

Extended constraint ranking models for frequency-sensitive accounts of syntax

Jonas Kuhn

Stanford University/University of Texas at Austin
jonask@mail.utexas.edu

Q|Tosnabrück
October 3, 2002

This work was supported by a postdoctoral fellowship of the German Academic Exchange Service (DAAD).

Overview

- **Optimality-Theoretic Syntax**
 - Optimality Theory and OT-LFG
 - constraint ranking vs. constraint weighting
- **Learning**
 - the constraint demotion algorithm
 - the generalized learning algorithm: Stochastic OT
- **Experiments**
 - corpus-based learning of clausal syntax of German
 - “supervised learning” vs. bidirectionality in learning
- **Discussion**
 - the issue of conflicting ranking arguments
 - potential alternatives

Constraint (re-)ranking in OT


Constraints (Grimshaw 1997, 374):

OP-SPEC Syntactic operators must be in specifier position.

OB-HD A projection has a head.

STAY Trace is not allowed.

R1: OP-SPEC \gg OB-HD \gg STAY: *English*


Input: read(x, y), $x = she$, $y = what$, tense = future Candidate set:	OP-SPEC	OB-HD	STAY
[IP she will [V ^P read what]]	*!		
[CP what e [IP she will [VP read t]]]		*!	*
 [CP what wil _i [IP she e _i [VP read t]]]			**

Constraint (re-)ranking in OT

Modifying the ranking:

R1: OP-SPEC \gg OB-HD \gg STAY: *English*

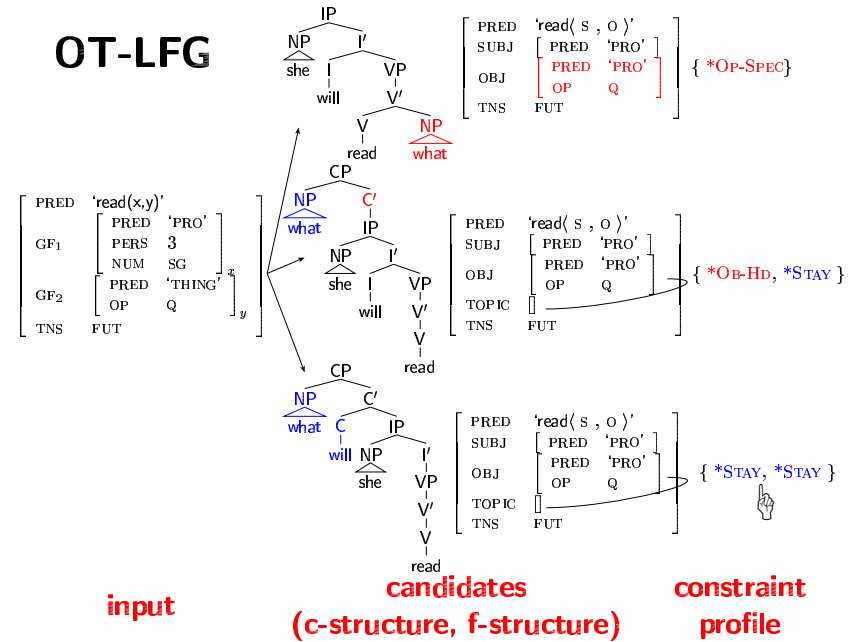
R2: STAY \gg OP-SPEC \gg OB-HD: *wh in situ language*

Input: read(x, y), $x = she$, $y = what$, tense = future Candidate set:	STAY	OP-SPEC	OB-HD
 [IP “she” “will” [VP “read” “what”]]		*	
[CP “what” e [IP “she” “will” [VP “read” t]]]	*!		*
[CP “what” “will _i ” [IP “she” e _i [VP “read” t]]]	*!*		

OT-LFG

- Declarative formalization of Optimality-Theoretic Syntax
- Based on the representation structures of *Lexical-Functional Grammar*
- Bresnan (1996, 2000), Kuhn (forthcoming), ...

