

*Embedding Grammar in a Quantitative Framework:
Case Studies from Phonology and Metrics*

Class 3: Phonotactics

1. Today

- Other theories of gradient constraint-based grammars.
- How the weights are found in maxent
- Phonotactics: a maxent approach

2. Readings

- Hayes and Wilson, sections 1-5
- Software for this paper, in user-friendly version, is available if you want to try it: course website

OTHER CONSTRAINT-BASED THEORIES OF GRADIENCE

3. OT with free-variation strata

- Anttila (1997a, 1997b)
- Group the constraints into strata; rank the strata, but rank at random within strata.
- This predicts a specific distribution of outputs.
- Very tightly constrained model (as Paul Kiparsky emphasizes); Hungarian
 - Hungarian (last time) is an example it seems unable to deal with.

4. Stochastic Optimality Theory

- Invented by Paul Boersma (1997); applied to phonology by Boersma and Hayes (2001).
- Give every constraint a “ranking value”.
- When you run the grammar, “jiggle” the weights by adding to each ranking value a small random quantity. Then sort them and apply standard OT to the resulting ranking.

5. The learnability situation for stochastic OT

- Boersma invented an algorithm (“Gradual Learning Algorithm”) for stochastic OT.
- It works pretty well for many simulations—though without maxent’s uncanny accuracy.
- Behaves very strangely for others (in my experience)
- and (*ouch!*) was found to fail to find the solution in a well-defined class of cases—Pater (2008), course web site
 - Pater, Joe. 2008. Gradual learning and convergence. *Linguistic Inquiry* 39. 334-345.

- Magri (ms.), course web site, has a beautiful demonstration of *why* the GLA fails: sometimes the right answer isn't even in its search space! (= grammars obtainable by legal ranking value adjustments).
- Magri has invented a better GLA, which he can prove to converge, but only for non-stochastic grammar.

6. Noisy Harmonic Grammar

- Paper by Boersma and Pater (course web site).
 - Boersma, Paul, and Joe Pater. 2008. Convergence properties of a gradual learning algorithm for Harmonic Grammar. Amsterdam and Amherst, MA: University of Amsterdam and University of Massachusetts ms. Rutgers Optimality Archive.
- This is like the simple Harmonic Grammar described last time (lowest penalty score wins), but as with Stochastic OT you add a bit of noise to each constraint weight when you “apply” the grammar.

7. The learnability situation for Noisy Harmonic Grammar

- Same as for stochastic OT: there is a learnability proof, but only for the non-stochastic applications

8. Where maxent differs sharply from these models

- **Harmonically bounded** candidates can semi-win (i.e. have more than zero probability)
 - A candidate is *harmonically bounded* if some other candidate has a strict subset of its violations.
- Scholars differ in whether harmonically bounded candidates should ever win.
 - Keller and Asudeh (*Linguistic Inquiry* 2002) thinks they should.
 - I've found slightly better performance in my metrical work if I let them win.¹
 - I'd say not letting them win is the majority current view.

9. Model-shopping: my own feelings

- Re. using algorithms that don't have a convergence proof: once burned, twice shy!
- I have some empirical worries re.
 - Constraint ganging (all versions of Harmonic Grammar)
 - Harmonically bounded semi-winners (maxent)

1

A QUICK OVERVIEW OF HOW LEARNING IN MAXENT WORKS

10. Source

- This discussion follows the attempted layman's explanation in Hayes and Wilson (2008) (course website).

11. Core idea: "Objective function"

- Defines the "goal" of learning.
- This is separated from the (varying) computational algorithms can be used to achieve it.
- *Maximize the predicted probability of the observed forms*
 - hence, minimizes the predicted probability of the unobserved forms
- Predicted probability of observed forms is quite calculable: calculate each one as given last time, then multiply them all together.

