Assembling Syntax: Modeling Constituent Questions in a Grammar Engineering Framework

Olga Zamaraeva

Department of Linguistics, University of Washington
Seminar on the Interactions between Formal and Computational Linguistics

June 1 2021

Part I: Introduction. Formal or Computational?

Rafael's School of Athens (*circa* 1510)

- Common point of view:
  - CL is a statistical field which could interact with formal linguistic theory more
  - "computational" = "statistical"

Part I: Introduction. Formal or Computational?

Rafael's School of Athens (*circa* 1510)

- Common point of view:
  - CL is a statistical field which could interact with formal linguistic theory more
  - "computational" = "statistical"
- Computational linguistics can be formal **and** statistical (not an "or")

Part I: Introduction. Formal or Computational?

Rafael's School of Athens (*circa* 1510)

- Common point of view:
  - CL is a statistical field which could interact with formal linguistic theory more
  - "computational" = "statistical"
- Computational linguistics can be formal **and** statistical (not an "or")
- **Any field** can!

Part I: Introduction. Formal or Computational?

Rafael's School of Athens (*circa* 1510)

- Common point of view:
  - CL is a statistical field which could interact with formal linguistic theory more
  - "computational" = "statistical"
- Computational linguistics can be formal **and** statistical (not an "or")
- **Any field** can!
- Formal is to statistical how rationalist is to empiricist

Part I: Introduction. Formal or Computational?

Rafael's School of Athens (*circa* 1510)

- Common point of view:
  - CL is a statistical field which could interact with formal linguistic theory more
  - "computational" = "statistical"
- Computational linguistics can be formal **and** statistical (not an "or")
- **Any field** can!
- Formal is to statistical how rationalist is to empiricist
- Rationalist is to empiricist how Idealist is to Materialist

Part I: Introduction. Formal or Computational?

Rafael's School of Athens (*circa* 1510)

- Common point of view:
  - CL is a statistical field which could interact with formal linguistic theory more
  - "computational" = "statistical"
- Computational linguistics can be formal **and** statistical (not an "or")
- **Any field** can!
- Formal is to statistical how rationalist is to empiricist
- Rationalist is to empiricist how Idealist is to Materialist
- **Ancient debate with no right answer**

Big research question

- What is the range of variation in human languages? 

Big research question

- What is the range of variation in human languages? 
- Many ways to approach and many relevant fields:

Big research question

- What is the range of variation in human languages? 
- Many ways to approach and many relevant fields:
  - Typology

Big research question

- What is the range of variation in human languages? 
- Many ways to approach and many relevant fields:
  - Typology
  - Syntactic, semantic, phonological, acquisition, and other theories

Big research question

- What is the range of variation in human languages? 
- Many ways to approach and many relevant fields:
  - Typology
  - Syntactic, semantic, phonological, acquisition, and other theories
  - Computational linguistics (formal and statistical)

Big research question

- What is the range of variation in human languages? 🌎
- Many ways to approach and many relevant fields:
  - Typology
  - Syntactic, semantic, phonological, acquisition, and other theories
  - Computational linguistics (formal and statistical)
  - Natural language processing

Big research question

- What is the range of variation in human languages? 🌎
- Many ways to approach and many relevant fields:
  - Typology
  - Syntactic, semantic, phonological, acquisition, and other theories
  - Computational linguistics (formal and statistical)
  - Natural language processing
- All of these fields are probably invested in understanding the range of variation in human language

Big research question

- What is the range of variation in human languages? 
- Many ways to approach and many relevant fields:
  - Typology
  - Syntactic, semantic, phonological, acquisition, and other theories
  - Computational linguistics (formal and statistical)
  - Natural language processing
- All of these fields are probably invested in understanding the range of variation in human language
- Today: Computational syntax with HPSG
  - Typologically informed

Natural Language Processing and CL

- CL may mean "answering linguistic questions with computational means"
  - E.g. Typological correlations
  - E.g. Grammar coverage and overgeneration
  - E.g. Lexical class membership

Natural Language Processing and CL

- CL may mean "answering linguistic questions with computational means"
  - E.g. Typological correlations
  - E.g. Grammar coverage and overgeneration
  - E.g. Lexical class membership
- NLP (often also called CL!) may or may not target "linguistic questions"

Natural Language Processing and CL

- CL may mean "answering linguistic questions with computational means"
  - E.g. Typological correlations
  - E.g. Grammar coverage and overgeneration
  - E.g. Lexical class membership
- NLP (often also called CL!) may or may not target "linguistic questions"
  - Learn models from data

Natural Language Processing and CL

- CL may mean "answering linguistic questions with computational means"
  - E.g. Typological correlations
  - E.g. Grammar coverage and overgeneration
  - E.g. Lexical class membership
- NLP (often also called CL!) may or may not target "linguistic questions"
  - Learn models from data
  - Use models to:

Natural Language Processing and CL

- CL may mean "answering linguistic questions with computational means"
  - E.g. Typological correlations
  - E.g. Grammar coverage and overgeneration
  - E.g. Lexical class membership
- NLP (often also called CL!) may or may not target "linguistic questions"
  - Learn models from data
  - Use models to:
    - Perform language tasks

Natural Language Processing and CL

- CL may mean "answering linguistic questions with computational means"
  - E.g. Typological correlations
  - E.g. Grammar coverage and overgeneration
  - E.g. Lexical class membership
- NLP (often also called CL!) may or may not target "linguistic questions"
  - Learn models from data
  - Use models to:
    - Perform language tasks
    - 💡Learn something about **faculties** involved in those tasks

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]

[1] **Look at any SCiL conference program**

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]

[1] **Look at any SCiL conference program**

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]
  - For performing tasks:
    - Can we just transfer from high-resource?

