Estimating Feature-based Distributions

Jeffrey Heinz and Cesar Koirala
heinz,koirala@udel.edu

University of Delaware

The 11th meeting of SIGMORPHON
co-located with the ACL
Uppsala, Sweden
July 15, 2010

The tension

- Phonologists want to state rules and constraints with *phonological features*
- But computational linguistics, which has a foundation in formal language theory, often has models with alphabets of *unique, wholly distinct symbols*

Challenge: Develop provably correct phonotactic learning algorithms which are based on phonological features and not wholly distinct symbols (cf. Hayes and Wilson 2008, Albright 2009)

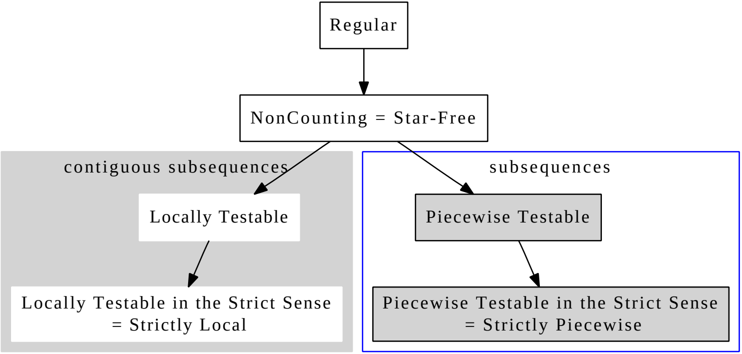
Technical results in this talk

1. Define feature-based Strictly k-Local distributions (n-gram models) and feature-based Stricly k-Piecewise distributions
2. The simplest model assumes there are *no* feature interactions!
3. Prove they are well-formed probability distributions with few parameters
4. Show how standard methods provably return the Maximum Likelihood (ML) estimate of a sample with respect to these families of distributions
5. The techniques draw upon *factoring distributions* (cf. Ghahramani and Jordan 1997, Saul and Jordan 1999, Dreyer et al. 2008, Dreyer and Eisner 2009, Heinz and Rogers 2010)

Conceptual goals of this talk

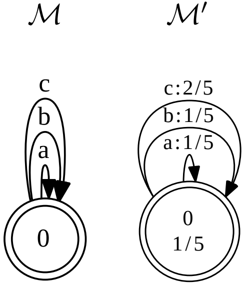
1. The lack of featural interaction captures exactly the intuition that symbols with like features ought to appear in like contexts.
2. The problem of feature-based learning is partially the problem of learning which features interact!
3. Show that how we factor the problem allows us to build in as much or as little feature interaction we think is neccessary.

Background: Subregular hierarchies


Regular
NonCounting = Star-Free
contiguous subsequences
Locally Testable
Locally Testable in the Strict Sense = Strictly Local
subsequences
Piecewise Testable
Piecewise Testable in the Strict Sense = Strictly Piecewise

(McNaughton and Papert 1971, Simon 1975, Rogers and Pullum 2007, Rogers et. al 2009, Heinz and Rogers 2010)

Background: ML Estimation of Subregular Distributions (structure is known, deterministic)



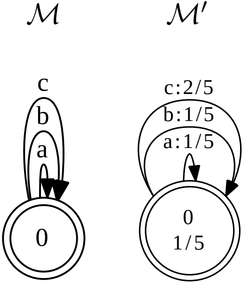
$\mathcal{M}$ represents a family of distributions with 4 parameters. $\mathcal{M}'$ represents a particular distribution in this family.

Theorem

For a sample S and deterministic finite-state acceptor $\mathcal{M}$, ***counting the parse of S through $\mathcal{M}$ and normalizing at each state*** *optimizes the maximum-likelihood estimate.*

(Geman and Johnson 2001, Vidal et. al 2005a, 2005b, de la Higuera 2010)

Background: ML Estimation of Subregular Distributions (structure is known, deterministic)



$\mathcal{M}$ represents a family of distributions with 4 parameters. $\mathcal{M}'$ represents a particular distribution in this family.

