Multilingual Learning with Parameter Co-occurrence Clustering

Jason Riggle, Max Bane, James Kirby, John Sylak
Department of Linguistics, University of Chicago
Correspondence: bane@uchicago.edu

NELS 39, Cornell
November 8, 2008

1 The problem of multilingual learning

This talk is about "multilingualism" in the broadest sense:

- The knowledge and use of many distinct linguistic systems by individual speakers (listeners).

Multilingualism under this definition is pervasive. Virtually all human beings can distinguish and use multiple linguistic systems — some quite similar to each other, others very different — at various points on the language–dialect–register continuum.

Some examples:

- Native bilingualism and code-switching.
- "Multi-dialectism". For instance, Clopper (2004) provides an extensive study of American English speakers' abilities to distinguish and categorize multiple dialects, as well as to employ multiple dialects natively, and even imitate them non-natively.
- Biber (1995) gives a detailed cross-linguistic survey of register variation. Language users make use of systematically different phonological, morphological, and syntactic forms — in effect, distinct linguistic systems — depending on social and conversational context.

On this view, what does one know when one knows, say, English? A collection of mostly overlapping but different systems of communication (registers, dialects, others' idiolects), perhaps acquired to different degrees, together with when to use them. A bilingual English-French speaker knows an even larger collection of this sort.

⇒ How are people able to acquire, and distinguish among, the member systems of these complex collections?

We present a strategy (class of algorithms) for modeling how this might be accomplished within a parameterized linguistic theory. Though the approach outlined here is applicable at any level of linguistic analysis (phonology, morphology, syntax, ...) and in any theory that has a notion of parameters (with some caveats), for purposes of concreteness and testing/evalution, we'll be working within an optimality theoretic (OT; Prince & Smolensky, 1993) framework for syllable shape phonotactics.

1.1 The monolingual problem

Linguistic systems are often formalized as mappings from some set of inputs (underlying representations) to a set of possible outputs (surface representations): $L : I \to O$

For example:

(1) Syllabification

Input		*Output*
CVCCV	↦	.CV.CCV.

(2) Morphology

Input		*Output*
PastPart(buy)	↦	bought (Most varieties of English)
PastPart(buy)	↦	boughten (Some varieties of English)
PastPartMasc(acheter)	↦	acheté (Most varieties of French)

(3) Syntax/Semantics

Input		*Output*
$\exists t(t < n \wedge \textsc{kiss}(\text{John})(\text{Mary})(t))$	$\mapsto$	Mary kissed John. (Most varieties of English)

The potentially infinite mapping L is usually described in terms of some finite set of parameters (i.e., a grammar) g that define it according to some theory $\mathcal{G}$. The problem faced by the learner of a language can be stated as:

- Given a finite sample $S = \{(i_1, o_1), \ldots, (i_n, o_n)\}$ of input-output pairs, what parameter settings g might define the mapping $L = \mathcal{G}(g)$ that generated them?

Whatever hypothesis the learner arrives at for L, it should be usable to predict what outputs correspond to inputs that were not seen in the sample (i.e., the learner *generalizes* from S).

- *Supervised Learning*: The learner's sample S contains (input, output)-pairs.
- *Unsupervised Learning*: S contains only outputs. The learner must hypothesize the underlying inputs in addition to the mapping that generates the observed outputs from them.

We will assume a supervised setting (following much recent linguistic work, e.g., Tesar & Smolensky 2000, Boersma & Hayes 2001, Riggle 2004).

1.2 The multilingual problem

The multilingual learning problem requires only a simple modification to this framework, but is significantly more difficult:

- Given a finite sample $S = \{(i_1, o_1), \ldots, (i_n, o_n)\}$ of input-output pairs drawn from *multiple languages*, what *set of grammars* might define those languages according to $\mathcal{G}$?

A major source of difficulty in this problem is illustrated by the hypothetical case in (4).

(4) a. L_1: /VC/ $\mapsto$ [CVC]
b. L_2: /VC/ $\mapsto$ [V]

Suppose one language represented in the sample, L_1, epenthesizes (inserts) onsets onto syllables lacking them (4a), while another, L_2, deletes syllable codas (4b). If the learner is to acquire both L_1 and L_2 as distinct languages, then his or her challenge is to avoid generalizing to a grammar of L_3, which does both:

$$L_3 : /\text{VC}/ \mapsto [\text{CV}]$$

Existing models of learning variation within a single language (e.g., Boersma & Hayes 2001), if adapted to the present situation, where the learning sample is a union of observations from several languages, will do just this. Some other strategy is needed.