[1] **Look at any SCiL conference program**

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]
  - For performing tasks:
    - Can we just transfer from high-resource?
    - From an **idealist/rationalist** 🏛 standpoint:

[1] **Look at any SCiL conference program**

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]
  - For performing tasks:
    - Can we just transfer from high-resource?
    - From an **idealist/rationalist** 🏛 standpoint:
    - Systematic knowledge is **core** to evaluating data-driven models

[1] **Look at any SCiL conference program**

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]
  - For performing tasks:
    - Can we just transfer from high-resource?
    - From an **idealist/rationalist** 🏛 standpoint:
    - Systematic knowledge is **core** to evaluating data-driven models
    - ...need to **reason** about what we've done

[1] **Look at any SCiL conference program**

Does NLP care about range of language variation?

- Linguists (including computational):
  - Finding/describing/analyzing range of variation is a **fundamental** goal
- NLP:
  - For learning about language faculties:
    - Cares similarly to how linguists do[1]
  - For performing tasks:
    - Can we just transfer from high-resource?
    - From an **idealist/rationalist** standpoint:
    - Systematic knowledge is **core** to evaluating data-driven models
    - ...need to **reason** about what we've done
    - PTB, PSD, UD (which were created by people) as **target** representations

[1] **Look at any SCiL conference program**

How to study languages?!

- Linguistic typology

How to study languages?!

- Linguistic typology
  - Study range of variation wrt **broad**, diverse 🌍 characteristics
- Syntactic (semantic, morphological, phonological, discourse...) theory 🏛

How to study languages?!

- Linguistic typology
  - Study range of variation wrt **broad**, diverse 🌍 characteristics
- Syntactic (semantic, morphological, phonological, discourse...) theory 🏛
  - **In-depth** inquiry into modeling phenomena "elegantly"

How to study languages?!

- Linguistic typology
  - Study range of variation wrt **broad**, diverse 🌏 characteristics
- Syntactic (semantic, morphological, phonological, discourse...) theory 🏛️
  - **In-depth** inquiry into modeling phenomena "elegantly"
- Computational 👩‍💻 linguistics
  - formal 🏛️ (**implement and test** theories)
  - statistical 🧮 (learn patterns bottom-up **from data** and generalize)

How to study languages?!

- Linguistic typology
  - Study range of variation wrt **broad**, diverse 🌏 characteristics
- Syntactic (semantic, morphological, phonological, discourse...) theory 🏛
  - **In-depth** inquiry into modeling phenomena "elegantly"
- Computational 👩‍💻 linguistics
  - formal 🏛 (**implement and test** theories)
  - statistical 🧮 (learn patterns bottom-up **from data** and generalize)
- Traditionally separate

How to study languages?!

- Linguistic typology
  - Study range of variation wrt **broad**, diverse 🌏 characteristics
- Syntactic (semantic, morphological, phonological, discourse...) theory 🏛
  - **In-depth** inquiry into modeling phenomena "elegantly"
- Computational 👩‍💻 linguistics
  - formal 🏛 (**implement and test** theories)
  - statistical 🧮 (learn patterns bottom-up **from data** and generalize)
- Traditionally separate
  - Can we combine them?

Combining breadth, depth, ideas, and empiricism

- Computational modeling
  - of the theory
  - ...for reproducibility and rigor
  - with emphasis on data patterns
  - for robustness
  - Goal: assemble fragments of in-depth analyses
  - ...into a typologically diverse system
  - ...which is fully implemented

Computational syntax

- Implement grammars on the computer 👩‍💻

Computational syntax

- Implement grammars on the computer 👩‍💻
- Run grammars automatically on sentences 📄

Computational syntax

- Implement grammars on the computer 👩‍💻
- Run grammars automatically on sentences 📄
  - ...as many sentences as you have 📄📄📄📄

Computational syntax

- Implement grammars on the computer 👩‍💻
- Run grammars automatically on sentences 📄
  - ...as many sentences as you have 📄📄📄📄
  - ...from as many languages as you have data from

Computational syntax

- Implement grammars on the computer 👩‍💻
- Run grammars automatically on sentences 📄
  - ...as many sentences as you have 📄📄📄📄
  - ...from as many languages as you have data from
  - ...including typologically diverse languages 🌏

Computational syntax

- Implement grammars on the computer 👩‍💻
- Run grammars automatically on sentences 📄
  - ...as many sentences as you have 📄📄📄📄
  - ...from as many languages as you have data from
  - ...including typologically diverse languages 🌏
  - ...as many times as you need

