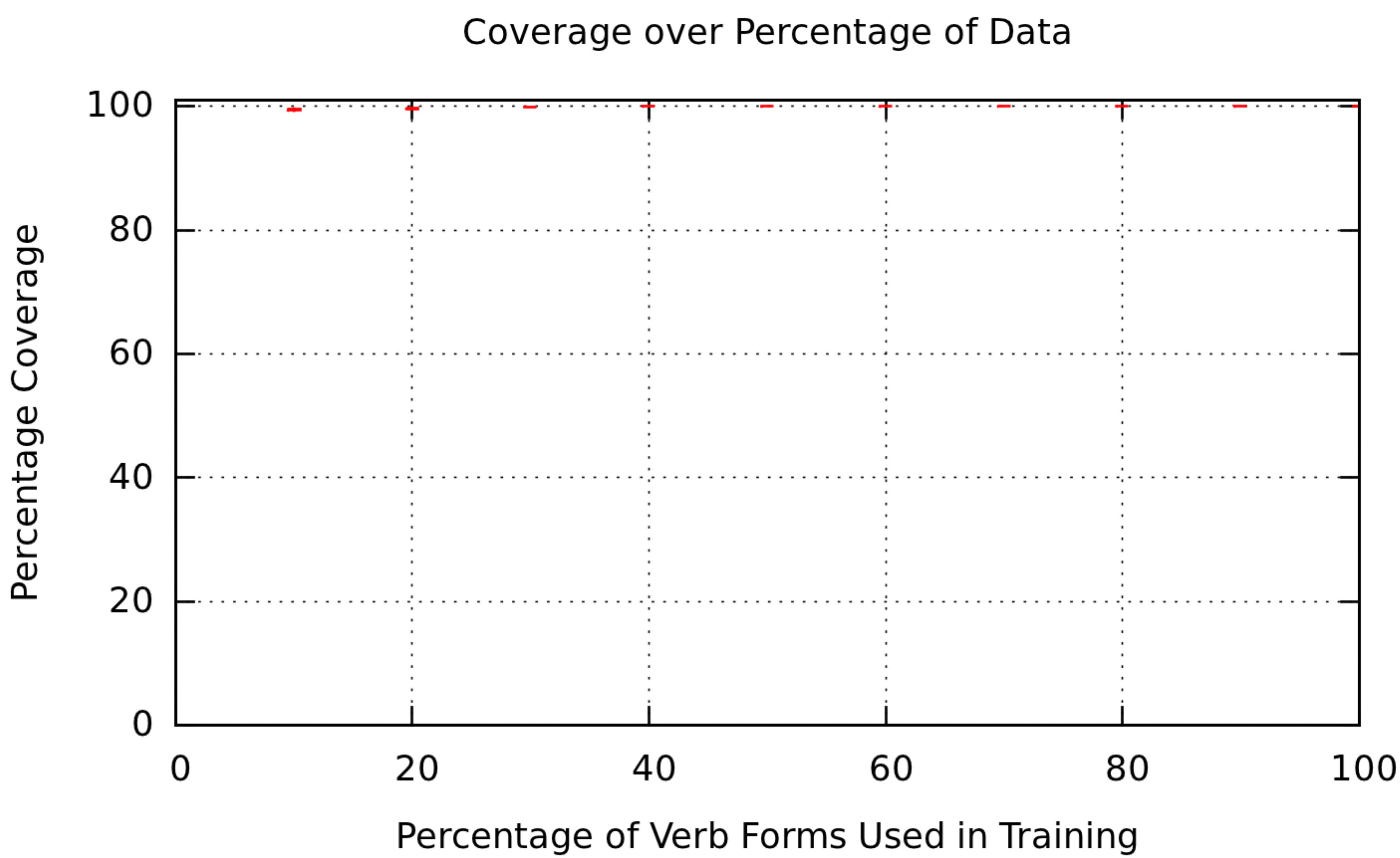
  ʒ(ə)-n(ə)-fɛ-(z)

  NOM.1SG-NEG-do.PRS-1SG

  ‘do’