12. Metaphor: the objective function is a mountain

- If we have just two constraints:
 - let North-South be the axis for Constraint1's weight
 - let East-West be the axis for Constraint2's weight
 - let height be the predicted probability of the observed data under any weight assignment.
- Find the weights that put you on top of the mountain (i.e., climb it!).

13. Two beautiful theorems

- The mountain has but one peak (=is convex; has no local maxima)
- The slope along any axis (if height expressed as a log) is *Observed Violations – Expected Violations* for the relevant constraint, a calculable quantity.
- So you can always reach the top, simply by persistently climbing uphill.
 - This may sound trivial but remember that the mountain actually exists in n -dimensional space, where n is the number of constraints.

14. The rest is implementation

- Ascending gradients efficiently is a popular challenge for computer scientists; both Goldwater and Johnson (2003) and the Maxent Grammar Tool adopt the "Conjugate Gradient" algorithm.

PHONOTACTICS

15. The problem of phonotactics

- **Definition:** the study of the principles of phonological well-formedness — what is the basis on which people can make judgments like ✓ [blɪk] vs. *[bnɪk]? (Chomsky and Halle 1965)

16. Phonotactics matter

- Characterizing phonotactics (knowledge of what forms are admissible) has always been a **core goal of phonological theorizing**.
- Phonotactic knowledge is arguably **the first phonological knowledge we acquire** (work by Jusczyk et al. 1993, Gerken 2004, etc.).
- Phonotactics evidently guides language learners as they try to figure out **phonological alternations** (Pater and Tessier 2003, and more distantly Kisseberth 1970)

17. Gradience in phonotactic intuitions

- Everybody who looks finds it. (Greenberg and Jenkins 1964, Ohala and Ohala 1986, Coleman and Pierrehumbert 1997, Vitevitch et al. 1997, Frisch et al. 2000, Treiman et al. 2000; Bailey and Hahn 2001, Hay, Pierrehumbert, and Beckman 2003, Hammond 2004)

EARLIER APPROACHES TO PHONOTACTICS

18. SPE era and immediately after

- Phonotactic constraints (on underlying forms of morphemes; Chomsky and Halle (1968)); surface (Shibatani 1973)²
- These lacked a theory of gradience.
- ... and a theory of learning.

19. Optimality theory (Prince and Smolensky 1993)

- Crucial idea is the **rich base**.
- Let anything be a possible underlying form
- but what is legal is what passes through the grammar—“filtering”
- High ranking of Faithfulness relative to Markedness allow more to pass through.

² Shibatani, Masayoshi (1973) The Role of Surface Phonotactic Constraints in Generative Phonology *Language* 49, 87-106.

20. Gradience

- You *could* attach this concept to the various theories of constraint-based stochastic grammar.
- But the derivational aspect of the Rich Base system makes a very odd prediction: that words should generally be in free variation with their “repaired forms”.
- E.g. [dwep] sounds funny so we occasionally say it as [dep].

21. The learning side

- This is a classical case of the **subset problem** (no negative evidence to keep you from thinking more forms are legal than are).
- Both Prince and Tesar (2004) and Hayes (2004), working in classical OT, invent a bunch of ad hoc heuristics to rank Faithfulness constraints as low as possible.
- This doesn’t seem to be gradientizable, and also seems a bit unprincipled...

A NEW DIRECTION

22. Why not do phonotactics as phonotactics?

- A maxent grammar can be arranged to assign a probability to any string of phones.
- It can do so without reference to any input form.

23. The scheme, roughly

- Suppose the world contains 100 speech sounds.
- Then there are 100 monosegmental words, 10000 bisegmental words, 1000000 trisegmental words, etc.
- We can cheat infinity by supposing some maximum—reasonably observable from the ambient language.
- Each possible word form violates a certain number of phonotactic (a.k.a. Markedness) constraints, so we can sum the penalties (weights times violations) in the usual way.
- And we can continue in the usual way, taking e to the negative of the result.
- Different: Z , the denominator, is not the sum across all candidates for a given input, but the sum across all possible word forms.
- End result: a probability for every word form.

