Learning grammar(s) statistically

Mark Johnson

Natural Language Group
Microsoft Research

Cognitive and Linguistic Sciences and Computer Science
Brown University

SFB 732 "Incremental Specification in Context"
Opening Colloquium
November 2006

joint work with Sharon Goldwater and Tom Griffiths

Statistics, grammar and acquisition

- Statistical methods have taken over (applied) computational linguistics
- Can they help us understand *language acquisition*?
  - Linguistic structures are compatible with statistics (Abney 1997)
  - Generative linguists are (or ought to be) Bayesians
  - Some simple statistical models don't work, and we can start to understand why
- Statistics and information theory may help us move from arm-chair philosophy to quantitative, empirical science

Outline

Statistical learning in computational linguistics

- Statistical learning exploits *distributional information* in the input
- Statistical learning is central to many practical applications
  - speech recognition
  - machine translation
  - search and related applications
- Statistical models can do (surprising?) linguistic things
  - build bi-lingual dictionaries
  - cluster words into broad lexical-semantic classes
  - find lexical properties, e.g., transitivity, verb-particle pairs
- But does it have anything deep to say about language?
  - What constitutes knowledge of language?
  - *How is it acquired?*
  - How is it put to use?

Humans are sensitive to statistics

- Lexical frequencies have huge impact on reaction times
- Infants can segment stream of nonsense syllables into "words" using statistical cues alone (Saffran 96)
  - But: statistical learning is much more than conditional probabilities!
- Order of acquisition is often determined by frequency (Lleó and Demuth 1999)

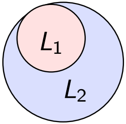
Statistics is compatible with linguistics

1. Colorless green ideas sleep furiously.
2. Furiously sleep ideas green colorless.

> *...It is fair to assume that neither sentence (1) nor (2) (nor indeed any part of these sentences) has ever occurred in an English discourse ... (Chomsky 1957)*

- A *class-based bigram model* predicts (1) is 2×10^5 more probable than (2) (Pereira 2000)
- We can define probability distributions over linguistically realistic structures
  - Maximum entropy models define probability distributions for *arbitrary grammars* (Abney 1997)
- *How do linguistic structures and constraints interact with distributional information in a statistical learner?*

Statistical learning and implicit negative evidence



- Logical approach to acquisition
  no negative evidence ⇒ *subset problem*
  guess L_2 when true lg is L_1
- statistical learning can use *implicit negative evidence*
  - if $L_2 - L_1$ is *expected* to occur but doesn't
    ⇒ L_2 is probably wrong
- Statistical learning can succeed where logical learning fails
  - Context-free grammars are not learnable from positive evidence alone, but *probabilistic* context-free grammars are
- because statistical learning models:
  - make stronger assumptions about input (follows distribution)
  - have weaker criteria for success (probabilistic convergence)

Probabilistic models and statistical learning

- Decompose learning problem into three components:
  1. class of *possible models*, i.e., (probabilistic) grammars and lexicons, from which learner chooses a model
  2. *objective function* (of model and input) that learning optimizes
     - e.g., *maximum likelihood*: find model that makes input as likely as possible
  3. *search algorithm* that finds optimal model(s) for input
- Using explicit probabilistic models lets us:
  - *combine models* for subtasks *in an optimal way*
  - better *understand* our learning models
  - diagnose problems with our learning models
    - distinguish *model errors* from *search errors*

Bayesian learning

$$\underbrace{\text{P(Hypothesis|Data)}}_{\text{Posterior}} \propto \underbrace{\text{P(Data|Hypothesis)}}_{\text{Likelihood}} \underbrace{\text{P(Hypothesis)}}_{\text{Prior}}$$

