The Grammar Matrix (Motivations, Technical Details)
Morphotactics in the Grammar Matrix

Ling 567
April 15, 2013

Overview

- Grammar Matrix: Motivation
- Grammar customization
- Morphology: Who's job is it anyway?
- Morphotactics in the customization system
- Morphotactics in customized grammars

Grammar Customization with the LinGO Grammar Matrix

MS/UW Symposium, January 22, 2010

Emily M. Bender

Joint work with: Scott Drellishak, Laurie Poulson, Antske Fokkens, Safiyyah Saleem, Michael Wayne Goodman, Kelly O'Hara, Joshua Hou, Daniel Mills, Dan Flickinger, Stephan Oepen

Precision grammars

- Map surface strings to syntactic and semantic representations, often bidirectionally
- Represent grammaticality
- Have been developed to broad coverage for a handful of languages in a handful of syntactic frameworks (Flickinger 2000, Siegel & Bender 2002, Müller & Kasper 2000)
- Can now parse efficiently (Oepen et al 2002)
- Scale more effectively than tree-bank derived grammars (in the sense of including new kinds of information)
- Can be made more robust with statistical lexical acquisition (Blunsom & Baldwin 2006) and other kinds of knowledge engineering/ML hybridization (Zhang & Kordoni 2008)
- ... but are expensive to build.

Human languages

- Vary along many dimensions, but not infinitely
- Can be seen as solving many of the same problems in different ways
- Just might share some core properties in common
- Can we leverage what's been learned in developing large-scale precision grammars for some languages to the development of grammars for others?

The Grammar Matrix

- Developed in the context of the DELPH-IN consortium (www.delph-in.net)
- Uses HPSG (Pollard and Sag 1994) and MRS (Copestake et al 2005)
- Core grammar originally abstracted from English Resource Grammar (Flickinger 2000) with reference to Jacy Japanese grammar (Siegel and Bender 2002)
- Aims to support both rapid initial development and long-term grammar build-out
- Promotes cross-grammar consistency in semantic representations
- Is also an exercise in exploration of potential universals
- http://www.delph-in.net/matrix

Customization System

- The Grammar Matrix core grammar is not itself a functioning grammar fragment
- Many phenomena are “widespread, but not universal” (Drellishak, 2009)
- Grammar customization is an approach to massively multilingual grammar code reuse
- Can the same analysis of e.g., SVO word order, split-ergativity, or “pro-drop” work in different languages?
- Web-based questionnaire elicits typological and lexical information, then outputs working “starter grammar”

Customization system: Current libraries

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

Evaluation: Do the existing libraries scale to unseen languages?

- Testsuites developed by (then) non-Matrix developer on the basis of descriptive grammars to cover phenomena represented in Matrix libraries
- Starter grammars developed through customization system
- Coverage, semantic accuracy, and overgeneration measured

Evaluation

Language	Coverage raw	Coverage treebanked	Overgeneration	Spurious ambiguity	Average readings
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

Evaluation

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 Object Agreement	+		+	+		+	+
Verb Subject Agreement	+	+	+	+		+	+
Person	+	+	+	+	+	+	+
Number	+	+	+	+	+	+	+/−
Gender	+	+	+	+	+	+	+

