Learning alternations in a maximum entropy model:

The role of perceptual similarity

James White

University of Ottawa

uOttawa

Role of perceptual similarity in phonological learning?

- Does perceptual similarity play a role during phonological learning?
  - If so, what is that role?

- Part of a larger question: Does phonetic substance play a role in phonological learning?
  - Are there substantive biases[1]?

1. Wilson, 2006

Steriade's P-map proposal

- **P-map**: a mental representation of the relative perceptual similarity between speech sounds in a given context.[1]
  - Perhaps based on an individual's prior perceptual experience.

- Steriade proposed that learners are biased by the P-map when learning phonological patterns.
  - Phonological processes assumed, *a priori*, to involve minimal perceptual modification.

1. Steriade, 2001/2008

Today's talk

- Goal: Investigate this issue by comparing learning models both with and without a similarity bias.

- Test case: Experimental data showing that adult learners disprefer saltatory alternations (White, 2014, *Cognition*).

  = a type of alternation involving not minimal change, but excessive change.

Roadmap

1. Overview of saltatory alternations

2. Overview of experimental results

3. Modeling

1. Saltatory alternation

Saltatory alternation

- Phonological alternation where an intermediate sound is “leaped over”.[1]
- Campidanian Sardinian:[2]
  - /p/ ⟶ [β] / V ___V
  - no change for /b/.

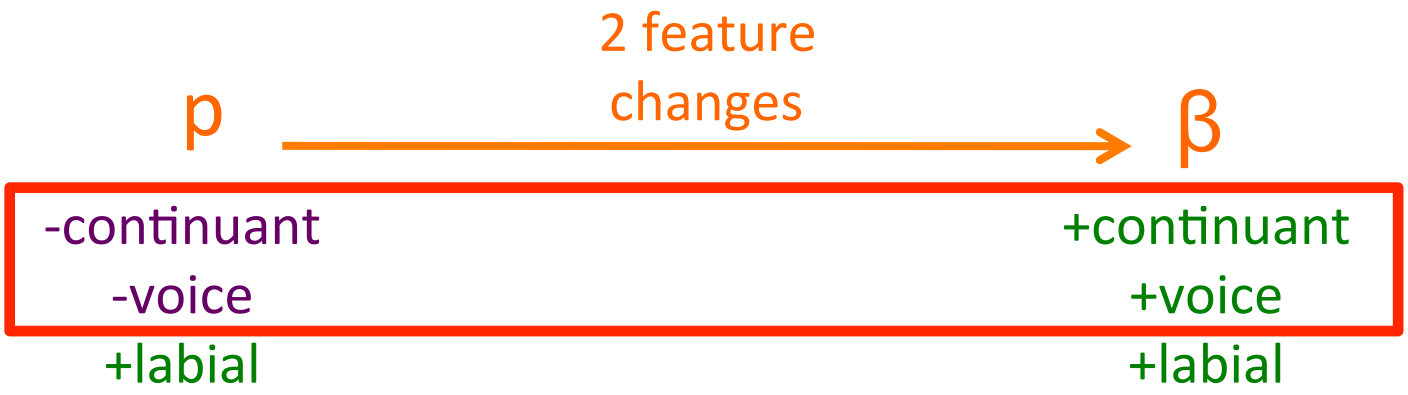
[pãi] ⟶ [s:u βãi] ‘the bread’

[bĩu] ⟶ [s:u bĩu] ‘the wine’

1. White, 2014
2. Bolognesi, 1998

Saltatory alternation

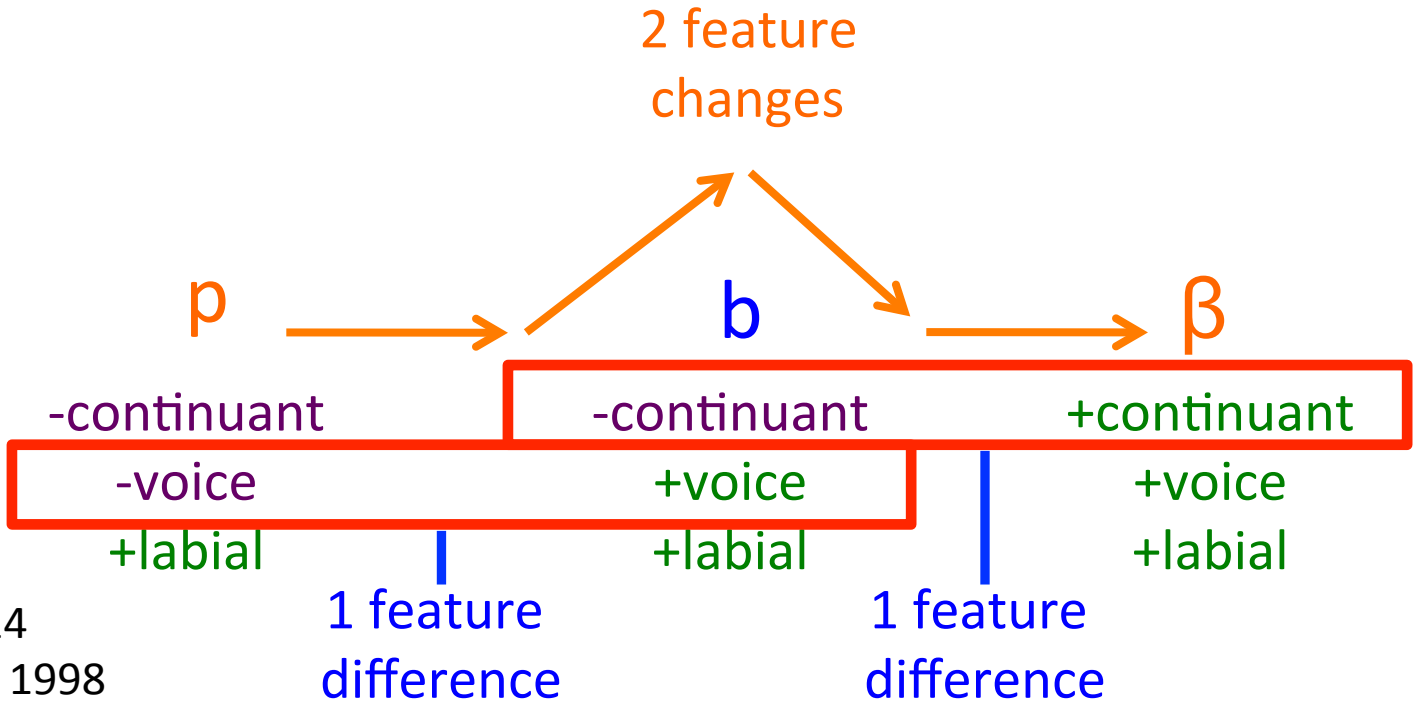
- Phonological alternation where an intermediate sound is “leaped over”.[1]
- <u>Campidanian Sardinian</u>:[2]