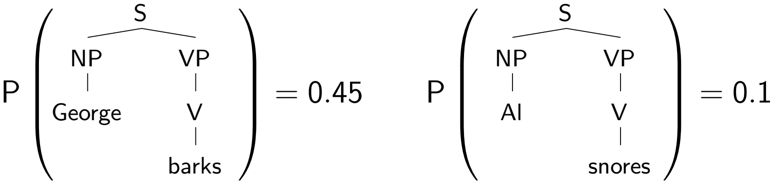
- Bayesian learning integrates information from *multiple information sources*
  - *Likelihood* reflects how well grammar fits input data
  - *Prior* encodes a priori preferences for particular grammars
- Priors can prefer
  - certain types of structures over others (informative priors)
  - smaller grammars over larger (Occam's razor, MDL)
- The *prior is as much a linguistic issue as the grammar*
  - Priors can be sensitive to linguistic structure (e.g., words should contain vowels)
  - Priors can encode *linguistic universals* and *markedness preferences*

Outline

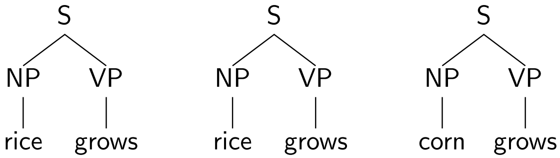
Probabilistic Context-Free Grammars

- The *probability* of a tree is the product of the probabilities of the rules used to construct it

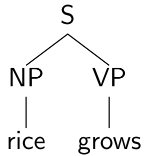
1.0	$S \longrightarrow NP\ VP$	1.0	$VP \longrightarrow V$
0.75	$NP \longrightarrow$ George	0.25	$NP \longrightarrow$ Al
0.6	$V \longrightarrow$ barks	0.4	$V \longrightarrow$ snores

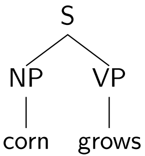


Learning PCFGs from trees (supervised)



Rule	**Count**	**Rel Freq**
S $\rightarrow$ NP VP	3	1
NP $\rightarrow$ rice	2	2/3
NP $\rightarrow$ corn	1	1/3
VP $\rightarrow$ grows	3	1

P () = 2/3

P () = 1/3

Rel freq is *maximum likelihood estimator* (selects rule probabilities that maximize probability of trees)

Learning from words alone (unsupervised)

- Training data consists of strings of words w
- Maximum likelihood estimator (grammar that makes w as likely as possible) no longer has closed form
- *Expectation maximization* is an iterative procedure for building unsupervised learners out of supervised learners
  - parse a bunch of sentences with current guess at grammar
  - weight each parse tree by its probability under current grammar
  - estimate grammar from these weighted parse trees as before
- Can incorporate Bayesian priors (e.g., prefer grammars whose rules have uniform head direction)

Dempster, Laird and Rubin (1977) "Maximum likelihood from incomplete data via the EM algorithm"

Expectation Maximization with a toy grammar

Initial rule probs

rule	**prob**
. . .	. . .
VP → V	0.2
VP → V NP	0.2
VP → NP V	0.2
VP → V NP NP	0.2
VP → NP NP V	0.2
. . .	. . .
Det → the	0.1
N → the	0.1
V → the	0.1

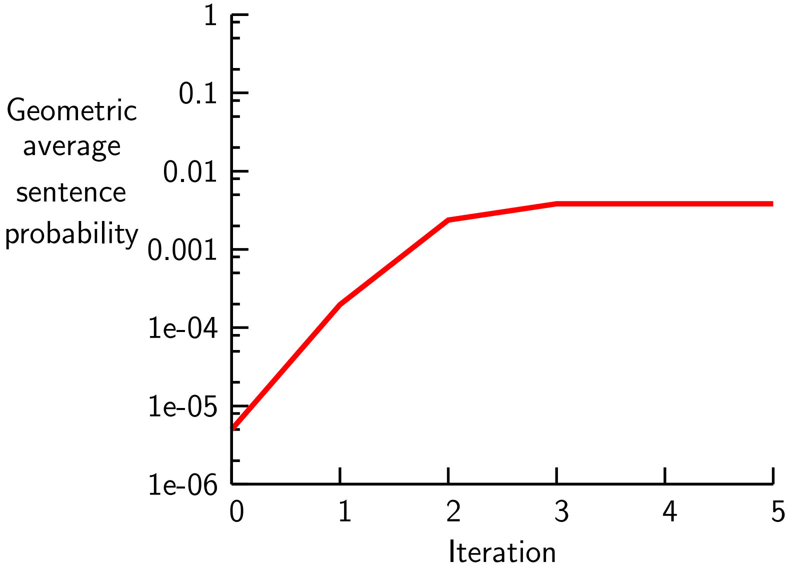
"English" input

the dog bites
the dog bites a man
a man gives the dog a bone
. . .