Computational syntax

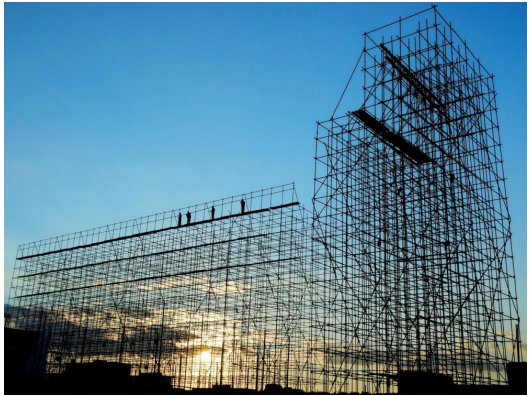
- Implement grammars on the computer 👩‍💻
- Run grammars automatically on sentences 📄
  - ...as many sentences as you have 📄📄📄📄
  - ...from as many languages as you have data from
  - ...including typologically diverse languages 🌏
  - ...as many times as you need
- Grow grammars and accumulate knowledge artifacts

Computational syntax

- Implement grammars on the computer
- Run grammars automatically on sentences
  - ...as many sentences as you have
  - ...from as many languages as you have data from
  - ...including typologically diverse languages
  - ...as many times as you need
- Grow grammars and accumulate knowledge artifacts
- Growing the area of applicability of a set of hypotheses which grammars represent

Philosophy: Method of fragments and Pragmatism

- Fully explicit **grammar fragments**[2] that can be extended
  - ...constitute research artifacts that can be literally built upon
  - ...together and over time, contribute to our understanding of syntax



https://www.theinformationlab.co.uk/2017/08/09/data-scaffolding-easy-steps-fill-missing-data/

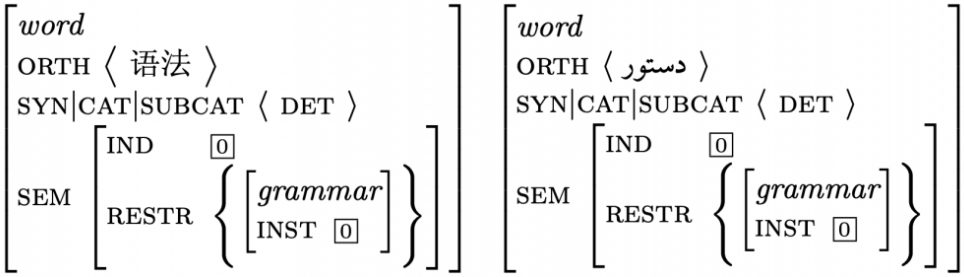
2 Montague 1974

Interim summary for Part I

- Philosophy of science applies to CL like to any field
- Empiricism and Rationalism have been at odds for centuries
- But they continue to walk hand in hand
- Next: Computational syntax
  - Implementing 🖼 formal 🏛 theories

Rafael's School of Athens (*circa* 1510)

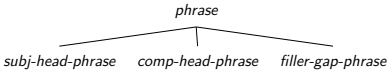
Part II: Formal depth, implemented on computer



https://hpsg.hu-berlin.de/hpsg2020/

Head-Driven Phrase Structure Grammar

- **Fully explicit** formalism[3]
- Lexicalist and surface-oriented
- A grammar is a hierarchy of **types** encoded as **feature structures** where features are constrained to have some values
- A structure licensing a sentence must be **well-formed**

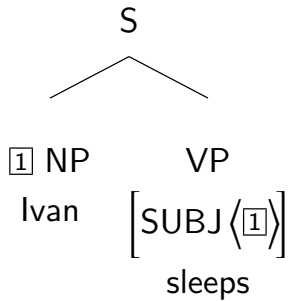


$$\begin{bmatrix} \textit{subj-head-phrase} & \\ \text{SUBJ} & \langle\rangle \\ \text{HEAD-DTR} & \boxed{2}\begin{bmatrix}\text{SUBJ} & \langle\, \boxed{1}\,\rangle\end{bmatrix} \\ \text{NON-HEAD-DTR} & \boxed{1} \\ \text{ARGS} & \langle\, \boxed{1}, \boxed{2}\,\rangle \end{bmatrix}$$

[3] Pollard and I. A. Sag 1994

HPSG Phrase Structure Rule

- Describes a feature structure that is a phrase and can be visualized as a tree
- "Mother" and "daughter" nodes
- Identities (tags)



$$\begin{bmatrix} \textit{subj-head-phrase} & \\ \text{SUBJ} & \langle\rangle \\ \text{HEAD-DTR} & \boxed{2}\begin{bmatrix}\text{SUBJ} & \langle\,\boxed{1}\,\rangle\end{bmatrix} \\ \text{NON-HEAD-DTR} & \boxed{1} \\ \text{ARGS} & \langle\,\boxed{1},\,\boxed{2}\,\rangle \end{bmatrix}$$

HPSG: Theoretical and Computational

- Both types of research exist
- HPSG formalism can be used to posit multiple theories
- DELPH-IN HPSG 🐬
  - International research consortium[4]
  - Restricted version of HPSG formalism

[4] https://github.com/delph-in, http://moin.delph-in.net/wiki/

DELPH-IN main projects

- The English Resource Grammar (ERG) [5]
  - **Broad** coverage; used in NLP[6]
    - Semantic representations (ERS, MRS, DMRS) used widely for **evaluating semantic parsers**[7]

