

## Lecture 20: From language acquisition to language change

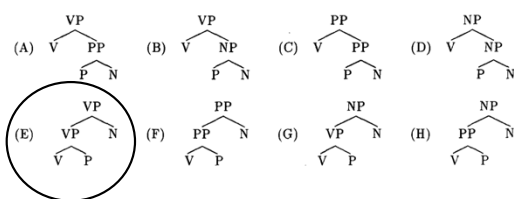
Professor Robert C. Berwick

## Menu

- Administrivia: Project endgame
- Why Bayesian CFG learning goes astray
- How we might repair it: the ‘classical’ linguistic methodology
- From the logical problem of language acquisition to the illogical problem of language change

6.863J SP11

## “walking on ice”



(A) Is the right structure. Why? Can a stochastic CFG learning algorithm find (A), rather than the other structures?

In fact, this turns out to be hard. The SCFG picks (E)! Why? Entropy of (A) turns out to be higher (worse) than (E)-(H). Learner that uses this will go wrong.

## Some terminology

- *Entropy* of a discrete random variable  $A$ , denoted  $H(A)$

$$H(A) \equiv \sum_a -p_A(a) \log p_A(a).$$

- *Cross-entropy between 2 distributions*  $p_A, p_B$ , a measure of how well distribution  $p_B$  predicts  $A$ , a minimum when distributions of  $A$  and  $B$  are identical:

$$\hat{H}_A(B) \equiv \sum_a -p_A(a) \log p_B(a),$$

- So if  $B$  is a PCFG, and  $A$  a corpus, this is a measure of how well the grammar models the corpus

6.863J SP11

### More terminology

- *Joint entropy* and *conditional entropy* of two random variables  $A, B$ :

$$H(A, B) \equiv \sum_{a,b} -p_{A,B}(a, b) \log p_{A,B}(a, b).$$

$$H(A|B) \equiv \sum_{a,b} -p_{A,B}(a, b) \log p_{A|B}(a, b).$$

- So this measures uncertainty in joint distribution & then the uncertainty of  $A$  given knowledge of  $B$
- $H(A, B) = H(A) + H(B|A) = H(B) + H(A|B)$
- Finally, we have

6.863J SP11

### Terminology

- *Mutual information* between  $A, B$ , a measure of the dependence between the two, 0 iff  $A, B$  independent; o.w. between 0 and  $\min(H(A), H(B))$

$$\begin{aligned} I(A, B) &\equiv \sum_{a,b} p_{A,B}(a, b) \log \frac{p_{A,B}(a,b)}{p_A(a)p_B(b)} \\ &= H(A) - H(A|B) \\ &= H(B) - H(B|A) \end{aligned}$$

- Note then that:

$$H(A, B) = H(A) + H(B|A) = H(A) + H(B) - I(A, B)$$

6.863J SP11

### Why is the entropy higher for the wrong structure?

Consider a similar example from word strings:

*Theguysawthedog*

We can find the 'break' between *the* and *guy*

BUT what about:

*Shewaspassedbytherunner*

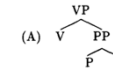
This will yield the incorrect 'word' *edby* because *ed* often precedes *by* in English

Similarly, verbs often followed by prepositions, because PPs adjoined to VPs...

6.863J SP11

### Generating this word sequence with a grammar

Now imagine a grammar generating the two-word sequence  $AB$ , via some rule  $X \rightarrow AB$ . We first generate  $A$ , chosen from distribution  $p_A$ , and then word  $B$ , from distribution  $p_B$ \* conditioned on  $p_A$  Q: can we get structure below?

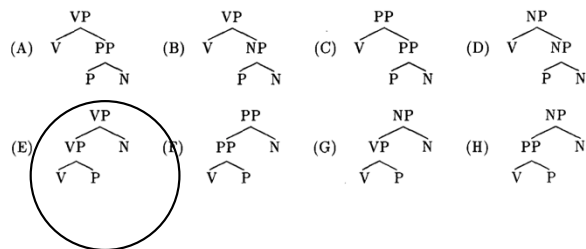


But in English, verbs+prepositions are more closely coupled than prepositions+nouns, semantically. So we expect the *mutual information* between the verb and the preposition to be *greater than* the mutual information between the preposition and the noun, and greater still than between the verb and the noun:

$$I(V, P) > I(P, N) > I(V, N)$$

6.863J SP11

Recall: (E) is preferred to (A)



6.863J SP11

Why structure (E) is preferred to (A)

Entropy for structure (A) is higher than that for (E)

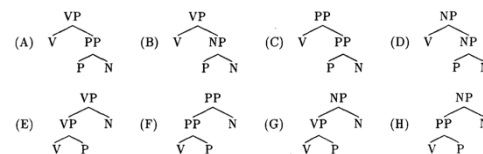
For structure (A), the entropy is:

$$(1) H(V)+H(P)+H(N|P)=H(V)+H(P)-I(P,N)$$

For structure (E), entropy is:

$$(2) H(V)+H(P)+H(P|V)=H(V)+H(P)-I(V,P)$$

But since we assumed  $I(V,P) > I(P,N)$ , so (2) beats (1)!