↓

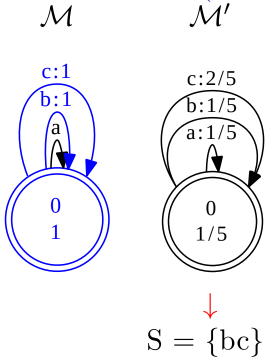
S = {bc}

Theorem

For a sample S and deterministic finite-state acceptor $\mathcal{M}$, ***counting the parse of S through $\mathcal{M}$ and normalizing at each state*** *optimizes the maximum-likelihood estimate.*

(Geman and Johnson 2001, Vidal et. al 2005a, 2005b, de la Higuera 2010)

Background: ML Estimation of Subregular Distributions (structure is known, deterministic)



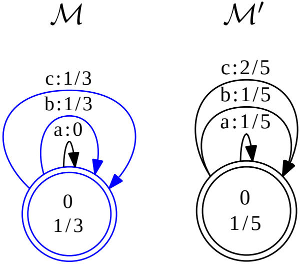
$\mathcal{M}$ represents a family of distributions with 4 parameters. $\mathcal{M}'$ represents a particular distribution in this family.

Theorem

For a sample S and deterministic finite-state acceptor $\mathcal{M}$, ***counting the parse of S through $\mathcal{M}$ and normalizing at each state*** *optimizes the maximum-likelihood estimate.*

(Geman and Johnson 2001, Vidal et. al 2005a, 2005b, de la Higuera 2010)

Background: ML Estimation of Subregular Distributions (structure is known, deterministic)



$\mathcal{M}$ represents a family of distributions with 4 parameters. $\mathcal{M}'$ represents a particular distribution in this family.

$\downarrow$

$S = \{bc\}$

Theorem

*For a sample S and deterministic finite-state acceptor $\mathcal{M}$, **counting the parse of S through $\mathcal{M}$ and normalizing at each state** optimizes the maximum-likelihood estimate.*

(Geman and Johnson 2001, Vidal et. al 2005a, 2005b, de la Higuera 2010)

Background: Feature Systems

	F	G
a	+	-
b	+	+
c	-	+

Table: An example of a feature system with $\Sigma = \{a, b, c\}$ and two features F and G.

Features

$$F : \Sigma \to \mathbb{V}_F$$

Feature Systems

$$\mathbb{F} = \langle F_1, \ldots, F_n \rangle$$

Background: Feature Notation

	F	G
a	+	-
b	+	+
c	-	+

Example 1: $G^{-1}(+) = \{b, c\}$.

Example 2: $\mathbb{F}^{-1}(\langle -F, +G \rangle) = \{c\}$.

Inverse feature functions Let F^{-1} be the inverse of F with domain V_F and codomain $\mathcal{P}(\Sigma)$.

Exhaustive feature systems are those for which, for all arguments $\vec{v}$, $\mathbb{F}^{-1}(\vec{v})$ is nonempty.

Distinctive feature systems are those for which, for all arguments $\vec{v}$, if $\mathbb{F}^{-1}(\vec{v})$ is nonempty then it is the case that $|\mathbb{F}^{-1}(\vec{v})| = 1$.

The features system above is distinctive, but not exhaustive.

Bigram models

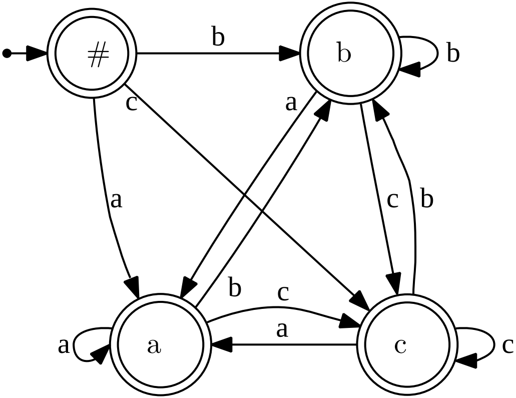
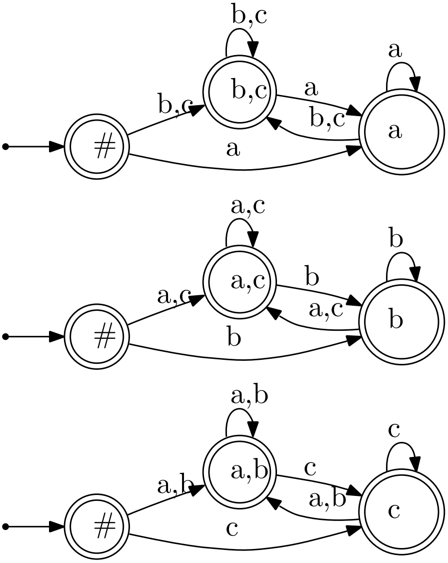


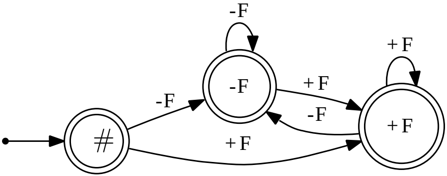
Figure: $\mathcal{M}_{\mathrm{SL2}}(\{a, b, c\})$ represents the structure of SL_2 distributions when $\Sigma = \{a, b, c\}$.