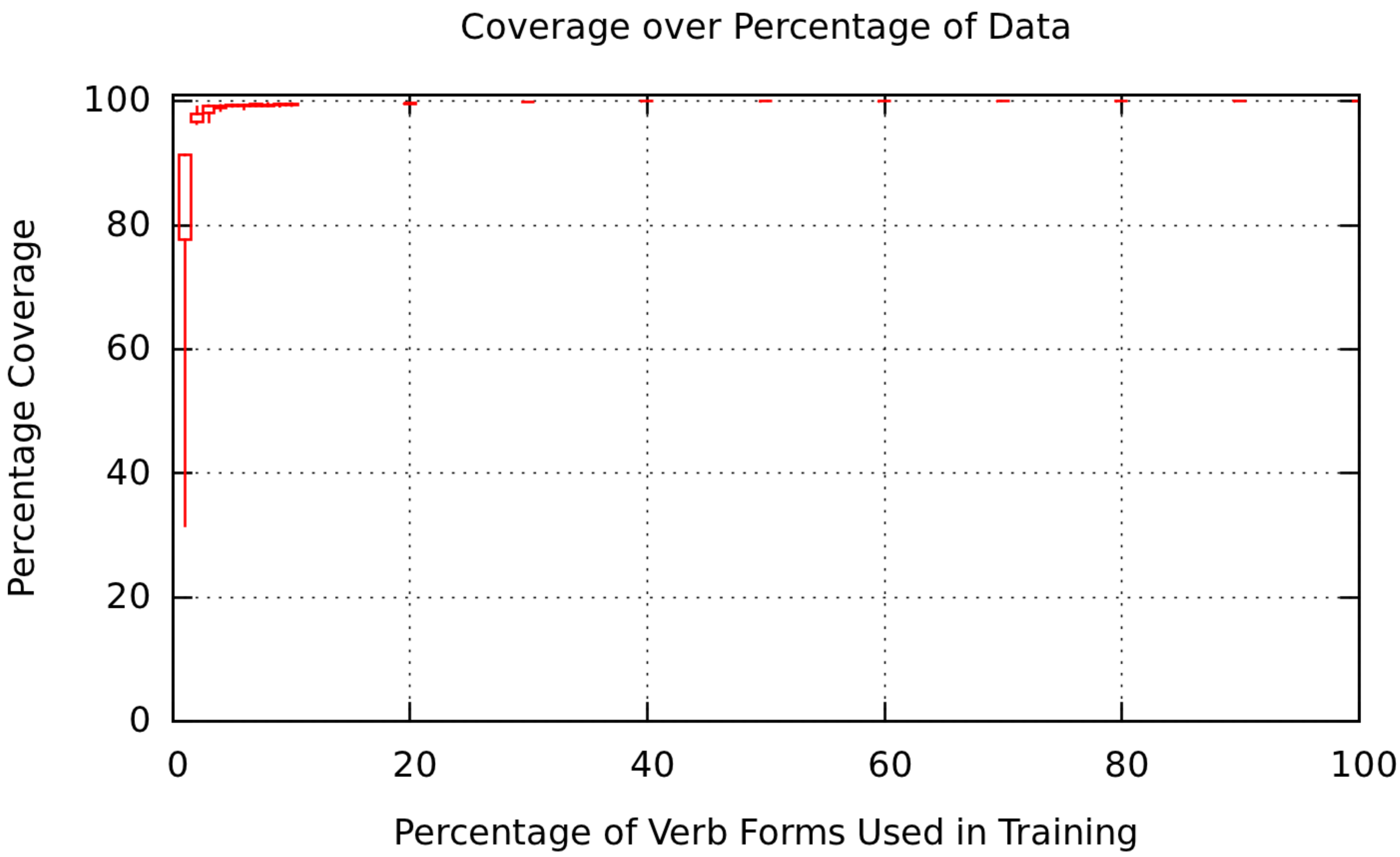
- 12,212 constructed incorrect forms built with same morphemes, but in the wrong sequence or with repetitions