OT-LFG



Constraint (re-)ranking in OT

OT: Ranking hypothesis

- OT hypothesis: linguistic competence can be modelled through constraint **ranking** rather than constraint **weighting**
 - violating a high-ranking constraint is worse than arbitrarily many violations of a lower-ranking constraint
- more restrictive model makes clear, testable predictions

Constraint (re-)ranking in OT

	CONSTR. 1	CONSTR. 2	CONSTR. 3
Candidate set:			
candidate A	*!	*	
candidate B			***

- if we observe candidate A, we know that
CONSTR. 3 \gg { CONSTR. 1, CONSTR. 2 }
- “**ranking argument**”

Constraint (re-)ranking in OT

(1) Candidate set: **CONSTR. 3** **CONSTR. 1** **CONSTR. 2** incompatible with data like

	CONSTR. 1	CONSTR. 2	CONSTR. 3
candidate A	*	*	*
candidate B	***		

(2) Candidate set: **CONSTR. 1** **CONSTR. 2** **CONSTR. 3**

	CONSTR. 1	CONSTR. 2	CONSTR. 3
candidate A'		*	*
candidate B'		*	

(3) Candidate set: **CONSTR. 1** **CONSTR. 2** **CONSTR. 3**

	CONSTR. 1	CONSTR. 2	CONSTR. 3
candidate A''			*
candidate B''	*		

- so if we observe such data, the constraint set must be incorrect

Constraint (re-)ranking in OT

- under a **constraint weighting** regime, (1) may be compatible with (2), with (3), with both, or none

(1) Candidate set: **CONSTR. 1** **CONSTR. 3** **CONSTR. 2**

	CONSTR. 1	CONSTR. 3	CONSTR. 2
-5 cand. A	*		*
-9 cand. B		***	

(2) Candidate set: **CONSTR. 1** **CONSTR. 3** **CONSTR. 2**

	CONSTR. 1	CONSTR. 3	CONSTR. 2
-3 cand. A'		*	
-4 cand. B'	*		

(3) Candidate set: **CONSTR. 1** **CONSTR. 3** **CONSTR. 2** not compatible

	CONSTR. 1	CONSTR. 3	CONSTR. 2
cand. A''		*	
cand. B''			*

Constraint (re-)ranking in OT

- under a **constraint weighting** regime, (1) may be compatible with (2), with (3), with both, or none

(1) Candidate set: **CONSTR. 1** **CONSTR. 2** **CONSTR. 3**

	CONSTR. 1	CONSTR. 2	CONSTR. 3
-11 cand. A	*	*	
-12 cand. B			***

(2) Candidate set: **CONSTR. 1** **CONSTR. 2** **CONSTR. 3**

	CONSTR. 1	CONSTR. 2	CONSTR. 3
-4 cand. A'			*
-6 cand. B'	*		

(3) Candidate set: **CONSTR. 1** **CONSTR. 2** **CONSTR. 3**

	CONSTR. 1	CONSTR. 2	CONSTR. 3
-4 cand. A''			*
-5 cand. B''		*	

Constraint (re-)ranking in OT

- under a **constraint weighting** regime, (1) may be compatible with (2), with (3), with both, or none

(1) Candidate set: **CONSTR. 3** **CONSTR. 1** **CONSTR. 2**

	CONSTR. 3	CONSTR. 1	CONSTR. 2
-3 cand. A		*	*
-9 cand. B	***		

(2) Candidate set: **CONSTR. 3** **CONSTR. 1** **CONSTR. 2** not compatible

	CONSTR. 3	CONSTR. 1	CONSTR. 2
-3 cand. A'	*		
-2 cand. B'		*	

(3) Candidate set: **CONSTR. 3** **CONSTR. 1** **CONSTR. 2** not compatible

	CONSTR. 3	CONSTR. 1	CONSTR. 2
-3 cand. A''	*		
-1 cand. B''			*

- under a constraint weighting scheme, the effect of choosing a particular constraint set is underdetermined

Constraint (re-)ranking in OT

OT: Ranking hypothesis

- choice of constraint set (and linguistic representation) has direct influence on predicted typology
- constraint set is assumed to reflect innate restrictions on possible grammars (Universal Grammar)

Constraint (re-)ranking in OT

Limitations of the Ranking hypothesis

- hard to derive optionality/variation (possibly with subtle differences in conditions on context of usage), no quantitative effects
 - (strict) ranking differentiates between any two candidates with a different constraint profile
 - no theoretical status for the second-best, third-best etc.

Candidate set:	1	2	3	4	5	6
cand. A	*!					
cand. B	*!	*	*			*
☞ cand. C		*	*			*
cand. D		*	*		*!	*
cand. E		*	*			**!