In general, this happens even with simple rule sets and grammars: convergence to suboptimal grammar

Example ('head' grammar)

$S \rightarrow AP$	$S \rightarrow CP$	$BP \rightarrow B CP$
$AP \rightarrow A BP$	$CP \rightarrow AP C$	$BP \rightarrow AP B$
$AP \rightarrow A CP$	$CP \rightarrow BP C$	$BP \rightarrow B$
$AP \rightarrow A$	$CP \rightarrow C$	

3 'parts of speech', A, B, C;

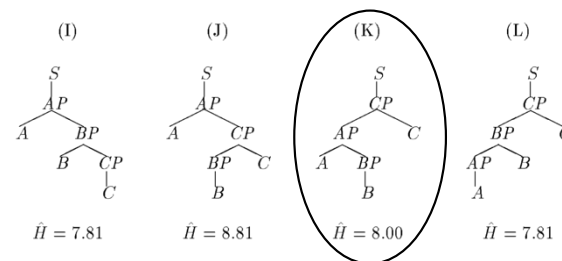
To model V, P, N: Assume:  $I(A,B)=1$ ;  $I(B,C)=0.188$ ;  $I(A,C)=0$

Assume uniform rule probabilities initially

sentence: ABC

6.863J SP11

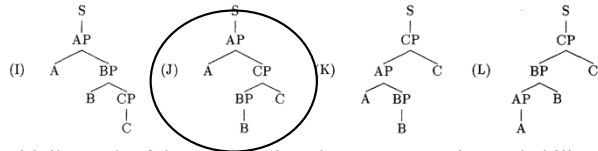
Here's what EM picks



It is 'attracted' to the wrong structure with suboptimal entropy (likelihood) value...let's see why

6.863J SP11

### 4 possible parses of "ABC"



Initially, each of these parses has the same posterior probability: Because we start with same pr's, and each parse uses the same number of rule expansions, and, finally, the bigrams are uniform at the start.

The estimated probability of a rule *after* the first pass is directly proportional to how many of these parse trees the rule occurs in

The rules that occur more than once are:  
 AP → A BP (parses I, K)  
 CP → BP C (parses J, L)      J, K have two  
 BP → B (parses J, K)

Why doesn't EM move to global optima from J to I, L?

If system starts at J, why can't it move to K, I, L?

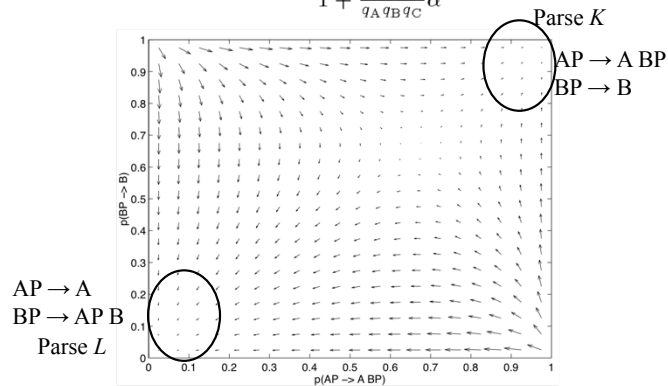
Answer: look at what has to change: 3 rules must have their nonterminals switched

(J)	(L)
$q_A: AP \rightarrow A BP$	$AP \rightarrow A$
$q_B: BP \rightarrow B$	$BP \rightarrow AP B$
$q_C: CP \rightarrow AP C$	$CP \rightarrow BP C$

6.863J SP11

### Update equation for the $q$ rule probabilities

$$q_A \cdot q_B \cdot q_C \leftarrow \frac{1}{1 + \frac{\bar{q}_A \bar{q}_B \bar{q}_C}{q_A q_B q_C} \alpha}$$



But there's another way to look at search by Bayesian methods....

6.863J SP11

We want the best (highest probability)  $G_i$  given the data sequence  $S_j$

$$\begin{aligned}
 p(G_i | S_j) &= \frac{p(G_i) \times p(S_j | G_i)}{p(S_j)} \\
 &= \frac{p(G_i) \times p(S_j | G_i)}{p(S_j)} \\
 \arg \max & \\
 &= \arg \max p(G_i) \times p(S_j | G_i) \quad (\text{since } S_j \text{ constant})
 \end{aligned}$$

And we can compute this! We just need to ‘search’ through all the grammars and find the one that maximizes this... can this be done? Horning has a general result for unambiguous CFGs; for a more recent (2011) approach that works with 2 simple grammar types & child language – see Perfors *et al.* Note: again, the  $G$ ’s only ‘approach’ the best  $G$  with increasing likelihood

6.863J SP11

Another view of this ‘maximize posterior probability’ view

$$= \arg \max p(G_i) \times p(S_j | G_i)$$

Now let’s assume:

- (1) that  $p(G_i) \propto 2^{-|G_i|}$  so that smaller grammars are more probable;
  - (2) by Shannon’s source coding theorem, optimal encodings of the data  $S_j$  wrt grammar  $G_i$  approaches  $-\log_2 p(S_j | G_i)$
- Then maximizing this ‘posterior probability’ becomes, after taking  $\log_2$ , is equivalent to finding the minimum of:  $|G_i| + |S|$  with  $S$  coded by  $G_i$ , which, by Shannon’s coding thm is equivalent to minimizing  $|G_i| - \log_2 p(S_j | G_i)$

This is usually called minimum description (MDL)

We want to find the shortest (smallest) grammar *plus* the encoding of the data using that grammar

6.863J SP11

- Most restrictive grammar just lists all possible utterances
- Only the observed data is grammatical, so it has a high probability
- A simple grammar could be made that allowed any sentences
- Grammar would have a high probability
- But data a *very* low one

MDL finds a middle ground between always generalizing and never generalizing

6.863J SP11

## Complexity and Probability

- More complex grammar
- Longer coding length, so lower probability
- More restrictive grammar
- Fewer choices for data, so each possibility has a higher probability

6.863J SP11

## Minimum description length as a criterion has a long pedigree...

Given the fixed notation, the criteria of simplicity governing the ordering of statements are as follows: that the shorter grammar is the simpler, and that among equally short grammars, the simplest is that in which the average length of derivation of sentences is least.

Chomsky, 1949, *Morphophonemics of Modern Hebrew*

So, this MDL criterion was there from the start:

Minimize the grammar size and  
Minimize the length of the exceptions that can't be compressed by the grammar + data that can be...

6.863J SP11

## Example of how grammar 'compresses' data: English auxiliary system

[0 auxiliary verbs]

1. John  $\emptyset$  eats

[1 aux verb]

2. John will eat $\emptyset$

3. John has eaten

4. John is eating

[2 aux verbs]

5. John M(odal) Be eating

6. John M H(ave) eaten

7. John H Be-en eating/en

[3 aux verbs]

8. John M H B eating

6.863J SP11

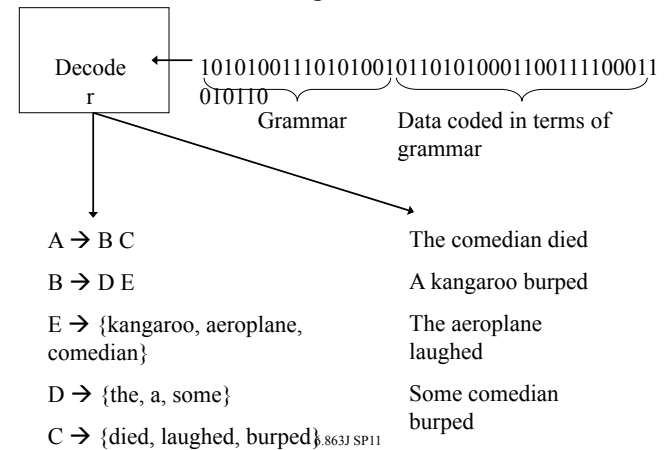
## 8 rules compressed to 1 = a *generalization*

Aux  $\rightarrow$  (M)(H)(B) Verb-stem

*Note* how the choice of a coding function matters!

6.863J SP11

## Encoding Grammars and Data



## Rampant Synonymy?

- Inductive inference (Solomonoff, 1960a)
- Kolmogorov complexity (Kolmogorov, 1965)
- Minimum Message Length (Wallace and Boulton, 1968)
- Algorithmic Information Theory (Chaitin, 1969)
- Minimum Description Length (Rissanen, 1978)
- Minimum Coding Length (Ellison, 1992)
- Bayesian Learning (Stolcke, 1994)
- Minimum Representation Length (Brent, 1996)

6.863J SP11

## Evaluation and Search

- MDL principle gives us an evaluation criterion for grammars (with respect to corpora)
  - But it doesn't solve the problem of how to find the grammars in the first place
- Search mechanism needed

6.863J SP11

## Two Learnability Problems

- How to determine which of two or more grammars is best given some data
- How to guide the search for grammars so that we can find the correct one, without considering every logically possible grammar

6.863J SP11

## MDL in Linguistics

- Solomonoff (1960b): 'Mechanization of Linguistic Learning'
- Learning phrase structure grammars for simple 'toy' languages: Stolcke (1994), Langley and Stromsten (2000)
- Or real corpora: Chen (1995), Grünwald (1994)
- Or for language modelling in speech recognition systems: Starkie (2001)

6.863J SP11

## Not Just Syntax!

- Phonology: Ellison (1992), Rissanen and Ristad (1994)
- Morphology: Brent (1993), Goldsmith (2001)
- Segmenting continuous speech: de Marcken (1996), Brent and Cartwright (1997)

6.863J SP11

## An Example

Learns simple phrase structure grammars

- Binary or non-branching rules:

$A \rightarrow B C$

$D \rightarrow E$

$F \rightarrow \text{tomato}$

- All derivations start from special symbol  $S$
- *null* symbol in 3<sup>rd</sup> position indicates non-branching rule