Initial results

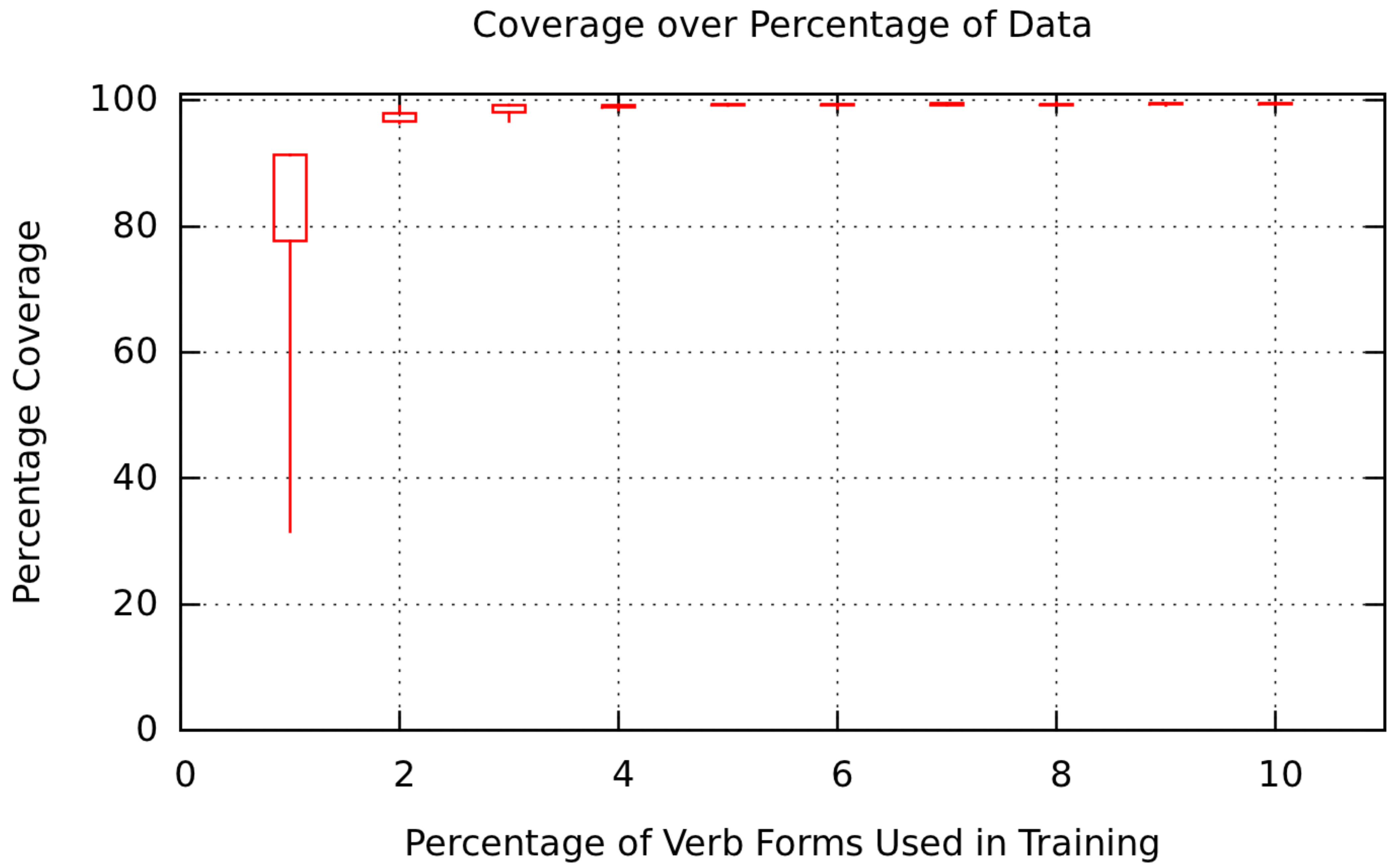


Overgeneration < 5%

Further investigation



Detailed view 0-10%



French case study: take-aways

- In order to learn morpheme order, even a tiny fraction of the full paradigm is enough
- Clean IGT is very nice to work with
- Next steps: Noisier IGT, gram interpretation

Learning case systems

None
(Verbal argument roles are determined only by word order, by intonation, or pragmatically.)

Nominative-accusative
S and A take a case named the ______ (e.g. nominative, subjective)
O takes a case named the ______ (e.g. accusative, objective)
*

Ergative-absolutive
A takes a case named the ______ (e.g. ergative, relative, narrative)
S and O take a case named the ______ (e.g. absolutive, nominative)
*

Tripartite
S takes a case named the ______ (e.g. nominative, subjective)
A takes a case named the ______ (e.g. ergative, agentive)
O takes a case named the ______ (e.g. absolutive, patientive)
*

Split-S
(The S argument of some intransitive verbs is marked by the same case as the agent of transitives, while for other verbs the S argument is marked by the sar
patient.)
A takes a case named the ______ (e.g. ergative, agentive)
O takes a case named the ______ (e.g. absolutive, patientive)
*

Fluid-S
(The S argument of some intransitive verbs is marked by the same case as the agent of transitives, while for other verbs the S argument is marked by the sar
patient, and for still other verbs the S argument can be marked by either case, depending on pragmatic factors (e.g. whether the S is perceived as being in co
action.)
A takes a case named the ______ (e.g. ergative, agentive)

Ta

A, O and S

- A = agent-like argument of two-argument verb
- O = patient-like argument of two-argument verb
- S = sole argument of one-argument verb
- Different case systems involve different alignments of these three
    - Is S treated like A? O? neither? variably like both?