Constraint (re-)ranking in OT

Overcoming the limitations!?

Modifications (not necessarily mutually exclusive)

- constraint ranking but no strict ranking Anttila (1997), Boersma (1998), Boersma and Hayes (2001);
- ranking-based model of linguistic competence embedded in a broader (weighting-based) context
- constraint weighting model with widely separated weights (i.e., coming close to a ranking model)
- ...?

How to decide between choices?

- different theoretical idealizations involved

Learning

- Error-based learning algorithms for Optimality Theory
 - initial ranking (e.g., all constraints ranked the same)
 - apply current ranking on data
 - adjust ranking if hypothesis about winner is wrong

Candidate set:	CONSTR. 1	CONSTR. 2	CONSTR. 3	CONSTR. 4	CONSTR. 5
candidate A		*	*		
observed: candidate B	*!	*			*

⇒ Incorrect ranking: CONSTR. 3 should outrank CONSTR. 1

Learning

Constraint Demotion Algorithm (Tesar and Smolensky 1998):

- (ignore constraints violated by both winner and observed output, and constraints violated by neither of the two)
- demote constraints violated by observed output just below highest-ranking constraint violated by putative winner

Learning

Constraint Demotion Algorithm

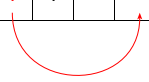
Candidate set:	CONSTR. 1	CONSTR. 2	CONSTR. 3	CONSTR. 4	CONSTR. 5
candidate A		*	*		
observed: candidate B	*	*			*

- ignore constraints violated by both winner and observed output, and constraints violated by neither of the two)

Learning

Constraint Demotion Algorithm

Candidate set:	CONSTR. 1	CONSTR. 2	CONSTR. 3	CONSTR. 4	CONSTR. 5
candidate A		*	*		
observed: candidate B	*	*			*



- demote constraints violated by observed output just below highest-ranking constraint violated by putative winner

Learning

Constraint Demotion Algorithm

Candidate set:	CONSTR. 2	CONSTR. 3	CONSTR. 1	CONSTR. 4	CONSTR. 5
candidate A	*	*!			
observed: candidate B	*		*		*

- Constraint Demotion Algorithm works for noise-free data
- cannot account for optionality/variation
- no sensitivity for frequencies

Learning

Gradual Learning Algorithm (GLA) (Boersma 1998, Boersma and Hayes 2001):

- robust modification of constraint demotion algorithm
- constraints are ranked on a continuous scale (i.e., still no weighting approach)
- some random noise (with normal distribution) added to constraint rank at evaluation time

~~CONSTR. 1~~ ~~CONSTR. 2~~ ~~CONSTR. 3~~ ~~CONSTR. 4~~ ~~CONSTR. 5~~

Learning step: all constraints differing in violation are slightly promoted/demoted

Learning

Gradual Learning Algorithm

	CONSTR. 1	CONSTR. 2	CONSTR. 3	CONSTR. 4	CONSTR. 5
Candidate set:					
candidate A		*	*		
observed: candidate B	*	*			*
	→	←			→

Learning

Gradual Learning Algorithm

- GLA can deal with optionality/variation (to a certain degree)

~~CONSTR. 1~~ ~~CONSTR. 2~~ ~~CONSTR. 3~~ ~~CONSTR. 4~~ ~~CONSTR. 5~~

- frequencies of alternatives in variable phenomena will be reproduced (Boersma and Hayes 2001) (with an appropriate set of constraints)
- application in syntax: OT-LFG (Bresnan and Deo 2001, Koontz-Garboden 2001, Dingare 2001, Bresnan et al. 2001)
- so far, accounts have focused on clear-cut grammar fragments

Experiments

Question: Can GLA be used for an exploratory analysis of a more complex cluster of interacting phenomena?

- start out with a certain set of linguistically well-understood constraints
- add further constraints (possibly *ad hoc* constraints) to explore interactions
- will robustness of algorithm allow for an insightful investigation of the learning behavior?
 - learning aspects of competence from performance data

(Hope: if the *ad hoc* constraints are not fully correct, they may still be sufficiently correlated with the unknown appropriate constraints.)