5 Flickinger 2000, 2011

6 Hajdik et al. 2019; Zamaraeva, Howell, and Rhine 2018; Buys and Blunsom 2017; Packard 2014; Bender, Flickinger, Oepen, et al. 2004; Morgado da Costa, Bond, et al. 2016; Morgado da Costa, Winder, et al. 2020

7 Oepen and Flickinger 2019

DELPH-IN main projects

- The English Resource Grammar (ERG) [5]
  - **Broad** coverage; used in NLP[6]
    - Semantic representations (ERS, MRS, DMRS) used widely for **evaluating semantic parsers**[7]

- **Medium**-size grammars of Japanese, Chinese, German, Spanish...[8]

5 Flickinger 2000, 2011

6 Hajdik et al. 2019; Zamaraeva, Howell, and Rhine 2018; Buys and Blunsom 2017; Packard 2014; Bender, Flickinger, Oepen, et al. 2004; Morgado da Costa, Bond, et al. 2016; Morgado da Costa, Winder, et al. 2020

7 Oepen and Flickinger 2019

8 Siegel et al. 2016; Fan 2018; Crysmann 2003; Marimon 2010

DELPH-IN main projects

- The English Resource Grammar (ERG) [5]
  - **Broad** coverage; used in NLP[6]
    - Semantic representations (ERS, MRS, DMRS) used widely for **evaluating semantic parsers**[7]

- **Medium**-size grammars of Japanese, Chinese, German, Spanish...[8]
- The Grammar Matrix:[9] Automated **starter** grammars; **typologically**-driven (Part III)

5 Flickinger 2000, 2011

6 Hajdik et al. 2019; Zamaraeva, Howell, and Rhine 2018; Buys and Blunsom 2017; Packard 2014; Bender, Flickinger, Oepen, et al. 2004; Morgado da Costa, Bond, et al. 2016; Morgado da Costa, Winder, et al. 2020

7 Oepen and Flickinger 2019

8 Siegel et al. 2016; Fan 2018; Crysmann 2003; Marimon 2010

9 Bender, Flickinger, and Oepen 2002; Bender, Drellishak, et al. 2010

DELPH-IN main projects

- The English Resource Grammar (ERG) [5]
  - **Broad** coverage; used in NLP[6]
    - Semantic representations (ERS, MRS, DMRS) used widely for **evaluating semantic parsers**[7]

- **Medium**-size grammars of Japanese, Chinese, German, Spanish...[8]
- The Grammar Matrix:[9] Automated **starter** grammars; **typologically**-driven (Part III)
  - Bootstrap grammar development for more languages[10]

5 Flickinger 2000, 2011

6 Hajdik et al. 2019; Zamaraeva, Howell, and Rhine 2018; Buys and Blunsom 2017; Packard 2014; Bender, Flickinger, Oepen, et al. 2004; Morgado da Costa, Bond, et al. 2016; Morgado da Costa, Winder, et al. 2020

7 Oepen and Flickinger 2019

8 Siegel et al. 2016; Fan 2018; Crysmann 2003; Marimon 2010

9 Bender, Flickinger, and Oepen 2002; Bender, Drellishak, et al. 2010

10 Bender 2010; Crowgey 2019; Inman 2019

Example: First cross-linguistic analysis of constituent (wh) questions in DELPH-IN HPSG[11]

- Classic 🏛 set of questions for syntactic theory:
  - How are question words ("wh-") distributed?
  - How to represent interrogative semantics?
    - Quantification, scope, wh-words as question parameters of different clauses...
  - How to model question word fronting (4)?
  - How to model optional fronting (5)?

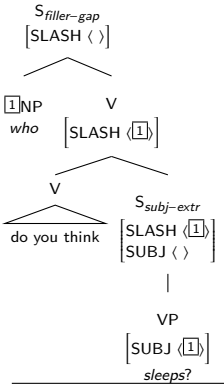
(4) Gde kto chto vidit?
where who.NOM what.ACC see.3SG
'Who sees what where?'[rus]

(5) Ty gde rabotaesh?
you where work.3SG
'Where do you work?'[rus]

11 Zamaraeva 2021

Fronting analysis in HPSG

- "Nonlocal" features are propagated up the tree[12]
- Feature SLASH creates a long-distance dependency at the level of the verb
- The *filler-gap* rule discharges the dependencies



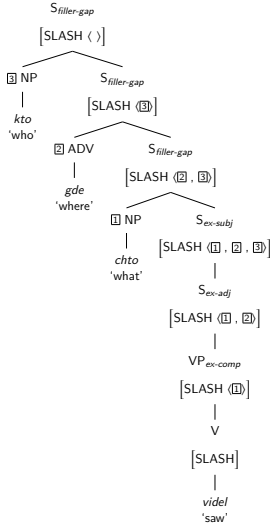
(5) Who do you think sleeps? [eng]

$$\begin{bmatrix} \textit{filler-gap-phrase} & \\ \text{SLASH} & \langle\ \rangle \\ \text{HEAD-DTR} & \begin{bmatrix}\text{SLASH} & \langle\ \boxed{1}\ \rangle\end{bmatrix} \\ \text{NON-HEAD-DTR} & \boxed{1} \end{bmatrix}$$

12 Pollard and I. A. Sag 1994; Ginzburg and I. Sag 2000

Extending the fragment: Multiple question fronting

- Recursive application of *filler-gap*[13]
- Theoretical takeaways arising from computational implementation:
  - "Optional" fronting is hard![14]
    - New evidence from HPSG
  - Simpler analysis of multiple fronting → less simple morphological marking[15]
    - Must a system be "elegant"?



