From Performance Errors to Optimal Competence Learnability of OT and HG with Simulated Annealing

TAMÁS BIRÓ

University of Amsterdam, Netherlands
t.s.biro@uva.nl

Can you look into the brain of the teacher?

Online learning from teacher’s performance.

Any need to learn until convergence on the grammar? Until the learner finds the teacher’s set of ranks?

String Grammar: “toy grammar” imitating typical OT phonology:

Competence as linguistic optimization

Generative linguistics as an optimization problem: how to map underlying form $U$ onto surface form $SF(U)$?

$SF(U) = \text{any opt. } H(w)$

Target function (“Harmony”) $H(w)$ derived from elementary functions $C_i(w)$ (“constraints” — a misnomer):

1. Hard constraints: $H(w) = C_i(w) \land C_i(w) \land \ldots \land \neg C_i(w)$ $\rightarrow$ P&P

2. Weighted sum: $H(w) = \sum_{i} w_i \cdot C_i(w) + \sum_{i} w_i \cdot C_i(w) \rightarrow$ HG

3. Exponential weights: $H(w) = C_i(w) \cdot C_i(w) \cdot \ldots \cdot C_i(w) \cdot q \rightarrow$ PH

4. OT tableau row: $H(w) = \sum_{i} C_i(w) \sum_{i} C_i(w) \sum_{i} C_i(w)$ $\rightarrow$ OT

Rank $r_i$ for each constraint $C_i$.

Grammar = set of ranks.

Set constraints by rank:

- OT hierarchy: $C_i \geq C_j$ iff $r_i > r_j$.
- k-best: $\prod_{i \in \mathcal{K}} q_i \rightarrow$ PH
- 2-best: $\prod_{i \in \mathcal{K}} q_i \rightarrow$ PH

In order of OT: grammatical forms, irregular forms and fast speech forms are returned (Biró 2007): OT much easier to learn than HG.

Distance of performances: JSD.

Learn until convergence: until no significant difference between two performance samples.

Converging on performance: $\Delta$ is Jensen-Shannon divergence

- Convergence criterion: JSD between sample produced by target grammar and sample produced by learner’s current grammar $\leq$ average JSD of two samples produced by target grammar. (Sample size = 100).

- A measure of the “distance” of two distributions $JSD(P(Q)) = \frac{1}{2} \sum (P \log \frac{P}{Q} + Q \log \frac{Q}{P})$.

- Symmetric: $JSD(P(Q)) = JSD(Q(P))$. Finite and non-negative: $0 \leq JSD(P(Q)) \leq 1$.

- $JSD(P(Q)) = 0$ if and only if $P(x) = Q(x)$, $\forall x$. $JSD(P(Q)) \neq 0$ if and only if $P(x) \neq Q(x)$, $\forall x$.

- Same sample: $JSD(L \| L) = 0$. Not a single overlap: $JSD(L \| L) = 1$.

- String Grammar: “toy grammar” imitating typical OT phonology:

  - Constraints: $\mathcal{G}(U) = \{0, \ldots, P - 1\}$. We have used $L = P = 4$ (0001, 0120, 0121, 3112).

  - Neighborhood structure: the candidate set $w$ and $w'$ neighbors of one-bit step transitions to $w'$:

    Basic step: change exactly one character $x_i$ (mod $P$) cyclically. Each neighbor with equal probability. Example: neighbors of 012 are exactly 1123, 3213, 0213, 0133, 0122 and 0121.

  - Constraints (for all $x_i \in \{0, \ldots, P-1\}$):

    - No-change (number of character $x_i$ in string): $\text{No-Ch}(u) = \sum_{i} (x_i = u_i)$.

    - No-initial-$x$: $\text{No-Init}(x_i) = (x_i = u_i)$.

    - No-final-$x$: $\text{No-Final}(x_i) = (x_i = u_i)$.

    - Assortment (number of adjacent character pairs): $\text{Assort}(u) = \sum_{i} (u_{i+1} = u_i)$.

    - Dissortation (number of adjacent dissimilar character pairs): $\text{Dissort}(u) = \sum_{i} (u_{i+1} \neq u_i)$.

    - Faithfulness to underlying form $U$ (using pointwise distance $\text{dist}(P)$)

    - $\text{Faith}(u) = \sum_{i} \text{dist}(u_i, u_i)$, where $\text{dist}(u_i, u_i) = \max \{a-b, 0\}$ and $P(i) = (b-a)$ and $P(i)$.

- Experiment: Measuring number of learning steps

  2000 times learning (red target, red underlying form) per grammar type, production method and learning method. Distribution of the number of learning steps until convergence:

  - Prod. method: deterministic.
  - Prod. method: stochastic.

- Observation: very long tail (significance based on Wilcoxon rank-sum test)

  - Performance errors make grammars more difficult to learn: $\text{gram} < 0.1 \text{sa} < 1$.

  - But not so for HG and SA (either significant, or not significant tendency). Why?

  - Biró’s update rule (M) quicker than Boersma’s (B) (extremely significant). Due to large update steps?

  - Grammar type (OT vs. HG): only minor influence (“hardly any”) and “small, but very significant”.

  - OT much easier to learn than HG (significant difference for sa cases). NB: also quicker to produce.

  - Does learning speed depend on initial grammar? On order of learning data?

  - New experiment: Run two learners learning the same target grammar.

  - With same initial grammar: strong correlation in their nr. of learning steps.

  - Learning data not the same: slightly decreased correlation.

  - With different initial grammars: correlation (almost) lost, large differences in learning time.

  - Con: long tail: children must start with same initial grammar, but need not receive same (correct or erroneous) data at asynchronous cases: SL is being born with initial grammar coming 20-30 times higher learning steps.

  - Stability of the algorithms: if learner reached target, will she stay there? Much improvement needed.

References