Experiments

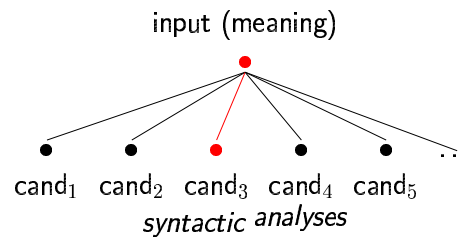
- set of phenomena should
 - display variation, but at the same time
 - clearly obey certain language-specific principles
- learn **clausal syntax of German**

Target information in learning

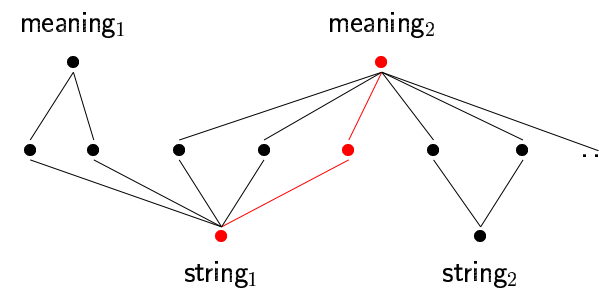
How much information should be provided to the learner?

- previous approaches (constraint demotion algorithm or GLA):
 - provide exact specification of target winner in candidate set
 - learning within generation-based optimization (expressive optimization)
- (over-)idealization
- **bidirectional** application of **optimization** should ultimately make this redundant:
 - **robust interpretive parsing** (Tesar and Smolensky 1998)
 - learner knows surface string
 - apply current constraint ranking to pick putative target winner among alternative parses

Target information in learning



Target information in learning

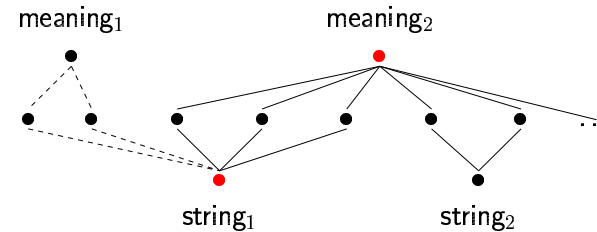


Target information in learning

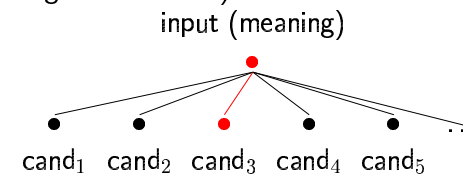
How much information should be provided to the learner?

- human learner can exploit semantic information and background knowledge
 - narrows down choices in interpretive parsing significantly
- ⇒ provide predicate-argument structure in learning experiments
- allows us to focus on syntactic learning
- parsing optimization has still a large set of candidates to choose from
 - alternative syntactic parses (c-structures)
with the same predicate-argument structure (f-structure)

Target information in learning



(compare full target annotation:)



Experimental set-up

Training data: Tiger treebank: syntactically annotated newspaper corpus of German (cf. Dipper et al. (2001), Zinsmeister et al. (2002))

- treebank includes constituency and grammatical relations
- only partially exploited for training data (as far as justified by non-syntactic information available to the human learner):
 - extracted full clauses (matrix and embedded clauses)
 - pre-bracketed embedded argument/modifier phrases (NPs, PPs, etc.), provided grammatical functions
 - no information provided about verbal constituents

Experimental set-up

- Der Vorstand der Firma hat gefordert, daß der *the board of the company has demanded that the* Geschäftsführer entlassen wird. *managing director laid off is*

gives rise to two training clauses:

- [Der Vorstand der Firma] hat gefordert, [daß ...]
- daß [der Geschäftsführer] entlassen wird

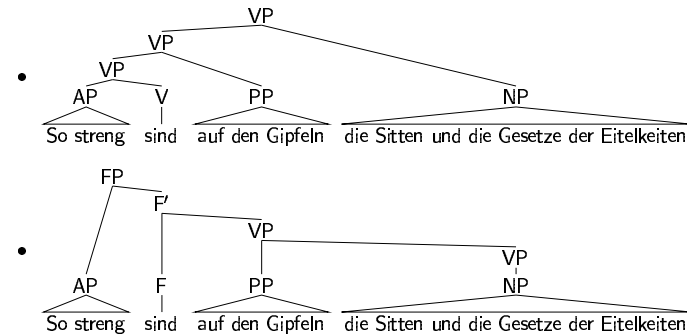
Experimental set-up

- highly underrestricted LFG grammar provides candidate analyses (all possible permutations of the input chunks are accepted)
 - grammar encodes extended X-bar scheme (inviolable principles)
 - all positions are optional, functional projections (IP, CP) can be freely filled with verbs, auxiliaries, complementizers
 - written with Xerox Linguistic Environment (XLE)