Factored Bigram models

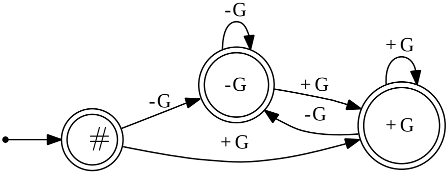


- The product of these machines yields the bigram machine on the previous slide.

Feature-based factors



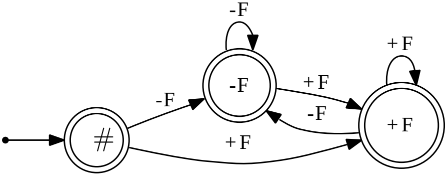
$\mathcal{M}_F$ represents a SL_2 distribution with respect to feature F.



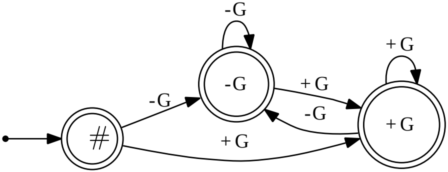
$\mathcal{M}_G$ represents a SL_2 distribution with respect to feature G.

1. Each machine encodes the probability of a feature value occuring given the the previous feature value (SL_2)
2. The alphabets are not the same so standard product will be the empty acceptor!

Feature-based factors



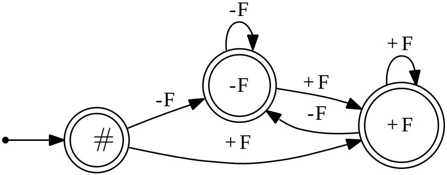
$\mathcal{M}_F$ represents a SL_2 distribution with respect to feature F.



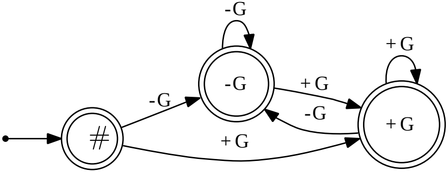
$\mathcal{M}_G$ represents a SL_2 distribution with respect to feature G.

1. Each machine encodes the probability of a feature value occuring given the the previous feature value (SL_2)
2. The alphabets are not the same so standard product will be the empty acceptor!

Feature-based factors



$\mathcal{M}_F$ represents a SL_2 distribution with respect to feature F.



$\mathcal{M}_G$ represents a SL_2 distribution with respect to feature G.

1. Each machine encodes the probability of a feature value occuring given the the previous feature value (SL_2)
2. The alphabets are not the same so standard product will be the empty acceptor!

The extended product of feature-based factors

Definition (Structure)

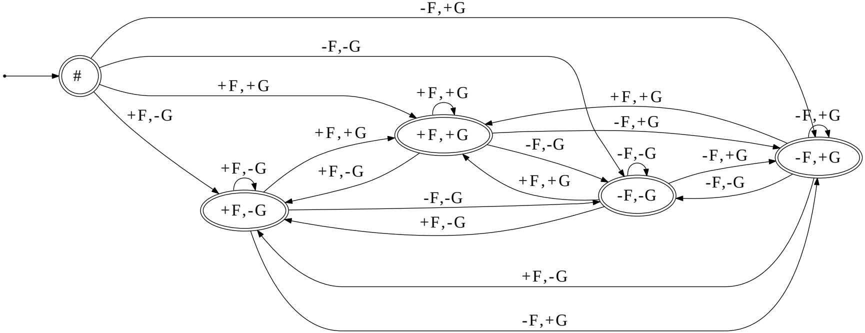
Let $\mathcal{M}_1 = \langle Q_1, \Sigma_1, q_{01}, \delta_1, F_1, T_1 \rangle$
and $\mathcal{M}_2 = \langle Q_2, \Sigma_2, q_{02}, \delta_2, F_2, T_2 \rangle$.