24. Example

- Constraints and weights:

*[σ η]	10
*h] σ	10
*NO ONSET	2
*CODA	2
*[-coronal]	1

Constraint: Weight:	*[σ]	*h] _σ	ONS	CODA	*[-cor]	Score
[ta]	10	10	2	2	1	0
[kup]				1	2	4
[ip]			1	1	1	5
[ŋah]	1	1				20

- and onward through the computation of $e^{-\text{Score}}$ and eventually probabilities.

25. Conceptual issues

- You had better get used to very small numbers!
- A really well-formed word like [kɪp] will still have a very low probability.
- But nowhere near as low as [bzɑɹʔk].
- A colossally low number is the probability of the training data, used in setting the weights.
 - My current work on Shakespeare (“metritactics”) gives his Sonnets a probability score of about $e^{-30,000}$.
- This is all fine; it’s what we use logarithms for...

26. Practical barriers

- Computing Expected values (for learning; see above) seems scary—even if we adopt a maximum word length, the number of strings is vast.
- Fortunately, the field of computational phonology is adept at evaluating vast numbers of strings at once, using finite state machines. We borrow ideas of Jason Eisner.

27. Other issues

- How would you find the phones?
 - Distributional learning (work of Frank Guenther, Ying Lin) has made some progress in reducing acoustic signals to proto-symbolic representations.
- How would you find the features?
 - These are traditionally taken to be innate, but see Lin, Jeff Mielke’s new book for learned feature systems.

WHERE DO CONSTRAINTS COME FROM?

28. The innatist view

- Various scholars propose that all the constraints are innate: Tesar and Prince 2000, McCarthy 2002.
- We are taking intermediate stance: not hostile to ideas of innate knowledge, but putting the burden of proof on them.
- Our model is an **inductive baseline**.

29. Observation

- Most feature systems only define a few hundred natural classes.
- They have vastly more formal expressions (3^n , if n binary features), but these multiply designate the natural classes.
- Thus for linear phonology, there aren't all that many possible constraints; maybe just millions or billions—which is very different from truly vast numbers.

30. Possible constraints

- A possible constraint is a **starred sequence of feature matrices**; e.g. in Korean:

$$*[\text{+syllabic}] \begin{bmatrix} \text{–voice} \\ \text{–aspirated} \\ \text{–tense} \end{bmatrix} [\text{+syllabic}]$$

- Feature matrices express **natural classes**; for example, the middle matrix above gives the set [p t tʃ k].

31. Search space size

- 200 natural classes, three-matrix limit: $200^3 + 200^2 + 200$ constraints = **8,040,200**
- This is not overwhelming, if you search it efficiently.
- Success depends in part in having a feature system that does not define too many natural classes.

32. What is needed for learning

- **Select** the constraints from a very large initial set (later)
- **Weight** the constraints (next)

33. Weighting the constraints: analysis by synthesis

- The system uses its current best-guess grammar to create a **sample** of pseudo-words. The better-formed a given pseudo-word is according to the current grammar, the more likely it will appear in the sample.

- It finds and adds a constraint that it thinks will be helpful.
- It modifies the weights to best fit the data.
- It keeps **iterating**.
- The samples will **come to look like the language**.
 - They will share a similar profile of violations
 - The distribution of segment sequences will be similar.

34. Demo: samples as English onsets are learned

- Most common 2-consonant onsets in the sample, columns are three stages of learning.

hʒ	ŋz	stʃ	bj	kj	bl
dŋ	hw	gn	θw	gl	kl
fð	nh	dw	sm	hj	sp
dð	hj	mj	dr	hw	sw
fʃ	jw	dʒj	hj	sm	tw
ʃ	jdʒ	tʃj	tw	sn	tr
rθ	rd	br	zʃ	fj	ʃr
d.ʒ	vj	sb	fj	pl	kw

35. Why use analysis by synthesis?

- This is how we overcome the “**no negative evidence**” **problem**—no one tells the child which words are ill-formed.
- The words in the samples include phonotactically bad ones, which will result in addition and assignment of high weights to constraints that rule them out.