6.863J SP11

## Encoding Grammars

Grammars can be coded as lists of three symbols

- First symbol is rules left hand side, second and third its right hand side

A, B, C, D, E, null, F, tomato, null

- First we have to encode the frequency of each symbol

6.863J SP11

## Encoding Data

1  $S \rightarrow NP VP$  (3) } Total frequency for  $S = 3$   
2  $NP \rightarrow \text{john}$  (2) } Total frequency for  $NP = 3$   
3  $NP \rightarrow \text{mary}$  (1) }  
4  $VP \rightarrow \text{screamed}$  (2) } Total frequency for  $VP = 3$   
5  $VP \rightarrow \text{died}$  (1) }

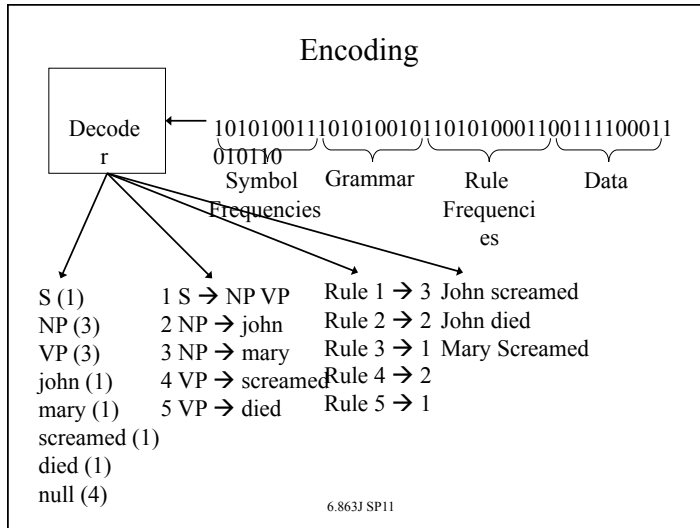
Data: 1, 2, 4, 1, 2, 5, 1, 3, 4

Probabilities: 1  $\rightarrow 3/3$ , 2  $\rightarrow 2/3$ , 4  $\rightarrow 2/3$ ,

1  $\rightarrow 3/3$ , 2  $\rightarrow 2/3$ ...

We must record the frequency of each rule

6.863J SP11



- ### Search Strategy
- Start with simple grammar that allows all sentences
  - Make simple change and see if it improves the evaluation (add a rule, delete a rule, change a symbol in a rule, etc.)
  - Annealing search
  - First stage: just look at data coding length
  - Second stage: look at overall evaluation
- 6.863J SP11

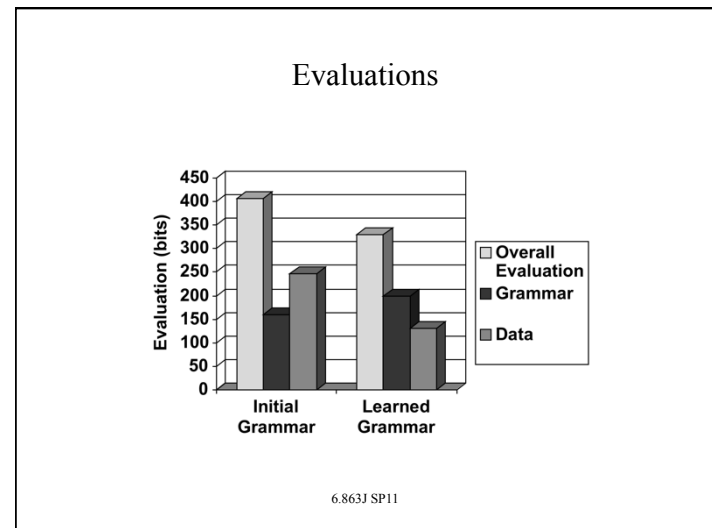
### Example: English

John hit Mary  
 Mary hit Ethel  
 Ethel ran  
 John ran  
 Mary ran  
 Ethel hit John  
 Noam hit John  
 Ethel screamed  
 Mary kicked Ethel  
 John hopes Ethel thinks Mary hit Ethel  
 Ethel thinks John ran  
 John thinks Ethel ran  
 Mary ran  
 Ethel hit Mary  
 Mary thinks John hit Ethel  
 John screamed  
 Noam hopes John screamed  
 Mary hopes Ethel hit John  
 Noam kicked Mary

#### Learned Grammar

S → NP VP  
 VP → ran  
 VP → screamed  
 VP → Vt NP  
 VP → Vs S  
 Vt → hit  
 Vt → kicked  
 Vs → thinks  
 Vs → hopes  
 NP → John  
 NP → Ethel  
 NP → Mary  
 NP → Noam

6.863J SP11



## Dative Alternation

- Children learn distinction between alternating and non-alternating verbs
- Previously unseen verbs are used productively in both constructions
  - New verbs follow regular pattern
- During learning children use non-alternating verbs in both constructions
  - U-shaped learning

6.863J SP11

## Training Data

- Three alternating verbs: *gave, passed, lent*
- One non-alternating verb: *donated*
- One verb seen only once: *sent*