Future work

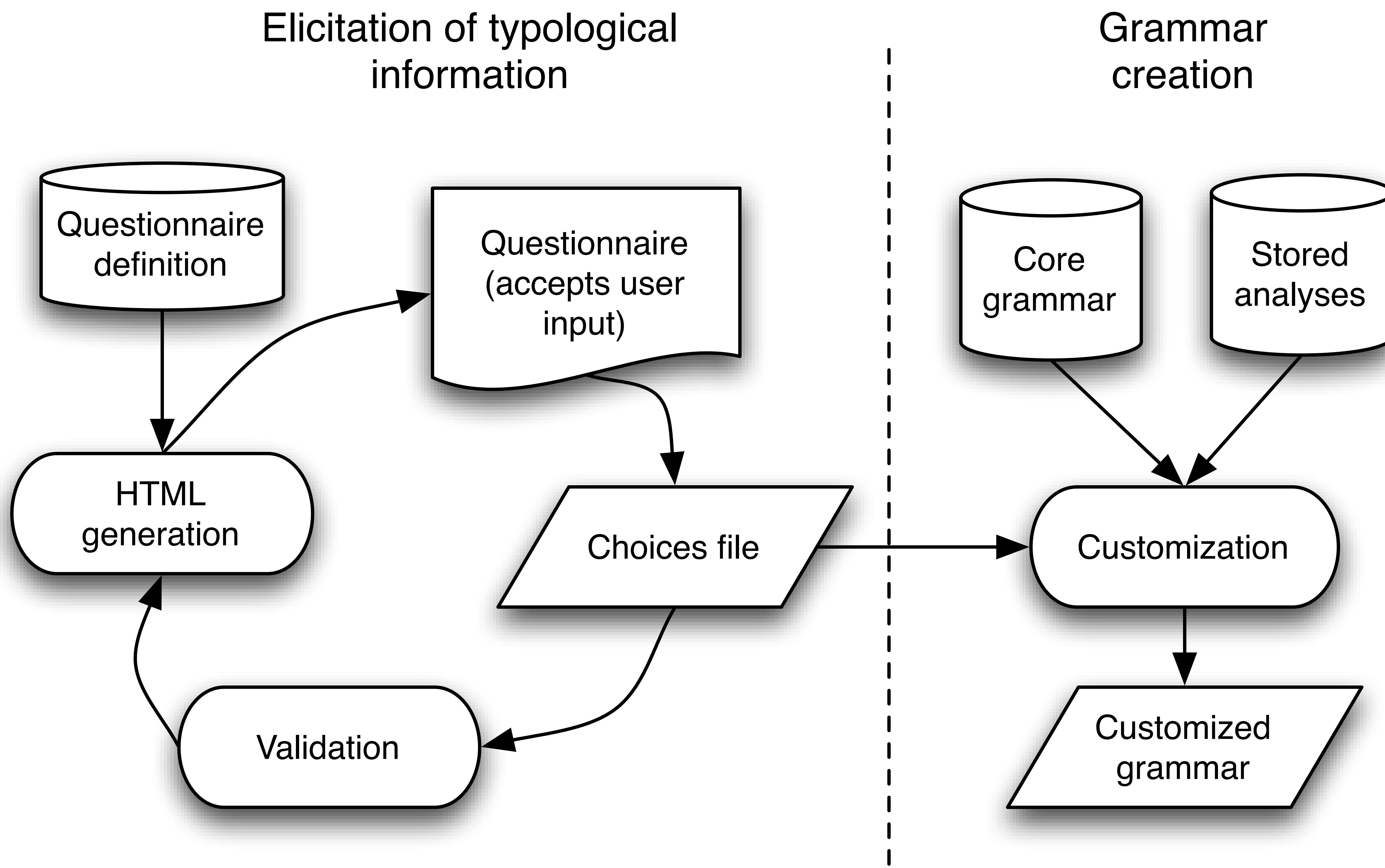
- More libraries: Modifiers, embedded clauses of various types, wh-questions, information structure, ...

- Lexical acquisition

- MOM (Matrix-ODIN Mash-up): Can the customization system questionnaire be filled out automatically on the basis of information in ODIN (Lewis & Xia 2008)?

Overview

- Grammar Matrix: Motivation
- Grammar customization
- Morphology: Who’s job is it anyway?
- Morphotactics in the customization system
- Morphotactics in customized grammars



(Bender et al 2010)

Overview

- Grammar Matrix: Motivation
- Grammar customization
- Morphology: Who’s job is it anyway?
- Morphotactics in the customization system
- Morphotactics in customized grammars

Morphology: Basics

- Morpheme: The smallest meaningful unit of language/smallest pairing of "form" and "meaning"
- But:
  - "form" can be lots of things, including empty but also messy changes to word form
  - "meaning" can be just syntactic features
- Morphotactics: Which morphemes can combine, in what order
- Morphophonology: Relationship between underlying word forms and surface forms
- Morphosyntax: Relationship between morphemes and syntactic and semantic features