1. White, 2014
2. Bolognesi, 1998

Saltatory alternation

- Phonological alternation where an intermediate sound is “leaped over”.[1]
- <u>Campidanian Sardinian</u>:[2]



1. White, 2014
2. Bolognesi, 1998

2. Experimental results

Learners prefer to avoid saltatory alternations.[1]

1. White, 2014

Artificial language experiments (adult learners)

1. Exposure phase

[kamap]

[kamavi]

Also: non-changing fillers like [luman] ⟶ [lumani]

Artificial language experiments (adult learners)

1. Exposure phase

[kamap]

[kamavi]

2. Verification phase

[kamap]

[kamapi]
or
[kamavi]???

3. Generalization phase

[lunub]

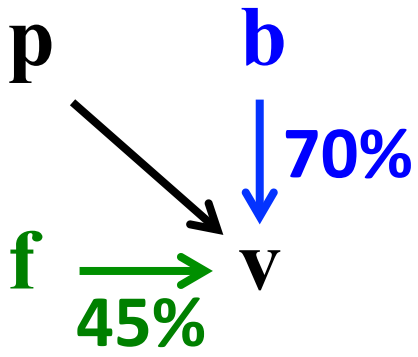
[lunubi]
or
[lunuvi]???

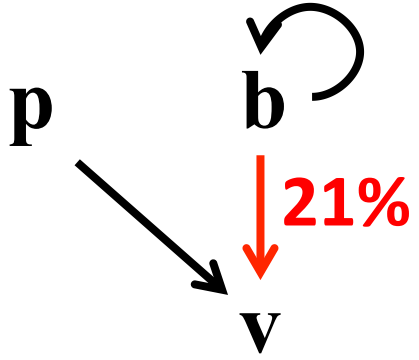
Crucial results to be accounted for

Ambiguous Saltation

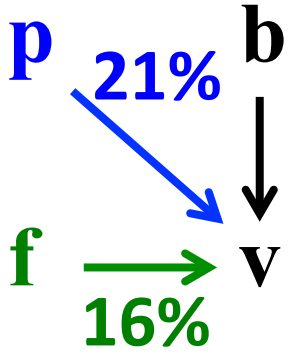
Explicit Saltation

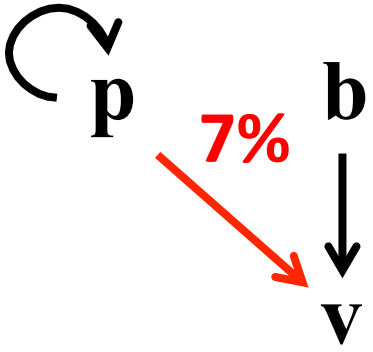
Test condition





Control condition





3. Maximum entropy learning model

For previous uses, see Goldwater & Johnson, 2003; Wilson, 2006; Hayes & Wilson, 2008; Hayes et al., 2009; Martin, 2012; others.

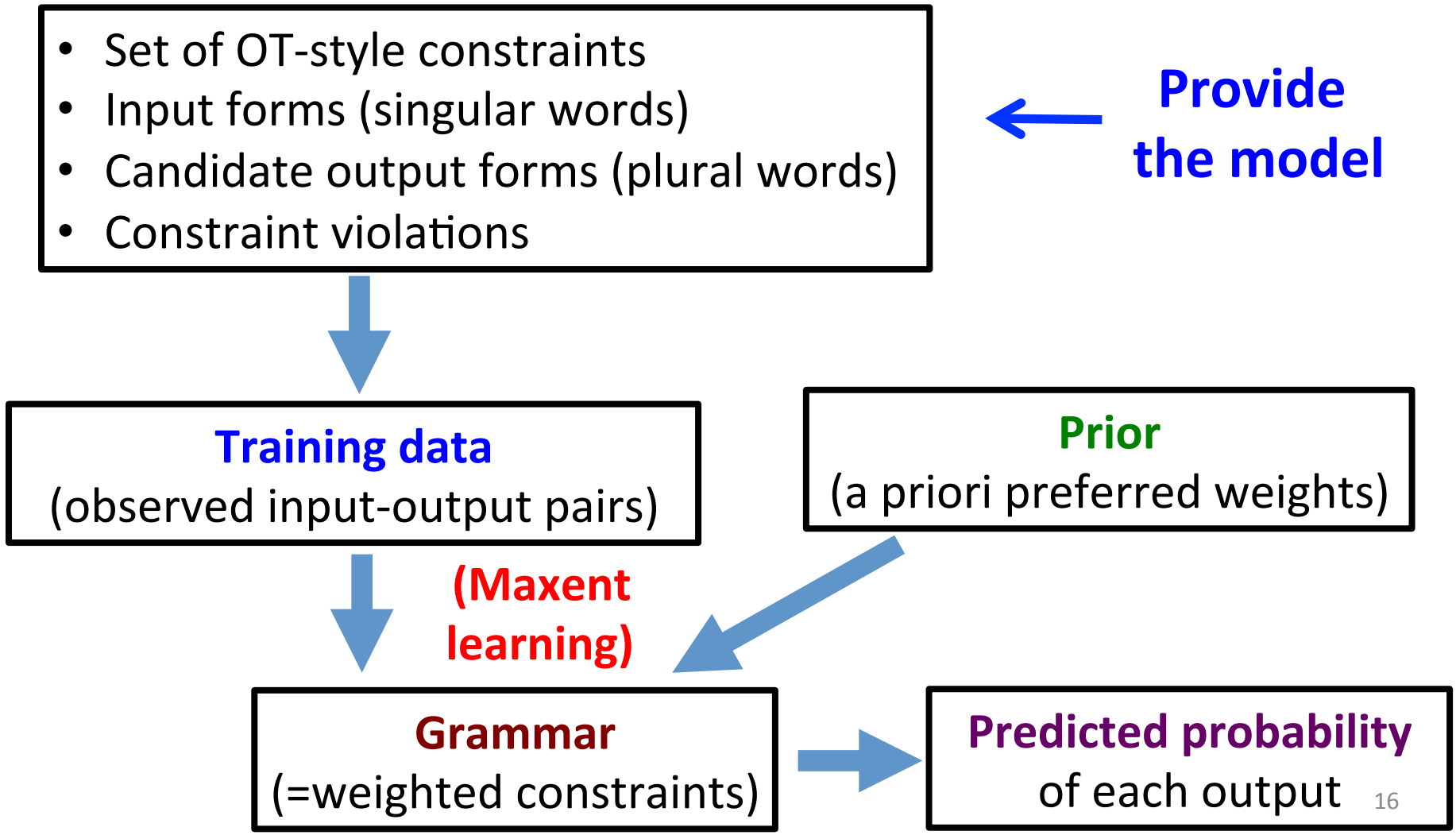
Implemented using the MaxEnt Grammar Tool (available at Bruce Hayes's webpage).

