Research seminar week 2

Tamás Biró
Humanities Computing
University of Groningen
`t.s.biro@rug.nl`

Niyogi, chapter 1

- Questions?
- Remarks?
- Criticism?

What is a *model* for some phenomenon?

A theoretical construction (with or without maths) whose predictions mimic the observable behaviour of the phenomenon.

$$\begin{array}{ccc} phenomenon & \rightarrow & model \\ \downarrow & & \downarrow \\ behaviour & \stackrel{=?}{\longrightarrow} & prediction \end{array}$$

Linguistic competence

Knowledge of language in brain (Chomsky).

- Its model: a *grammar*.
- Grammar generates for each input the form that is grammatical in the language being described.
- grammatical form $=?$ correct form by native speaker

Correct form according to native speaker, not to academy.

Linguistic performance

- Utterences actually produced by native speaker (standard approach).
- Computation in brain during language production (approach of T.B.).
- Performance model: predicts the forms actually uttered/observable.

Competence vs. performance

- Competence: a function
  $$input \mapsto grammatical\ form.$$
- Performance: an algorithm realizing (implementing) this function.
  Ideally, it only returns grammatical form.

Behaviour of an algorithm

- Precision and error rate.
- Run time, complexity (time, storage space).

What about precision, run time and complexity of human linguistic performance (computation in one's brain)?

Approaches to linguistic competence

- Harmony Grammar (Smolensky)
- Optimality Theory (Prince and Smolensky)
- Hard constraints (Chomsky, among others)
- Principles and Parameters (Chomsky)

(Not in historical order...)

Harmony Grammar (in OT-style)

- Input: I
- *Set of candidates* C generated by GEN function: $C = GEN(I)$.
- *Harmony* $H(c)$: a real-valued function on elements c in C.

- Output (grammatical form, prediction for correct form):

$$O = \arg\max_{c \in C} H(c).$$

$$O(I) = \arg\max_{c \in GEN(I)} H(c).$$

Languages differ because they have different grammars, i.e., different $H(c)$ functions.

Example for HG: word stress

Language A: *banána*, Language B: *bánana*.

[banana]	H_A	H_B
bánana	3.7	**4.4**
banána	**5.9**	-2.3
bananá	-2.1	0.9

Original idea: $H(c)$ defined by sum of weights in a connectionist network representing the candidate.

Harmony Grammar (in OT-style)

$H(c)$: weighted sum of elementary functions ($constraints$) defined on the candidate:

$$H(c) = w_1 \cdot c_1(c) + w_2 \cdot c_2(c) + ... + w_n \cdot c_n(c)$$

Constraints are universal, weights are language dependent.

Example for HG: word stress

$c_1(c)$: number of syllables intervening between left edge of c and stess.

$c_2(c)$: number of syllables intervening between stress and right edge of c.

$c_3(c)$: number of stress on the last syllable of c (either 0 or 1).

Example for HG: word stress

Language A: $w_1 = -0.5$, $w_2 = -1.2$, $w_3 = -3.1$.

Language B: $w_1 = -4.8$, $w_2 = -1.2$, $w_3 = -0.1$.

[banana]	c_1	c_2	c_3	H_A	H_B
bánana	0	2	0	-2.4	**-2.4**
banána	1	1	0	**-1.7**	-6.0
bananá	2	0	1	-4.1	-9.7

HG with exponential weights

$$H(c) = w_1 \cdot c_1(c) + w_2 \cdot c_2(c) + ... + w_n \cdot c_n(c)$$

where $w_i = -q^i$ for some $q > 1$.

$$H(c) = -c_n(c) \cdot q^n - ... - c_2(c) \cdot q^2 - c_n(c) \cdot q$$

If q very large: Optimality Theory.

Optimality Theory (OT)

- No weights w_i, but *constraint hierarchy*.
- *Strict domination*.
- Only the best candidates survive a constraint (at least one; single best or equally good ones).

Optimality Theory: example 1

$c_1 \gg c_2 \gg c_3$

[banana]	c_1	c_2	c_3
bánana	0	2	0
banána	1!	1	0
bananá	2!	0	1

Winner: *bánana* $\rightarrow$ Language B.

Optimality Theory: example 2

$c_3 \gg c_2 \gg c_1$

[banana]	c_3	c_2	c_1
bánana	0	2!	0
banána	0	1	1
bananá	1!	0	2

Winner: *banána* $\rightarrow$ Language A.

Pseudo-code for Optimality Theory

```
ALGORITHM Evaluation in OT
   Input: input form I, constraint hierarchy CH

CS <-- candidate set corresponding to input I

FOR constraint con in CH (strongest to weakest)
    CS <-- subset of CS that is best for con
NEXT con

Return CS      # the best candidate(s)
               #  corresponding to I w.r.t. CH
```

OT and HG

- Constraints and candidates: universal.
- Cross-linguistic differences due to weights/hierarchies.
- HG: any weights;
  OT: large weights = strict domination.
- Grammatical form can violate constraints.

Hard constraints

- Grammatical form *cannot* violate constraints.
- Constraints are language specific.

Hard constraint: example

Language A: *banána*

- Constraint 3: no stress on last syllable.
- Constraint 2: minimize number of syllables between stress and right edge.

Language B: *bánana*

- C 1: no syll between left edge and stress.

Principles and Parameters (P&P)

- System in the set of constraints.
- Principles: universal constraints, with a detail left open.
- Parameters: language specific ways of filling that detail; usually $binary$.

Principles and Parameters: example

- Principle 1: minimize the distance number of syllables intervening between stress and [Parameter 1: left/right] edge.
- Principle 2: stress on last syllable is [Parameter 2: yes/not] allowed.

	Param 1 = left	Param 1 = right
Param 2 = yes	*bánana*	*bananá*
Param 2 = no	*bánana*	*banána*

Language typology

Languages can be divided into 3 types:

- Type 1: stress on first syllable.
- Type 2: stress on last syllable.
- Type 3: stress on penultimate syllable.
- (And more...) But (almost) no language with stress on second syllable as a rule!

Language typology

Languages predicted possible by a model:

- HG: use all possible weight combinations.
- OT: use all possible hierarchies.
- P&P: use all possible parameter combinations.

Home work (send me as an email)

Show that both the OT model and the P&P model explain why there are three types of languages, and why the fourth type does not exist. Hint: use a four-syllable-word!

If you like math: prove also that even our HG model gives the same prediction.

Next week:

- More examples.
- Performance models.

Start implementing of OT / HG in groups.