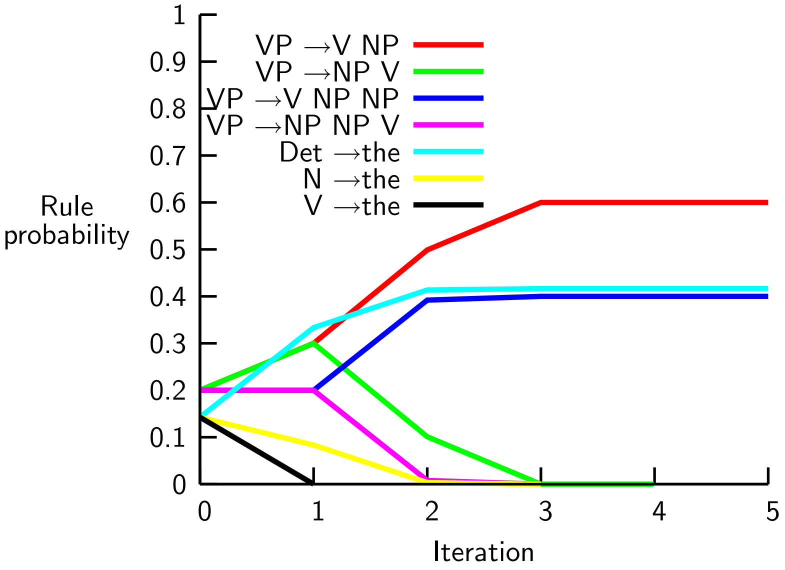
"pseudo-Japanese" input

the dog bites
the dog a man bites
a man the dog a bone gives
. . .

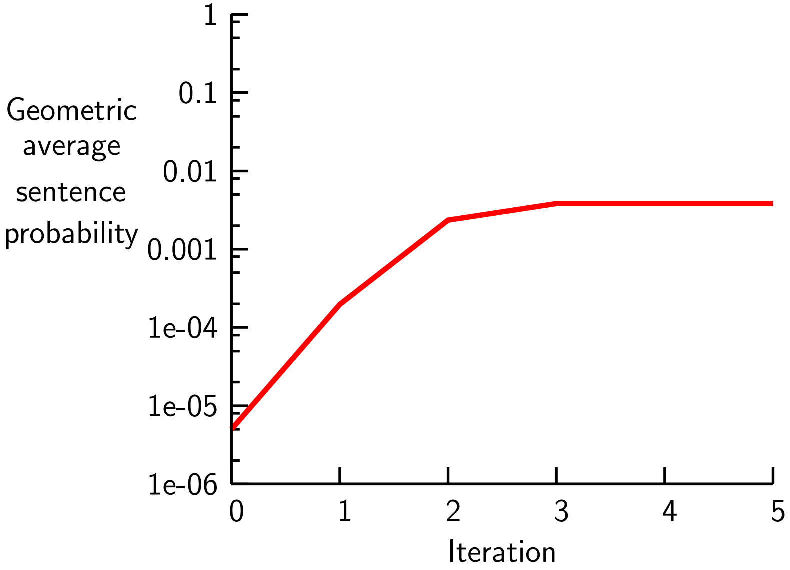
Probability of "English"