2 things the learning model should account for

1. Saltatory alternations exist and thus must be learnable for the child.

2. This type of alternation is dispreferred by learners.
   - Can we model the experimental results?

Overview of the model



Constraint set

- Feature-based markedness constraints, which motivate alternating.
  - *V [–voice] V
  - *V [–continuant] V

- Correspondence constraints banning alternations between specific pairs of sounds.[1]
  - E.g., *MAP(p,v) = don't have an alternation between [p] and [v].

1. Zuraw, 2007, 2013

Why *MAP constraints?

- Traditional faithfulness constraints (classical OT) cannot generate saltation.[1, 2]
  - /p/ → [v], when [b] is legal, results in a gratuitous faithfulness violation.
  - True, even with weighted constraints.

- Straightforward implementation of the similarity bias.

1. Lubowicz, 2002 2. Ito & Mester, 2003

Training data

- Same as the training data in the experiments.

- Ambiguous Saltation experiment:

input	possible outputs	# observed
p	v	18
	p	0
	f	0
	b	0

input	possible outputs	# observed
t	ð	18
	t	0
	θ	0
	d	0

Prior (= the bias)

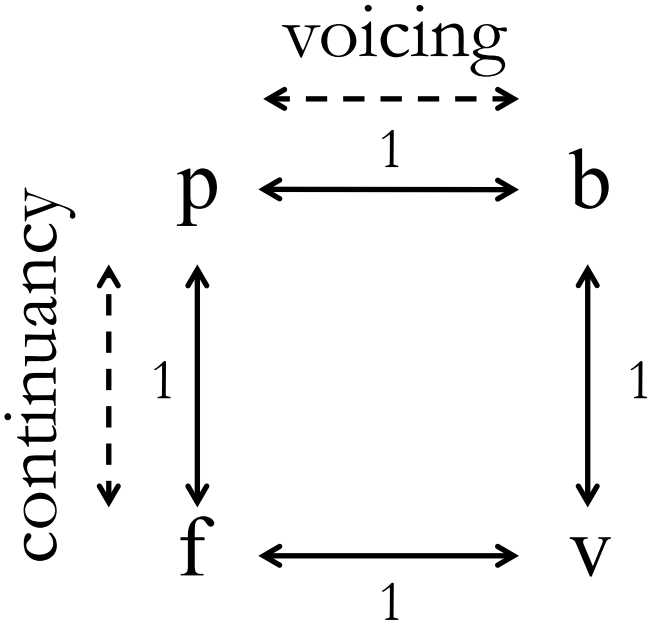
- Biases the learning towards certain outcomes, based on *a priori* assumptions.
  - In this case, based on perceptual similarity.
  - Soft bias, not an absolute restriction.

- The prior is Gaussian. Provide:
  - μ = preferred weight for each constraint.
  - σ^2 = how tightly constraints are held to their preferred weight.

Measure of similarity

- How people judge the similarity of speech sounds is likely complex.[1, 2, 3]
- One possiblity:

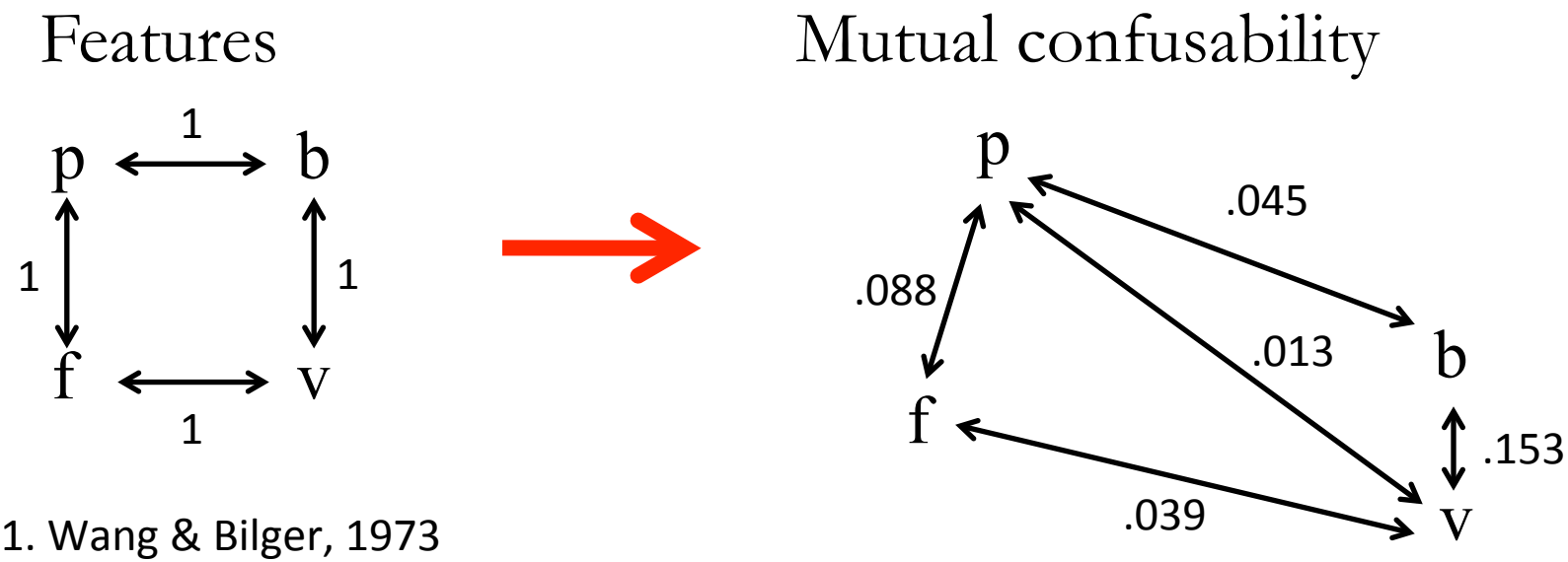
Featural similarity



1. Steriade, 2001/2008 2. Mielke, 2008 3. Cristia et al., 2013

Perceptual similarity

- I use mutual confusability as a simplified measure of perceptual similarity.
  - Data from published confusion experiments with adult English speakers.[1]
- Example:



1. Wang & Bilger, 1973

Setting the prior

- Goal: go directly from confusion proportions to prior constraint weights. (No cherry-picking weights).
- Solution: Train up a separate maxent model intended solely to generate prior weights (μ), based on confusion probabilities.