36. The procedure in more detail

- Constraint-picking: from the set of possible constraints, use **heuristics** (below) to pick one that is likely to help a lot and add it to the grammar.
- Using methods described at the start of this handout, **adjust the weights** so that, using this constraint, the next sample will be a better match to the training data.
- Create a **new sample**, based on the new weights.
- Repeat.
- **Termination point**: no further good constraints (as defined below) are available.

37. Heuristics for picking constraints

- Install **accurate** constraints first — meaning the lowest **Observed/Expected**
 - **Observed** = violations in the training set
 - **Expected** = violations in the sample

- For equally accurate constraints, install **general** constraints first (few feature matrices, broad natural classes)
 - Idea: if you pick these, you'll cover the data before you ever have to use nongeneral constraints ...
 - ... and the grammar will generalize to unheard cases

38. Terminating the algorithm as a whole

- Termination occurs when no more constraints can be found whose Observed/Expected value satisfies (is less than) a specified criterion.

39. Summary



The phonotactic learning system involves:

- A format for constraints, defining the space of possible constraints.
- Heuristics for constraint selection
- A system for constraint weighting
- A criterion for when to terminate

SIMULATION: ENGLISH ONSETS

40. The Data

- Word-initial syllable onsets (maximal consonant sequences) in English.

b	l	ɪ	k	<i>blick</i>	well-formed
					
b	n	ɪ	k	* <i>bnick</i>	ill-formed
					

41. Earlier analytic work on the English onset inventory

- Bloomfield 1933, Whorf 1940, O'Connor and Trim 1953, Fudge 1969, Selkirk 1982, Clements and Keyser 1983, Hammond 1999

42. Corpus of training data

- Word-initial onsets in the Carnegie-Mellon Pronouncing Dictionary (<http://www.speech.cs.cmu.edu/cgi-bin/cmudict>).
- They are given to the algorithm in proportion to their **type frequencies** (number of words with a particular onset).
- These were “cleansed” of **exotic** (foreign, or rare-words only) clusters, like *sphere* [sf] or *Puerto Rico* [pw]

- (We also tried it putting in all the exotica; performance of grammars thus trained was worse).

43. The training data

[k] (2764), [r] (2752), [d] (2526), [s] (2215), [m] (1965), [p] (1881), [b] (1544), [l] (1225), [f] (1222), [h] (1153), [t] (1146), [pr] (1046), [w] (780), [n] (716), [v] (615), [g] (537), [dʒ] (524), [st] (521), [tr] (515), [kr] (387), [ʃ] (379), [gr] (331), [tʃ] (329), [br] (319), [sp] (313), [fl] (290), [kl] (285), [k] (278), [j] (268), [fr] (254), [pl] (238), [bl] (213), [sl] (213), [dr] (211), [kw] (201), [str] (183), [θ] (173), [sw] (153), [gl] (131), [hw] (111), [sn] (109), [skr] (93), [z] (83), [sm] (82), [θr] (73), [skw] (69), [fj] (55), [tw] (55), [mj] (54), [spr] (51), [hj] (50), [kj] (45), [ʃr] (40), [pj] (34), [spl] (27), [bj] (21), [ð] (19), [dw] (17), [gw] (11), [vj] (6), [spj] (5), [skj] (4), [θw] (4), [skl] (1)

43.1. Feature system

- To minimize natural class count, we **underspecified** (e.g. Archangeli 1984), but left **sonority features** richly specified, under the view (Steriade 1999) that these are important for segmental licensing.