(6) kto gde chto vidit?
who.NOM where what.ACC see.3SG
'Who sees what where?'[rus]

13 Zamaraeva and Emerson 2020; Crysmann 2015

14 Zamaraeva 2021

15 Zamaraeva to appear

Interim Summary

- HPSG is a fully-explicit syntactic formalism
- Can implement grammar gragments rigorously
- Most insights come from **assembling** the fragments into a system
  - ...intractable by hand
  - 🤦🏽‍♀️
- Next: The Grammar Matrix: A typologically driven meta-grammar engineering framework

Assembling and evaluating typologically diverse analyses

Assembling grammars systematically

- **Goal**: Make implemented linguistic grammars bigger and more accessible to broader research community

Assembling grammars systematically

- **Goal**: Make implemented linguistic grammars bigger and more accessible to broader research community
- **Method**: Meta-grammar engineering with Head-Driven Phrase Structure Grammar

Assembling grammars systematically

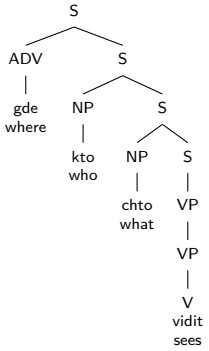
- **Goal**: Make implemented linguistic grammars bigger and more accessible to broader research community
- **Method**: Meta-grammar engineering with Head-Driven Phrase Structure Grammar
- **Project**: Analysis of constituent questions for a grammar engineering system

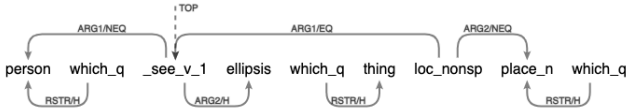
Assembling grammars systematically

- **Goal**: Make implemented linguistic grammars bigger and more accessible to broader research community
- **Method**: Meta-grammar engineering with Head-Driven Phrase Structure Grammar
- **Project**: Analysis of constituent questions for a grammar engineering system
- **Result**:
  - New library in the system; more complex hypotheses can be tested
  - Archived test suites and analyses for several languages
  - Some takeaways regarding the interaction of different analyses

Questionnaire – Analysis – Semantics

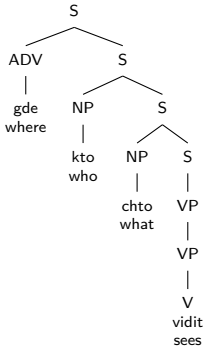
(4) Gde kto chto vidit?
where who.NOM what.ACC see.3SG
'Who sees what where?'[rus]





Questionnaire – Analysis – Semantics

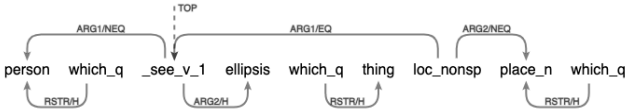
(4) Gde kto chto vidit?
where who.NOM what.ACC see.3SG
'Who sees what where?'[rus]



Choices regarding the position of question phrases

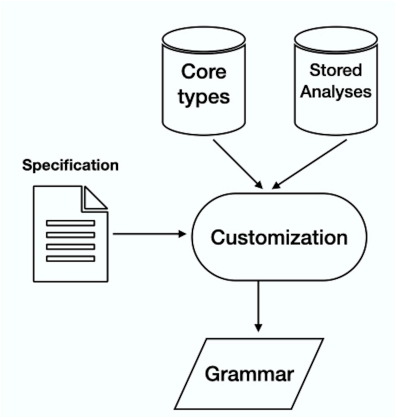
Question phrases can appear at the left edge of the sentence regardless of the position the questioned constituent would appear in (*Who did you see? I know who you saw* etc.):

- ○ Only one question phrase can be fronted
- ◉ All question phrases can be fronted
- ○ Question phrases cannot be fronted (stay *in situ*)



The Grammar Matrix

- Meta-grammar engineering framework[16]
- Input: Typological specification, lexicon, morphological rules
- Output: Implemented HPSG grammar fragment
  - Parse and generate sentences
  - Output syntactic and **semantic** representations
- Many syntactic phenomena are supported[17]

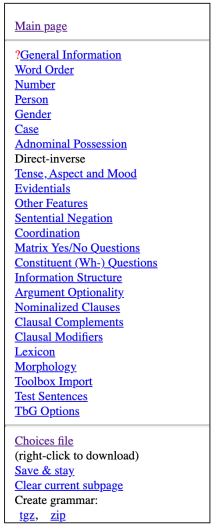


16 https://matrix.ling.washington.edu/customize/matrix.cgi

17 Zamaraeva 2021; Zamaraeva, Howell, and Bender 2019; Howell and Zamaraeva 2018; Saleem 2010; Song 2014; Nielsen 2018; Drellishak and Bender 2005; Crowgey 2013; Bender and Flickinger 2005

Matrix libraries

- Specify several phenomena at the same time
- Click to **get a grammar fragment** covering all of them[18]
- Test hypotheses **in interaction**
- Parse and **generate** data within fragment's area of coverage
  - Large lexicons can be imported



[18] There may be bugs 🐞

Matrix library development: Bottom up!

