Language Acquisition in Optimality Theory

Hal Daumé III
ISI@USC

Language Acquisition Reading Group

5 July 2001

Outline

- What is Optimality Theory?
- Computability in OT
- Learning in OT
- Shortcomings
- Possible Solutions
- More Problems

What is OT?

- What is it that all grammars have in common?

  - A set of contraints on well-formedness

- How may languages differ?

  - In which constraints have priority in the case of conflict

What is OT?

- There is a (small) finite set of "universal constraints", $C = \{c_{i_1}, c_{i_2}, \ldots, c_{i_n}\}$
- Each language orders these constraints differently, eg: $\langle c_{i_5}, c_{i_1}, \ldots, c_{i_n}, c_{i_2} \rangle$.
- Constraints are violable
- For constraints $\langle c_1, c_2, c_3 \rangle$, we write, eg, $\langle 2, 0, 1 \rangle$ if c_1 failed twice, c_2 succeeded and c_3 failed once.

More Formally...

- C is a set of constraints
- Gen takes a semantic input and produces structural outputs
- $Marks$ ranks an output according to the constraints
- $Eval$ chooses the best one

For Example...

- Input: $\langle sing(x), x = \text{topic}, x = \text{he}, tense = \text{pres. perf.}\rangle$
- Candidates:

a.	$[_{IP}$	has [		sung]]	
b.	$[_{IP}$ he$_i$	has [	t_i	sung]]	
c.	$[_{IP}$	has [[	t_i	sung]	he$_i$]
d.	$[_{IP}$ it	has [[	t_i	sung]	he$_i$]
e.	...				

Example Constraints

- SUBJ: The highest A-specifier in an extended projection must be filled
- DEPIO: Elements of the output must exist in the input
- DROP-TOP: Arguments coreferent with the topic are structurally unrealized
- AL-FOC: The left edge of a focussed constituent is aligned with the right edge of a maximal projection
- MAXIO: Elements of the input must exist in the output

Example Ranking

	MaxIO	Subj	DepIO	Drop-Top	Al-Foc
a. $[_{IP}$ has [sung]]	*!	*			
⇒ b. $[_{IP}$ he$_i$ has $[t_i$ sung]]				*	
c. $[_{IP}$ has $[[t_i$ sung] he$_i$]		*!		*	
d. $[_{IP}$ it has $[[t_i$ sung] he$_i$]			*!	*	

Remember:

- Subj: The highest A-specifier in an extended projection must be filled
- DepIO: Elements of the output must exist in the input
- Drop-Top: Arguments coreferent with the topic are structurally unrealized
- MaxIO: Elements of the input must exist in the output

Computational Issues

- $Marks$ and $Eval$ are computable, assuming reasonable constraints
- Generation from fully specified f-structures is decidable
- Generation from underspecified f-structures is undecidable

But! infinite structured search spaces may be better than finite non-structured spaces.

We require:

- Relaxed offline parsability/generability
- Restriction on constraints

Learning in OT

Given:

- a set of constraints, C
- a set of positive examples of a language, D

Goal: determine the appropriate ordering of the constraints.

Learning in OT

We decompose the problem into:

- Robust Interpretive Parsing
- Learning the Grammar

Use an iterative algorithm (like EM)

Learning in OT

Two major approaches:

- Constraint demotion (Tesar 1995; Tesar and Smolensky 1998)
- Gradual learning (Boersma 1998)

Intelligent ISI Systems Division

Constraint Demotion

Initially, all constraints are unranked.

We look at each winner/loser pair; eg:

winner	$*c_1$	$*c_2$		$*c_4$	$*c_5$	
loser	$*c_1$		$*c_3$	$*c_4$	$*c_5$	$*c_5$

- Winner-mark
- Loser-mark
- Irrelevant

Constraint Demotion

Minimal demotion: Each of the winner-mark constraints is demoted minimally

winner	$*c_1$	$*c_2$		$*c_4$	$*c_5$	
loser	$*c_1$		$*c_3$	$*c_4$	$*c_5$	$*c_5$

If the previous ranking was $c_2 > c_3 > c_5$, the constraint c_2 would be demoted between c_3 and c_5, yeilding $c_3 > c_2 > c_5$.