	p	t	tʃ	k	b	d	dʒ	g	f	θ	s	ʃ	h	v	ð	z	ʒ	m	n	ŋ	l	r	j	w	
cons	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	-	-	-	
appr	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	+	+	+
son	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	+	+	+	+	+	+	+	
cont	-	-	-	-	-	-	-	-	+	+	+	+	+	+	+	+	+								
nas																		+	+	+					
voice	-	-	-	-	+	+	+	+	-	-	-	-	-	+	+	+	+								
spread													+												
lab	+				+				+				+					+						+	
cor		+	+			+	+			+	+	+				+	+	+	+		+	+			
ant		+	-			+	-			+	+	-			+	+	-		+		+	-			
strid		-	+			-	+			-	+	+			-	+	+		-		-	-			
lat																					+				
dors				+				+												+					
high																							+	+	
back																							-	+	

44. Constraint format used

- Up to three feature matrices.

$$* \begin{bmatrix} \alpha F & \gamma H & \varepsilon J \\ \beta G & \delta I & \zeta K \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \end{bmatrix}$$

- At most one matrix can be a **complement class**:

$$* \left[\begin{array}{l} \wedge \text{-voice} \\ +\text{ant} \\ +\text{strid} \end{array} \right] [+nas]$$

“Nothing can precede a nasal except [s]” (*[pn], ✓[sn])

45. Running the algorithm; termination

- We ran an implemented version of the algorithm above.
- We let the algorithm terminate when it couldn't discover any more constraints with an Observed/Expected lower than 0.3.
- This occurred after 24 constraints.

46. Grammar learned: sample constraints

- Here are the six constraints with the highest weights:

<i>Constraint</i>	<i>Weight</i>	<i>Comment</i>	<i>Examples</i>
1. *[+son][]	6.66	Sonorants may only be onset-final	*rt
2. * $\left[\begin{array}{l} \wedge \text{-voice} \\ +\text{ant} \\ +\text{strid} \end{array} \right] [-\text{approx}]$	5.91	Nasals and obstruents may only be preceded (within the onset) by [s].	*kt, *kk, *skt
3. *[+son,+dors]	5.64	*[?]	*?, *s?
4. *[][+voice]	5.37	Voiced obstruents may not cluster with preceding C.	*sb, *sd, *sgr
5. *[][+cont]	5.17	Fricatives may not cluster with preceding C.	*sf, *s?, *sh, *sfl
6. *[][-back]	5.04	[j] may not cluster with a preceding C; see above for assumed syllabic parsing of [ju].	*[bj] _{ons}

47. Grammar learned: more samples

- The five that are **violated in the training data** (responsible for gradient intuitions):

7. *[+cont,+voice,+cor]	2.69	*voiced coronal fricative (violable)	ð, z, *? (see also #2)
8. *[+strid][-ant]	2.10	In effect: [ʔ] is rare (violable).	ʔr vs. fr
9. * $\left[\begin{array}{l} +\text{cont} \\ -\text{strid} \end{array} \right] \left[\begin{array}{l} \wedge +\text{approx} \\ -\text{ant} \end{array} \right]$	2.06	[ʔ, ð] may only be followed by [r] (violable).	?w vs. ?r (see also #21)
10. *[][+cor] $\left[\begin{array}{l} \wedge +\text{approx} \\ -\text{ant} \end{array} \right]$	2.06	In effect: only [r] after [st]	?stw vs. skw, str (see also #23)
11. *[+cont,-strid]	1.84	[ʔ, ð] are rare (violable).	? vs. f, s
12. * $\left[\begin{array}{l} -\text{cont} \\ -\text{voice} \\ +\text{cor} \end{array} \right] \left[\begin{array}{l} \wedge +\text{approx} \\ -\text{ant} \end{array} \right]$	1.70	In effect: [t] can only be followed by [r] (violable).	tw vs. tr

48. There is ganging

8, 14, 22 gang up to give gives *[stʔ] the bad score of 6.21.

It would be hard to show that ganging is essential, however.