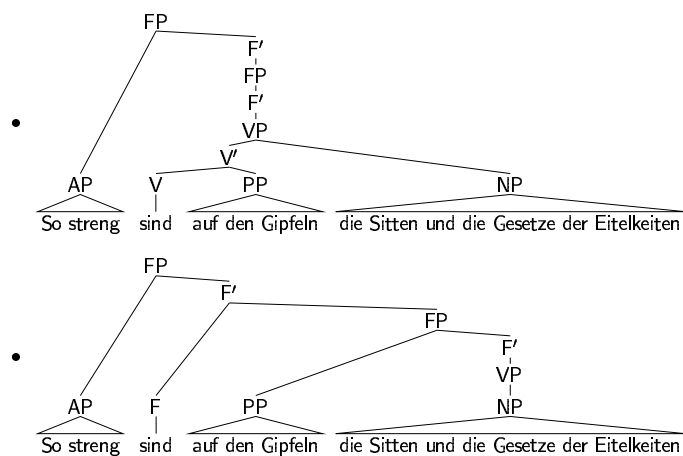
The base grammar

Even for correct surface order, many different underlying analyses possible:

[So streng] [sind] [auf den Gipfeln] [die Sitten und die Gesetze der Eitelkeiten]
So strict are on the summits the customs and the rules of vanities



[So streng] [sind] [auf den Gipfeln] [die Sitten und die Gesetze der Eitelkeiten]
So strict are on the summits the customs and the rules of vanities



Experimental set-up

- core constraints inspired by OT accounts of clausal syntax (Grimshaw 1997, Sells 2001)
- further constraints added to ensure distinguishability of candidates
- c. 90 constraints, based on X-bar configurations, precedence relations of grammatical functions/NP types (pronominal vs. full), etc.

Experimental set-up

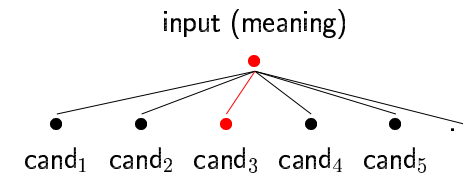
- **Generation-based learning**
 - all possible analyses for all string permutations are generated
 - GLA is applied: i.e., current ranking is used to determine winner
 - check against target winner (constraint demotion/promotion in case of mismatch)
- Goal: generate only strings that are contextually appropriate generation alternatives in German

Learning schemes

Comparison between three schemes:

1. "fully supervised":

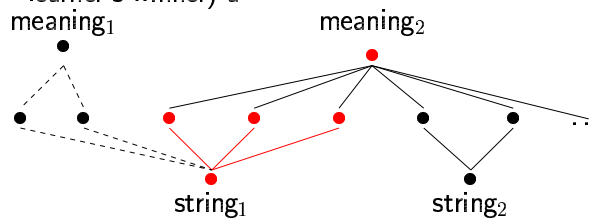
- manual annotation of exact target structure for training clauses
- any other winner counts as error – even with correct word order
- constraints violated only by target winner are demoted



Learning schemes

2. "string-as-target":

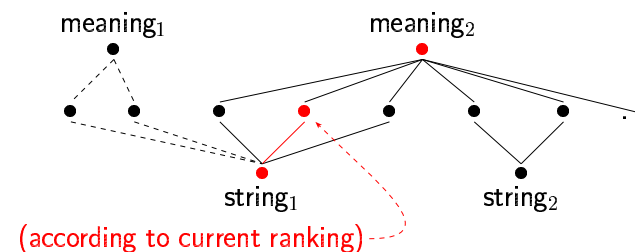
- no manual annotation, all candidates with the right word order count as target winners
- incorrect surface order counts as error
- constraints violated by **any of the target winners** (and not the learner's winner) are demoted