Then the *extended product* is
$\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2 = \langle Q, \Sigma, q_0, \delta, F, T \rangle$ where

1. Q and q_0 are defined in terms of the standard DFA product over the state space $Q_1 \times Q_2$ (Hopcroft et al. 2001).
2. $\Sigma = \Sigma_1 \times \Sigma_2$
3. For all $\langle q_1, q_2 \rangle \in Q$ and $\langle \sigma_1, \sigma_2 \rangle \in \Sigma$,
   $\delta(\langle q_1, q_2 \rangle, \langle \sigma_1, \sigma_2 \rangle) = \langle q_1', q_2' \rangle$ iff $\delta_1(q_1, \sigma_1) = q_1'$ and
   $\delta_2(q_2, \sigma_2) = q_2'$.

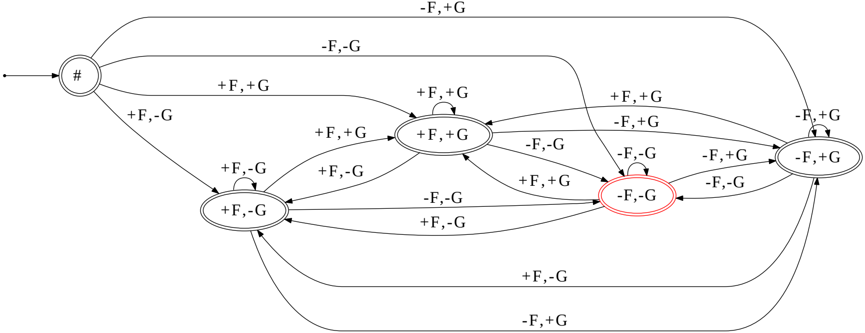
The extended product of $\mathcal{M}_F$ and $\mathcal{M}_G$

	F	G
a	+	-
b	+	+
c	-	+



The extended product of $\mathcal{M}_F$ and $\mathcal{M}_G$

	F	G
a	+	-
b	+	+
c	-	+
d	-	-



Extended normalized co-emission product

Let $\mathcal{M}_1 = \langle Q_1, \Sigma_1, q_{01}, \delta_1, F_1, T_1 \rangle$
and $\mathcal{M}_2 = \langle Q_2, \Sigma_2, q_{02}, \delta_2, F_2, T_2 \rangle$.

Let $CT(\sigma_1, \sigma_2, q_1, q_2) = T_1(q_1, \sigma_1) \cdot T_2(q_2, \sigma_2)$
and $CF(q_1, q_2) = F_1(q_1) \cdot F_2(q_2)$.

Definition (Probabilities)

The *extended normalized co-emission product* is the extended product $\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2 = \langle Q, \Sigma, q_0, \delta, F, T \rangle$. For all $\langle q_1, q_2 \rangle \in Q$,

1. Let $Z(\langle q_1, q_2 \rangle) = CF(q_1, q_2) \; + \; \sum_{\langle \sigma_1, \sigma_2 \rangle \in \Sigma} CT(\sigma_1, \sigma_2, q_1, q_2)$
2. $F(\langle q_1, q_2 \rangle) = \frac{CF(q_1, q_2)}{Z(\langle q_1, q_2 \rangle)}$; and
3. for all $\langle \sigma_1, \sigma_2 \rangle \in \Sigma$, $T(\langle q_1, q_2 \rangle, \langle \sigma_1, \sigma_2 \rangle) = \frac{CT(\sigma_1, \sigma_2, q_1, q_2)}{Z(\langle q_1, q_2 \rangle)}$

Feature-based SL$_k$ distributions

$$\mathbb{F} = \langle F_1, \ldots, F_m \rangle$$

$$w = \sigma_1 \ldots \sigma_n$$

Probability of words

$$Pr(w) = Pr(\sigma_1 \mid \#) \cdot \ldots \cdot Pr(\sigma_n \mid \sigma_{n-1}) \cdot Pr(\# \mid \sigma_n) \qquad (1)$$

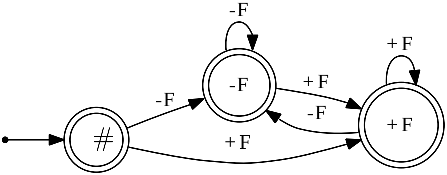
Featural decomposition

$$Pr(\sigma \mid \tau) = \frac{\prod_{1 \leq i \leq m} Pr(F_i(\sigma) \mid F_i(\tau))}{Z(\tau)} \qquad (2)$$

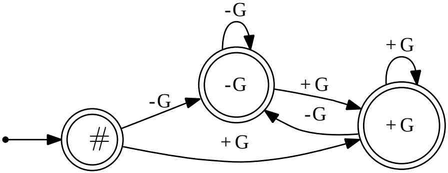
Theorem (Well-formed probability distribution)

Equations 1 and 2 define a well-formed probability distribution over Σ^*.