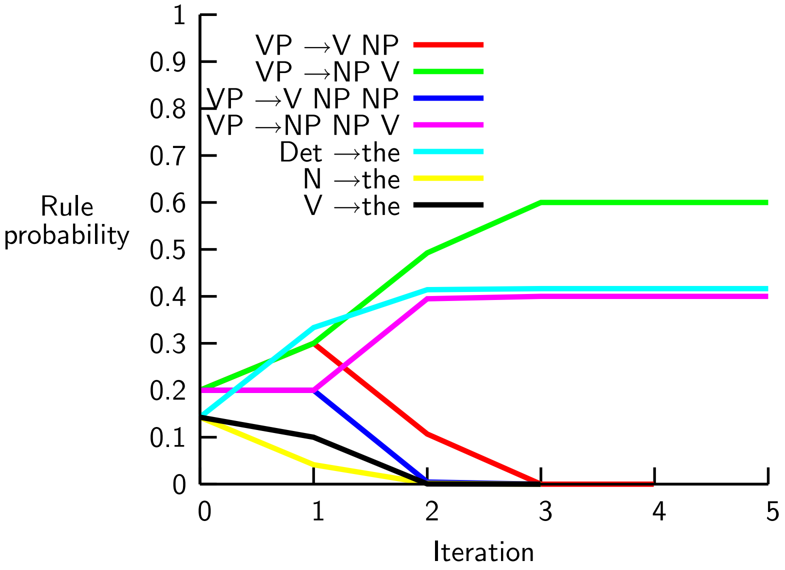
Rule probabilities from "English"



Probability of “Japanese”



Rule probabilities from "Japanese"

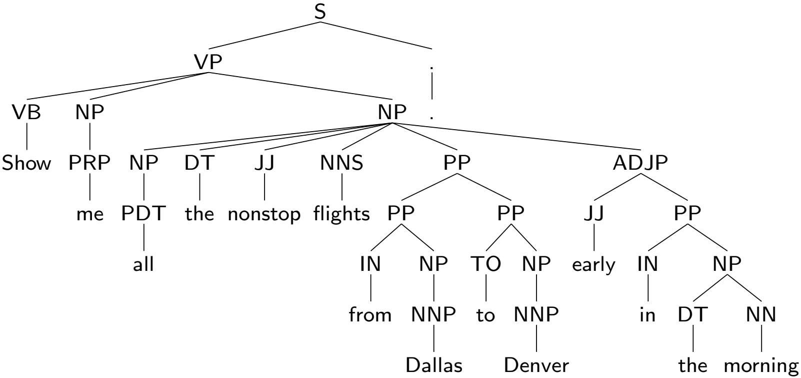


Statistical grammar learning

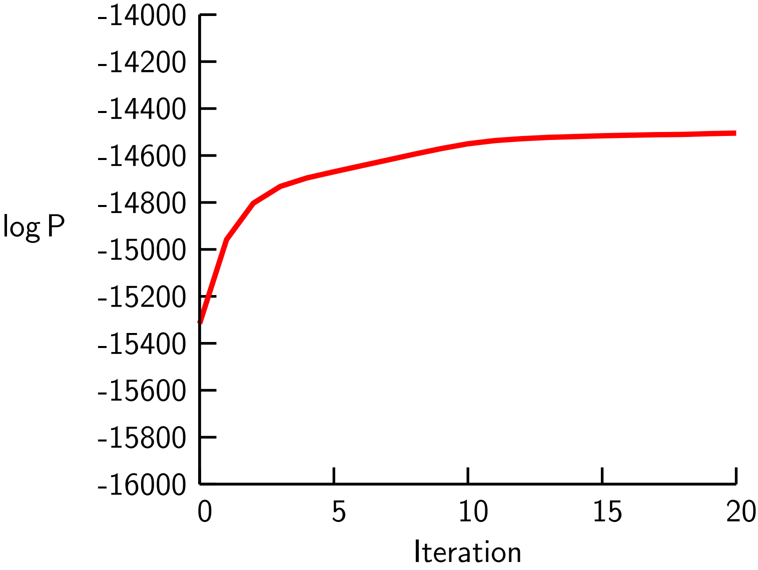
- Simple algorithm: learn from your best guesses
  - requires learner to parse the input
- "Glass box" models: learner's prior knowledge and learnt generalizations are *explicitly represented*
- Optimization of smooth function of rule weights ⇒ learning can involve small, incremental updates
- Learning structure (rules) is hard, but ...
- Parameter estimation can approximate rule learning
  - start with "superset" grammar
  - estimate rule probabilities
  - discard low probability rules