Learning schemes

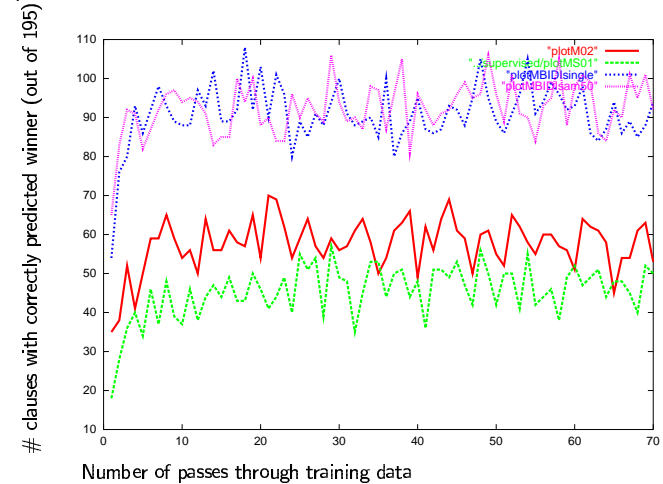
3. bidirectional optimization ("bootstrapping"):

- current ranking determines target winner among parsing alternatives
- all other candidates (possibly with correct surface order) count as errors



Results

Learning curves: supervised, string-as-target, and bidirectional learning
(additional variant: with sampling of stochastic ranking in parsing)



Results

Evaluation schemes:

- percent acceptable orderings on unseen data (manually labeled)
- disambiguation of unseen parsing ambiguities
 - adopting a **bidirectional optimization** scheme (i.e., using the constraint ranking from generation or disambiguation in parsing)
 - using only ambiguous sentences (case ambiguity nom./acc.(/dat.))

daß [die Bundesregierung] [die militärische Zusammenarbeit] wiederbelebt hat
that the federal government the military cooperation revitalized has

nominative – accusative vs. accusative – nominative

Results

Evaluation results (for a specific series of experiments):

- percent acceptable orderings on unseen data

initial ranking	“string-as-target”	bidirectional	“supervised”
34%	66%	87%	90%

- disambiguation of unseen parsing ambiguities

initial ranking	“string-as-target”	bidirectional	“supervised”
54%	76%	84%	83%

Bidirectional bootstrapping approach seems very promising.

Discussion

Limitations:

- robustness of GLA doesn't make it possible to deal with **conflicting ranking arguments** in the general case (without perfect constraint set)
- large set of constraints makes exploratory analysis somewhat opaque

Conflicting ranking arguments

Keller (2000), Keller and Asudeh (2001): set of data unlearnable for GLA

	C_1	C_2	C_3	Candidate occurs (as target winner) with . . .
candidate a.		*		high frequency
candidate b.		*	*	medium frequency
candidate c.	*			low frequency

- Under no ranking will candidate b. come out as the winner
 - b. is *harmonically bounded* by a.: it has all of a.'s constraint violations and some more
 - constraint pattern does occur in linguistic data (for word ordering constraints standardly assumed for German)
- Keller and Asudeh conclude from this that the GLA is incorrect and that furthermore, cumulative effects of constraint violations should be allowed (contra the strict ranking assumption).

Conflicting ranking arguments

- However: harmonic bounding can be detected by inspection of learning data ("off-line")

	C_1	C_2	C_3	Candidate occurs with . . .
candidate a.		*		high frequency
candidate b.		*	*	medium frequency
candidate c.	*			low frequency

impossible winner

- Alternative interpretation of the situation: learning data indicate the lack of a constraint in the constraint set
 - with an adequate constraint set, the data are learnable
 - constraints sensitive to information-structural differences

Conflicting ranking arguments

- More problematic cases: lack of constraint not detectable "off-line", i.e., by inspection of individual data

Categorical case ("classical" OT):

		C_4	C_5	C_6		
(1)	observed	candidate a.		*		$\Rightarrow C_4 \gg C_5$
		candidate b.	*			
			C_4	C_5	C_6	
(2)	observed	candidate a.			*	$\Rightarrow C_5 \gg C_6$
		candidate b.		*		
			C_4	C_5	C_6	
(3)	observed	candidate a.	*			$\Rightarrow C_6 \gg C_4$ – inconsistency!
		candidate b.			*	

Conflicting ranking arguments

- Even in Stochastic OT the problem doesn't necessarily go away:

	C_4	C_5	C_6	Candidate occurs ...	
(1) candidate a.		*		<i>always</i>	⇒ C_4 C_5
candidate b.	*			<i>never</i>	
(2) candidate a.			*	<i>always</i>	⇒ C_5 C_6
candidate b.		*		<i>never</i>	
(3) candidate a.	*			<i>half the time</i>	⇒ C_4
candidate b.			*	<i>half the time</i>	