Two opposing pressures:

- The need to distinguish or separate the languages represented in the sample.
- The need to accomodate the possibility that the target languages might be highly similar, and overlap significantly.

2 A strategy: Tracking parameter co-occurrence

Our strategy depends on the following assumption:

(5) Given a single (input, output)-pair, it is possible to determine (efficiently) which parameter settings are consistent with that pair — i.e., which grammars define a language containing that pair.

- For example, within Finite State OT, we can use Prince's (2002) Elementary Ranking Conditions (ERCs) to express which grammars (constraint orderings) are consistent with a given observation.

With this assumption met, the following batch-learner strategy for multilingual learning is possible:

1. Begin with an empty, unweighted, undirected "co-occurrence graph".
2. For each observation (i, o) in the sample:
   (a) Construct a list of statements about which properties a grammar would have to possess in order to be consistent with seeing that observation (in OT, a list/conjunction of ERCs).
   (b) For each statement in that list, add a node to the co-occurrence graph, and add an edge between between each pair of those nodes.

   After doing this for the entire sample of observations, the co-occurrence graph reflects which grammatical properties were seen to be consistent with the sample, and the edges in the graph indicate which grammatical properties were seen to be *simultaneously required for a single observation.* See Figure 1.

   Hypothesis: Intuitively, the "dense" or highly connected, mutually consistent, regions of the graph tend to correspond to the grammars of the individual languages from whose mixture the sample was drawn.
3. Apply some heuristic to identify the dense regions, or clusters, of the co-occurence graph, and adopt hypothesis grammars that are consistent with the grammatical properties specified by those clusters.

- Essentially, steps 1 and 2 of the strategy serve to reduce the multilingual learning problem to one that is heavily studied in the fields of computational learning theory and data mining: clustering.
- ...But with the added requirements that the clusters might overlap, and should specify a mutually non-contradictory set of grammatical properties.
- Depending on one's clustering heuristic, there is often no need to presuppose the number of languages represented by the sample.

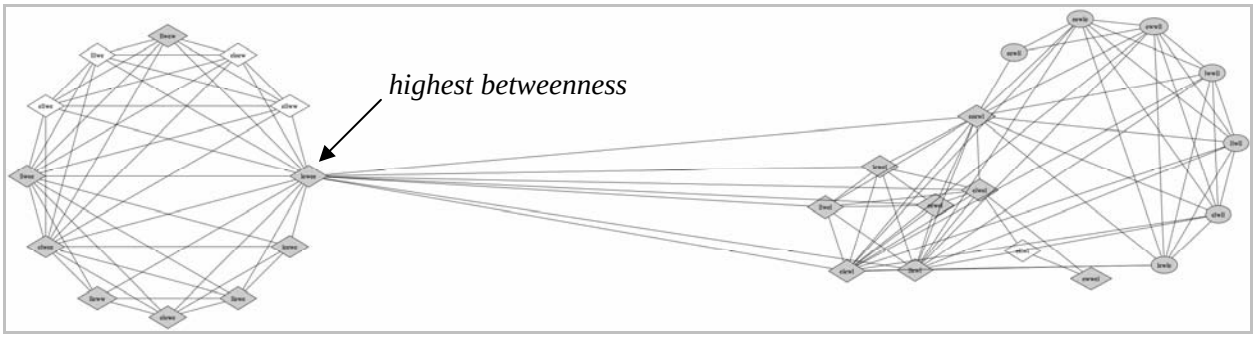

Figure 1: A co-occurrence graph constructed from a sample of the language mixture of two 5-constraint Finite State OT grammars. Note that there are at least two dense regions. The learning strategy would be to adopt these regions of grammatical properties as defining the learner's hypothesis about the grammars that produced the mixed sample.

- There are many possible clustering heuristics for Step 3.
- We've experimented with a rather simplistic one based on the graph theoretic notion of "centrality".

The "betweenness centrality" of a node in a graph (Freeman, 1977):

- The betweenness centrality of a node n in a graph G is the proportion of shortest paths between pairs of nodes in G that pass through n.

Intuitively speaking, betweenness centrality can be used to identify dense regions in a graph by locating the nodes which lie on their edges, that is, the nodes *between* dense areas. Figure 1 indicates the "betweenest" node in an example co-occurrence graph.

