Foundational Models of Language I

IPAM Graduate Summer School:
Probabilistic Models of Cognition

Sharon Goldwater

School of Informatics
University of Edinburgh
sgwater@inf.ed.ac.uk

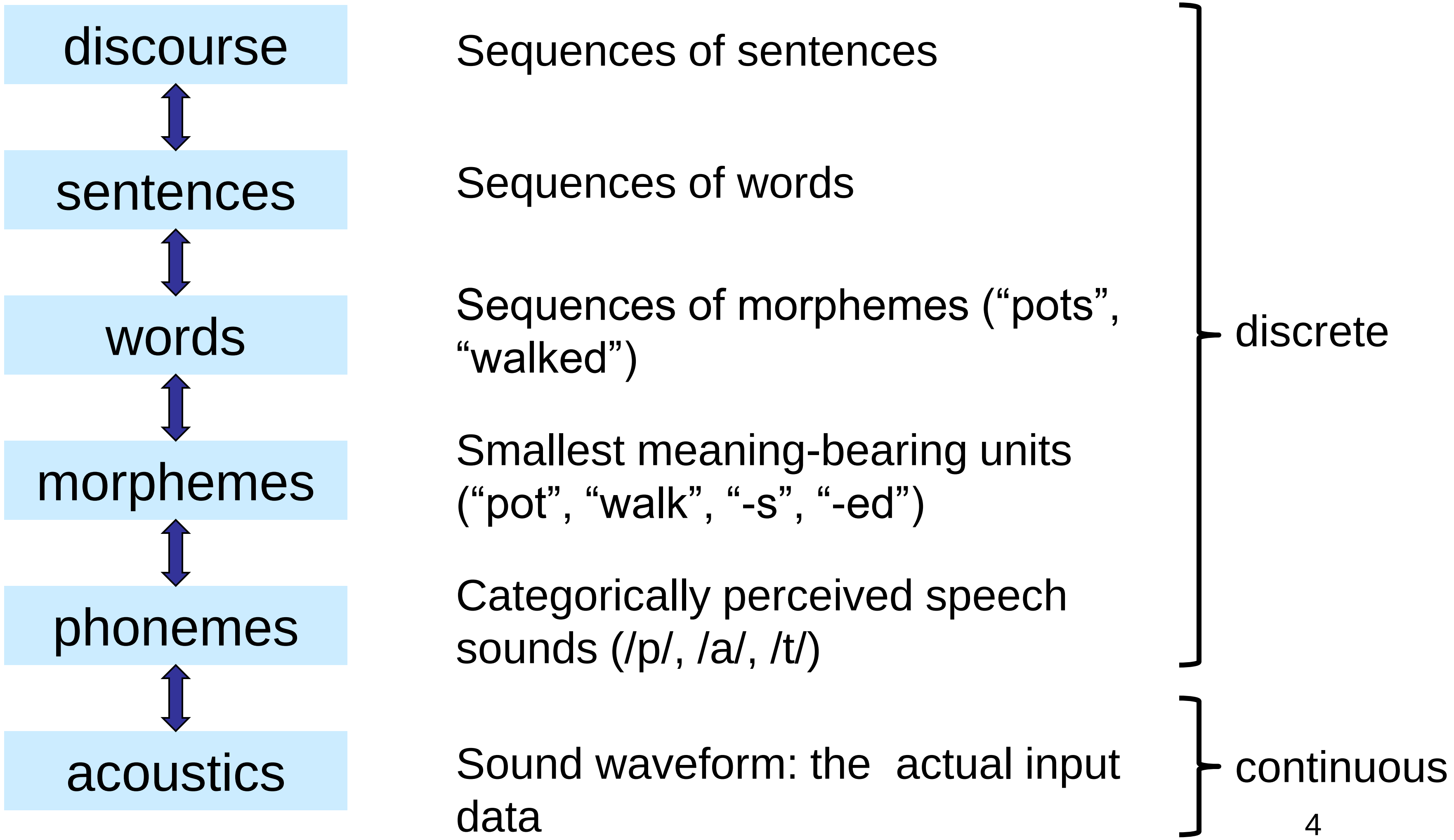
What is special about language?