- Intuitively: Represents the listener's perceptual experience, a computational version of the P-map.

Prior weights

Constraints	Substantively Biased
*V [-voice] V	0
*V [-cont] V	0
*MAP(p, f)	1.34
(p, b)	2.44
(p, v)	3.65
(b, v)	1.30
(b, f)	1.96
(f, v)	2.56

2 other models compared

1. **Anti-alternation**: All *MAP constraints start with the same weight.
   - prior weight = average of prior weights in the substantively biased model.
   - *A priori* preference to avoid any alternation equally, regardless of similarity.

2. **Unbiased**: Prior weight of 0 for all constraint.

Prior weights

Constraints	Substantively Biased	Anti-alternation	Unbiased
*V [-voice] V	0	0	0
*V [-cont] V	0	0	0
*MAP(p, f)	1.34	2.27	0
(p, b)	2.44	2.27	0
(p, v)	3.65	2.27	0
(b, v)	1.30	2.27	0
(b, f)	1.96	2.27	0
(f, v)	2.56	2.27	0

Learning procedure

- Find the set of constraint weights that maximizes this function:

$$[\sum_{j=1}^{n} \log \Pr(y_j \mid x_j)] - [\sum_{i=1}^{m} \frac{(w_i - \mu_i)^2}{2\sigma_i^2}]$$

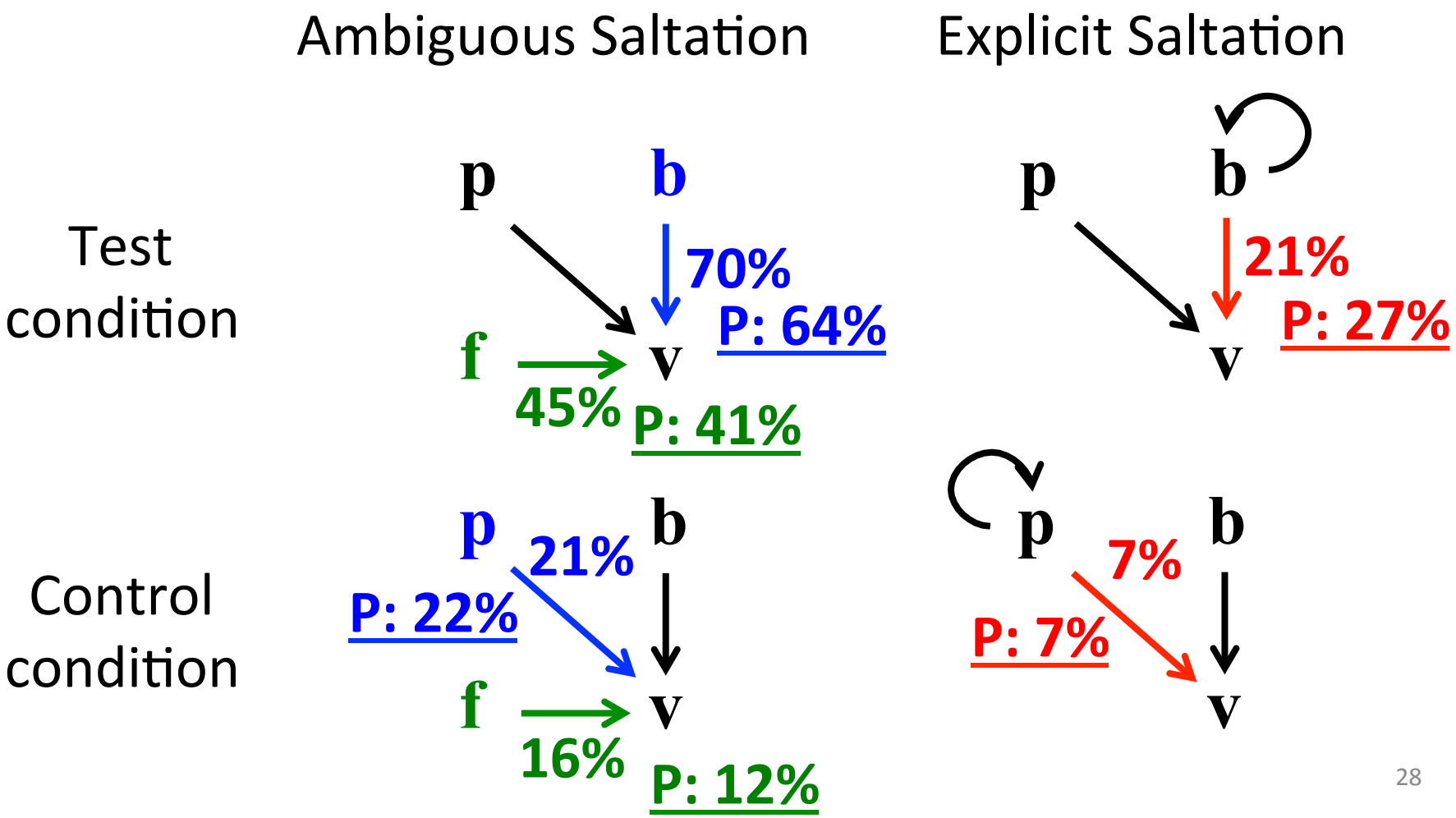
Maximize the likelihood of the training data