Our algorithm:

(6) **Betweenness centrality clustering heuristic**

1. Begin with an empty set H of hypothesis grammars.
2. For each connected component C of the co-occurrence graph G:
   (a) If C represents a set of mutually consistent grammatical properties, construct a hypothesis grammar consistent with C and add it to H. Remove C from G.
   (b) Otherwise, find the node v in G with the greatest betweenness centrality, and tentatively remove it from G to determine what new connected components $k_1, \ldots, k_n$ appear in G as a result of its loss. Add copies of v back to each of $k_1, \ldots, k_n$, reconnecting those copies to whichever nodes in $k_1, \ldots, k_n$ shared an edge with v before its removal. Return to Step 2.
3. Greedily attempt to merge any hypothesis grammars in H that agree with each other on all inputs.

3 Testing the strategy

We have been experimenting with a number of Monte Carlo simulations to assess the learning algorithm's behavior.

- The general experimental setup:
  - Finite State OT grammars of syllable shape phonotactics, similar to those described by Riggle (2004) — 13 constraints over the alphabet $\{\mathrm{C}, \mathrm{V}\}$, with inputs drawn from a fixed lexicon of all forms from $\{\mathrm{C}, \mathrm{V}\}^*$ with length 5 or less (62 forms). 679 distinct possible languages.
  - A learner applying algorithm (6) to learn the underlying grammars from a mixed sample of multiple languages.
  - The learner's sample contains (input, output)-pairs for only a portion of the lexicon.
  - Target language grammars randomly generated each trial.
  - Results averaged over many trials (100 to 1,000, depending on complexity of experiment).
- The algorithm performs reasonably well on small collections of target languages, though it tends to overestimate the number of languages present in the sample.
- We can assess how different a set of hypothesis grammars is from the target set by calculating an "agreement displacement" quantity, which indicates the degree to which the grammars in the two sets disagree on how (input, output)-pairs should be mapped.
- Figures 3–4 take a closer look at the bilingual case (i.e., the sample contains a mixture of two languages).
  - Figure 3 shows that, in terms of the agreement displacement of its hypotheses, the algorithm performs best with around 10–20 samples (out of a lexicon of 62 forms).
  - Figure 4 indicates that when the algorithm acquires hypotheses different from the target set, they are nontheless likely to be similar to the target set.

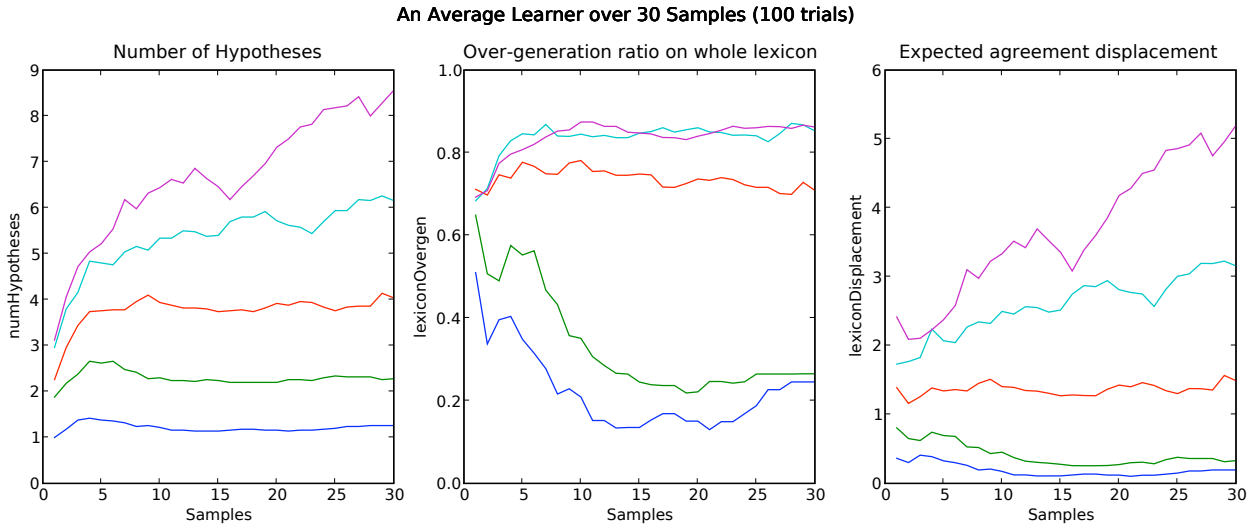

Figure 2: The algorithm tends to infer too many hypothesis grammars.