Matrix library development: Bottom up!

- Questionnaires are designed based on **surveys of typological literature** 🌍

Matrix library development: Bottom up!

- Questionnaires are designed based on **surveys of typological literature**
- Mapping of questionnaire answers to implemented grammar fragments is done with a hybrid, **data-driven + formal** methodology[19]
  - Use theory but goal is to grow coverage and minimize overgeneration over **test suite**

[19] Bender, Drellishak, et al. 2010; Oepen and Flickinger 1998

Matrix library development: Bottom up!

- Questionnaires are designed based on **surveys of typological literature**
- Mapping of questionnaire answers to implemented grammar fragments is done with a hybrid, **data-driven + formal** methodology[19]
  - Use theory but goal is to grow coverage and minimize overgeneration over **test suite**

[19] Bender, Drellishak, et al. 2010; Oepen and Flickinger 1998

Matrix library development: Bottom up!

- Questionnaires are designed based on **surveys of typological literature**
- Mapping of questionnaire answers to implemented grammar fragments is done with a hybrid, **data-driven + formal** methodology[19]
  - Use theory but goal is to grow coverage and minimize overgeneration over **test suite**
- Libraries are evaluated with **held-out languages**

[19] Bender, Drellishak, et al. 2010; Oepen and Flickinger 1998

Matrix library development: Bottom up!

- Questionnaires are designed based on **surveys of typological literature**
- Mapping of questionnaire answers to implemented grammar fragments is done with a hybrid, **data-driven + formal** methodology[19]
  - Use theory but goal is to grow coverage and minimize overgeneration over **test suite**
- Libraries are evaluated with **held-out languages**
- Growing **regression testing** base
  - Language specs + test suites paired with "gold" semantic representations
  - Check **automatically** how any small change affects the **all** of the pairings

[19] Bender, Drellishak, et al. 2010; Oepen and Flickinger 1998

Meta-grammar engineering research question

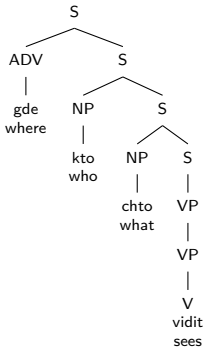
RQ: *What constitutes a model of a range of typologically attested ways of forming constituent questions within the given framework?*

Evaluation: How well does the analysis generalize to a set of randomly picked "held-out" languages?

Constituent questions in the Matrix

(4) Gde kto chto vidit?
where who.NOM what.ACC see.3SG
'Who sees what where?'[rus]

- New library[20]
- Typological scope:[21]
  - Position of question phrase
    - **Fronting**, *in situ*
    - Fronting optionality
  - Morphological marking
  - Question particles
    - position
    - obligatoriness
  - Question words



18 Zamaraeva 2021

19 König and Siemund 2007; Hagège 2008; Miyagawa 1987, *inter alia*

Assembling 🖼 fronting

Specification	Types	Core?	New?	Custom features
single oblig. front.	*wh-ques-phrase*	no	yes	SLASH
	subj-, obj-, adj-ex.	no	no	none
multi oblig. front.	*wh-ques-phrase*	no	yes	HDR|QUE MODIFIED *hasmod*
	subj-, obj-, adj-ex.	no	no	none
single opt. front.	*wh-ques-phrase*	no	yes	SLASH
	in-sutu-phrase	no	yes	none
	subj-, obj-, adj-ex.	no	no	none
multi opt. front.	*wh-ques-phrase*	no	yes	MODIFIED *hasmod*
	in-sutu-phrase	no	yes	HDR|L-QUE –
	subj-, obj-, adj-ex.	no	no	HDR|L-QUE –
in situ (no front.)	*in-sutu-phrase*	no	yes	none

The position of question phrases customization summary[22]

22 Zamaraeva 2021

Evaluation

- “Freeze” the analyses and the development

Evaluation

- “Freeze” the analyses and the development
- Grab reference grammars from “unseen” language families

Evaluation

- "Freeze" the analyses and the development
- Grab reference grammars from "unseen" language families
- Grab all examples of constituent questions from those books

Evaluation

- "Freeze" the analyses and the development
- Grab reference grammars from "unseen" language families
- Grab all examples of constituent questions from those books
- Fill out the questionnaire so as to cover the examples

Evaluation

- "Freeze" the analyses and the development
- Grab reference grammars from "unseen" language families
- Grab all examples of constituent questions from those books
- Fill out the questionnaire so as to cover the examples
- Click "create grammar", run the grammar on the examples