with a penalty for weights that vary from the prior

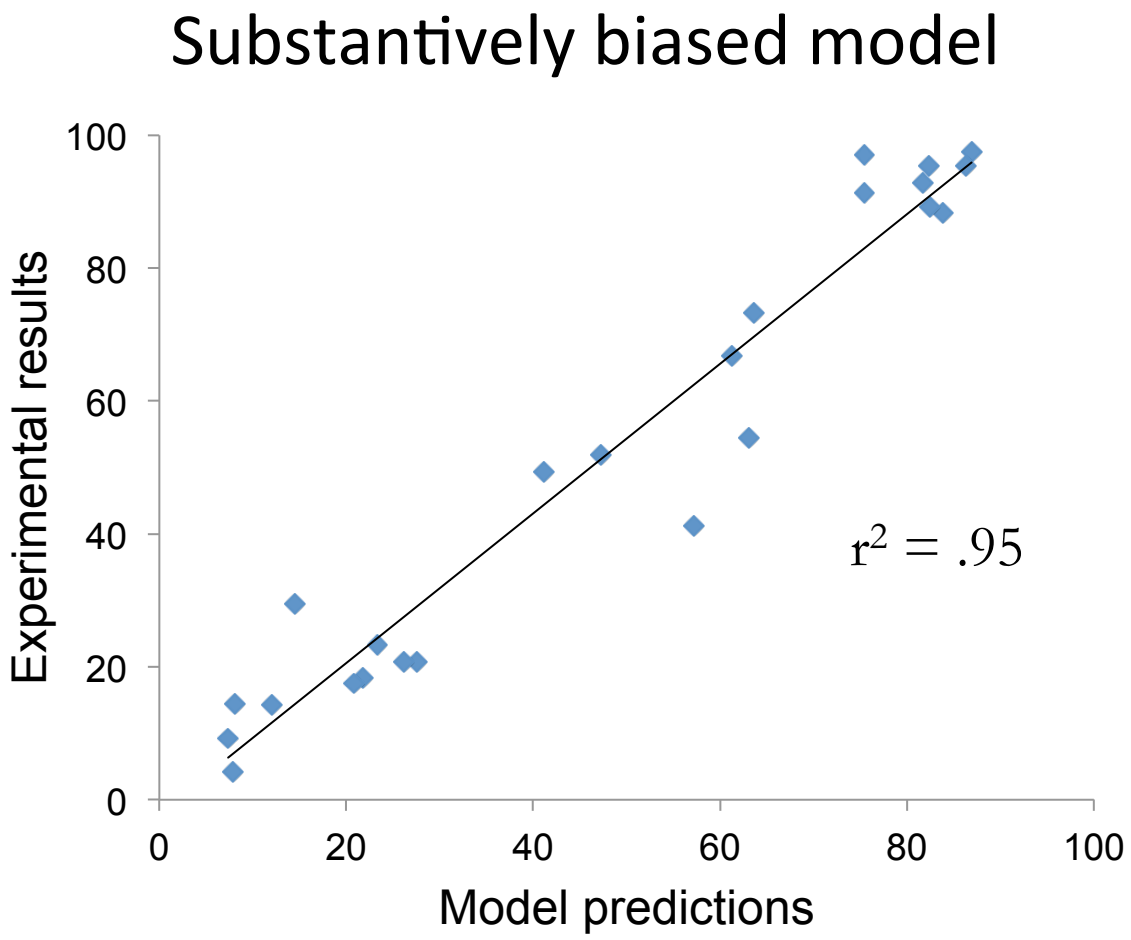
- Model will provably always succeed at finding the "best" grammar by this criterion.[1]

1. Berger et al., 1996

Returning to the crucial experimental results



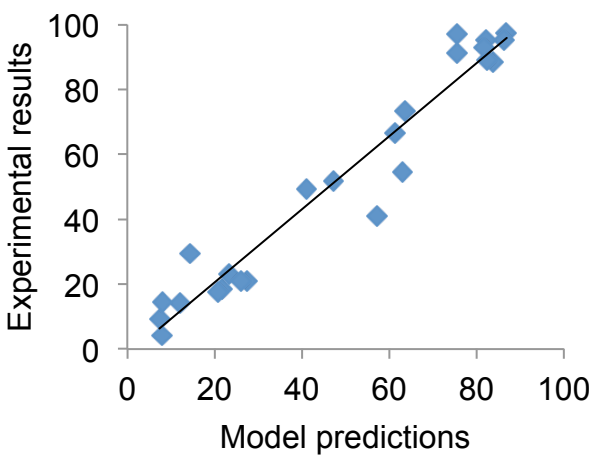
Model performance



Recall: No manipulation of prior weights by hand!

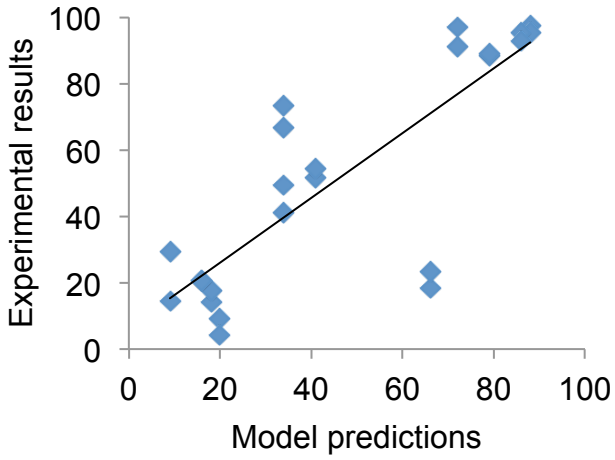
Compared to other models

Substantively biased



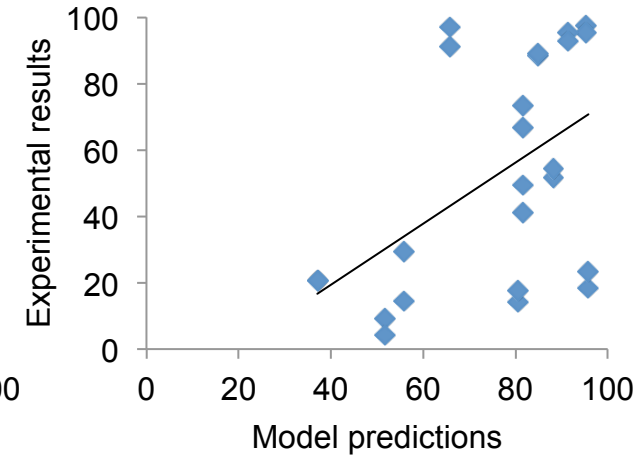
$r^2 = .95$

Anti-alternation



$r^2 = .67$

Unbiased



$r^2 = .25$

Where do the control models go wrong?

- Unbiased model:
  - In general, predicts too much generalization – no reason to avoid positing new alternations.