Learning from real data

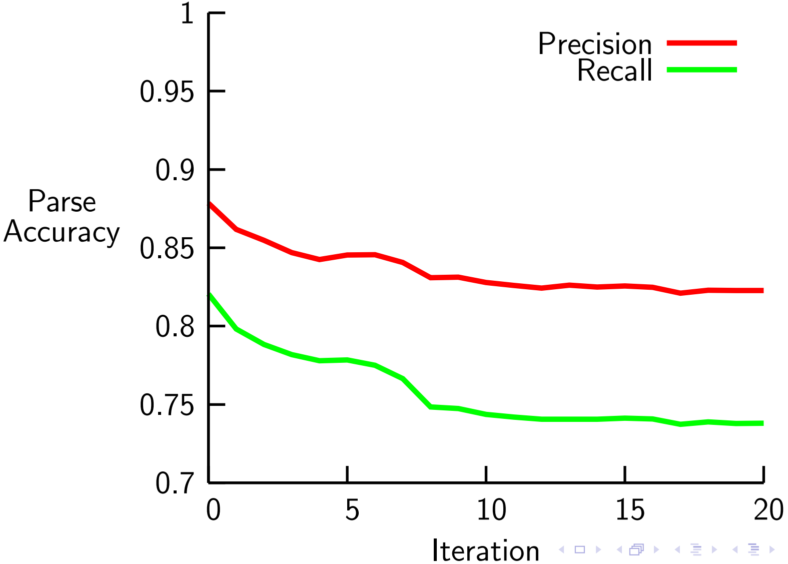
- ATIS treebank consists of 1,300 hand-constructed parse trees
- input consists of POS tags rather than words
- about 1,000 PCFG rules are needed to build these trees



Probability of training strings



Accuracy of parses produced using the learnt grammar



The PCFG model is wrong

- EM learner initialized with *correct parse trees* for sentences
  - given true rules and their probabilities
    $\Rightarrow$poor performance not due to search error
- Learner was evaluated on training data
  - poor performance not due to over-learning
- Parse accuracy drops as likelihood increases
  - higher likelihood $\not\Rightarrow$ better parses
  - *the statistical model is wrong*

Why doesn't a PCFG learner work on real data?

- higher likelihood $\not\Rightarrow$ parse accuracy
  $\Rightarrow$ *probabilistic model and/or objective function are wrong*
- Bayesian prior preferring smaller grammars doesn't help
- What could be wrong?
  - Wrong kind of grammar (Klein and Manning)
  - Wrong probabilistic model (Smith and Eisner)
  - Wrong training data (Yang)
  - Predicting word strings is wrong objective
  - Grammar *ignores semantics* (Zettlemoyer and Collins)

de Marken (1995) "Lexical heads, phrase structure and the induction of grammar"

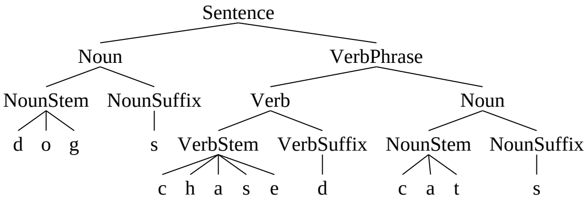
Outline

Research strategy

- Start with phonology, morphology and lexicon; leave syntax and semantics until later
  - children learn (some) words and inflections before they learn what they mean
  - child-directed speech corpora are readily available; contextual information is not
- Goal of this research (as yet unachieved):

  Input: "d o g s c h a s e d c a t s"
  (actually use unsegmented broad phonemic transcription)

  Output:

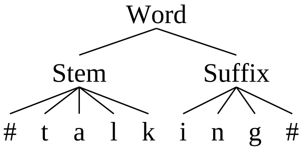


A grammar for concatenative morphology

- Too many things could be going wrong in learning syntax start with something simpler!
- Input data: regular verbs (in broad phonemic representation)
- Learning goal: segment verbs into stems and inflectional suffixes

