Multilingual Learning with Parameter Co-occurrence Clustering

Max Bane, James Kirby, Jason Riggle, John Sylak

University of Chicago

1. Introduction

Multilingualism may be viewed in the broadest sense as the knowledge and use of many distinct, though possibly overlapping linguistic systems by individual speakers. It is, according to this view, a pervasive phenomenon, with virtually all language users—even those commonly called monolingual—being able to distinguish and employ multiple linguistic systems at various points on the language-dialect-register continuum. In addition to the canonical examples of native bilingualism, code-switching, etc., this definition will include such things as the "multi-dialectism" described in Clopper's (2004) extensive study of American English speakers' abilities to distinguish and categorize multiple dialects, as well as to produce multiple dialects natively and even imitate them non-natively. Also included is the case of register variation, wherein speakers make use of systematically different phonological, morphological, and syntactic forms and processes (in effect, distinct systems of communication) depending on social and conversational context; Biber (1995) gives a detailed cross-linguistic survey. In this view, then, a speaker who "knows a language" like English, Cantonese, Palauan, or Guaraní, in fact knows a collection of mostly overlapping, yet distinct, systems of communication, including registers, dialects, and others' idiolects, some perhaps acquired to different degrees, or only passively (i.e., to be recognized and distinguished, but not produced), along with some specification of their contexts of use.

A fundamental question, then, is how language learners in a pervasively multilingual environment, where they receive mixed samples from many distinct linguistic systems, can manage to distinguish the component systems and acquire them separately. For example, if a learner is exposed to languages L_1 and L_2, where L_1 epenthesizes onsets and L_2

Draft of March 1, 2009. To appear in NELS 39.

deletes codas, how can it avoid generalizing to a third language L_3, which does both?

(1) a. L_1: /VC/ → [CVC]
b. L_2: /VC/ → [V]
c. *L_3: /VC/ → [CV]

The computational task of language learning has long been a central issue in theoretical linguistics, and most work has focused on its monolingual formulation, in which the learner's sample is drawn from a single target language (though see Müller-Lancé 2003, Genzel 2005, Snyder and Barzilay 2008 for some recent work on the multilingual task). This paper considers a minimal extension of the usual monolingual formulation to accommodate the multilingual setting, and presents a novel strategy for discriminating and learning languages within it by clustering grammatical properties according to their co-occurrence in the sample. The heuristic that we propose is generic in the sense that it is applicable within any parameterized linguistic theory for which it is feasible to compute the possible parameter-settings implied by observing a single input-output mapping; for purposes of concreteness and evaluation, we present the algorithm within the framework of Optimality Theory (OT; Prince and Smolensky 1993), using syllable structure grammars as a case study.

In what follows, Section 2 reviews the computational problem of language learning (§2.1), describes how we approach multilingual learning in this framework (§2.2), and briefly introduces Prince's (2002) method of parameterizing OT grammars within a 3-valued logic (§2.3), which will prove useful for our case study. The learning algorithm we propose is then presented in Section 3, beginning with the central idea of creating a graph of parameter co-occurrence relations (§3.1), which essentially reduces the multilingual learning problem to that of detecting clusters in a graph. We detail one possible means of performing this detection, based on a graph theoretic measure of centrality (§3.2), and report some encouraging results of this method, as assessed by a battery of Monte Carlo simulations applied to the syllable structure case study (§3.3). We then discuss a straightforward way in which our strategy can be extended by the incorporation of extralinguistic information into the graph-building phase of the algorithm (§3.4). Section 4 finally offers some summarizing and concluding remarks.

2. The Multilingual Learning Problem

2.1 Learning as Parameter Estimation

We will represent a generative linguistic system L as a mapping from some set of inputs I (underlying representations) to a set of possible outputs O (surface representations), $L : I \rightarrow O$. The input and output sets I and O are typically infinite, corresponding to all valid linguistic representations in some domain (phonological, syntactic, etc.) With this formalization, the learner of a language faces the following problem.

(2) Given some finite sample $S = \{(i_1, o_1), \ldots, (i_n, o_n)\}$ of example input-output pairs ($i_1, \ldots, i_n \in I$; $o_1, \ldots, o_n \in O$), what is the mapping L that generated them? I.e., what is the L such that $\{(i_1, L(i_1)), \ldots, (i_n, L(i_n))\} = S$?

Given that the set of inputs to be mapped to outputs is infinite (or in the case of a finite set, assuming that the sample does not exemplify the mapping on every input), whatever hypothesis the learner arrives at for L will necessarily predict outputs for novel inputs not present in the sample; that is, the learner must *generalize* from the sample.

This framing of the problem corresponds to the "supervised" case, in which the learner is assumed to observe inputs and must only infer the relationship between inputs and outputs, as opposed to the "unsupervised" case, in which only output forms are present in the learner's sample, so that the input forms themselves must also be inferred. We follow the general trend of recent linguistic learnability work (e.g., Tesar and Smolensky 2000, Boersma and Hayes 2001, Riggle 2004) in focusing on the supervised case here. The monolingual learner, then, is represented as an algorithm, A, that is fed a sample, S, of input-output pairs drawn from some target mapping (language) L. This algorithm returns a hypothesized mapping, H, whose similarity to the target may be assessed in a variety of ways, as a function of sample length and other considerations (cf. Gold 1967, Angluin 1980 and others).