Conflicting ranking arguments

- ⇒ If constraint set that was used to generate the data is unknown, the learner needs the capability of inducing new constraints/adjusting the violation profile

Potential alternatives to GLA model

- Split model:
 - core OT model, based on fine-grained contextual constraints
 - background model for learning the violation pattern for contextual constraints

problem: it is typically underdetermined which part needs adjustment
- Move from constraint ranking to a constraint weighting scheme (but: typological predictions?); preliminary experiments with log-linear model developed for disambiguation of LFG parsing (Johnson et al. 1999)
 - using training data from “supervised” experiment
 - higher fit on training data
 - comparative evaluation not straightforward

Discussion

Outlook

- Planned: experiments allowing for a more focused analysis of the constraints relevant for variable phenomena
 - start with a classical, manually written grammar for German
 - significantly smaller set of candidates
 - focus on pragmatically motivated generation alternatives (information structural alternatives in word order etc.) – all of which are grammatical
 - establish a set of suitable contextual constraints usable on real corpus data

References

- Anttila, Arto. 1997. *Variation in Finnish Phonology and Morphology*. PhD thesis, Stanford University.
- Boersma, Paul. 1998. *Functional Phonology. Formalizing the interactions between articulatory and perceptual drives*. PhD thesis, University of Amsterdam.
- Boersma, Paul, and Bruce Hayes. 2001. Empirical tests of the gradual learning algorithm. *Linguistic Inquiry* 32(1):45–86.
- Bresnan, Joan. 1996. LFG in an OT setting: Modelling competition and economy. In M. Butt and T. H. King (eds.), *Proceedings of the First LFG Conference*, CSLI Proceedings Online.
- Bresnan, Joan. 2000. Optimal syntax. In Joost Dekkers, Frank van der Leeuw, and Jeroen van de Weijer (eds.), *Optimality Theory: Phonology, Syntax, and Acquisition*. Oxford University Press.
- Bresnan, Joan, and Ashwini Deo. 2001. Grammatical constraints on variation: 'be' in the Survey of English Dialects and (Stochastic) Optimality Theory. Ms., Stanford University.
- Bresnan, Joan, Shipra Dingare, and Christopher Manning. 2001. Soft constraints mirror hard constraints: Voice and person in English and Lummi. In *Proceedings of the LFG 01 Conference*. CSLI Publications. To appear.
- Dingare, Shipra. 2001. The effect of feature hierarchies on frequencies of passivization in English. Master's thesis, Stanford University.

- Dipper, Stefanie, Thorsten Brants, Wolfgang Lezius, Oliver Plaehn, and George Smith. 2001. The TIGER treebank. Ms., Universität Potsdam (Inst. f. Germanistik), Saarbrücken (CoLi), Stuttgart (IMS).
- Grimshaw, Jane. 1997. Projection, heads, and optimality. *Linguistic Inquiry* 28(3):373–422.
- Johnson, Mark, Stuart Geman, Stephen Canon, Zhiyi Chi, and Stefan Riezler. 1999. Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL '99)*, College Park, MD, pp. 535–541.
- Keller, Frank. 2000. *Gradiance in Grammar: Experimental and Computational Aspects of Degrees of Grammaticality*. PhD thesis, University of Edinburgh.
- Keller, Frank, and Ash Asudeh. 2001. Probabilistic learning algorithms and Optimality Theory. Ms., Saarbrücken, Stanford University.
- Koontz-Garboden, Andrew. 2001. A stochastic OT approach to word order variation in Korlai Portuguese. paper presented at the 37th annual meeting of the Chicago Linguistic Society, Chicago, IL, April 20, 2001.
- Kuhn, Jonas. forthcoming. *Optimality-Theoretic Syntax: a Declarative Approach*. Stanford, CA: CSLI Publications.
- Sells, Peter. 2001. *Alignment Constraints in Swedish Clausal Syntax*. Stanford: CSLI Publications.
- Tesar, Bruce B., and Paul Smolensky. 1998. Learnability in Optimality Theory. *Linguistic Inquiry* 29(2):229–268.

- Zinsmeister, Heike, Jonas Kuhn, and Stefanie Dipper. 2002. Utilizing LFG parses for treebank annotation. In *LFG 2002, Athens*.