<i>Constraint</i>	<i>Weight</i>	<i>Comment</i>	<i>Examples</i>
13. *[+ant,+strid][-ant]	2.80	Anteriority assimilation	*sr vs. ʔr
14. *[+strid][-ant]	2.10	In effect: [ʔr] is rare (violable).	ʔr vs. fr
15. *[-strid]	1.31	Stridents must be initial in a cluster.	*stʔ

49. Assessing the English onset simulation I: separation

- Test all 14,424 strings of consonants up to length three.
- The best scores of the bad: [stw] 3.76, [dl] 4.40, [hl] 4.82, [hr] 4.82, [vl] 4.84, [vr] 4.84, [ʔl] 4.84, [ʔw] 4.84, [sr] 4.90, [fw] 4.96, [pw] 4.96, and [spw] 4.96.
- The worst scores of the good: [ð] 4.54, [ʔw] 3.91, [skl] 3.05, [dw] 2.97, [gw] 2.97, [z] 2.69, [ʔr] 2.10, [ʔ] 1.85, [ʔr] 1.85, [tw] 1.70.
- Main reassurance: no impossible clusters slipped through the cracks.

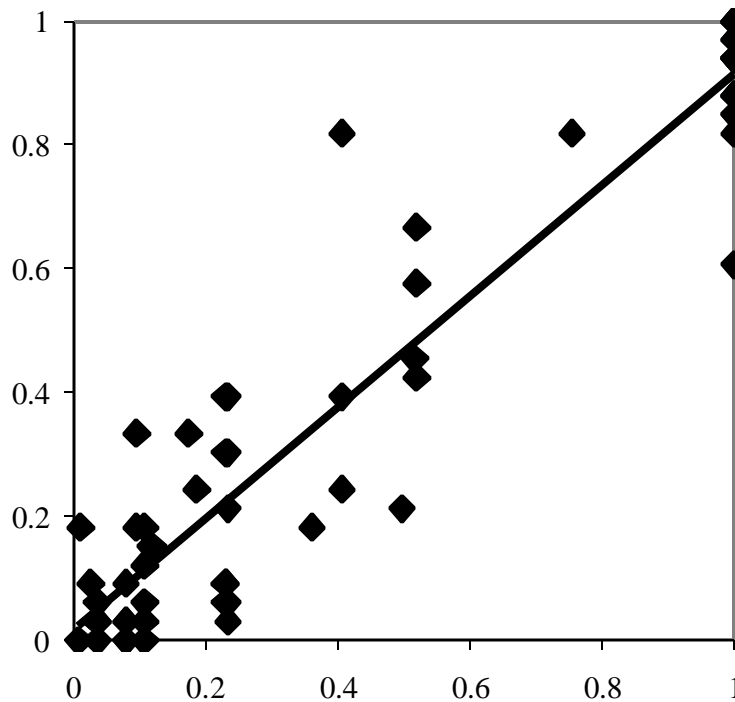
50. Assessing the English onset simulation II: experimental data

- We used “blick” test data from Scholes (1966).
- 33 seventh graders gave up-or-down ratings on 66 words; asked whether they were likely to be usable as words of English.
- Onsets were varied; rhymes were just few and deliberately bland. E.g. [znet], [ztɪn], [trʌn], [ʃkip], etc.
- Statistic recorded: how many of the 33 voted “yes”.

51. How well did our model do?

- We let the learned grammar be an “experimental subject”, calculating its scores for all test words.
- We used the equation $e^{\text{Score}/T}$ to turn scores into maxent values.
- We then computed the correlation of the maxent values with the Scholes experimental data: $r = 0.94$.

52. Scattergram



53. Our model outperforms other models

- Correlations (r) with the Scholes data:

Our model	0.94
Constraints proposed by Clements and Keyser (1983), with maxent weighting	0.93
Coleman and Pierrehumbert (1997)	0.89
N-gram program (industrial standard, “GRM Library” from ATT Labs)	0.89
Our model, but no features (e.g. [t] = “[+t]”)	0.88
Analogical model of Bailey and Hahn 2001	0.83

SCALING UP THE MODEL: THE ROLE OF PHONOLOGICAL THEORY

54. Signpost

- The English onset system is a test of the **baseline version** of our model (representations modeled on Chomsky and Halle 1968).
- This version turns out to need amplification, due to issues of **locality** and **counting**.