Morphology: Example

slolmáyaye
slol-ma-ya-yÁ
know-1SG.PAT-2SG.AGT-know

‘you know/knew me’ [lkt]

Morphology: Example

slolmáyaye

slol-ma-ya-yÁ

know-1SG.PAT-2SG.AGT-know

‘you know/knew me’ [lkt]

- Infixation, vowel harmony: Morphophonology

Morphology: Example

slolmáyaye
slol-ma-ya-yÁ
know-1SG.PAT-2SG.AGT-know

'you know/knew me' [lkt]

- Infixation, vowel harmony: Morphophonology
- Relative order of PAT and AGT marker, optionality of same: Morphotactics

Morphology: Example

slolmáyaye
slol-ma-ya-yÁ
know-1SG.PAT-2SG.AGT-know

‘you know/knew me’ [lkt]

- Infixation, vowel harmony: Morphophonology
- Relative order of PAT and AGT marker, optionality of same: Morphotactics
- Mapping to constraints that the patient argument be 1sg and the agent 1pl: Morphosyntax

Morphology: Example

slolmáyaye
slol-ma-ya-yÁ
know-1SG.PAT-2SG.AGT-know

'you know/knew me' [lkt]

- Infixation, vowel harmony: Morphophonology
- Relative order of PAT and AGT marker, optionality of same: Morphotactics
- Mapping to constraints that the patient argument be 1sg and the agent 1pl: Morphosyntax
- Actually parsing the string: priceless!

What morphophonolgy can the LKB & the customization system handle?

	LKB	Customization System
polite concatenative morphology	✓	✓
zero morphemes	✓	✓
morphologically conditioned allomorphy	✓	✓
phon. chnages at morpheme boundary	✓	
ablaut		
infixation		
vowel harmony		
suppletion		

Assume a morphophonological analyzer...

- Morphophonological analyzers map surface forms to underlying strings of morphemes
- FSTs are up to the task (except for open-class reduplication)
  - XFST (Beesley & Karttunen 2003) is a very linguist-friendly set up; FOMA (Holden & Algeria 2010) is a open-source package with similar functionality
- But you don't need to build one for this class!
- Use the morpheme segmented line of your IGT to represent what it would map to, and then (if you have any interesting morphophonology) have that line be the target for your grammar.

Morphophonology/morphosyntax boundary: Where to draw the line?

- Underlying morphemes can be represented as a sequence of phonemes or as as symbols representing morphological features.
- A canonical XFST-derived analyzer will also include POS tags as a morphological feature in the underlying form.
- From the point of view of the LKB:
  - The POS tag adds nothing
  - Spelling the morphemes as morphological features adds nothing: we still need a lexical rule that maps those strings to constraints on avms

Morphophonology/morphosyntax boundary: Where to draw the line?

- On the other hand: for XFST/FOMA, the POS tags (and maybe features) can be useful intermediate stages in processing
- The features can make it easier to create gloss lines automatically.
- On the third hand: using sequences of morphemes might make LKB input/output comprehensible to speakers
- So what should the upper tape have?

Overview

- Grammar Matrix: Motivation
- Grammar customization
- Morphology: Who’s job is it anyway?
- Morphotactics in the customization system
- Morphotactics in customized grammars

Basic concepts

- Position class: A supertype to lexical rules which fit in the same slot
- Lexical rule type: *lex-rule* and its subtypes, all have DTR feature
- Lexical rule instance: A grammar entity (manipulatable by the LKB) which inherits from a lexical rule type and specifies a spelling change (including no change).
- Forbids constraint: A specification in the customization system stating that a stem lexical rule type (including a position class) cannot co-occur with another lexical rule type, instance, pc or stem.
- Requires constraint: A specification in the customization system stating that a stem lexical rule type (including a position class) must co-occur with another lexical rule type, instance, pc or stem.