Proposed process

- Find sample of two-argument verbs, with both arguments nouny and overt
- Find a sample of one-argument verbs, with sole argument nouny and overt
- Extract grams from each argument: could be on head noun, determiners, adjectives, adpositions
- Discard grams known to not involve case (e.g., SG, DEF, 1st)
- Assign each argument of two argument verbs to “A” or “O” role
- Discard grams appearing on both A and O arguments (with same tense and person value)
- Compare remaining grams on O and A to S

Answering the questionnaire

- No such grams: No case
- S=A, O is different: Nominative-accusative
- S=O, A is different: Ergative-absolutive
- S,A,O all different: Tripartite
- Some S like A, some like O, depending on verb: Split-S
- Some S like A, some like O, even within the same verb: Fluid-S
- S=A or O depending on noun features (person, pronominality): Split-N
- S=A or O depending on verb features (TMA): Split-V
- Language is Austronesian: Focus case

Data will noisy!

- Combine heuristics described here with stochastic processing
- Bias the system with priors reflecting typological prevalence of different systems

Planned initial tests

- Ling 567 grammars (31 languages):
  - Testsuites and choices files constructed by students
  - Very clean IGT, but small (100-200 examples)
- ODIN data + 567 choices files
- ODIN data + WALS (Haspelmath et al 2008)

Scaling up

- When we can answer the full questionnaire and extract lexical information reliably test by comparing MRS output by customized grammar to translation line in IGT
  - Or even ERG's MRS for that line!
- But: the Grammar Matrix customization system still only covers a small handful of phenomena
- Interesting coverage over collected narratives will require at least:
  - More valence frames for verbs, modification, non-verbal predicates, discourse particles

Overview

- Precision grammars
- The Grammar Matrix
- Grammar customization
- How to make a grammar library
- Extensions 1: CLIMB
- Extensions 2: AGGREGATION

References

Alahverdzhieva, K., & Lascarides, A. (2011). An hpsg approach to synchronous speech and deixis. In S. . Müller (Ed.), *Proceedings of the 18th international conference on head-driven phase structure grammar (hpsg)* (pp. 6–24). Seattle: CSLI Publications.

Bateman, J. A. (1997). Enabling technology for multilingual natural language generation: the KPML development environment. *Journal of Natural Language Engineering*, *3*, 15–55.

Beaver, D. I., & Clark, B. Z. (2008). *Sense and sensitivity: How focus determines meaning.* Wiley-Blackwell.

Bender, E. M. (2008). Grammar engineering for linguistic hypothesis testing. In N. Gaylord, A. Palmer, & E. Ponvert (Eds.), *Proceedings of the texas linguistics society x conference: Computational linguistics for less-studied languages* (pp. 16–36). Stanford: CSLI Publications.

Bender, E. M. (2011). On achieving and evaluating language independence in NLP. *Linguistic Issues in Language Technology*, *6*, 1–26.

Bender, E. M., Drellishak, S., Fokkens, A., Poulson, L., & Saleem, S. (2010). Grammar customization. *Research on Language & Computation*, 1-50. Available from `http://dx.doi.org/10.1007/s11168-010-9070-1` (10.1007/s11168-010-9070-1)

Bender, E. M., & Flickinger, D. (2005). Rapid prototyping of scalable grammars: Towards modularity in extensions to a language-independent core. In *Proceedings of the 2nd international joint conference on natural language processing ijcnlp-05 (posters/demos).* Jeju Island, Korea.

Bender, E. M., Flickinger, D., & Oepen, S. (2002). The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In J. Carroll, N. Oostdijk, & R. Sutcliffe (Eds.), *Proceedings of the workshop on grammar engineering and evaluation at the 19th international conference on computational linguistics* (pp. 8–14). Taipei, Taiwan.