The museum lent Sam a painting

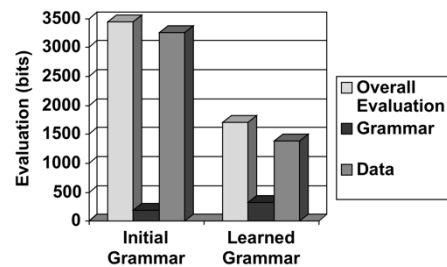
John gave a painting to Sam

Sam donated John to the museum

The museum sent a painting to Sam

6.863J SP11

## Dative Evaluations



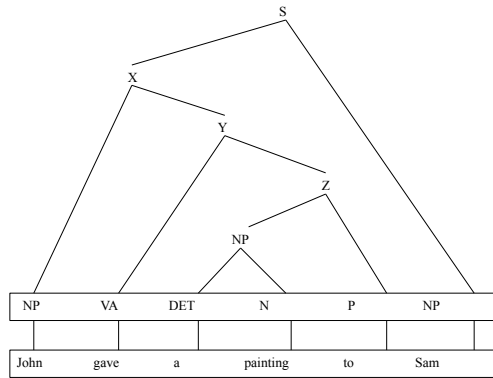
6.863J SP11

## Grammar Properties

- Learned grammar distinguishes alternating and non-alternating verbs
- *sent* appears in alternating class
- With less data, only one class of verbs, so *donated* can appear in both constructions
- All sentences generated by the grammar are grammatical
- But structures are not right

6.863J SP11

## Learned Structures



6.863J SP11

## Regular and Irregular Rules

- Why does the model place a newly seen verb in the regular class?

$Y \rightarrow VA NP$   
 $Y \rightarrow VA Z$   
 $Y \rightarrow VP Z$   
 $VA \rightarrow passed$   
 $VA \rightarrow gave$   
 $VA \rightarrow lent$   
 $VP \rightarrow donated$

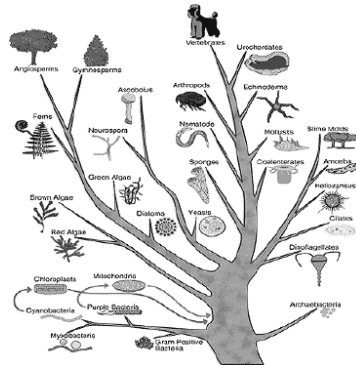
	<i>sent doesn't alternate</i>	<i>sent alternates</i>
Overall Evaluation (bits)	1703.6	1703.4
Grammar (bits)	322.2	321.0
Data (bits)	1381.4	1382.3

VA / VP  $\rightarrow$  sent

Regular constructions are preferred because the grammar is coded statistically

6.863J SP11

Why are there distinct species?  
Why are there distinct *language* species?



6.863J SP11

Nature 449, 665-667 (11 October 2007)

“A language is a dialect with an army and a navy”

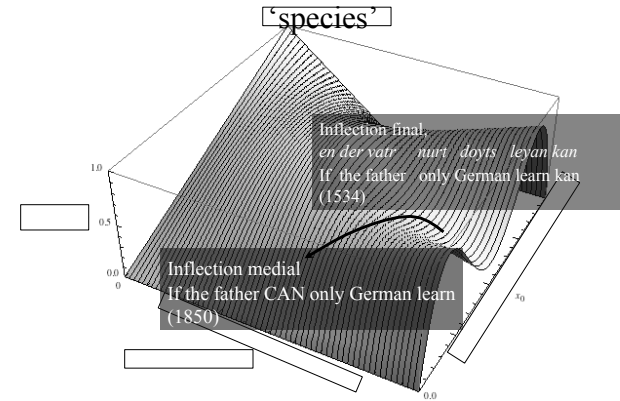
6.863J SP11

### Why are there species at all?

- For Darwin, evolution was the conversion of standing variation *among individuals* into variation *between groups* in space and time
- Question: But what about human language?
- Answer: Distinct human language ‘species’ arise due to individual variation in language ‘trait inheritance’ (learning), paralleling Darwin’s theory (a kind of ‘symmetry breaking’ in the dynamical systems mapping ‘genotype’ to ‘phenotype’ from generation to generation, based on the linguistic notion of a ‘parameter’)

6.863J SP11

### Result: ‘Attractor’ basins for possible language



(Yiddish, Data from Santorini, 1993)

6.863J SP11

### ‘Analysis of variance’ accounting for observed distribution of organisms

Factors (neither exclusive nor exhaustive):

1. *Where* we are in the ‘tree of life’;
  2. *What* biochemical/physical options are there at this point;
  3. *What* is selected (sieved) at this point
- What proportion of the ‘observed variance’ does each factor account for?

Guess: Factor 1, 80–85%

Factor 2, 18%

Factor 3, 2%

That is, *contingent history*

Is the same true for the array of possible human languages?