- Unique to humans
  - Open question: which components of language are shared with other species?
- “Infinite use of finite means” (Hauser, Chomsky, & Fitch, 2002).
- Complex system learned quickly and easily by infants with almost no instruction, yet difficult for adults.
- Possibility of specialized cognitive processes.

What is special about language data?

- Hierarchical (compositional) structure.
- Sparse distributions (Zipf’s law).
- Created by humans for humans (a product of our brains, not the external world).

Language is hierarchical



discourse	Sequences of sentences	discrete
sentences	Sequences of words	
words	Sequences of morphemes ("pots", "walked")	
morphemes	Smallest meaning-bearing units ("pot", "walk", "-s", "-ed")	
phonemes	Categorically perceived speech sounds (/p/, /a/, /t/)	
acoustics	Sound waveform: the actual input data	continuous

Language is also non-hierarchical

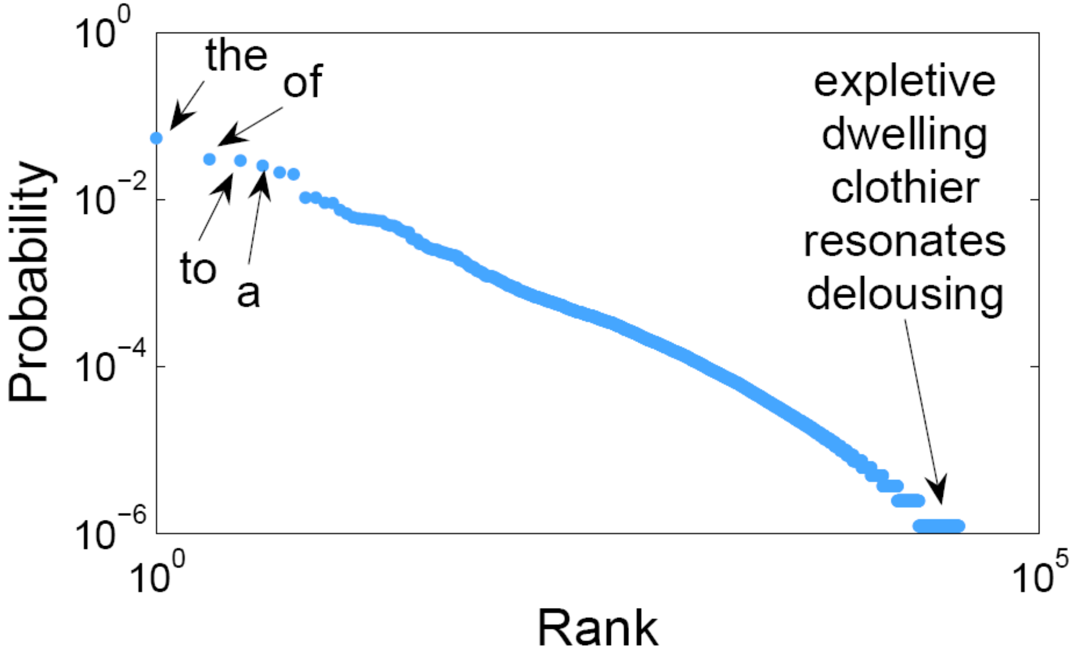
- Language has several levels of structure, each composed of discrete units from the level below.

  but

- Within each level, compositional structure may or may not be hierarchical.
- Different kinds of structures can occur at the same level (e.g. syntactic structure vs. intonational structure in sentences).

Language is sparse

- Distribution over words follows Zipf's law:



  - A power-law distribution: $P(n_w = x) \propto x^{-g}$. Different languages have different exponents, but same pattern.