Bender, E. M., Schikowski, R., & Bickel, B. (2012). Deriving a lexicon for a precision grammar from language documentation resources: A case study of Chintang. In *Proceedings of COLING 2012.*

Bender, E. M., Wax, D., & Goodman, M. W. (2012). From IGT to precision grammar: French verbal morphology. In *Lsa annual meeting extended abstracts 2012.*

Bergmair, R. (2008). Monte Carlo semantics: McPIET at RTE4. In *Text analysis conference (tac 2008) workshop-rte-4 track. national institute of standards and technology.*

Bresnan, J., & Mchombo, S. A. (1995). The lexical integrity principle: Evidence from Bantu. *Natural Language & Linguistic Theory*, *13*, 181-254. Available from `http://dx.doi.org/10.1007/BF00992782` (10.1007/BF00992782)

Butt, M., Dyvik, H., King, T. H., Masuichi, H., & Rohrer, C. (2002). The parallel grammar project. In J. Carroll, N. Oostdijk, & R. Sutcliffe (Eds.), *Proceedings of the workshop on grammar engineering and evaluation at the 19th international conference on computational linguistics* (pp. 1–7).

Callmeier, U. (2002). Preprocessing and encoding techniques in pet. In S. Oepen,

D. Flickinger, J. Tsujii, & H. Uszkoreit (Eds.), *Collaborative language engineering. A case study in efficient grammar-based processing.* Stanford, CA: CSLI Publications.

Copestake, A. (2002a). Definitions of typed feature structures. In S. Oepen, D. Flickinger, J. Tsujii, & H. Uszkoreit (Eds.), *Collaborative language engineering* (pp. 227–230). Stanford, CA: CSLI Publications.

Copestake, A. (2002b). *Implementing typed feature structure grammars.* Stanford, CA: CSLI Publications.

Copestake, A., Flickinger, D., Pollard, C., & Sag, I. A. (2005). Minimal recursion semantics: An introduction. *Research on Language & Computation*, *3*(4), 281–332.

Crowgey, J. (2012). *The syntactic exponence of negation: A model for the LinGO grammar matrix.* Unpublished master's thesis, University of Washington.

Doran, C., Egedi, D., Hockey, B. A., Srinivas, B., & Zaidel, M. (1994). XTAG system: A wide coverage grammar for English. In *Proceedings of the 15th conference on computational linguistics - volume 2* (pp. 922–928). Stroudsburg, PA, USA: Association for Computational Linguistics. Available from `http://dx.doi.org/10.3115/991250.991297`

Drellishak, S. (2009). *Widespread but not universal: Improving the typological coverage of the Grammar Matrix.* Unpublished doctoral dissertation, University of Washington.

Drellishak, S., & Bender, E. M. (2005). A coordination module for a crosslinguistic grammar resource. In S. Müller (Ed.), *The proceedings of the 12th international conference on head-driven phrase structure grammar, department of informatics, university of lisbon* (pp. 108–128). Stanford: CSLI Publications. Available from `http://cslipublications.stanford.edu/HPSG/6/`

Dryer, M. S. (2011). Negative morphemes. In M. S. Dryer & M. Haspelmath (Eds.), *The world atlas of language structures online.* Munich: Max Planck Digital Library. Available from `http://wals.info/feature/112A`

Farrar, S., & Langendoen, T. (2003). A linguistic ontology for the semantic web. *Glot International*, *7*, 97–100.

Flickinger, D. (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering, 6 (1) (Special Issue on Efficient Processing with HPSG)*, 15–28.

Flickinger, D. (2011). Accuracy v. robustness in grammar engineering. In E. M. Bender & J. E. Arnold (Eds.), *Language from a cognitive perspective: Grammar, usage and processing* (pp. 31–50). Stanford, CA: CSLI Publications.

Fokkens, A. (2011, June). Metagrammar engineering: Towards systematic exploration of implemented grammars. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 1066–1076). Portland, Oregon, USA: Association for Computational Linguistics. Available from `http://www.aclweb.org/anthology/P11-1107`

Fokkens, A. S. (2010). *Documentation for the Grammar Matrix word order library* (Tech. Rep.). Saarland University.

Frank, A., Krieger, H., Xu, F., Uszkoreit, H., Crysmann, B., Jörg, B., et al. (2007). Question answering from structured knowledge sources. *Journal of Applied Logic, Special Issue* *on Questions and Answers: Theoretical and Applied Perspectives*, *5*, 20–48.

Goodman, M. W., & Bender, E. M. (2010). *What's in a word? redefining the morphotactic infrastructure in the LinGO Grammar Matrix customization system.* (Poster presented at the Morphology and Formal Grammar Workshop at HPSG 2010)

Götz, T., & Meurers, W. D. (1997). The ConTroll system as large grammar development platform. In *Proceedings of the Workshop on Computational Environments for Grammar Development and Linguistic Engineering* (pp. 38 – 45). Madrid, Spain.