6.863J SP11

### Adds a new explanatory condition: the relevance of contingent history and ‘gaps’ in the ‘fossil record’ (no, not *that* kind of gap)

- “Phenotypic inertia”: how much of the observable distribution of (biological) species is fixed by *where* it is on the ‘tree of life’
- You can’t get there from here: some possible species are *unattainable*, not due to biological/physical possibility, but due to historical contingency
- Initial conditions can, and do, *matter*

6.863J SP11

### Explanation by dynamical systems

- Only *certain* ‘language species’ will be on possible dynamical trajectories; others might be ‘inaccessible’
- Only *certain* distinctions (‘parameters’) will lead to phase changes; others will only lead to gradual clines
- The Arrow of Time: as a result, sometimes it might be impossible to ‘go backwards’ (eg, loss of verb 2nd is *stable*)
- How many of the ‘gaps’ in observable languages are due to dynamical system trajectory constraints alone?

6.863J SP11

### A famous quote about idealization

“Linguistic theory is concerned primarily with an ideal speaker-listener, in a completely homogenous speech-community, who knows and acquires its language perfectly and instantaneously [... ]no cogent reason for modifying [this position] has been offered.” – N. Chomsky, 1965.

What is *not* Darwinian here?  
How can we make linguistics more ‘biolinguistic’?

6.863J SP11

These idealizations of modern linguistic theory *block* a fully Darwinian view & lead to ‘the paradox of language change’

‘Galilean’ idealizations:

Homogenous language community  
Perfect language acquisition (learning)  
Instantaneous language acquisition

An explanatory paradox: given *perfect* language learning from generation to generation, there could be no possibility language change, and no distinct language species would ever emerge

*Relax* these constraints – and we obtain language ‘species’ w/ possible phase transitions (no, not *that* kind of phase)

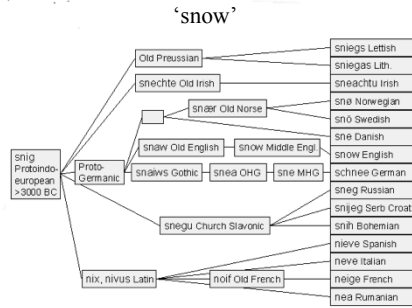
6.863J SP11

### Resolving the paradox: the remainder of this talk

1. The machinery of linguistic trait inheritance (learning replaces Mendelism)
2. Two ways of incorporating population & individual variation in dynamical system models
  - a) Version 1: Learning from 1 person, “Iterated Learning” (aka, ‘single parenting’)
  - b) Version 2: Learning from 2 or more people, “Social Learning”
3. How linguistic ‘species’ emerge as fixed points from model version 2 (but not from version 1)
4. A new explanatory adequacy?

6.863J SP11

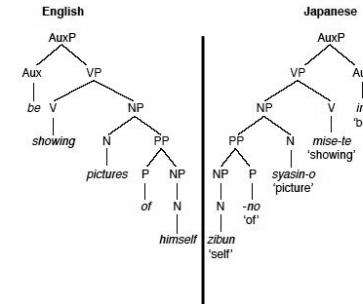
The conventional ‘language taxonomy’ is based on ‘external sound shapes traits’ (usually ‘word cognates’)



6.863J SP11

Reminder re ‘linguistic traits’ (salivate if it rings a bell)

- One single “deep” switch only (like Pax-6 *eyeless* gene, though there are *important* differences)



Mirror image

6.863J SP11

Modern Italian, Spanish, French, European Portuguese (It, Sp, Fr, Ptg), Latin (Lat), Classical Greek (ClG), New Testament Greek Koiné (NT), Modern Greek (Grk), Gothic (Got), Old English (OE), Modern English, German (E, D), Modern Bulgarian (Blg) and Serbo-Croat (SC), Standard Arabic (Ar)

Reminder 2: There are other language ‘traits’...

	It	Sp	Fr	Ptg	Lat	ClG	NT	Grk	OE	E	D	Blg	SC	Ar
1. = gramm. number	+	+	+	+	+	+	+	+	+	+	+	+	+	+
2. = gramm. count / null article	+	+	+	+	+	+	+	+	+	+	+	+	+	+
3. = gramm. def in DP	+	+	+	+	+	+	+	+	+	+	+	+	+	+
4. = gramm. sample in D -3	+	+	+	+	+	+	+	+	+	+	+	+	+	+
5. = strong inf. def in D -3	+	+	+	+	+	+	+	+	+	+	+	+	+	+
6. = number affix on N (BNS)	+	+	+	+	+	+	+	+	+	+	+	+	+	+
7. = ambiguous singulars =-1	+	+	+	+	+	+	+	+	+	+	+	+	+	+
8. = def checking APs	+	+	+	+	+	+	+	+	+	+	+	+	+	+
9. = Aux = PP =-art <sup>1</sup> , -12	+	+	+	+	+	+	+	+	+	+	+	+	+	+
10. = def spread on APs =+3 or =8	+	+	+	+	+	+	+	+	+	+	+	+	+	+
11. = def spread on Mod =+3 or =8	+	+	+	+	+	+	+	+	+	+	+	+	+	+
12. = morphic def. formatc	+	+	+	+	+	+	+	+	+	+	+	+	+	+
13. = inversely ordered As	+	+	+	+	+	+	+	+	+	+	+	+	+	+
14. = N over acc. arg.	+	+	+	+	+	+	+	+	+	+	+	+	+	+
15. = N over GenO =+4	+	+	+	+	+	+	+	+	+	+	+	+	+	+
16. = N over low As =-15, -13	+	+	+	+	+	+	+	+	+	+	+	+	+	+
17. = N over MD As =-16	+	+	+	+	+	+	+	+	+	+	+	+	+	+
18. = N over high As =-17	+	+	+	+	+	+	+	+	+	+	+	+	+	+
19. = free APs in Mod	+	+	+	+	+	+	+	+	+	+	+	+	+	+
20. = free Gens (non-agr)	+	+	+	+	+	+	+	+	+	+	+	+	+	+
21. = structural Gens =-23	+	+	+	+	+	+	+	+	+	+	+	+	+	+
22. = group. Gens (vs. Infl. Gens) =-24	+	+	+	+	+	+	+	+	+	+	+	+	+	+
23. = GenO =-20 or (=21, =22)	+	+	+	+	+	+	+	+	+	+	+	+	+	+
24. = Gens =-20 or (=21, =22)	+	+	+	+	+	+	+	+	+	+	+	+	+	+
25. = possessive pronouns	+	+	+	+	+	+	+	+	+	+	+	+	+	+
26. = obliq. poss. pronouns =-25	+	+	+	+	+	+	+	+	+	+	+	+	+	+
27. = def checking poss. =-25, =-3	+	+	+	+	+	+	+	+	+	+	+	+	+	+
28. = context. affix. As	+	+	+	+	+	+	+	+	+	+	+	+	+	+
29. = Coconsistency principle =-13	+	+	+	+	+	+	+	+	+	+	+	+	+	+
30. = def. checking dem.	+	+	+	+	+	+	+	+	+	+	+	+	+	+