A crucial factor affecting the learnability of a class of languages is the way in which it is *parameterized*. We assume that each language mapping, though potentially infinite in extent, is describable by a finite grammar, or collection of parameters, g. The language determined by any particular grammar is defined by a function $\mathscr{G}$, which we can identify with Universal Grammar: $L = \mathscr{G}(g)$. In general, many distinct grammars (parameter settings) may determine the same input-output mapping. The learner's problem amounts to estimating the parameters under which $\mathscr{G}$would yield a language consistent with the sample it has observed:

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

2.2 Multilingual Learning: Learning Mixtures

The multilingual learning problem can be described in the same vein, but now we suppose that the learner's sample contains input-output pairs drawn from some *set* of languages. It is this set, rather than any particular language, that constitutes the learner's "target."

(4) 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 $\mathscr{G}$? That is, hypothesize some set of grammars $\{g_1, \ldots, g_k\}$ yielding a set of languages $\mathscr{L} = \{L_1 = \mathscr{G}(g_1), \ldots, L_k = \mathscr{G}(g_k)\}$ such that for each $(i_j, o_j) \in S$ there is an $L_m \in \mathscr{L}$ such that $L_m(i_j) = o_j$.

One source of difficulty for the learner is that we do not assume any prior knowledge of how many target languages are represented in the sample; this is part of what the learner must infer, in addition to the natures of the languages themselves. This may be likened to learning a mixture of parameterized functions for which the number of mixture components is not known in advance. In general, one can suppose that not every language in the target set is represented with equal frequency by the sample, so that the learner must additionally

determine the relative weight, or probability of representation, of each language. For simplicity, though, we will assume throughout this paper that each language in the target set is equally represented in the learner's sample; that is, each language contributes the same number of example input-output pairs (though some languages may agree with each other on the output for a given input, so that a pair might exemplify several languages though it was contributed to the sample by just one).

A major part of the learner's task, then, is deciding how to "carve up" the sample it receives, solely on the basis of the information contained within it, into the portions that represent the different hypothesis languages that generated them. While doing so, it faces two opposing pressures: the need, on the one hand, to separate the different languages that are present, and on the other hand, the need to accommodate the possibility that some of the target languages might overlap to some degree, agreeing on the output forms for potentially many inputs. To be successful it must avoid positing spuriously many hypothesis languages, making more distinctions than are present in the target set, and it must avoid hypotheses that "merge" what should be distinct languages that happened not to conflict with each other in the sample received. This last sort of error is partly exemplified by the case in (1), and is worth detailing further.

Suppose that the learner's sample represents two languages, one of which, L_1, adds onsets to syllables that lack them underlyingly, while the other, L_2, deletes underlying codas. Then the learner might receive a sample like the following (subscripts indicate the source language of the input-output pair):

(5) $S = \{(/\text{VC}/, [\text{CVC}])_{L_1}, (/\text{CV}/, [\text{CV}])_{L_1}, (/\text{VC}/, [\text{V}])_{L_2}, (/\text{CVC}/, [\text{CV}])_{L_2}\}$

Here, depending on the particular parameterization employed, the learner may have no trouble inferring that two distinct languages are represented, since the pairs (/VC/,[CVC]) and (/VC/,[V]) disagree with each other on the same input, implying that one language epenthesizes while some other language deletes.[1] Such distinguishing and helpful conflicts are not guaranteed to occur in the learner's sample, however. Consider the following:

(6) $S = \{(/\text{V}/, [\text{CV}])_{L_1}, (/\text{CV}/, [\text{CV}])_{L_1}, (/\text{VC}/, [\text{V}])_{L_2}, (/\text{CVC}/, [\text{CV}])_{L_2}\}$

Now the learner observes both onset epenthesis (/V/→[CV]) and coda deletion (/VC/→[V], /CVC/→[CV]), but no input on which there is a conflict between these two processes. The learner may then very well posit that the sample represents a language that performs both, though such a language is not in fact in the target set.

Within the context of OT, there are existing models that represent free variation as a kind of "multilingualism," with variation arising from a multitude of grammars that have been acquired by learners (e.g., Anttila 1997, Boersma and Hayes 2001, Anttila 2008), and a natural question is whether they might gainfully be applied to the present case of multilingual learning in general. However, if adapted to the multilingual learning problem as formulated here, these models will always commit this kind of "merge" error. This is by design, because the models are intended to capture free variation by learners acquiring what we might call an *inclusive union* of grammars, whereas our problem calls for learning

[1] This implicitly assumes that language mappings are not parameterized in such a way as to allow both outputs for the same input. This is the case, for instance, in OT, where it is usually intended that a total ordering of the constraints picks out exactly one optimal output per input.

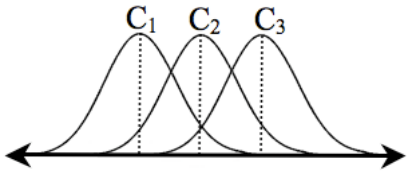

Figure 1: In a model like that of Boersma and Hayes (2001), variation between grammars $C_1 \gg C_2 \gg C_3$ and $C_3 \gg C_2 \gg C_1$ implies non-zero probability for all six possible grammars over the three constraints, since the Gaussian distribution of each constraint's ranking value must overlap with the others.

exclusive disjunctions of grammars. In a Stochastic OT-style model (Boersma and Hayes 2001), for instance, variation between the rankings $C_1 \gg C_2 \gg C_3$ and $C_3 \gg C_2 \gg C_1$ must imply non-zero probability for each of the six possible rankings of $\{C_1, C_2, C_3\}$ (see Figure 1).