Haspelmath, M., Dryer, M. S., Gil, D., & Comrie, B. (Eds.). (2008). *The world atlas of language structures online.* Munich: Max Planck Digital Library. (http://wals.info)

Hellan, L., & Haugereid, P. (2003). NorSource: An exercise in Matrix grammar-building design. In E. M. Bender, D. Flickinger, F. Fouvry, & M. Siegel (Eds.), *Proceedings of the workshop on ideas and strategies for multilingual grammar development, esslli 2003* (pp. 41–48). Vienna, Austria.

Kordoni, V., & Neu, J. (2005). Deep analysis of Modern Greek. In K.-Y. Su, J. Tsujii, & J.-H. Lee (Eds.), *Lecture notes in computer science* (Vol. 3248, pp. 674–683). Berlin: Springer-Verlag.

Marimon, M., Bel, N., & Seghezzi, N. (2007). Test-suite construction for a Spanish grammar. In T. H. King & E. M. Bender (Eds.), *Proceedings of the geaf 2007 workshop*. Stanford, CA: CSLI Publications.

Nichols, E., Bond, F., Tanaka, T., Fujita, S., & Flickinger, D. (2006, July). Multilingual ontology acquisition from multiple MRDs. In *Proceedings of the 2nd workshop on ontology learning and population: Bridging the gap between text and knowledge* (pp. 10–17). Sydney, Australia: Association for Computational Linguistics. Available from `http://www.aclweb.org/anthology/W/W06/W06-0502.pdf`

Oepen, S. (2001). [incr tsdb()] — *Competence and performance laboratory. User manual* (Technical Report). Saarbrücken, Germany: Computational Linguistics, Saarland University.

Oepen, S., Velldal, E., Lønning, J. T., Meurer, P., Rosn, V., & Flickinger, D. (2007). Towards hybrid quality-oriented machine translation. On linguistics and probabilities in MT. In *The 11th international conference on theoretical and methodological issues in machine translation (tmi-07)*. Skövde, Sweden.

O'Hara, K. (2008). *A morphotactic infrastructure for a grammar customization system.* Unpublished master's thesis, University of Washington.

Pollard, C., & Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar.* Chicago, IL and Stanford, CA: The University of Chicago Press and CSLI Publications.

Poulson, L. (n.d.). Meta-modeling of tense and aspect in a cross-linguistic grammar engineering platform. *UW Working Papers in Linguistics*, *28*.

Saleem, S., & Bender, E. M. (2010, August). Argument optionality in the lingo grammar matrix. In *Coling 2010: Posters* (pp. 1068–1076). Beijing, China: Coling 2010 Organizing Committee. Available from `http://www.aclweb.org/anthology/C10-2123`

Schlangen, D. (2003). *A coherence-based approach to the interpretation of non-sentential utterances in dialogue.* Unpublished doctoral dissertation, University of Edinburgh.

Siegel, M., & Bender, E. M. (2002). Efficient deep processing of Japanese. In *Proceedings of the 3rd workshop on asian language resources and international standardization at the 19th international conference on computational linguistics.* Taipei, Taiwan.

Suppes, P., Flickinger, D., Macken, B., Cook, J., & Liang, T. (2012). Description of the EPGY Stanford University online courses for mathematics and language arts. In *International society for technology in education annual (iste) 2012 conference.* San Diego CA.

Toutanova, K., Manning, C. D., Flickinger, D., & Oepen, S. (2005). Stochastic HPSG parse disambiguation using the Redwoods corpus. *Research on Language & Computation*, *3*(1), 83–105.

Velldal, E. (2008). *Empirical realization ranking.* Unpublished doctoral dissertation, University of Oslo, Department of Informatics.

Xia, F., & Lewis, W. (2007). Multilingual structural projection across interlinearized text. In *Naacl-hlt 2007.* Rochester, NY.

Zhang, Y., & Kordoni, V. (2006). Automated deep lexical acquisition for robust open texts processing. In *Proceedings of the fifth international conference on language resources and evaluation (lrec 2006)* (pp. 275–280). Available from `http://hnk.ffzg.hr/bibl/lrec2006/`

Zhang, Y., & Krieger, H.-U. (2011, October). Large-scale corpus-driven PCFG approximation of an HPSG. In *Proceedings of the 12th international conference on parsing technologies* (pp. 198–208). Dublin, Ireland: Association for Computational Linguistics. Available from `http://www.aclweb.org/anthology/W11-2923`