Summary of feature-based distributions



$\mathcal{M}_F$ represents a SL$_2$ distribution with respect to feature F.



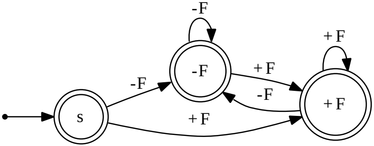
$\mathcal{M}_G$ represents a SL$_2$ distribution with respect to feature G.

1. Parameters:
   $Pr([+F] \mid [+F])$
   $Pr([+F] \mid [-F])$
   $\dots$
2. For n binary features, there are $8n + 1$ parameters.
3. Feature independence.
   Example: $Pr(a|b) =$

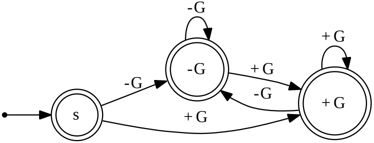
$$Pr([+F, -G] \mid [+F, +G]) =$$

$$\frac{Pr(+F \mid +F) \cdot Pr(-G \mid +G)}{Z}$$

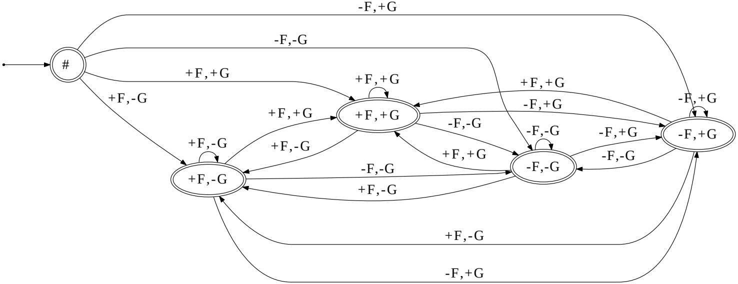
Estimating feature-based distributions



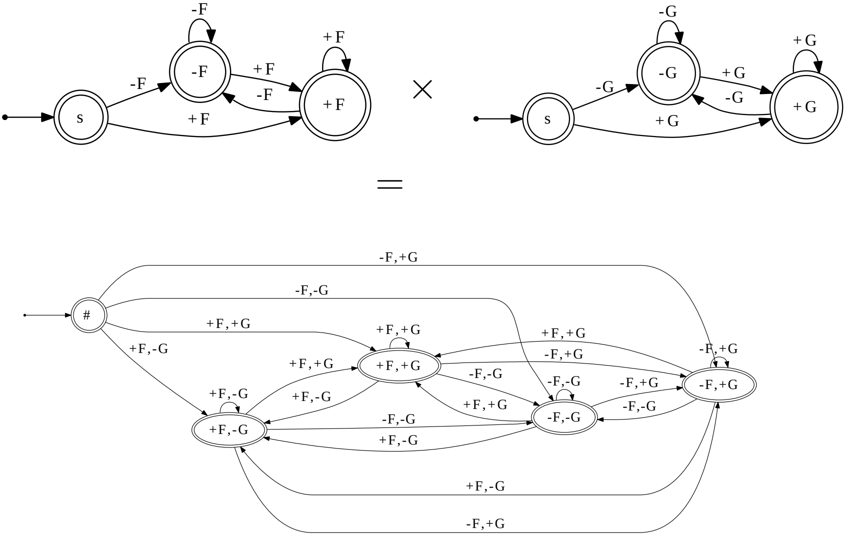
×



=



Estimating feature-based distributions



Estimate the factors, not the product!

Estimating feature-based distributions

Theorem

Let $\mathcal{M} = \mathcal{M}_1 \times \ldots \times \mathcal{M}_n$. The ML estimate of a sample S with respect to the feature-based distribution $\mathcal{M}$ is obtained by obtaining the ML estimation of S with respect to each $\mathcal{M}_i$.