The multilingual learning problem as we've laid it out here requires the learner to identify the set of languages from which its sample is drawn based solely on the information offered by the mappings exemplified in that sample. That is, no recourse to extralinguistic information is possible. There is in fact some evidence that such information can be exploited by human learners (e.g., Weikum et al. 2007), and as it turns out, the learning heuristic that we describe below can rather straightforwardly be adapted to benefit from something like extralinguistic cues, however one wishes to parameterize them (detailed in §3.4).

2.3 Elementary Ranking Conditions: Inferring Parameters in OT

In Section 3 below, a crucial assumption of the multilingual learning algorithm that we describe is that one can make inferences about parameter settings on the basis of individual observed input-output pairs. This assumption can be met in Optimality Theory by the use of Elementary Ranking Conditions (ERCs; Prince 2002), which we review briefly here.

In OT, the output form that a language L maps an input to is determined by a ranking of a set of constraints on such pairings, CON. This constraint set is typically supposed constant across languages (which is to say it is incorporated into $\mathscr{G}$), so that the grammar g of a language is simply a total ordering of CON; a language $L = \mathscr{G}(g)$ maps input i to whichever member $o \in O$ yields the pair (i,o) that incurs the fewest violations of the highest ranked (by g) constraint on which (i,o) differs from any other (i,o_j) for $o_j \in O, o_j \neq o$. Thus, modulo a fixed CON, a language is fully parameterized in OT by an ordering/ranking g.

An ERC describes the parameters (i.e., rankings) under which, for a given input, some output o_1 would be chosen over another output o_2 as optimal according to the fixed constraint set CON. It takes the form of a tuple of length $k = |\text{CON}|$, each of whose coordinates is one of the symbols W, L, or e. Each coordinate of the k-tuple corresponds to one of the k constraints in CON. The meaning of an ERC is that *at least one* of the constraints whose coordinates in the tuple contain a W must outrank *all* of the constraints whose coordinates contain an L in order for o_1 to be favored over o_2. Instances of e in the tuple indicate constraints that favor neither o_1 nor o_2. We can define a function *erc* that com-

putes the ERC implied by a given pair of winning and losing outputs, written $erc(o_1 \succ o_2)$, as in (7).

(7) THE *erc* FUNCTION:
Given a set CON of k constraints indexed $\{1,\ldots,k\}$ and a pair of candidate outputs o_1, o_2 for the same input i, the function $erc(o_1 \succ o_2)$ returns an ERC $\alpha = \langle \alpha_1, ..., \alpha_k \rangle$ describing rankings under which o_1 is more harmonic than o_2.
$erc(o_1 \succ o_2) = \langle \alpha_1, ..., \alpha_k \rangle$, where $\begin{cases} \alpha_j = \text{W}: & \text{if } C_j((i,o_1)) < C_j((i,o_2)) \\ \alpha_j = \text{L}: & \text{if } C_j((i,o_1)) > C_j((i,o_2)) \\ \alpha_j = e: & \text{if } C_j((i,o_1)) = C_j((i,o_2)) \end{cases}$
and where $C_j(x)$ indicates the number of violations that constraint indexed j assigns to input-output pair x.

Thus the ERC $erc(o_1 \succ o_2)$ can be said to represent a disjunction of partial orderings with which any total ordering would have to be consistent in order to predict o_1 rather than o_2 as the output for some common input. The parameters under which an output o_1 will be selected over *all other* candidate outputs for some input, then, are exactly those rankings consistent with the *conjunction* of disjunctions of partial orderings implied by comparing o_1 to every other candidate[2] in the comparative tableau for that input. For example, if there are three possible candidates o_1, o_2, o_3 for an input i, with violations as shown in (8), then if we observe o_1 as the actual output for i, two ERCs are implied, one for each comparison of o_1 to the two losing candidates. The rankings implied by observing the pair (i, o_1), then, are just those consistent with both of these ERCs; in the case of (8), these would be any rankings in which C_1 outranks both C_2 and C_4, and simultaneously at least one of C_1 or C_2 outranks both of C_3 and C_4.

(8) COMPARATIVE TABLEAU:

Input /i/	C_1	C_2	C_3	C_4
☞Candidate output o_1	0	1	1	2
Candidate output o_2	2	0	1	1
Candidate output o_3	1	2	0	1

$erc(o_1 \succ o_2) = \langle \text{W,L},e,\text{L} \rangle$
$erc(o_1 \succ o_3) = \langle \text{W,W,L,L} \rangle$

Prince (2002) describes in detail how ERCs can form the basis of a 3-valued logic for reasoning about constraint rankings from data. For the purposes of this paper it is sufficient to establish that they offer a concise and computable representation of what parameter settings (rankings) in OT are implied by individual input-output observations in a learner's sample.

3. Algorithm and Results

3.1 Graphing Parameter Co-occurrence

The multilingual learning algorithm we present here bears some relation to existing conceptions of how to represent varieties of linguistic systems like dialects and registers; Biber

[2] In fact, only the non-harmonically bounded candidates must be considered.