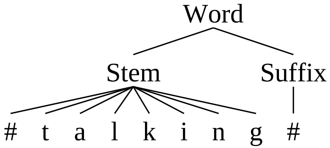
$$\begin{aligned} \text{Verb} &\rightarrow \text{Stem Suffix} & \\ \text{Stem} &\rightarrow w & w \in \Sigma^\star \\ \text{Suffix} &\rightarrow w & w \in \Sigma^\star \end{aligned}$$

Data = t a l k i n g



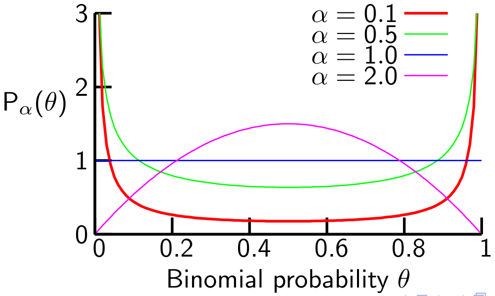
Maximum likelihood estimation won't work

- A *saturated model* has one parameter (i.e., rule) for each datum (word)
- The grammar that analyses *each word as a stem with a null suffix* is a saturated model
- Saturated models in general have highest likelihood
  - ⇒ saturated model *exactly replicates* (memorizes) training data
  - ⇒ doesn't "waste probability" on any other strings
  - ⇒ maximizes likelihood of training data



Bayesian priors for sparse grammars

- The saturated grammar has a rule for every word
- Factoring words into stems and suffixes should require *fewer rules*
- We can use Bayesian priors to prefer grammars with few rules
- We have developed MCMC algorithms for sampling from the posterior distribution of trees given strings with a *Dirichlet prior* on rule probabilities



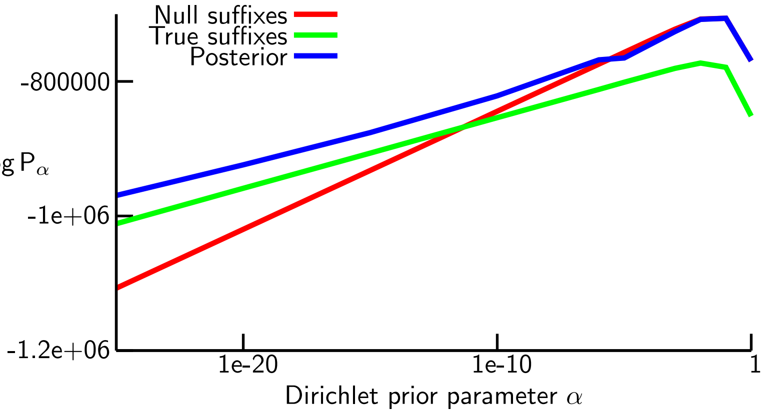
Morphological segmentation experiment

- Bayesian estimator with *Dirichlet prior* with parameter α
  - prefers sparser solutions (i.e., fewer stems and suffixes) as $\alpha \to 0$
- Component-wise Gibbs sampler samples from posterior distribution of parses
  - reanalyses each word based on parses of the other words
- Trained on orthographic verbs from U Penn. Wall Street Journal treebank
  - behaves similarly with broad phonemic child-directed input

Posterior samples from WSJ verb tokens

$\alpha = 0.1$	$\alpha = 10^{-5}$		$\alpha = 10^{-10}$		$\alpha = 10^{-15}$	
expect	expect		expect		expect	
expects	expects		expects		expects	
expected	expected		expected		expected	
expecting	expect	ing	expect	ing	expect	ing
include	include		include		include	
includes	includes		includ	es	includ	es
included	included		includ	ed	includ	ed
including	including		including		including	
add	add		add		add	
adds	adds		adds		add	s
added	added		add	ed	added	
adding	adding		add	ing	add	ing
continue	continue		continue		continue	
continues	continues		continue	s	continue	s
continued	continued		continu	ed	continu	ed
continuing	continuing		continu	ing	continu	ing
report	report		report		report	