- Anti-alternation model:
  - Unable to account for subtler variations between segments (e.g., [b] changed to [v] more often than [f]).
  - Ends up predicting more generalization in the control case than in the saltation case – opposite of the actual results!!

Discussion

- Anti-alternation model outperforms Unbiased model.
  - Consistent with a general preference for avoiding alternations (e.g., OO-Faith set high by default).[1, 2]
- Substantively biased model outperforms Anti-alternation model.
  - Evidence that perceptual similarity plays a role even beyond a general preference to avoid alternations.
- General framework for looking at the role of perceptual similarity in phonological learning.

1. Hayes, 2000 2. Tessier, 2012

Prior as a soft bias

- The prior is crucial to the model's success. It allows the desired learning pattern:
  - Alternations between dissimilar sounds are initially dispreferred.
  - But with enough training data, the prior can be overcome – they are learnable.

- The anti-saltation effect seen in the experiments seems to fall out from a more general similarity bias.

Evidence with infant learners?

- 12-month-olds learning potentially saltatory alternation generalize to intermediate sounds.[1]

- 12-month-old English-learning infants know [d ~ ɾ], but not [t ~ ɾ], despite greater support for [t ~ ɾ] in their input.[2]

1. White & Sundara, 2012, submitted
2. Sundara et al., 2013 (BUCLD)

Thank you!

- Acknowledgments:
  - For help and discussion, special thanks to Bruce Hayes, Megha Sundara, Kie Zuraw, Robert Daland, Sharon Peperkamp, Adam Albright, and audiences at UCLA, the University of Ottawa, and Stony Brook University.
  - Thanks to my undergraduate research assistants as well as research assistants at the UCLA Langauge Acquisition Lab.

Creating the prior (details)

- Input = identification data from confusion experiments.[1]

  E.g.,

Stimulus	Responses				Stimulus	Responses			
	p	b	f	v		t	d	θ	ð
p	1844	54	159	26	t	1765	107	92	26
b	206	1331	241	408	d	91	1640	75	193
f	601	161	1202	93	θ	267	118	712	135
v	51	386	127	1428	ð	44	371	125	680

- Each *MAP constraint is weighted according to how likely its sounds are to be confused for each other.
- Output weights → Prior of main model

1. Wang & Bilger 1973

Effect of different confusion data

Source	Table #	Context	In noise?	r^2
WB 1973	2-3	CV and VC	white noise	.95
WB 1973	2	CV	white noise	.93
WB 1973	3	VC	white noise	.92
WB 1973	6-7	CV and VC	none	.93
WB 1973	6	CV	none	.82
WB 1973	7	VC	none	.96
MN 1955	2-6	CV	white noise	.94
C-etal 2004	----	CV and VC	babbled noise	.82
C-etal 2004	----	CV	babbled noise	.79
C-etal 2004	----	VC	babbled noise	.77
Unbiased model (for comparison)				.25