(1995, p. 30) summarizes these well in his introduction to a "multi-dimensional" analysis of register variation:

> [W]hen analyses are based on the co-occurrence and alternation patterns within a group of linguistic features, important differences across registers are revealed. … Ervin-Tripp (1972) and Hymes (1974) identify 'speech styles' as varieties that are defined by a shared set of co-occurring linguistic features. Halliday (1988, 162) defines a register as '*a cluster of associated features having a greater-than-random … tendency to co-occur*.' [Emphasis ours.]

Our algorithm seeks to detect such clusters of associated features in the data, and to identify those clusters with target languages; features in this case are statements about the grammatical properties implied by individual input-output pairs, i.e., descriptions of the sets of parameter settings that would generate that pair.

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

In an OT model, these statements can be computed and represented as ERCs as described above.

The central idea of the algorithm, then, is to keep track of these statements about the grammatical properties implied by individual input-output observations, and to take note of which statements are seen to be *simultaneously implied* by single observations. The intuition is that groups of statements that are strongly associated with each other by virtue of being simultaneously required by many individual input-output pairs will tend to be groups of statements that are all true of one of the target languages present in the sample. We track the co-occurrence of statements by building a graph whose nodes are the statements and whose edges indicate which pairs of statements were at some point implied together by an individual input-output pair in the sample. Strongly associated groups of grammatical properties then appear as dense clusters or "communities" of linked nodes within the graph. The whole algorithm is outlined in (10).

(10) OUTLINE OF THE ALGORITHM:
   a. Begin with an empty, unweighted, undirected "co-occurrence graph."
   b. For each observed input-output pair in the sample:
      i. Construct a list of statements about which properties a grammar would need to have in order to be consistent with seeing that observation (in OT, this is a list or conjunction of ERCs).
      ii. For each statement in that list, add a node to the co-occurrence graph, and add an edge 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*.

Hypothesis: Intuitively, the "dense," or highly connected, mutually consistent, regions of the graph correspond to the grammars of the individual languages from whose union the sample was originally taken.

c. Apply some heuristic to identify the dense regions, or *clusters*, of the co-occurrence graph, and adopt hypothesis grammars that are consistent with the grammatical properties specified by the statements in those clusters.

Figure 2 illustrates what a co-occurrence graph might look like in a particular case. Steps (10a) and (10b) of the algorithm essentially reduce the multilingual learning problem to one that is well studied in the fields of graph theory and network analysis: detecting clusters, or "community structure" in a graph. There are some complications, however. The first is that it is possible for some statements about grammatical properties to contradict each other, and so we must avoid adopting hypotheses built on contradictory statements. For example, the two ERCs in (11) contradict each other; (11a) specifies that one of constraints C_1 or C_5 must dominate both C_2 and C_3, while (11b) says that C_3 dominates both C_1, C_4, and C_5, a flat contradiction. Furthermore, it is possible for a set of ERCs to be inconsistent or contradictory as a whole, without any particular pair of ERCs contradicting each other. In (12), for instance, for all three ERCs to be true it would be necessary for C_1 to outrank C_2, C_2 to outrank C_3, and for C_3 to outrank C_1—a contradiction by circularity.

(11) TWO INCOMPATIBLE ERCS:

a. $\langle \mathrm{W}, \mathrm{L}, \mathrm{L}, e, \mathrm{W} \rangle$

b. $\langle \mathrm{L}, e, \mathrm{W}, \mathrm{L}, \mathrm{L} \rangle$

(12) AN INCONSISTENT SET OF ERCS:
$\{\langle \mathrm{W}, \mathrm{L}, e \rangle, \langle e, \mathrm{W}, \mathrm{L} \rangle, \langle \mathrm{L}, e, \mathrm{W} \rangle\}$

As a result, it is possible that whatever groups of nodes in the co-occurrence graph we identify as clusters (step (10c)) may not correspond to internally consistent sets of grammatical properties. Every immediately linked pair of nodes will necessarily be consistent with each other, since the linkage between them means that they are both implied by a single observation in the sample. But if two or more of the target languages happen to have some grammatical properties in common, while disagreeing on others, some properties that conflict across languages may end up sharing a component in the graph, linked to each other at a distance via edges with otherwise compatible nodes—a chain of pairwise consistent statements that are nonetheless contradictory when taken as a whole, as in (12). Thus the cluster detection heuristic of (10c) must be careful to only find mutually consistent groups of nodes to form the basis of the hypothesized grammars.

An additional complication is how one constructs hypothesis grammars from the set of clusters that have been identified in the co-occurrence graph (10c). In general, the set of grammatical properties in one of these clusters may not be sufficiently exhaustive to specify exactly one grammar. In OT, for example, the conjunction of disjunctions of partial orders described by the set of ERCs in a cluster may leave some constraints unranked relative to each other, so that multiple total orderings of the constraint set would be consistent with the cluster. This is expected because the learner's sample may not be complete enough to fully exemplify the behavior of every target language on every input; some method of choosing