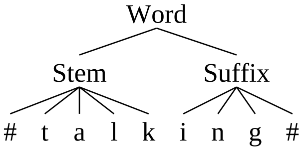
Log posterior of models on token data



- Correct solution is nowhere near as likely as posterior

⇒ no point trying to fix algorithm because *model is wrong!*

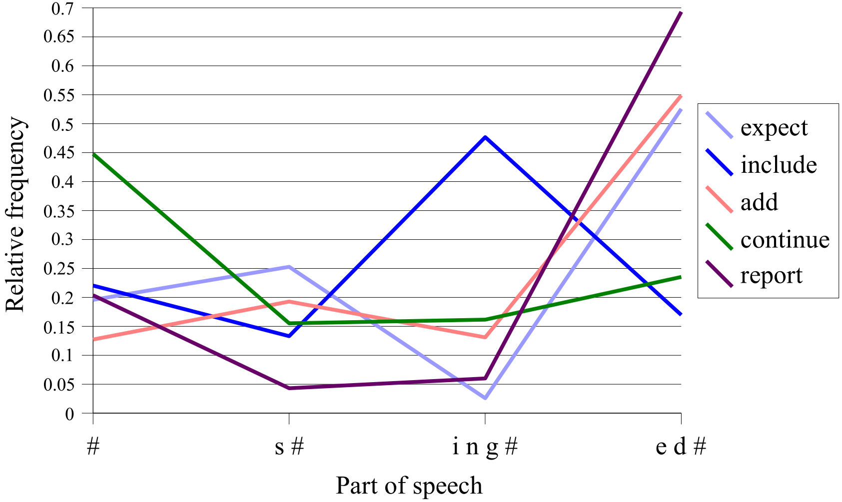
Independence assumptions in PCFG model



$$\text{P(Word)} = \text{P(Stem)P(Suffix)}$$

- Model expects relative frequency of each suffix *to be the same for all stems*

Relative frequencies of inflected verb forms



Types and tokens

- A word *type* is a distinct word shape
- A word *token* is an occurrence of a word

Data	=	"the cat chased the other cat"
Tokens	=	"the" 2, "cat" 2, "chased" 1, "other" 1
Types	=	"the" 1, "cat" 1, "chased" 1, "other" 1

- Using word types instead of word tokens effectively normalizes for frequency variations

Posterior samples from WSJ verb *types*

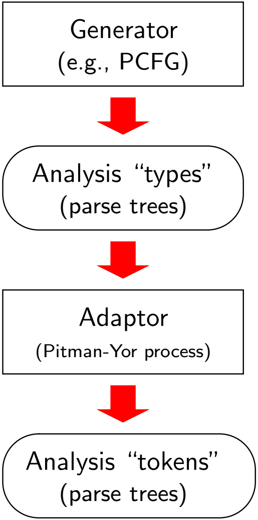
$\alpha = 0.1$		$\alpha = 10^{-5}$		$\alpha = 10^{-10}$		$\alpha = 10^{-15}$	
expect		expect		expect		exp	ect
expects		expect	s	expect	s	exp	ects
expected		expect	ed	expect	ed	exp	ected
expect	ing	expect	ing	expect	ing	exp	ecting
include		includ	e	includ	e	includ	e
include	s	includ	es	includ	es	includ	es
included		includ	ed	includ	ed	includ	ed
including		includ	ing	includ	ing	includ	ing
add		add		add		add	
adds		add	s	add	s	add	s
add	ed	add	ed	add	ed	add	ed
adding		add	ing	add	ing	add	ing
continue		continu	e	continu	e	continu	e
continue	s	continu	es	continu	es	continu	es
continu	ed	continu	ed	continu	ed	continu	ed
continuing		continu	ing	continu	ing	continu	ing
report		report		repo	rt	rep	ort

Learning from types and tokens

- Overdispersion in suffix distribution can be ignored by learning from types instead of tokens
- Some psycholinguistics claim that children learn morphology from types (Pierrehumbert 2003)
- To identify word types the input must be segmented into word tokens
- But the input doesn't come neatly segmented into tokens!
- We have been developing *two stage adaptor models* to deal with type-token mismatches

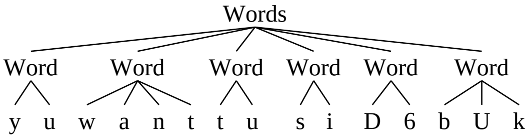
Two stage adaptor framework

- *Generator* determines set of possible structures
- *Adaptor* replicates them an arbitrary number of times (determines their probability)
- "Noisy channel" Bayesian inversion used to train generator and adaptor
  - Generator learns structure from "types"
  - Adaptor learns (power law) frequencies from tokens