Language is sparse

- Sparsity also applies to other linguistic units, whether the possibilities are finite or infinite.
  - Phonemes (finite), syntactic constructions (infinite).
- Regardless of the amount of data seen, models must still generalize to rare and unseen cases.

Colorless green ideas sleep furiously.
*Green furiously sleep ideas colorless.

- Building assumptions of sparsity into a model can improve performance.

Language is made by people

- Most of our cognitive systems are adapted to learn from data from the external world.
    - Learning and processing biases reflect existing structure.
- Language data is produced by other people.
    - Biases can be arbitrary (?) as long as they match other people's biases.
- Understanding people's interactions is important for studying many aspects of language, incl. learning, evolution, and change.
    - Difficult! But, can also learn by modeling individuals.

Today's lectures

- Phenomena we might want to model.
  - A sampling of questions from psycholinguistics.
- Hierarchical and non-hierarchical models for language.
  - Four kinds of probabilistic models every computational linguist should know.
  - Many of next week's language lectures will further explore and build on these models, with specific applications.
- Dealing with sparse distributions.
  - How Bayesian modeling can help with this.
  - Again, more next week.

Areas of linguistics

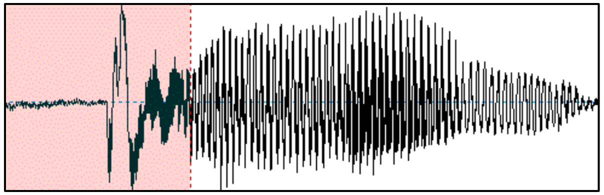
pragmatics	Conventional meaning and discourse effects
semantics	Literal meaning of words and sentences
syntax	Structure of words in sentences
morphology	Structure of morphemes in words
phonology	Structure of sounds in morphemes, words, and sentences (incl. intonation)
phonetics	Relationship of acoustics to phonemes.

Most models deal with only one or two areas at once, though interactions between levels (joint models) are an interesting and growing research area.

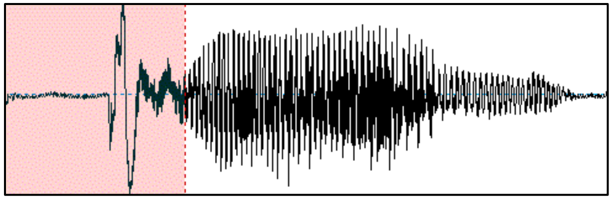
[More later: Feldman]

Phonetics and phonology

- Different acoustic input is perceived as the 'same' sound category. How?
  - Random pronunciation variation.

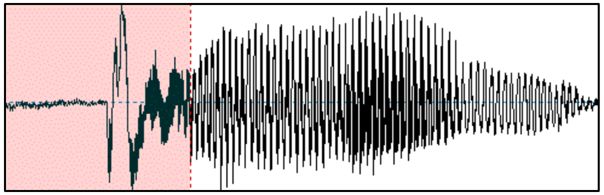


"pan"

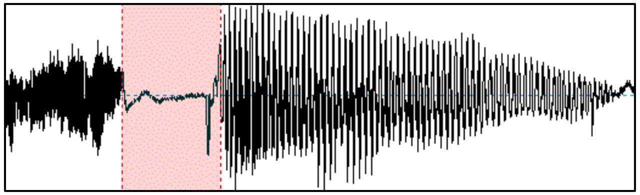


"pan"

  - Systematic non-linguistic variation (e.g., speaker differences).
  - Systematic linguistic variation (e.g., context).



"pan"



"span"

[More later: Johnson, O'Donnell]

Morphemes and words

- How do infants learn to identify word and morpheme boundaries in continuous speech?
- How do we recognize words, both in isolation and in continuous speech?
  - Lots of ambiguity in both boundaries and phonemes.

It's hard to recognize speech.
It's hard to wreck a nice beach.

- How are morphologically complex words stored, understood, and generated?
  - Do we store individual morphemes and use rules to combine them, or store whole words? What about irregular forms (*walk-walked* vs. *run-ran*)?