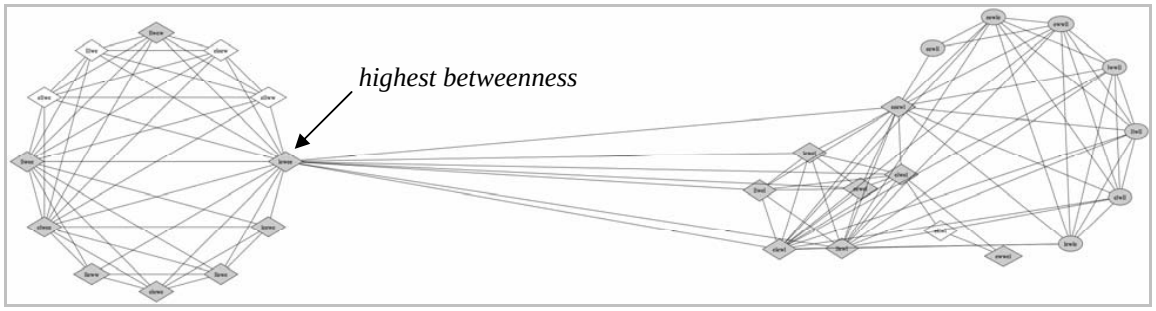

Figure 2: A co-occurrence graph as constructed in one of the Monte Carlo trials for the syllable structure case study with two target languages (described in §3.3). Each node represents a single ERC, and the edges indicate ERCs that were simultaneously implied by individual observations. Two dense regions are visually apparent in this layout, corresponding to the grammatical properties (ERCs) of the two languages. The node with the greatest betweenness centrality is pointed out.

among possible grammars consistent with a cluster of properties is necessary. As a simple baseline, in the OT case study below we just choose a total ranking uniformly at random from all those consistent with all the ERCs in a cluster.[3]

3.2 Clustering with Betweenness Centrality

It remains to provide an implementation of step (10c) in the outline of the algorithm. This must be a heuristic for identifying "dense" groups of grammatical properties in the co-occurrence graph that are free form internal contradictions. A dense region, or cluster, can be defined as a set of nodes with a large number of connections between each other, and a minimal number of connections to nodes outside the set (see, e.g., Newman 2003, 2004, Schaeffer 2007 for more background on graph clustering). Quite a number of methods have been developed for this general task, and we adopt a fairly simple one here based on a measure due to Freeman (1977) called *betweenness centrality*. This is a quantity that can be calculated for each node in a graph, roughly expressing the degree to which that node lies "between" many other nodes. It is defined as in (13).

(13) BETWEENNESS CENTRALITY:
The betweenness centrality of a node n in a graph G is the proportion of all shortest paths between pairs of nodes in G that pass through n.

This measure can be used to detect clusters by identifying the nodes that sit at their boundaries. An example of a node with high betweenness centrality is indicated in Figure 2.

Our clustering heuristic proceeds by checking each connected component of the co-occurrence graph, and seeing if the nodes contained in it specify mutually consistent grammatical properties. If so, it adopts the component as the basis for a hypothesis grammar, and if not, it splits the component apart into clusters by finding the node with the

[3]This can be accomplished in a brute force manner by generating random linearizations until one is found to be consistent with the ERC set; this suffices for our case study. In general, with more constraints, a more efficient method will be necessary, perhaps along the lines described by Matthews (1991).

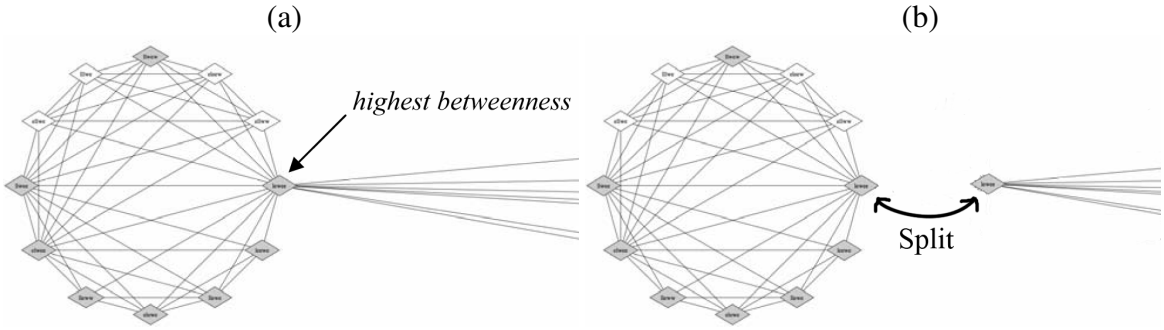

Figure 3: A node with high betweenness centrality is identified in (a); (b) illustrates the process of "splitting" that node and distributing a copy to each new connected component that would result from its removal.

greatest betweenness centrality and making a division there (see Figure 3). It does this iteratively until the entire graph has been broken into mutually consistent components that are adopted as hypotheses.

(14) CLUSTERING HEURISTIC FOR (10C):[4]

a. Begin with an empty set H of hypothesis grammars.

b. For each connected component C of the co-occurrence graph G:

i. 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.

ii. Otherwise, find the node v in G with the greatest betweenness centrality, and tentatively remove it from G to determine which 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 $k_1, \ldots, k_n$ and reconnect those copies to whichever nodes in $k_1, \ldots, k_n$ shared an edge with v before its removal. Return to step (14b).