Evaluation

- "Freeze" the analyses and the development
- Grab reference grammars from "unseen" language families
- Grab all examples of constituent questions from those books
- Fill out the questionnaire so as to cover the examples
- Click "create grammar", run the grammar on the examples
- **Coverage**: % of examples the grammar actually covered
- **Overgeneration** % of ungrammatical examples parsed
  - covered = gave **correct** semantic representation

Evaluation

- "Freeze" the analyses and the development
- Grab reference grammars from "unseen" language families
- Grab all examples of constituent questions from those books
- Fill out the questionnaire so as to cover the examples
- Click "create grammar", run the grammar on the examples
- **Coverage**: % of examples the grammar actually covered
- **Overgeneration** % of ungrammatical examples parsed
  - covered = gave **correct** semantic representation
- **How well can the system handle examples from an "unseen" language, as it is described in the reference grammar?**

Evaluating the constituent questions library[22][23]

Language	ISO-639-3	Family	Gram. items	Coverage%	Question typology
Apinajé	[apn]	Macro-Jê	17	82.3	single front.
Makah	[myh]	Wakashan	14	78.5	morphological,int. verbs
Pacoh	[pac]	Austroasiatic	26	84.6	single opt. front.
Paresi-Haliti	[pab]	Arawakan	64	56.0	single front., int. verbs
Jalkunan	[bxl]	Mande	33	78.8	*in situ*, particle, int. verbs

✓ Single fronting, particles, morphological marking, interrogative verbs

✗ Question words as predicates

Grammars made correct predictions where I didn't expect them to!

Bugs consistent with theoretical predictions

22 Zamaraeva 2021

Conclusion

- Formal approaches to syntax are an important part of linguistics and NLP

Conclusion

- Formal approaches to syntax are an important part of linguistics and NLP
- Fully explicit formalisms like HPSG allow us to implement grammars on the computer and rigorously test them

Conclusion

- Formal approaches to syntax are an important part of linguistics and NLP
- Fully explicit formalisms like HPSG allow us to implement grammars on the computer and rigorously test them
- **Example:** An analysis of constituent questions integrated into the Grammar Matrix system
  - Clear area of applicability as archived in the specifications, test suites, and the version of the Matrix system

Conclusion

- Formal approaches to syntax are an important part of linguistics and NLP
- Fully explicit formalisms like HPSG allow us to implement grammars on the computer and rigorously test them
- **Example:** An analysis of constituent questions integrated into the Grammar Matrix system
  - Clear area of applicability as archived in the specifications, test suites, and the version of the Matrix system
- **Future:** Robust NLP evaluation and precision in NLP applications for more languages 

Conclusion

- Formal approaches to syntax are an important part of linguistics and NLP
- Fully explicit formalisms like HPSG allow us to implement grammars on the computer and rigorously test them
- **Example:** An analysis of constituent questions integrated into the Grammar Matrix system
  - Clear area of applicability as archived in the specifications, test suites, and the version of the Matrix system
- **Future:** Robust NLP evaluation and precision in NLP applications for more languages
- **Future:** Encoded and tested sets of hypotheses for more languages

Conclusion

- Formal approaches to syntax are an important part of linguistics and NLP
- Fully explicit formalisms like HPSG allow us to implement grammars on the computer and rigorously test them
- **Example:** An analysis of constituent questions integrated into the Grammar Matrix system
  - Clear area of applicability as archived in the specifications, test suites, and the version of the Matrix system
- **Future:** Robust NLP evaluation and precision in NLP applications for more languages
- **Future:** Encoded and tested sets of hypotheses for more languages
- **Assembling fragments of our understanding of language**

Acknowledgments

- The work presented today was partially funded by the United States National Science Foundation under Grant No. BCS-1561833 (PI Bender).
- This work would not be possible without the DELPH-IN community

References I

Montague, Richard (1974). *Formal Philosophy: Selected Papers, RH Thomason, Ed.* Yale University Press, New Haven, Conn.

Pollard, Carl and Ivan A. Sag (1994). *Head-Driven Phrase Structure Grammar*. Studies in Contemporary Linguistics. Chicago, IL and Stanford, CA: The University of Chicago Press and CSLI Publications.

Flickinger, Dan (2000). "On building a more efficient grammar by exploiting types". In: *Natural Language Engineering* 6.01, pp. 15–28.

— (2011). "Accuracy v. Robustness in Grammar Engineering". In: *Language from a Cognitive Perspective: Grammar, Usage and Processing*. Ed. by Emily M Bender and Jennifer E. Arnold. Stanford, CA: CSLI Publications, pp. 31–50.

Hajdik, Valerie et al. (2019). "Neural text generation from rich semantic representations". In: *arXiv preprint arXiv:1904.11564*.

Zamaraeva, Olga, Kristen Howell, and Adam Rhine (2018). "Improving Feature Extraction for Pathology Reports with Precise Negation Scope Detection". In: *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 3564–3575.

Buys, Jan and Phil Blunsom (2017). "Robust Incremental Neural Semantic Graph Parsing". In: *ACL (1)*.

Packard, Woodley (2014). "UW-MRS: Leveraging a Deep Grammar for Robotic Spatial Commands". In: *SemEval 2014*, p. 812.