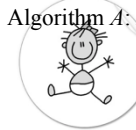
(Guardiano and Longobardi 2003)

1. How does ‘language inheritance’ work?



Data stream  $D = \{s_1, s_2, \dots\}$

Algorithm A:  $D \rightarrow \text{Language } L$



Let’s formalize this a bit more...

6.863J SP11

General learning model & heterogeneous speaker community: a generational picture

- Each such  $g \in \mathcal{G}$  yields a potentially infinite set of expressions  $L_g \in \Sigma^*$ , generated by  $g$ .
- Not all expressions are produced with the same frequency by users of  $g$
- Let  $P_g$  be a probability distribution over the set  $L_g$  such that  $s \in L_g$  is produced with probability  $P_g(s)$  by speakers of  $g$
- $P^{(t)}(g)$  is the proportion of the population using grammar  $g$

6.863J SP11

Example distribution & dfn of learning algorithm

- A heterogeneous population with equal numbers of two languages,  $g$  and  $h$  could be characterized by a distribution  $P^{(t)}$  where  $P^{(t)}(h) = P^{(t)}(g) = \frac{1}{2}$
- The learner is an acquisition algorithm that maps linguistic experience onto linguistic knowledge

$$\mathcal{A} : \cup_{i=1}^{\infty} (\Sigma^*)^i \rightarrow \mathcal{G}$$

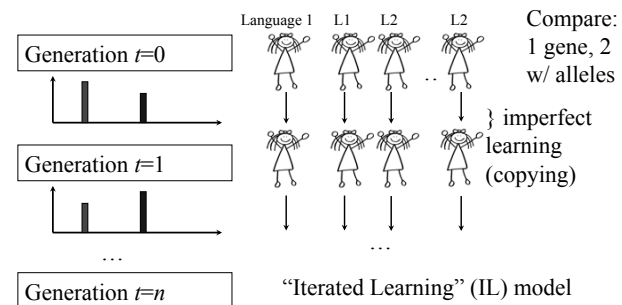
6.863J SP11

This is very general!

- The learning algorithm can be *anything* computable: corpus-based, Bayesian, neural net, class of probabilistic context-free grammars ...
- With respect to linguistic theories, if we allow only a finite number of grammars, then most current 'parameterized' theories are allowed (LFG, HPSG, GB, MP, ....)

6.863J SP11

2a. The simplest population model of 'imperfect' language inheritance: multiple 'agents', with learning from just 1 agent



6.863J SP11

## The population biology ‘sandbox’: simplified system – background assumptions

As in population genetics, initially:

1. Infinite population size
2. Perfect mixing
3. No migration, selection, geographic distortion (gene flow, barriers,...) etc.

Note: all of these can be relaxed, but we start here to gain basic insight

6.863J SP11

## Iterated Learning is Simple

- Single agent and a grammar  $g_1 \in \mathcal{G}$
- Agent then produces  $n$  example sentences, the linguistic experience  $D = (s_1, \dots, s_n)$
- Learner applies the map  $\mathcal{A}$  to attain a mature grammar  $\mathcal{A}(D) = g_2 \in \mathcal{G}$  and proceeds to produce  $D$  for a single agent in the next generation
- Yields the sequence  $g_1 \rightarrow g_2 \rightarrow g_3 \rightarrow \dots$

6.863J SP11

## Iterated learning = a Markov chain

- This is a Markov chain whose state space is  $\mathcal{G}$
- The chain’s state denoted by  $g_t \in \mathcal{G}$  at each point in time  $t$
- The transition matrix of the Markov chain is as follows,
- Where  $T[g, h]$  is the probability the learner would acquire  $h$  given data  $D$  provided to it by an agent using the grammar  $g$  and generating the primary linguistic data  $D$  according to  $P_g$ .