A simple example

	F	G
a	+	-
b	+	+
c	-	+
d	-	-

$$S = \{aaab,\ caca,\ acab,\ cbb\}$$

$P(\sigma \mid \tau)$		σ					σ				
		a	b	c	d	#	a	b	c	d	#
τ	a	0.29	0.29	0.29	0.	0.14	0.22	0.43	0.17	0.09	0.09
	b	0.	0.25	0.	0.	0.75	0.32	0.21	0.09	0.13	0.26
	c	0.75	0.25	0.	0.	0.	0.60	0.40	0.	0	0.
	d	0.	0.	0.	0.	0.	0.33	0.67	0	0	0
	#	0.5	0.	0.5	0.	0.	0.25	0.25	0.25	0.25	0.
		Segment-based ML estimation					Feature-based ML estimation				

Matching acceptability judgements

Hayes and Wilson (2008) induce maxent grammar with possible constraint types: Strictly *k*-Local constraints stated with features and complements of features.

Hayes and Wilson maxent models	example constraints	r
features & complement classes	*[ˆ +F][-F,+G]	0.946
no features & complement classes	*[ˆ a][b]	0.937
features & no complement classes	*[+F][-F,+G]	0.914
no features & no complement classes	*[a][b]	0.885

Table: Correlations of different settings versions of HW maxent model with Scholes (1966) data. Models allowed constraints up to length 3 and were trained on adapted version of words in CMU dictionary.

Number of parameters and performance

Are the additional parameters that buy an additional gain in performance worth it?

- Wilson and Obdeyn (2009) provide an excellent discussion of the model comparison literature and provide a rigorous comparative analysis of computational modeling of OCP restrictions.
- The feature-based SL_2 distribution, when trained on words in CMU dictionary, only obtain $r = 0.751$ with Scholes' data, but with far fewer parameters.
- While unknown whether the additional parameters are worth it, we still learn something: only $\sim$ a quarter of the variation of the data in Scholes (1966) is due to feature interaction!

Feature interaction in phonology

1. Different features may be prohibited from occuring simultaneously in certain contexts.
   *English: attested #[+nasal] and #[+dorsal] but *#[+nasal,+dorsal]*
2. Specific languages may prohibit different features from simultaneously occuring in all contexts.
   **[+syllabic,-sonorant] (English)*
3. Different features may be universally incompatible
   **[+high,+low]*
4. Different features may be prohibited from occuring syntagmatically.
   **[+nasal][-voice]*

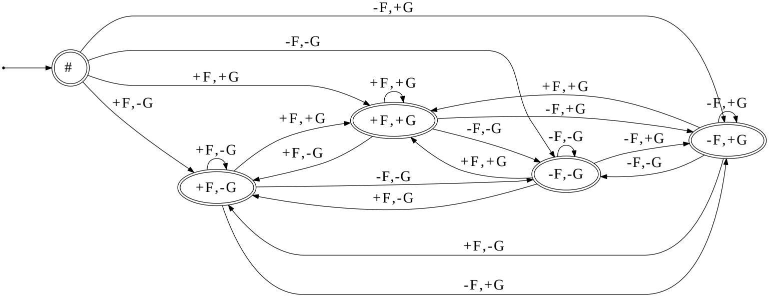
Independence assumption is too strong but still useful

1. Researchers can quantify the extent to which data can be explained without invoking featural interaction
2. In principle, researchers can detect when feature interaction ought to be invoked (because the expected number of occurences does not match the actual number)

Capturing feature-interaction with different factors

$$\mathbb{F} = \langle F, G, H, \ldots, M \rangle$$

Prior knowledge that F and G potentially interact suggests using the machine below as a factor in the model!



Full interaction of features yields the segment-based distribution.

Summary

1. We can fully integrate a feature-system into SL_k distributions (k-gram models) and SP_k ones too.
2. We proved we have well-formed distributions and proved that that ML estimation of the parameters is possible (by estimating the factors).
3. 10 binary features describe up to 1,024 symbols. Bigram models have $(1,024)^2 = 1,050,625$ parameters, but the feature-based bigram model has only 81.
4. It follows these distributions are less expressive, and can be estimated with less data (like sounds occur in like contexts).
5. Understanding how learners identify featural interactions is *the issue* in understanding generalizations on the basis of phonological features.