[More later: Johnson, O’Donnell, Levy]

Syntax

- How do we learn syntactic categories (parts of speech) like nouns and verbs?

> I asked her not to lumple. But yesterday…

- How do we learn which sentences are legal?

She saw Kim with Sandy.	Who did she see Kim with?
She saw Kim and Sandy.	*Who did she see Kim and?

- How do we deal with ambiguity in processing sentences?

[More later: Liang]

Semantics and pragmatics

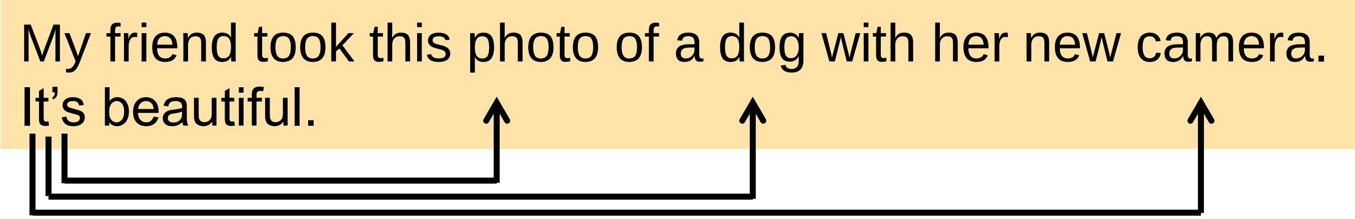
- How do we learn the meanings of individual words?

“Look at the doggie”

- What do words actually mean anyway?

My team is likely to win the playoff.

- How do we resolve ambiguities in discourse?



My friend took this photo of a dog with her new camera.
It’s beautiful.

Language models

- A language model (LM) defines a probability distribution over a sequence of words* $\mathbf{w} = w_1 \ldots w_N$.
  - Uses a probabilistic generative process.
  - May include hidden variables: parts of speech, syntactic relationships, semantic features, etc.

- Not all models for language are language models.
  - Clustering model for phonetic category learning.
  - Vector space/geometric model for word meanings.

*or phonemes, morphemes, etc. We'll assume words here.

Four kinds of language models

- Bag-of-words
  - Independent draws.
- *N*-gram
  - sequence model, no latent variables.
- Hidden Markov model (HMM)
  - sequence model with latent variables.
- Probabilistic context-free grammar (PCFG)
  - hierarchical model with latent variables.

Bag-of-words model

- Simplest kind of LM: words are generated iid.
  - Draw words from a “bag” with replacement.

$$P(\mathbf{w}) = \prod_{i=1}^{N} P(w_i)$$

OR

$$w_i \mid \theta \sim \mathrm{Multinomial}(\theta)$$

- Very bad model of syntax, but useful for semantics.
  - Latent Dirichlet Process (LDA) model: a “topic” is a bag of words. A document contains words from one or more topics. Use to compute document similarity (information retrieval) or word similarity (word association, priming, etc.)

For more on LDA in cognitive modeling, see Griffiths et al. (2007).

N-gram model

- Simplest way to incorporate context: generate each word conditioned on the previous *n*-1 words.
  - Bigram model (*n*=2): $P(\mathbf{w}) = \prod_{i=1}^{N} P(w_i \mid w_{i-1})$
  - Unigram model is just bag of words.
- Not cognitively plausible for syntax: no long-range dependencies.

The girl is nice vs. The girls are nice

The girl over there is nice.
The girl over there walking the dog is nice.
The girl over there walking the dog and looking confused is nice.

N-grams are useful

- Many cognitive models require estimates of word probabilities in context, use *n*-grams.
  - Ex. Pronunciation variability (word duration).
- *N*-grams used as a kind of filter in NLP applications.
  - Ex. Machine translation, speech recognition; *n* = 3 to 7.