Bender, Emily M, Dan Flickinger, Stephan Oepen, et al. (2004). "Arboretum: Using a precision grammar for grammar checking in CALL". In: *InSTIL/ICALL*.

Morgado da Costa, Luis, Francis Bond, and Xiaoling He (2016). "Syntactic well-formedness diagnosis and error-based coaching in computer assisted language learning using machine translation". In: *NLPTEA*, pp. 107–116.

Morgado da Costa, Luis, R. Winder, et al. (2020). "Automated Writing Support Using Deep Linguistic Parsers". In: *LREC*, pp. 369–377.

Oepen, Stephan and Dan Flickinger (2019). "The ERG at MRP 2019: Radically Compositional Semantic Dependencies". In: *CoNLL 2019*, p. 40.

Siegel, Melanie, Emily M Bender, and Francis Bond (Nov. 2016). *Jacy: An Implemented Grammar of Japanese*. CSLI Studies in Computational Linguistics. Stanford CA: CSLI Publications. ISBN: 9781684000180.

Fan, Zhenzhen (2018). "Building an HPSG Chinese grammar (Zhong)". PhD thesis. Nanyang Technological University.

Crysmann, Berthold (2003). "On the Efficient Implementation of German Verb Placement in HPSG". In: *Proceedings of RANLP 2003*. Borovets, Bulgaria, pp. 112–116.

Marimon, Montserrat (2010). "The Spanish Resource Grammar.". In: *LREC*.

Bender, Emily M, Dan Flickinger, and Stephan Oepen (2002). "The Grammar Matrix: An Open-Source Starter-Kit for the Rapid Development of Cross-Linguistically Consistent Broad-Coverage Precision Grammars". In: *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*. Ed. by John Carroll, Nelleke Oostdijk, and Richard Sutcliffe. Taipei, Taiwan, pp. 8–14.

References II

Bender, Emily M, Scott Drellishak, et al. (2010). "Grammar Customization". In: *Research on Language & Computation* 8.1, pp. 23–72. ISSN: 1570-7075.

Bender, Emily M (2010). "Reweaving a grammar for Wambaya". In: *Linguistic Issues in Language Technology* 3.1.

Crowgey, Joshua (2019). "Braiding Language (by Computer): Lushootseed Grammar Engineering". PhD thesis. University of Washington.

Inman, David Anthony (2019). "Multi-predicate Constructions in Nuuchahnulth". PhD thesis. University of Washington.

Zamaraeva, Olga (2021). "A cross-linguistic analysis of constituent questions for the Grammar Matrix". PhD thesis. University of Washington.

Ginzburg, Jonathan and Ivan Sag (2000). *Interrogative investigations*. Stanford: CSLI publications.

Zamaraeva, Olga and Guy Emerson (2020). "Multiple Question Fronting without Relational Constraints: An analysis of Russian as a basis for cross-linguistic modeling". In: *Proceedings of the 27th International Conference on Head-Driven Phrase Structure Grammar*. Virtual conference, to appear.

Crysmann, Berthold (2015). "Resumption and Extraction in an Implemented HPSG grammar of Hausa". In: *Proceedings of the Grammar Engineering Across Frameworks (GEAF) 2015 Workshop*, pp. 65–72.

Zamaraeva, Olga (to appear). "Morphological marking of questions: A case for nonlocal amalgamation". In: *Proceedings of the 28th International Conference on Head-Driven Phrase Structure Grammar*, to appear.

Zamaraeva, Olga, Kristen Howell, and Emily M Bender (2019). "Modeling clausal complementation for a grammar engineering resource". In: Proceedings of the 2nd meeting of the Society for Computation in Linguistics, pp. 39–49.

Howell, Kristen and Olga Zamaraeva (2018). "Clausal modifiers in the Grammar Matrix". In: *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 2939–2952.

Saleem, Safiyyah (2010). "Argument optionality: A new library for the Grammar Matrix customization system". MA thesis. University of Washington.

Song, Sanghoun (2014). "A grammar library for information structure". PhD thesis. University of Washington.

Nielsen, Elizabeth K (2018). "Modeling adnominal possession in the lingo grammar matrix". MA thesis. University of Washington.

Drellishak, Scott and Emily M Bender (2005). "Coordination Modules for a Crosslinguistic Grammar Resource". In: *Departamento de Informática Faculdadede Ciências da Universidadede Lisboa Campo Grande, 1749–016 Lisboa Portugal*, p. 29.

Crowgey, Joshua (2013). "The syntactic exponence of sentential negation: A model for the LinGO Grammar Matrix". PhD thesis. University of Washington.

References III

Bender, Emily M and Dan Flickinger (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.

Oepen, Stephan and Dan Flickinger (1998). "Towards systematic grammar profiling. test suite technology 10 years after". In: *Computer Speech & Language* 12.4, pp. 411–435.

König, Ekkehard and Peter Siemund (2007). "Speech act distinctions in grammar". In: *Language typology and syntactic description* 1, pp. 276–324.

Hagège, Claude (2008). "Towards a typology of interrogative verbs". In: *Linguistic typology* 12, pp. 1–44.

Miyagawa, Shigeru (1987). "LF affix raising in Japanese". In: *Linguistic Inquiry* 18.2, pp. 362–367.