55. Why it's bad for learning to have to count high

- The **number of possible constraints** is roughly equal to:
 - C^m where
 - C = number of natural classes
 - m = number of feature matrices permitted in a constraint
- This is a geometric progression and quickly becomes prohibitively high for learning.

56. But phonology *can* scan long distances

- **Vowel harmony** can affect vowels across long consonant clusters:
- **Stress.** The constraint enforcing the three-syllable window of Spanish, counted in segments, can target the sixth-to-last segment:

57. One remedy: phonological theory

- Tiers, grids make what might appear nonlocal be local formally—part of their original intent.
- This works suggests a clear payoff in terms of learnability.

58. Vowel harmony

- Provide a “projection” consisting only of the vowels, and let learning take place on that projection as well.
- Shona (Bantu) has height harmony (roughly: high after high, mid after mid)

59. Shona vowel distribution

- e, o* may occur as follows:
 - in initial syllables, as in *beka* ‘belch’, *gondwa* ‘become replete with water’.
 - e* may occur non-initially if the preceding vowel is *e* or *o*, as in *cherenga* ‘scratch’, *fovedza* ‘dent’.
 - o* may occur non-initially only if the preceding vowel is *o*, as in *dokonya* ‘be very talkative’.
- i, u* may occur as follows.
 - in initial syllables, as in *gwisha* ‘take away’, *huna* ‘search intently’.
 - i* may occur non-initially unless the preceding vowel is *e* or *o*, as in *kabida* ‘lap (liquid)’, *bhigidza* ‘hit with thrown object’, *churidza* ‘plunge, dip’.
 - u* may occur non-initially unless the preceding vowel is *o*, as in *baduka* ‘split’, *bikura* ‘snatch and carry away’, *chevhura* ‘cut deeply with sharp instrument’, *dhuguka* ‘cook for a long time’.
- a* is freely distributed.³

³ However, in our learning data, final vowels are always /a/, since the dictionary entries for verbs all end with the suffix /-a/.

60. Shona vowel distribution: corpus data

<i>Vowel sequence</i>	<i>Count</i>	<i>Ad hoc O/E</i>	<i>Status</i>	<i>Classification</i>
a a	1443	1.03	✓	
a e	3	0.02	*	Noninitial <i>e</i> without harmony trigger
a i	500	1.69	✓	
a o	0	0.00	*	Noninitial <i>o</i> without harmony trigger
a u	568	1.24	✓	
e a	639	0.77	✓	
e e	587	5.30	✓	
e i	2	0.01	*	<i>i</i> not lowered after <i>e</i>
e o	0	0.00	*	Noninitial <i>o</i> without harmony trigger
e u	260	0.96	✓	<i>e</i> not a lowering trigger for back vowels
i a	1130	1.14	✓	
i e	0	0.00	*	Noninitial <i>e</i> without harmony trigger
i i	478	2.29	✓	
i o	0	0.00	*	Noninitial <i>o</i> without harmony trigger
i u	175	0.54	✓	
o a	638	0.75	✓	
o e	153	1.35	✓	
o i	23	0.13	?	<i>i</i> not lowered after <i>o</i> (weak trigger)
o o	694	6.56	✓	
o u	20	0.07	?	<i>u</i> not lowered after <i>o</i> (weak trigger)
u a	1737	1.14	✓	
u e	4	0.02	*	Noninitial <i>e</i> without harmony trigger
u i	175	0.55	✓	
u o	1	0.005	*	Noninitial <i>o</i> without harmony trigger
u u	811	1.63	✓	

61. Learned vowel projection grammar for Shona: harmony constraints

<i>Constraint</i>	<i>Weight</i>	<i>Comment</i>
a. *[^-high,-low][-high,-low]	5.017	*mid unless preceded by mid
b. *[^-low,+back][-high,-low,+back]	4.429	*o unless preceded by o
c. *[-high,-back][+high,-back]	1.909	*ei
d. *[-high,-low][+high,-back]	2.331	*[eo]i
e. *[-high,-low,+back][+high,+back]	2.265	*ou

where ^ means “unless”

- This is more complicated than the traditional description, but it gets the nuances: [o] is a slightly weak trigger.