Step (14b.ii) accomplishes the splitting of a betweenest node by distributing copies of it to any new connected components that would result from its removal. An advantage of this approach is that copies of a single node may end up contributing to multiple hypotheses—that is, if several of the target languages "overlap" or have grammatical properties in common, this can result in the algorithm identifying hypothesis grammars that also share those properties. This is the mechanism by which languages may be identified and discriminated even if they exhibit similar behavior in the learner's sample.

3.3 Evaluating the Algorithm

We have tested the learning algorithm in an OT setting with ten constraints on syllable structure, summarized in Table 1, a simplified subset of those described by Prince and Smolensky (1993, part II). The alphabet of representation for inputs in this model is {C,V}, representing consonants and vowels. The possible symbols for output strings are {C,V, .}:

[4]This heuristic bears some significant resemblance to the centrality-based Girvan-Newman algorithm (Girvan and Newman 2002), which successively removes high-betweenness edges to separate clusters.

CONSTRAINT	PENALIZES...
*C	Each instance of a C.
ONSET	Each syllable without an onset.
*CODA	Each syllable with a coda.
*COMPLEXCODA	Each syllable with a multi-consonantal coda.
DEP	Each insertion of a segment.
DEPC	Each insertion of a C.
DEPV	Each insertion of a V.
MAX	Each deletion of a segment.
MAXC	Each deletion of a C.
MAXV	Each deletion of a V.

Table 1: The constraint set of the OT syllable structure case study.

consonants, vowels, and syllable edges. A language in this model thus maps strings in $I = \{\mathrm{C}, \mathrm{V}\}^*$ to strings in $O = \{\mathrm{C}, \mathrm{V}, .\}^*$. For simplicity we assume that all languages have a shared finite lexicon of inputs, consisting of all strings up to length five over {C,V}, which makes for a total of $2+2^2+2^3+2^4+2^5 = 62$ input forms. Furthermore, we assume that only output candidates adhering to the structure in (15) are considered for any input.

(15) FILTER ON OUTPUT CANDIDATES:
$[.(\mathrm{C})(\mathrm{C})(\mathrm{V})\mathrm{V}(\mathrm{C})(\mathrm{C}).]^*$
That is, candidates containing zero or more syllables of the form (C)(C)(V)V(C)(C).

For each input, the set of non-harmonically bounded output candidates consistent with (15) was computed using finite state methods (Riggle 2004).

To test the algorithm in this setting, we applied it to samples of input-output pairs drawn from mixtures of between one and five target languages. In each trial, each target language was generated by selecting a random linearization of the ten constraints. One hundred trials were run for each number of target languages (i.e., 100 trials with one randomly chosen target language per trial, 100 trials with two randomly chosen target languages, etc.). In each trial, the learner's performance was assessed on different sized samples drawn from that trial's set of target languages, allowing us to see how the performance of the algorithm varies with increasingly large samples relative to the size of the target lexicons, up to a maximum sample size of 30 (out of 62 lexical items); here sample size refers to the number of example input-output pairs drawn from *each* target language, so for example, 20 samples from four target languages is a total of 80 input-output pairs. All possible input-output pairs were equally likely to appear in a sample, being chosen uniformly at random.

At each trial, three quantities were measured to track the algorithm's performance:

(16) The number of hypothesis grammars that the algorithm returns.

(17) The "over/under-generation ratio" of the set of hypothesis grammars. This is the percentage of inputs in the lexicon for which either:

a. at least one of the learner's hypothesis grammars yields an output that none of the target languages map the input to (the learner over-generates as in (1)), or

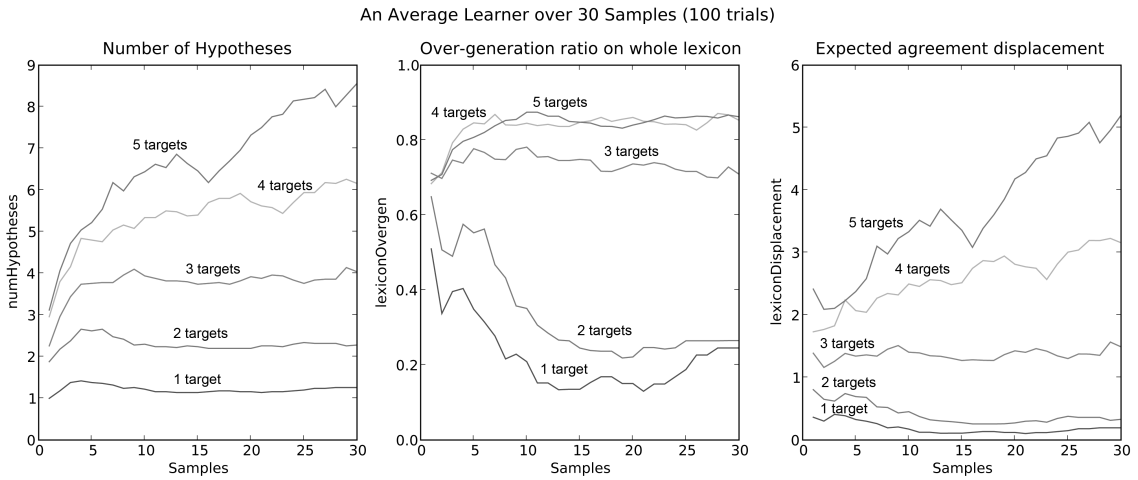

Figure 4: Average performance of the algorithm over 100 trials with 2 to 30 samples each.

b. at least one target language yields an output that none of the learner's hypothesis grammars map the input to (the learner under-generates).