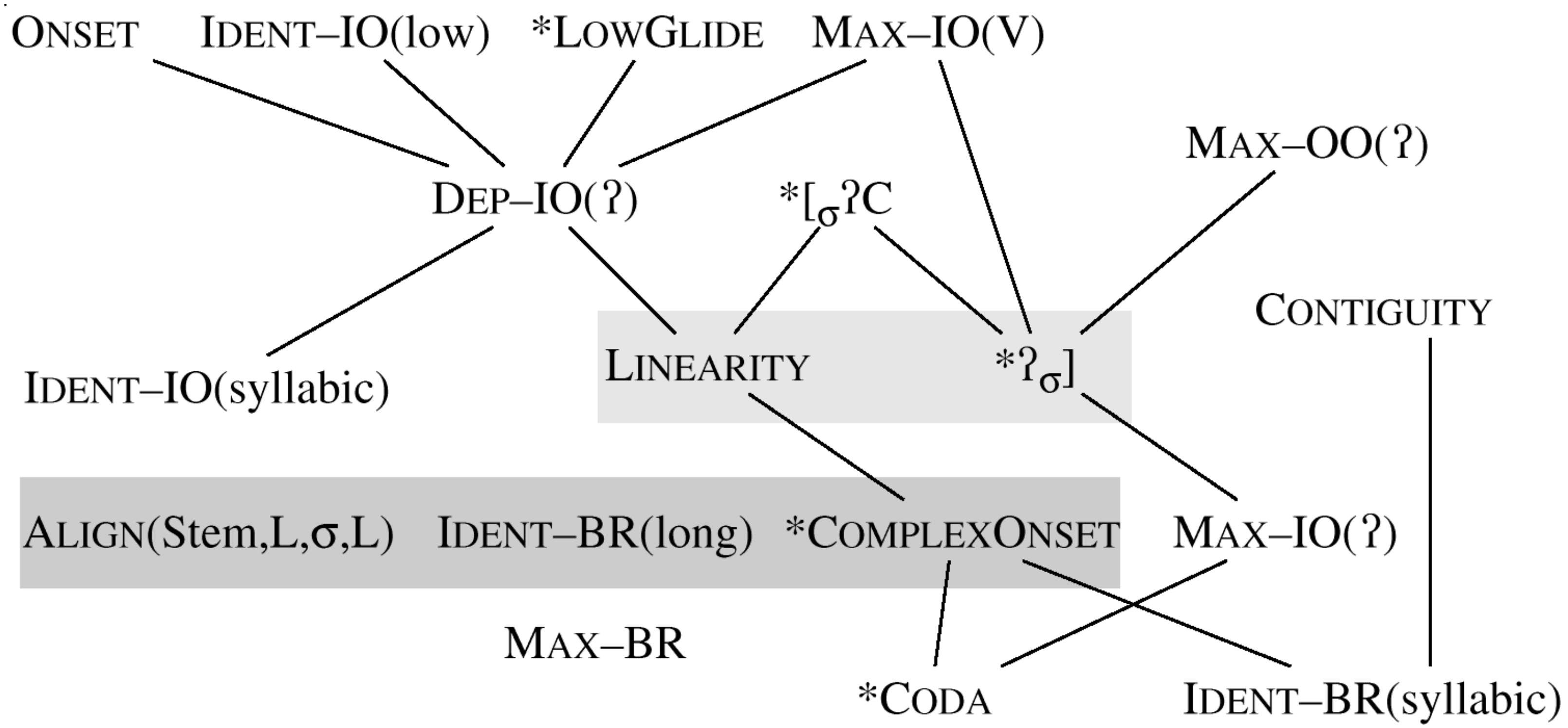
Problems with Constraint Demotion

- Cannot handle noisy data
- Could yield an infinite loop of demotions

Gradual Learning

- Gradual learning algorithm is robust
- Constraints are ranked on a continuous scale
- Constraint evaluation is stochastic

Maximal promotion/demotion system: all winner-marks are demoted slightly and all the loser-marks are promoted slightly.