62. System gets lost without the vowel projection

VCCCV is possible in Shona, and the search space is huge (recall: Cⁿ)

63. An approach that might be able to learn the projection.

see Goldsmith and Xanthos (2009)⁴

QUESTIONS AND FUTURE DIRECTIONS

64. Hidden structure

- Syllables, onsets, rhymes, affricate/cluster distinction
- These help us make sense of phonotactics, but are not observable.
- See Tesar and Smolensky (2000) *Learnability in Optimality Theory* for one approach.

65. How does this fit into the architecture of phonological theory?

- From the viewpoint of Optimality Theory, this is a sore thumb.
- Why? Phonology does two things:
 - Account for the phonological well-formedness of words, phrases, etc.
 - Account for **alternation**: same entity appears in different forms in different styles and contexts: [tɹɪmɒ] vs. [dʒʌmpɪ]
- OT does both at once. We say [dʒʌmpɪ], not [dʒʌmpɒ], because [dʒʌmpɒ] is phonotactically impossible.
- I'm ambivalent about whether this is an advantage—much alternation has no phonotactic basis.

66. Learning-theoretic phonology

- Learn the phonotactics first
 - It can be done early, when you've only accomplished word segmentation.
 - It probably *is* done early; perhaps around 9 months (Juszyck and colleagues)
- Phonotactic knowledge then serves as a guide to learning alternations. ("Hmm, my emerging grammar would lead me to say [dʒʌmpɒ], but I know that's very unlikely.")
- We'll see an actual tiny implementation of this tomorrow, when we start learning alternations.

67. Does the system learn junk?

- Our whole-language study was Wargamay (Australia, Dixon 1981).
- We caught everything Dixon did—and he is pretty good!
- But we caught 57 more constraints, which tend to be complicated may be accidentally true.

⁴ John Goldsmith and Aris Xanthos. "Learning Phonological Categories." *Language* 85.1 (2009): 4-38

68. Let's look at English

- For this lecture I found a file with 3800 English monosyllables and caused the system to learn 80 constraints.
- Here are some of them, with my seat-of-the pants classification.
- Sensible and straightforward

*[-round,-low,+back][-consonantal]	3.217	0		*Δr
*[+nasal,+coronal][+labial]	3.124	0	nasal place assimilation	*ɪnp
*[+nasal,+coronal][+dorsal]	2.568	0	nasal place assimilation	*ɪnk
*[+word_boundary][+word_boundary] (vowel tier)	4.153	0	*no vowels	*pst

- Sensible and surprising

*[-back][+diphthong]	2.75	0	yaw you yay	[jaɪ, jaʊ, jɔɪ]
----------------------	------	---	-------------	-----------------

- Accidentally true?

*[+continuant,+voice,+strident][+diphthong]	2.214	0	zoy, zie, zow	[zɔɪ, zɑɪ, zɑʊ]
*[+continuant,+voice,+coronal][+back,- tense]	2.777	1	zup, zop	[zʌp]

69. Remedies for the accidentally true

- Computational: Colin Wilson has recently figured how to:
 - calculate the predicted probability of the training data⁵
 - Use this to do significance testing of added constraints.
- Theoretical/UG: vet the constraints for phonetic sensibleness

70. What I'd like to do

- A wug test including many words like *zie*.
- If they sound ok to the native speakers, explore ways to avoid making the wrong prediction.

⁵ Unknown to us in the published work; it sufficed to know the gradients to climb the mountain...