Learning tiers for long-distance phonotactics

An open question in the acquisition of long-distance phonological phenomena is, to what extent is tier information present in Universal Grammar? Is it possible to *learn* tiers? This paper introduces a new algorithm, based in formal language theory, which suggests a positive answer to the latter question, at least regarding phonotactics. The learner, the TSL_2 Learning Algorithm (TLA), based on the Tier-based Strictly Local (TSL) formal languages (Heinz et al., 2011) can from positive data induce a tier over which long-distance phonotactic generalizations can be made.

Long-distance phonotactics are those which involve dependences between segments that ignore intervening material. Material not ignored can be thought of as on a 'tier.' A classic example is Finnish vowel harmony, in which all vowels in a word must be $[+\text{back}]$ or $[-\text{back}]$, excluding the transparent vowels /e/ and /i/, as seen in (1). This intervening material hampers the search for finding dependencies in input data; for example, Hayes and Wilson (2008)'s learner cannot find vowel coocurrence restrictions in Shona $[\pm\text{ATR}]$ harmony until it is given the vowel tier *a priori*. However, different languages use different tiers—Finnish and Shona have different transparent vowels, and Latin liquid dissimilation requires a liquid tier (Jensen, 1974; Odden, 1994). Thus, it is worth pursuing ways to discover tiers; Goldsmith and Riggle (2012) offer one approach.

The TLA can also do this, by learning the TSL_2 class of formal languages. TSL_2 languages, designed to model locality on a tier, are those whose grammar is a tuple (T,S) where T is the *tier*, or a subset of the full alphabet Σ of symbols, and S are the allowable sequences of length 2 of symbols from T (Heinz et al., 2011). These sequences specify which members of the tier are allowed to be 'adjacent' to one another, ignoring any intervening symbols not on the tier. For example, let $G = (T,S)$ as defined in (2). *ulkota* is in $L(G)$, because ignoring all symbols not in T, *u* and *o* are adjacent (*u*$\underline{l}$*k*$\underline{o}$*t*$\underline{a}$), *o* and *a* are adjacent (*ulk*$\underline{o}$*t*$\underline{a}$), and *uo* and *oa* are both in S. A string like *ylkota* is not in $L(G)$, because *y* and *o* are adjacent on the tier but *yo* is not in S.

The TLA can provably learn such a grammar from positive data without knowing T in advance. It does so by remembering *paths*, or precedence relations that also keep track of sets of intervening symbols. For example, $\langle u, \{l,k\}, o\rangle$ is in the paths of *ulkota*, because *u* precedes *o* in the string and $\{l,k\}$ is the set of symbols that come between them. The TLA begins by guessing that $T = \Sigma$ and then recurses through a process in which it removes a symbol from T when there are no longer any restrictions on its distribution, relativized to T (i.e., when considering paths whose intervening set is a subset of $\Sigma - T$). Like Goldsmith and Riggle (2012)'s algorithm, the TLA finds tiers based on dependencies between symbols, although in a discrete, and not statistical, manner.

For success to be guaranteed, the algorithm requires that at least one phoneme to appear adjacent to all other phonemes in the input corpus, in order to begin removing symbols from T. As natural language data also exhibits local coocurrence restrictions, this requirement might not always be met. Therefore, to demonstrate the algorithm on natural language data, natural classes were used to simplify 9,700 harmonic Finnish forms taken from the corpus used in Goldsmith and Riggle (2012). The alphabet was collapsed into natural classes as follows: stops $\{p,t,d,k,g\}$ were changed to *t*, fricatives $\{v,h,s\}$ were set to *s*, nasals $\{m,n\}$ set to *n*, and other sonorants $\{r,l,j\}$ set to *j*. Vowels were left unsimplified. With this, the TLA learned the grammar in (3), for which T only includes the harmonizing vowels and S only harmonic vowel sequences. The natural class stipulation was necessary, but, significantly, transparent vowels were also correctly removed from the tier. Future work can explore ways to replace this stipulation, such as learning local coocurrence restrictions (Rogers and Pullum, 2011) and factoring them out of the long-distance learning. Regardless, this work contributes a method for inducing tiers from positive data.

(1) Finnish vowel harmony (Nevins, 2010; Odden, 1994)

		Vowel features:	
a. pöytä-nä ‘table-ESS’	c. ulko-ta ‘outside-ABL’	[−back]:	y, ö, ä
b. väkkärä-nä ‘pinwheel-ESS’	d. pappi-na ‘priest-ESS’	[+back]:	u, o, a

(2) $\Sigma = \{i, y, u, e, ö, o, ä, a, l, k, t, p, n\}$
$T = \{y, u, ö, o, ä, a\}$
$S = \{yy, yö, yä, öy, öö, öä, äy, äö,ää, uu, uo, ua, ou, oo, oa, au, ao, aa\}$

(3) $T = \{a, ä, o, u, ö, y\}$ $(\Sigma = \{t, s, n, j, i, y, u, e, ö, o, ä, a, \})$
$S = \{aa, ao, au, öä, ää, oo, äö, yy, öy, yö, yä, oa, uu, uo, äu, äy, öö, ou, ua\}$

References

Goldsmith, J. and Riggle, J. (2012). Information theoretic approaches to phonological structure: the case of finnish vowel harmony. *Natural Language & Linguistic Theory*, 30(3):859–896.

Hayes, B. and Wilson, C. (2008). A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry*, 39:379–440.

Heinz, J., Rawal, C., and Tanner, H. G. (2011). Tier-based strictly local constraints for phonology. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 58–64, Portland, Oregon, USA. Association for Computational Linguistics.

Jensen, J. (1974). Variables in phonology. *Language*, 50:675–686.

Nevins, A. (2010). *Locality in Vowel Harmony*. Number 55 in Linguistic Inquiry Monographs. MIT Press.

Odden, D. (1994). Adjacency parameters in phonology. *Language*, 70(2):289–330.

Rogers, J. and Pullum, G. (2011). Aural pattern recognition experiments and the subregular hierarchy. *Journal of Logic, Language and Information*, 20:329–342.