Position classes, inputs and lexical rule hierarchies

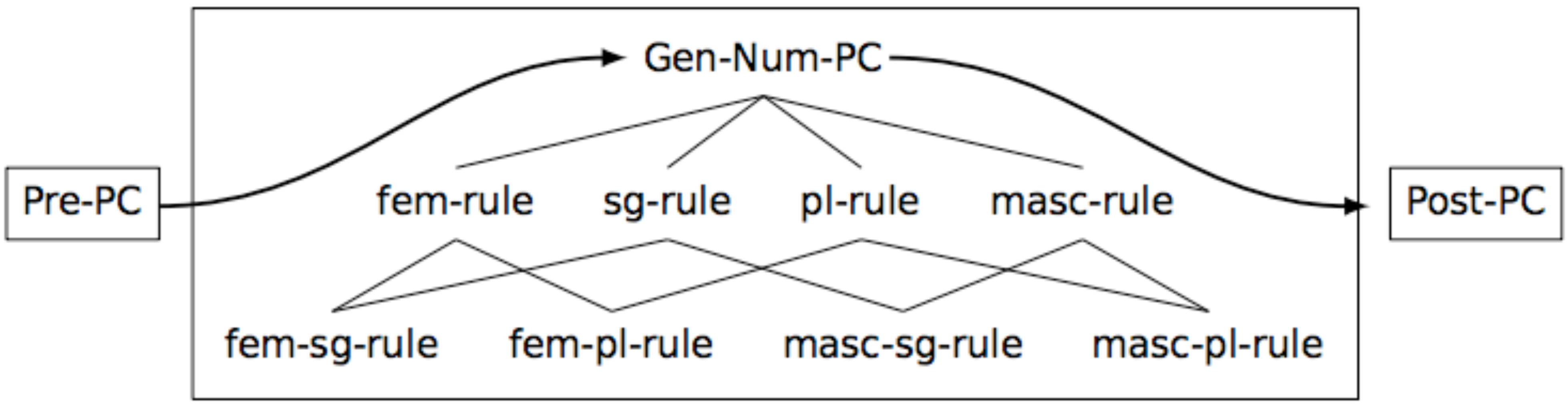


Figure 9: Example lexical rule type hierarchy in a position class

(Goodman 2013)

To define a position class

- Required:
    - Whether or not it is obligatory
    - Possible inputs and prefix/suffix
        - = position in the string
- Optional:
    - Requires/forbids constraints

To define a lex rule type

- Required
  - Nothing (though defaults fill in)
- Optional
  - Name
  - Supertype (if it doesn’t inherit directly from its position class)
  - Requires/forbids constraints

To define a lex rule instance

- Required
  - Affix v. no affix
  - Spelling for affix
- Optional
  - Nothing

Overview

- Grammar Matrix: Motivation
- Grammar customization
- Morphology: Who’s job is it anyway?
- Morphotactics in the customization system
- Morphotactics in customized grammars

tdl files

- matrix.tdl: Supertypes for lex-rules, which handle the copying up of everything you’re not changing
- my_language.tdl: Position classes and lex rule types defined through the customization system; features for inside INFLECTED
- lrules.tdl: Instances for non-spelling-changing lex rules (zero morphemes)
- irules.tdl: Instances for spelling-changing lex rules

Handling of morphotactics

- Rule order handled through super types and typing the DTR feature
- Requires/forbids through the INFLECTED feature

```
case-lex-rule-super := Representative-rule-dtr &
                       add-only-no-ccont-rule &
                       noun-telic-rule-dtr &
[ INFLECTED [ CASE-FLAG +,
              INNER-NEGATION-FLAG #inner-negation,
              NUMBERED-FLAG #numbered ],
  DTR case-rule-dtr &
      [ INFLECTED [ INNER-NEGATION-FLAG
                                #inner-negation,
                    NUMBERED-FLAG #numbered ] ] ].
```

Overview

- Grammar Matrix: Motivation
- Grammar customization
- Morphology: Who’s job is it anyway?
- Morphotactics in the customization system
- Morphotactics in customized grammars