6.863J SP11

## The Markov chain transition matrix

The probability of mapping from  $g$  to  $h$  is given by:

$$T[g, h] = \text{prob}[g \rightarrow h] = \text{prob}[\mathcal{A}(D) = h \mid D \text{ generated by } P_g]$$

Here’s a simple example...

6.863J SP11

### Iterated learning

- Consider a population whose initial state is  $P^{(0)}$
- $P^{(0)}(g)$  is the proportion of  $g$  grammar users in generation 0
- If there is just one teacher, then the distribution evolves as:  

$$P^{(t+1)}(h) = \sum_{g \in \mathcal{G}} P^{(t)}(g) T[g, h]$$
- This has just a single stable equilibrium from all initial conditions
- There are no bifurcations (sudden changes)

6.863J SP11

### Summary: Mathematics of the IL model

$\alpha_t(g)$  = proportion of Language  $g$  speakers in generation  $t$ ;  
 $\alpha_{t+1}(g)$  = proportion of Language  $g$  speakers in generation  $t+1$ :

$T[g, h]$  = probability that learner hears speech from language  $g$  and learns (inherits) language  $h$  – usually the same language as  $g$  (NB:  $T$  is a stochastic matrix)

State update equation (how many people speak the language in the next generation):

$$\alpha_{t+1} = \sum_g \alpha_t(g) T[g, h]$$

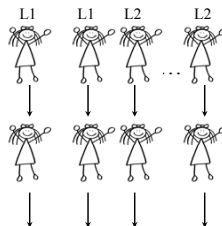
$T$  is a Markov transition matrix;  $\alpha_t$  evolves as a Markov chain according to a linear rule roughly no matter what the ‘learning algorithm’ is – it can be *any* computable function

6.863J SP11

### Example:

The picture with 1 ‘gene’, 2 ‘alleles’ (languages)

$\alpha_t(1)$  = proportion of language 1 speakers, generation  $t$



$\alpha_{t+1}(1)$  = proportion in generation  $t+1$

This is a dynamical system – what are its properties?

6.863J SP11

### The simplest case:

1 ‘gene’ with 2 ‘alleles’ (2 languages, L1, L2)  
 & ‘inheritance’ error  $\epsilon$

$T[1, 1]$  = Probability of hearing Italian, learning Italian

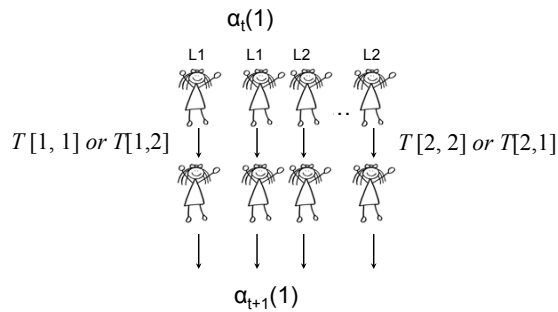
$T[1, 2]$  = Probability of hearing Italian, but (mis)learning English, with some (presumably small) error,  $\epsilon$  (NB, except for me, where  $\epsilon$  is estimated at 0.99 from reliable observers)

$T[2, 2]$  = Probability of hearing English, learning English

$T[2, 1]$  = Probability of hearing English, mislearning Italian, wolog also =  $\epsilon$

6.863J SP11

The picture with 1 'gene', 2 'alleles' (languages)



6.863J SP11

Dynamics of the IL model

Question: Can the IL model yield distinct language 'species'?

Answer: No! It yields neither stable language species, nor admits the possibility of rapid language change

But why?

6.863J SP11

Given IL, what is the steady-state language mixture after many generations?

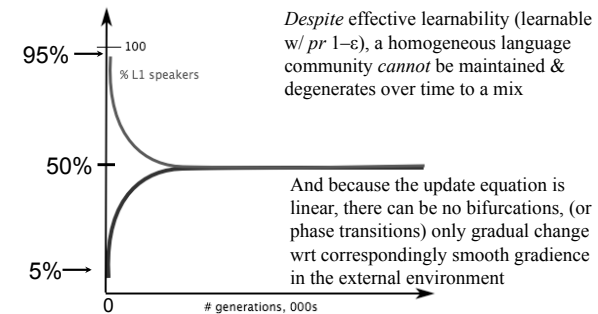
(Perhaps surprising) outcome: just a *single* equilibrium (steady-state) from *all* initial conditions (possible initial distributions of Italian, English speakers), a 50-50 mix of the two languages:

$$\alpha_* = \frac{T(2,1)}{T(2,1) + (1 - T(1,1))} = \frac{\epsilon}{\epsilon + \epsilon} = \frac{1}{2}$$

(Plausible? Do you think that if we all sat around in this villa long enough we'd all wind up in this state after about 10 generations?)

6.863J SP11

Dynamics of Iterated Learning model: cannot account for language stability



6.863J SP11