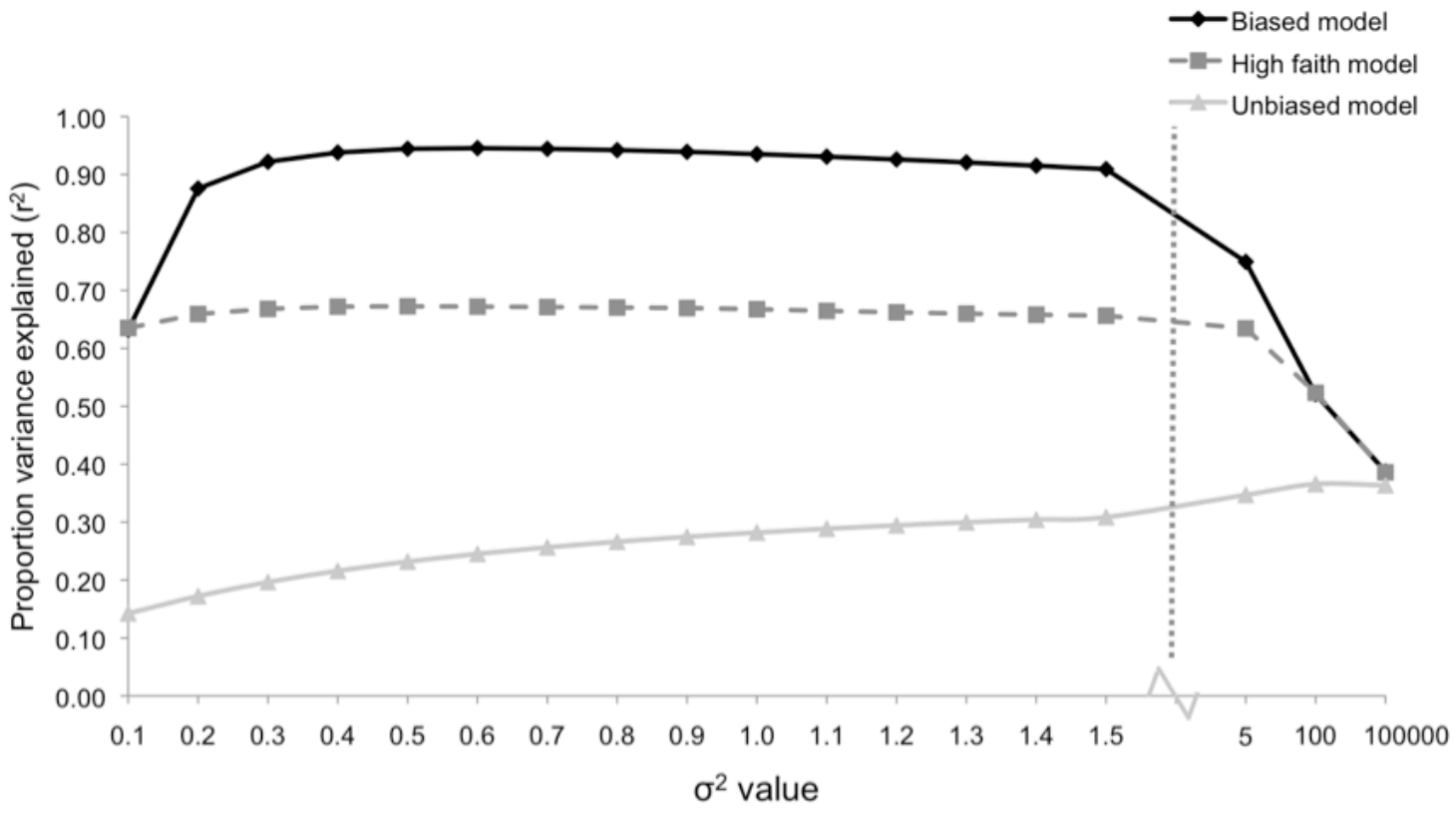
How the weights change during learning

Explicit Saltation experiment

p → v , b → b

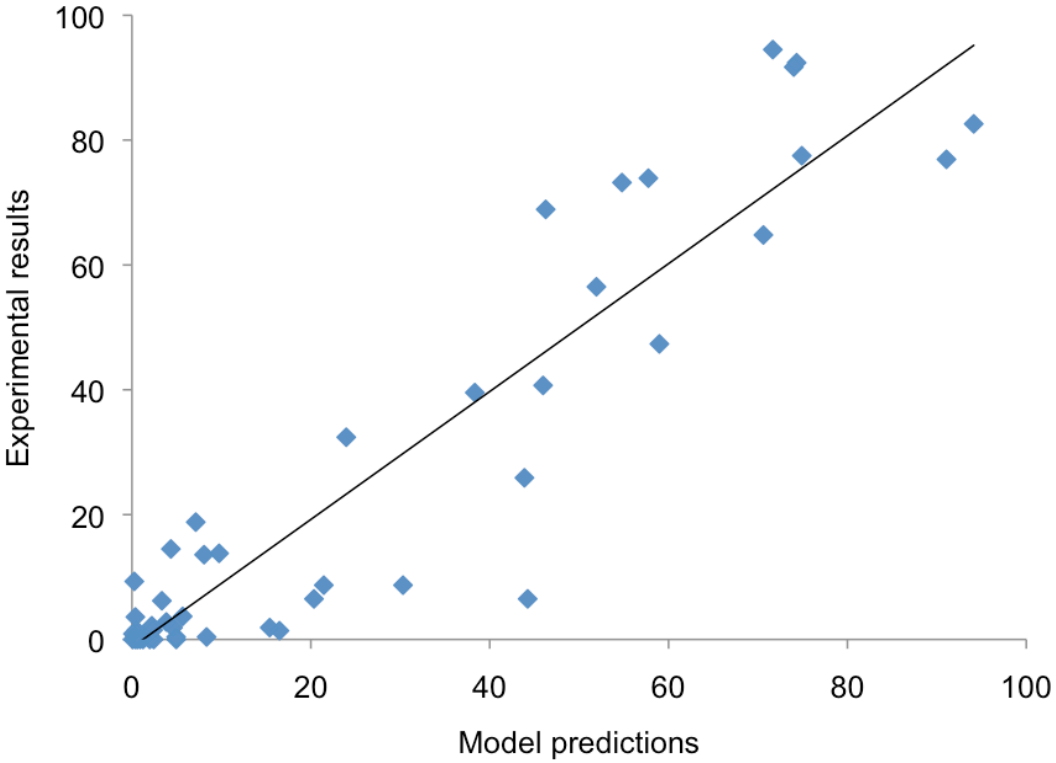
Constraints	Prior weight	Post-learning weight
*V [-voice] V	0	2.45
*V [-cont] V	0	1.05
*MAP(p, f)	1.34	1.74
(p, b)	2.44	2.94
(p, v)	**3.65**	**1.96**
(b, v)	**1.30**	**2.02**
(b, f)	1.96	2.02
(f, v)	2.56	2.56

Effect of σ


Biased model
High faith model
Unbiased model
Proportion variance explained (r^2)
1.00
0.90
0.80
0.70
0.60
0.50
0.40
0.30
0.20
0.10
0.00
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1
1.2
1.3
1.4
1.5
5
100
100000
σ^2 value

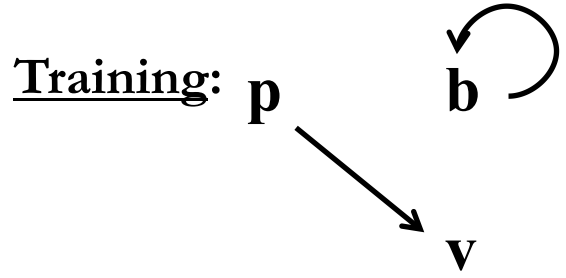
Production study

Figure 12. Predictions of the substantively biased model plotted against the experimental results from the production experiment. Overall $r^2 = .87$.



Predicting probabilities from the grammar

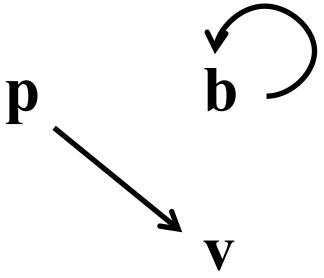
Explicit Saltation experiment



/kama**p**/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
kama**v**i			*			
kama**p**i	*			*		

Predicting probabilities from the grammar

Explicit Saltation experiment

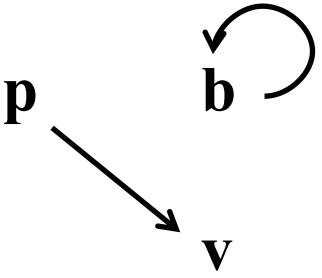


/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
kamavi			*			
kamapi	*			*		

Input form and output candidates

Predicting probabilities from the grammar

Explicit Saltation experiment

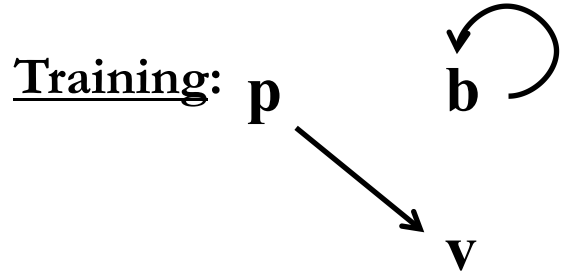


Constraint weights

/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
kamavi			*			
kamapi	*			*		

Predicting probabilities from the grammar

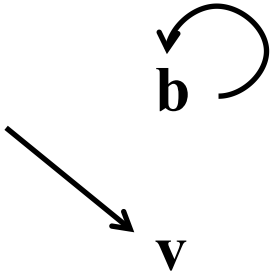
Explicit Saltation experiment



/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
kamavi			1.96			
kamapi	2.45			1.05		