Gradual Learning Results

Working with phonology data from Ilokano; hand-ranking:

Gradual Learning Results

Automatic ranking (with 21,000 data points):

Constraint	Ranking Value	Constraint	Ranking Value
ONSET	164.00	CONTIGUITY	108.00
LOWGLIDE	164.00	ALIGN	87.96
IDENT-IO(LOW)	164.00	COMPLEXONSET	86.64
MAX-IO	162.00	IDENT-BR(LONG)	85.64
$[_\sigma ?C$	142.00	MAX-IO	80.00
MAX-OO	138.00	MAX-BR	67.60
DEP-IO	130.00	IDENT-BR(SYLL)	55.60
$?]_\sigma$	111.08	CODA	40.36
LINEARITY	110.92	IDENT-IO(SYLL)	-56.00

Gradual Learning Results

Accuracy of predictions:

Input	Output	Actual	Simulation Result
/ta?o-en/	taw.?en	50%	52.5%
	ta?.wen	50%	47.9%
/HRED-bwaja/	bu:.bwa.ja	33.3%	36.7%
	bwaj.bwa.ja	33.3%	31.2%
	bub.wa.ja	33.33%	32.1%
/pa?lak/	pa.lak	100%	100%
/lab?aj/	lab.?aj	100%	100%
/trabaho/	tra.ba.ho	100%	100%

Language-Particular Ineffability

- In a particular language, there may be no grammatical way of stating a certain state of affairs that is perfectly possible in another language:

  Input: $\langle$ eat(x, y); $x =$who, $y =$what $\rangle$

  English: Who ate what?

Italian:	*	Chi	ha	mangiato	che cosa?
		who	has	eaten	what

- This is a well-known problem in OT: optimization is defined in such a way that there is at least one winning candidate for *every* competition
- Assuming competitions are universal, this grammaticality difference is unexpected (and unexplainable)

Possible Solutions

- Assume a highly abstract input, called $Input$ to circumvent potential derivational connotations (Legendre, Smolensky and Wilson 1998)
  - $Input$ defines the candidate sets for optimization
  - It is unrelated to anything playing a role in a particular application of the grammar in language production
  - $Input$ allows candidates in a competition to no longer have the same interpretation

Possible Solutions

Suppose we have two competitions with the following two inputs:

a. $\langle Q_i, Q_j, \text{eat}(x_i, y_i)\rangle$

b. $\langle Q_i, \text{eat}(x_i, y), y = \text{something(indefinite)}\rangle$

In English, the two competitions have two different winners:

a. Who ate what?

b. Who ate anything?

In Italian, neutralization occurs and we get the same same candidate for both:

Chi	ha	mangiato	qualche cosa?
who	has	eaten	anything

What Happened There?

Consider the constraints:

- DEPIO: Elements of the output must exist in the input
- MAXIO: Elements of the input must exist in the output
- ABSORB: Only one *wh*-element is allowed

{DEPIO, MAXIO} =FAITH

In English, FAITH$\ll$ABSORB but in Italian ABSORB$\ll$FAITH

More Problems

- $Input$ works fine for an OT system with a fixed constraint ranking
- It also works for learning with "tagged" data
- *However,* it is totally unnatural that information about $Input$ is context-deducible for the learner
  - By assumption, there is no connection between the utterance situation and $Input$
  - i.e., when an adult Italian speaker says "who ate anything," the learner cannot tell which competition it won
  - Then, there is no way of learning a language with a low-ranking faithfulness constraint
- Why should make the learner assume the more complicated competition rather than the simpler one?

Questions?