(18) The "expected agreement displacement" of the set of hypothesis grammars. This measures the overall distance, or dissimilarity, between the set of languages defined by the learner's hypothesis grammars and the set of actual target languages. First, for any pair of languages L_1 and L_2, we can compute their "expected agreement" $E_{\text{agr}}(L_1, L_2)$, which is the probability that they will agree with each other on the output for an input chosen uniformly at random from the lexicon. If $T = \{T_1, \ldots, T_n\}$ is the set of target languages, then for each T_i we compute an expected agreement vector $\vec{V}_{\text{agr}}(T_i) = \langle E_{\text{agr}}(T_i, T_1), \ldots, E_{\text{agr}}(T_i, T_n) \rangle$, encoding T_i's degree of agreement with each other target language. All of these vectors together form a matrix M_T, describing the expected agreement between any pair of target languages. Similarly, if $H = \{H_1, \ldots, H_m\}$ is the set of hypothesized languages, we calculate an expected agreement vector $\vec{V}_{\text{agr}}(H_i) = \langle E_{\text{agr}}(H_i, H_1), \ldots, E_{\text{agr}}(H_i, H_m) \rangle$ for each hypothesis language H_i, giving a matrix M_H of agreement between pairs of hypothesis languages. Our final metric, then, is a measure of how different these two matrices, M_T and M_H, are. It is defined as the total distance that the vectors in the larger of the two would have to be displaced to be transformed into the nearest vectors of the smaller matrix.

Figure 4 shows the 100-trial average behavior of the algorithm, as quantified by these measures, on random samples of 2 to 30 input-output pairs from 1 to 5 target languages. It can be seen that the algorithm tends to overestimate the number of languages present, with this tendency increasing as the actual number of target languages grows. There may also be a qualitative difference in how it responds to target sets of one or two languages, versus sets of three or more, as measured by the over/under-generation ratio and expected agreement displacement. Further detailed analysis of different test cases will be necessary to explicate this.

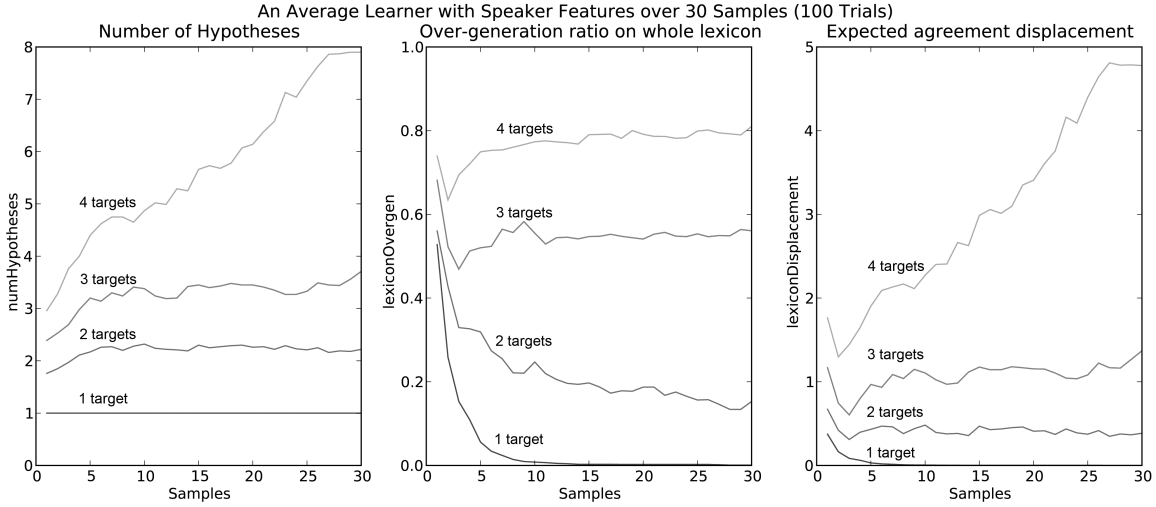

Figure 5: Average performance of the algorithm with speaker features.

3.4 Incorporating Extralinguistic Information

One advantage of the cluster-of-parameters approach to multilingual learning is that it is generic with respect to what kinds of objects the parameters are. So far we have experimented with the case in which they are statements about possible constraint rankings, but in principle any kind of parameter or feature deemed relevant to distinguishing languages may be incorporated. For instance, one might suppose that in addition to the internal co-occurrence of purely grammatical properties, a variety is also defined by some relation of these properties to external sociolinguistic variables such as the identities of speakers and listeners, properties of the social context, and so on. Assuming that the learner can observe the state of such variables, they may contribute their own nodes to the co-occurrence graph that our algorithm constructs, and thus play a role in distinguishing one language from another.

To experiment with this possibility, we tested the algorithm exactly as described above, but with the inclusion of what might be called "speaker features." The idea is that each target language is produced by a distinct "speaker," and that when the learner observes an input-output pair, it also observes which speaker produced it. When the learner computes the list of grammatical properties (ERCs) implied by the pair, it includes among these a feature or parameter identifying which speaker produced it. This speaker feature is added as a node to the co-occurrence graph just like the other grammatical properties, and is linked to them, having just co-occurred with them on a single input-output pair. Subsequent observations from that speaker will continue to link that same speaker feature node to new grammatical properties, possibly providing crucial additional structure to the graph for purposes of identifying the clusters.