4 Future Work

- Larger experiments testing the limits of the strategy.
- Weighted co-occurrence graphs and more sophisticated clustering techniques (spectral, Bayesian, etc.)
- Corpus analysis: e.g., distinguishing the registers present in a large body of transcribed telephone conversations.

References

Biber, D. (1995). *Dimensions of Register Variation*. Cambridge University Press.

Boersma, P. & B. Hayes (2001). Empirical tests of the Gradual Learning Algorithm. *Linguistic Inquiry* 32, pp. 45–86.

Clopper, C. G. (2004). *Linguistic Experience and the Perceptual Classification of Dialect Variation*. Ph.D. thesis, Indiana University.

Freeman, L. C. (1977). A set of measures of centrality based on betweenness. *Sociometry* 40, pp. 35–41.

Prince, A. (2002). Entailed ranking arguments. *Rutgers Optimality Archive* ROA-500.

Prince, A. & P. Smolensky (1993). Optimality theory: Constraint interaction in generative grammar. Ms., Rutgers University and University of Colorado, Boulder.

Riggle, J. (2004). *Generation, Recognition, and Learning in Finite State Optimality Theory*. Ph.D. thesis, UCLA.

Tesar, B. & P. Smolensky (2000). *Learning in Optimality Theory*. MIT Press.

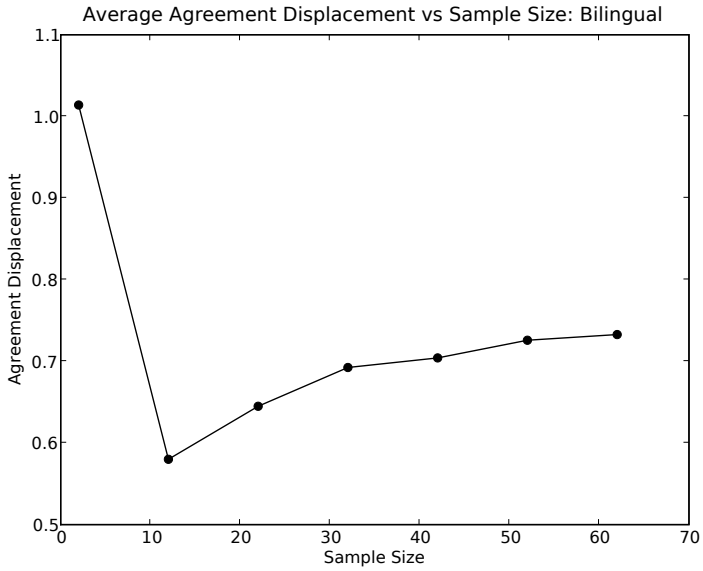

Figure 3: The algorithm performs best with around 10–20 samples.

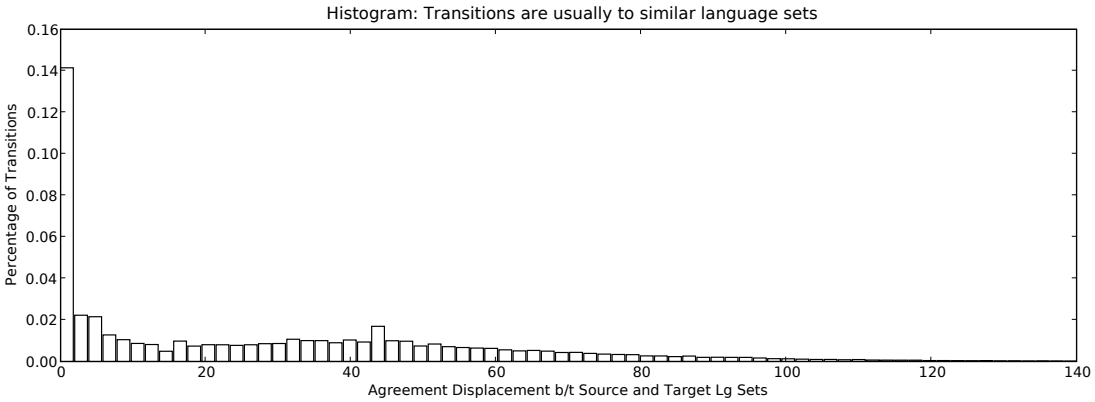

Figure 4: The algorithm is likely to acquire sets of langauges similar to the target set.