Outline

Grammars for word segmentation

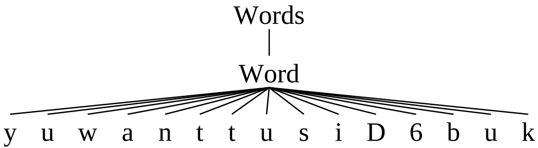


Sample input = y u w a n t t u s i D 6 b u k

Utterance $\rightarrow$ Word Utterance
Utterance $\rightarrow$ Word
Word $\rightarrow w, \qquad w \in \Sigma^\star$

- These are *unigram models* of sentences
  (each word is *conditionally independent* of its neighbours)
- This assumption is standardly made in models of word segmentation (Brent 1999), but is it accurate?

Saturated grammar is maximum likelihood grammar



- Grammar that generates each utterance as a single word exactly matches input distribution

$\Rightarrow$ saturated grammar is maximum likelihood grammar

$\Rightarrow$ use Bayesian estimation with a sparse Dirichlet process prior

- "Chinese Restaurant Process" used to construct Monte Carlo Sampler

Segmentations found by unigram model

yuwant tu si D6bUk	lUk D*z 6b7 wIT hIz h&t
&nd 6dOgi	yu wanttu lUk&tDIs
lUk&tDIs	h&v6 drINk
oke nQ	WAtsDIs
WAtsD&t	WAtIzIt
lUk k&nyu tek ItQt	tek D6dOgi Qt

- Trained on Brent broad phonemic child-directed corpus
- Tends to find *multi-word expressions*, e.g, *yuwant*
- Word finding accuracy is less than Brent's accuracy
- *These solutions are more likely under Brent's model than the solutions Brent found*

$\Rightarrow$ Brent's search procedure is not finding optimal solution

Contextual dependencies in word segmentation

- Unigram model assumes words are independently distributed
- but words in multiword expressions are not independently distributed
  - if we train from a corpus in which the words are randomly permuted, the unigram model finds correct segmentations
- Bigram models capture word-word dependencies $P(w_{i+1}|w_i)$
- straight-forward to build a Gibbs sampler,
  even though we don't have a fixed set of words
  - Each step reanalyses a word or pair of words using the analyses of the rest of the input

Segmentations found by bigram model

yu want tu si D6 bUk	lUk D*z 6 b7 wIT hIz h&t
&nd 6 dOgi	yu want tu lUk&t DIs
lUk&t DIs	h&v 6 drINk
oke nQ	WAts DIs
WAts D&t	WAtIz It
lUk k&nyu tek It Qt	tek D6 dOgi Qt

- Bigram model segments much more accurately than unigram model and Brent's model

$\Rightarrow$ *conditional independence alone is not a sufficient cue for accurate word segmentation*

Outline

Conclusion

- We have mathematical and computational tools to connect learning theory and linguistic theory
- Studying learning via *explicit probabilistic models*
  - is compatible with linguistic theory
  - permits quantitative study of models and information sources
  - helps *understand* why a learning model succeeds or fails
- Bayesian learning lets us combine statistical learning with with prior information
  - priors can encode "Occam's razor" preferences for sparse grammars, and
  - *universal grammar* and *markedness preferences*
  - evaluate usefulness of different types of linguistic universals are for language acquisition

Future work

- Integrate the morphology and word segmentation systems
  - Are their *synergistic interactions* between these components?
- Include other linguistic phenomena
  - Would a *phonological component* improve lexical and morphological acquisition?
- Develop more *realistic training data corpora*
  - Use *forced alignment* to identify pronunciation variants and prosodic properties of words in child-directed speech
- Develop priors that encode *linguistic universals* and *markedness preferences*
  - quantitatively evaluate their usefulness for acquisition

Thank you!

- This research was supported by:
  - The National Science Foundation
  - The National Institutes of Health
  - Brown University