Figure 5 shows the results of this for one to four target languages. With speaker parameters, the algorithm is better able to discriminate between three or fewer target languages. In addition, the over/under-generation ratio and expected agreement displacement are notably reduced for three or fewer target languages, though the case of four targets still

proves difficult for the algorithm. In principle, one could incorporate any type of parameter encoding features that systematically co-occur with a language, dialect, or register, be they sociolinguistic, morphosyntactic, etc.

4. Conclusions and Future Work

We have provided a preliminary demonstration of clustering on parameter co-occurrence as a generic strategy for discriminating and acquiring multiple languages from a mixed sample. While this demonstration establishes the viability of such an approach generally, at least two defects are immediately apparent, motivating further refinement:

i. The algorithm has a tendency to overestimate the number of target languages present in the sample. This likely accounts for much of the over/under-generation and expected agreement displacement, since carving the nodes of the co-occurrence graph into too many clusters means that each hypothesis will be based on a smaller, vaguer set of grammatical properties, allowing for hypotheses that diverge significantly from the full targets. To ameliorate this, some method of coalescing and recombining too-small clusters may be useful.

ii. Currently, grammatical properties are simply linked in the graph if they are seen to co-occur at all on any input-output pair. Superior results should result from actually tracking the *frequency* of their co-occurrence, so that a *weighted* graph is constructed, with the weight on each edge indicating the number of times that the linked properties co-occurred in the sample. A weight-sensitive clustering heuristic may then be applied to take advantage of this additional information.

Further work of this nature will be necessary to address the question of how numerous linguistic systems can be simultaneously acquired in a pervasively multilingual environment.

References

Angluin, Dana. 1980. Inductive inference of formal languages from positive data. *Information and Control* 45:117–135.

Anttila, Arto. 1997. Variation in Finnish phonology and morphology. Doctoral Dissertation, Stanford.

Anttila, Arto. 2008. Phonological constraints on constituent ordering. In *Proceedings of the 26th West Coast Conference on Formal Linguistics (WCCFL 26)*, ed. Charles B. Chang and Hannah Haynie, 51–59. Somerville, MA: Cascadilla Proceedings Project.

Biber, Douglas. 1995. *Dimensions of register variation*. Cambridge University Press.

Boersma, Paul, and Bruce Hayes. 2001. Empirical tests of the Gradual Learning Algorithm. *Linguistic Inquiry* 32:45–86.

Clopper, Cynthia G. 2004. Linguistic experience and the perceptual classification of dialect variation. Doctoral Dissertation, Indiana University, Bloomington, IN.

Ervin-Tripp, Susan M. 1972. Alternation and co-occurrence. In *Directions in sociolinguistics: The ethnography of communication*, ed. J.J. Gumperz and D. Hymes, 218–250. New York: Holt, Rinehart and Winston.

Freeman, Linton C. 1977. A set of measures of centrality based on betweenness. *Sociometry* 40:35–41.

Genzel, Dmitriy. 2005. Inducing a multilingual dictionary from a parallel multitext in related languages. In *Proceedings HLT/EMNLP*, 875–882.

Girvan, M., and M.E.J. Newman. 2002. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA* 99:7821–7826.

Gold, E. M. 1967. Language identification in the limit. *Information and Control* 10:447–474.

Halliday, Michael Alexander Kirkwood. 1988. *New developments in systemic linguistics*. London: Pinter.

Hymes, Dell. 1974. *Foundations in sociolinguistics: An ethnographic approach*. University of Pennsylvania Press.

Matthews, Peter. 1991. Generating a random linear extension of a partial order. *Annals of Probability* 19:1367–1392.

Müller-Lancé, Johannes. 2003. A strategy model of multilingual learning. In *The multilingual lexicon*, ed. Jasone Cenoz, Britta Hufeisen, and Ulrike Jessner, 117–132. Springer Netherlands.

Newman, M.E.J. 2003. The structure and function of complex networks. *SIAM Review* 45:167–256.

Newman, M.E.J. 2004. Detecting community structure in networks. *Eur. Phys. J. B* 38:321–330.

Prince, Alan. 2002. Entailed ranking arguments. *Rutgers Optimality Archive* ROA-500.

Prince, Alan, and Paul Smolensky. 1993. Optimality Theory: Constraint interaction in generative grammar. Ms., Rutgers University and University of Colorado, Boulder.

Riggle, Jason. 2004. Generation, recognition, and learning in Finite State Optimality Theory. Doctoral Dissertation, University of California, Los Angeles.

Schaeffer, Satu Elisa. 2007. Graph clustering. *Computer Science Review* 1:27–64.

Snyder, Benjamin, and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In *Proceedings of ACL-08: HLT*, 737–745. Association for Computational Linguistics.

Tesar, Bruce, and Paul Smolensky. 2000. *Learning in Optimality Theory*. MIT Press.

Weikum, W.M., A. Vouloumanos, J. Navarra, S. Soto-Faraco, N. Sebastián-Gallés, and J.F. Werker. 2007. Visual language discrimination in infancy. *Science* 316:1159.

Department of Linguistics
University of Chicago
Chicago, IL 60637

jriggle@uchicago.edu
All source code produced in the course of this work is available upon request.