Predicting probabilities from the grammar

Explicit Saltation experiment



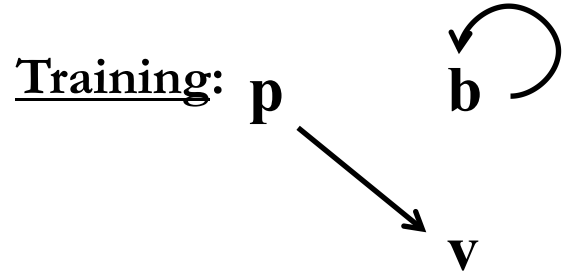
/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	**Total penalty**	Predicted output
kamavi			1.96		**1.96**	
kamapi	2.45			1.05	**3.50**	

Sum

$$\frac{e^{(-\text{penalty})}}{\sum e^{(-\text{penalty})}}$$

Predicting probabilities from the grammar

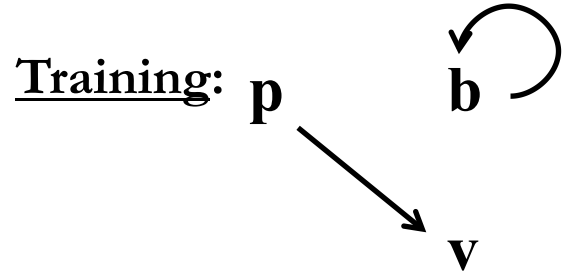
Explicit Saltation experiment



/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
kamavi			1.96		1.96	82 %
kamapi	2.45			1.05	3.50	18 %

Predicting probabilities from the grammar

Explicit Saltation experiment

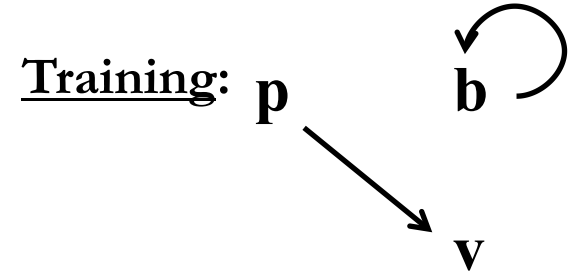


/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	**Predicted output**
kamavi			1.96		1.96	**82 %**
kamapi	2.45			1.05	3.50	**18 %**

/lunub/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
lunuvi		2.02			2.02	27 %
lunubi				1.05	1.05	73 %

Predicting probabilities from the grammar

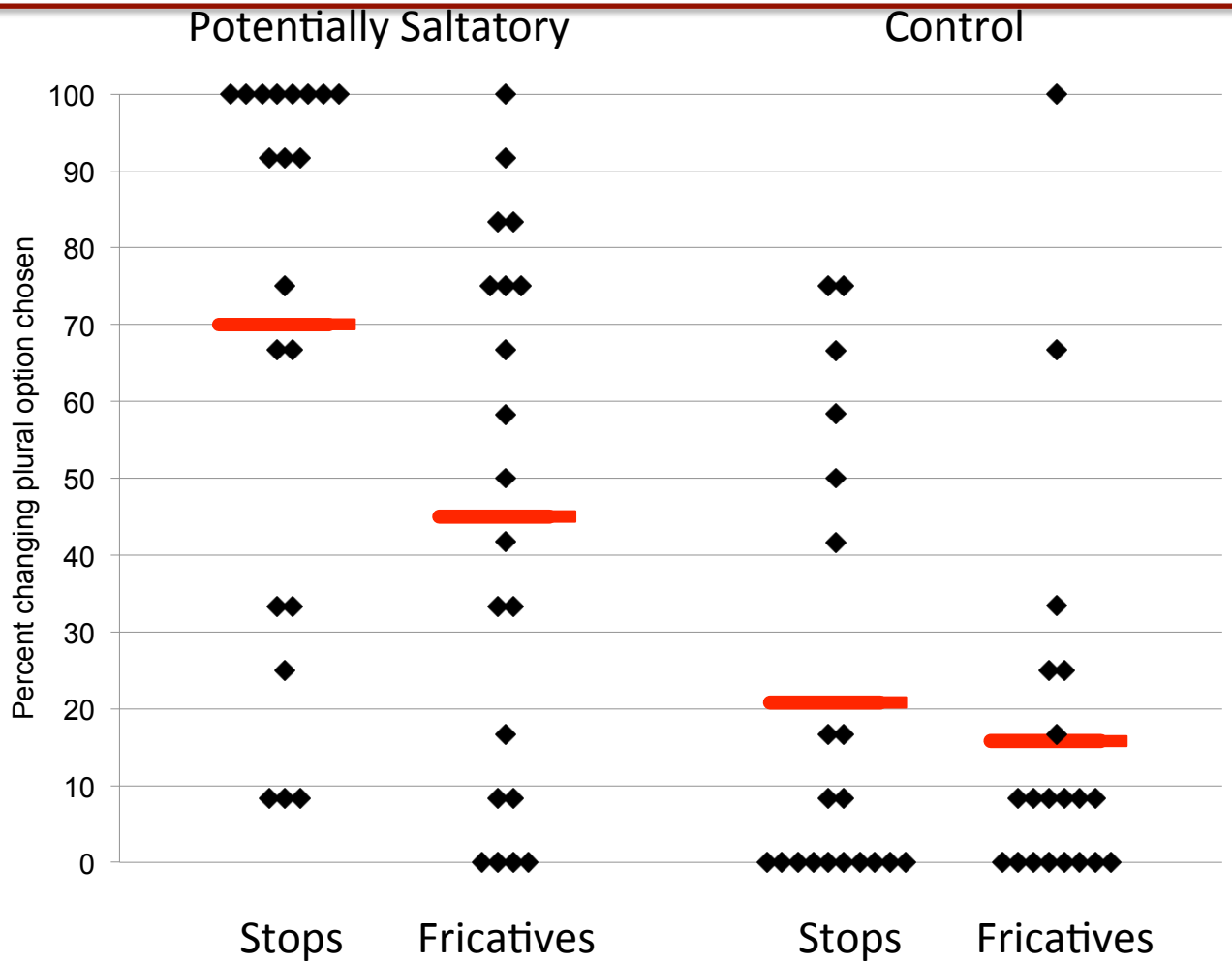
Explicit Saltation experiment



/kamap/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
kamavi			1.96		1.96	82 %
kamapi	2.45			1.05	3.50	18 %

/lunub/	*V[-voice]V 2.45	*MAP(b, v) 2.02	*MAP(p, v) 1.96	*V[-cont]V 1.05	Total penalty	Predicted output
lunuvi		2.02			2.02	27 %
lunubi				1.05	1.05	73 %

Exp. 1 Results



Exp. 2 Results