It’s hard to wreck a nice beach
It’s are direct ton eyes peach
It’s hard tour reckon ice beach
It’s hard to recognize speech

- Plausible cognitive model for parts of phonology (phoneme *n*-grams).

Estimating parameters

- How to determine $\theta^{(j)}$ (distribution of words after w_j)?
    - To simplify, assume we split data, so $\mathbf{w}$ is now all words with context w_j. Then estimate θ for unigram model of $\mathbf{w}$.
- How to determine θ?
    - One way: use empirical probabilities (relative frequencies) from a large corpus.

$$P(w_i = k) = \frac{n_k}{N}$$

Actual bigram model:

$$P(w_i = k \mid w_{i-1} = k') = \frac{n_{k,k'}}{n_k}$$

- This is maximum-likelihood estimation.

$$P(w_i = k) = \hat{\theta}_k \quad \text{where} \quad \hat{\theta} = \arg\max_{\theta} P(\mathbf{w} \mid \theta)$$

Smoothing

- Problem: higher order *n*-gram → better model, but more sparse data. MLE gets zero counts, overfits.
- Lots of NLP research on how to smooth estimates of θ.
  - Add-λ smoothing (*W* = vocab size):

$$P(w_i = k) = \frac{n_k + \lambda}{N + W\lambda}$$

  - Interpolation (here for bigrams, can generalize to higher order):

$$P(w_i \mid w_{i-1}) = \lambda P(w_i \mid w_{i-1}) + (1 - \lambda) P(w_i)$$

    where each $P(.)$ is estimated using MLE.
  - Many fancier methods.

Bayesian estimation

- Many smoothing methods can be reinterpreted as Bayesian estimation.
  - Don’t estimate θ, what we really want is $P(w_{N+1}|\mathbf{w})$.
  - Compute weighted average over values of θ:

$$P(w_{N+1} \mid \mathbf{w}) = \int P(w_{N+1} \mid \theta) P(\theta \mid \mathbf{w})\, d\theta$$

  - Requires a prior distribution over θ. Dirichlet prior is convenient, makes the integral easy to compute.

Dirichlet distribution

- Dirichlet is a distribution over distributions.
  - Samples from a *K*-dimensional Dirichlet with parameters $\alpha = \alpha_1 \ldots \alpha_K$ are parameters $\theta = \theta_1 \ldots \theta_K$ of multinomial.

$$P(\theta) \propto \prod_{j=1}^{K} \theta_j^{\alpha_j - 1}$$

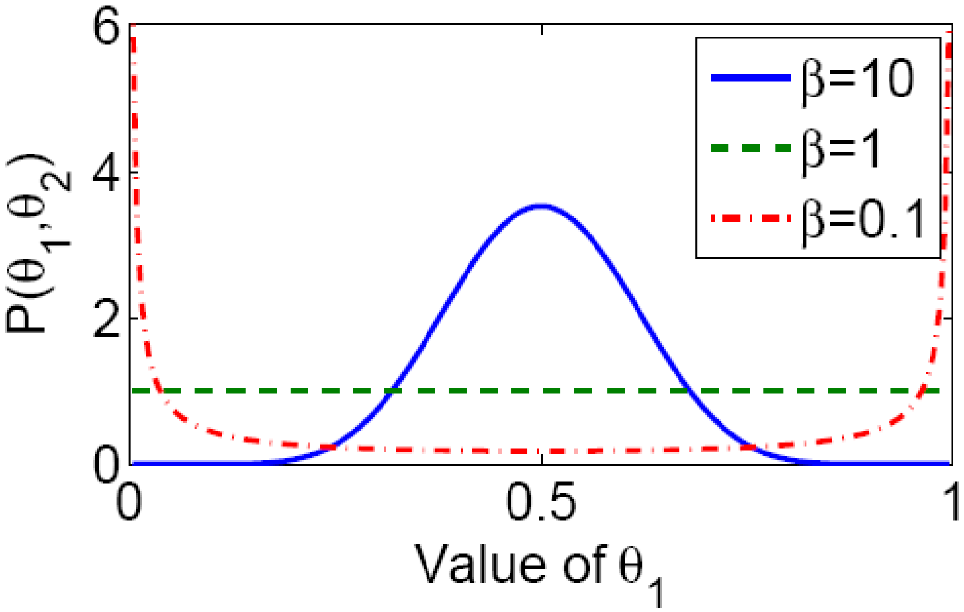
  - We usually use a symmetric Dirichlet, where $\alpha_1 \ldots \alpha_K$ are all equal to β. Write Dirichlet(β) to mean Dirichlet($\beta, \beta, \ldots, \beta$).
  - New model:

$$\theta \mid \beta \sim \text{Dirichlet}(\beta)$$

$$w_i \mid \theta \sim \text{Multinomial}(\theta)$$

Dirichlet distribution

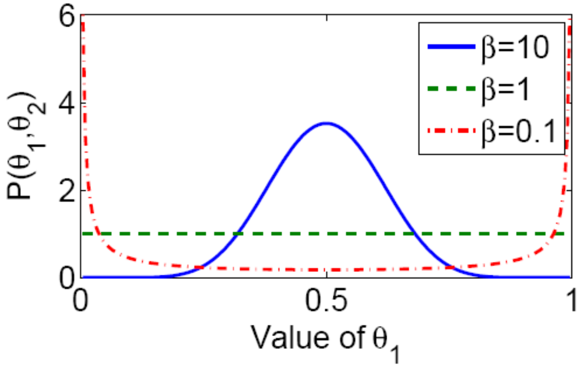
- A 2-dim. symmetric Dirichlet(β) prior over $\theta = (\theta_1, \theta_2)$*:



  - $\beta > 1$: prefer uniform distributions
  - $\beta = 1$: no preference
  - $\beta < 1$: prefer sparse (skewed) distributions

*Normally, the 2-dim. Multinomial is called Binomial; the 2-dim. Dirichlet is called Beta.

Example: coin factory



- What is the prior distribution over θ_h, the probability of flipping heads using a coin from the factory?
    - Factory makes weighted coins, but we don’t know the weight.

      β = 1: an uninformative prior.
    - Factory normally makes fair coins, but occasionally the equipment is misaligned.

      β > 1: we think coins are fair (unless we get a lot of evidence to the contrary).
    - Someone tampered with the equipment.

      β < 1: we think coins are biased, but don’t know which way. (A little evidence suggests which way, a lot of evidence required to convince us coins are actually fair.)

Dirichlet-multinomial model

- Dirichlet distribution is useful prior for multinomial because they are conjugate distributions.
  - Posterior distribution has same form as prior distribution.*

$$\overset{\text{Dirichlet}}{\overset{\searrow}{P(\theta \mid \mathbf{w})}} \propto \overset{\text{Multinomial}}{\overset{\downarrow}{P(\mathbf{w} \mid \theta)}}\overset{\text{Dirichlet}}{\overset{\swarrow}{P(\theta)}}$$

- This makes integration work out nicely. Specifically,

$$P(w_{N+1} = k \mid \mathbf{w}) = \underset{\text{Asymmetric case}}{\frac{n_k + \alpha_k}{N + \sum_{j=1}^{W} \alpha_j}} = \underset{\text{Symmetric case}}{\frac{n_k + \beta}{N + W\beta}}$$

*In particular, if P(θ) is Dirichlet($\alpha_1 \ldots \alpha_W$), then P($\theta | \mathbf{w}$) is Dirichlet($\alpha_1 + n_1, \ldots, \alpha_W + n_W$), where is n_k is the number of times the *k*th lexical item occurs in $\mathbf{w}$.

Back to smoothing

- Predictive distribution from Dirichlet-multinomial model is just the same as MLE with add-β smoothing:

$$P(w_{N+1} = k \mid \mathbf{w}) = \frac{n_k + \beta}{N + W\beta}$$

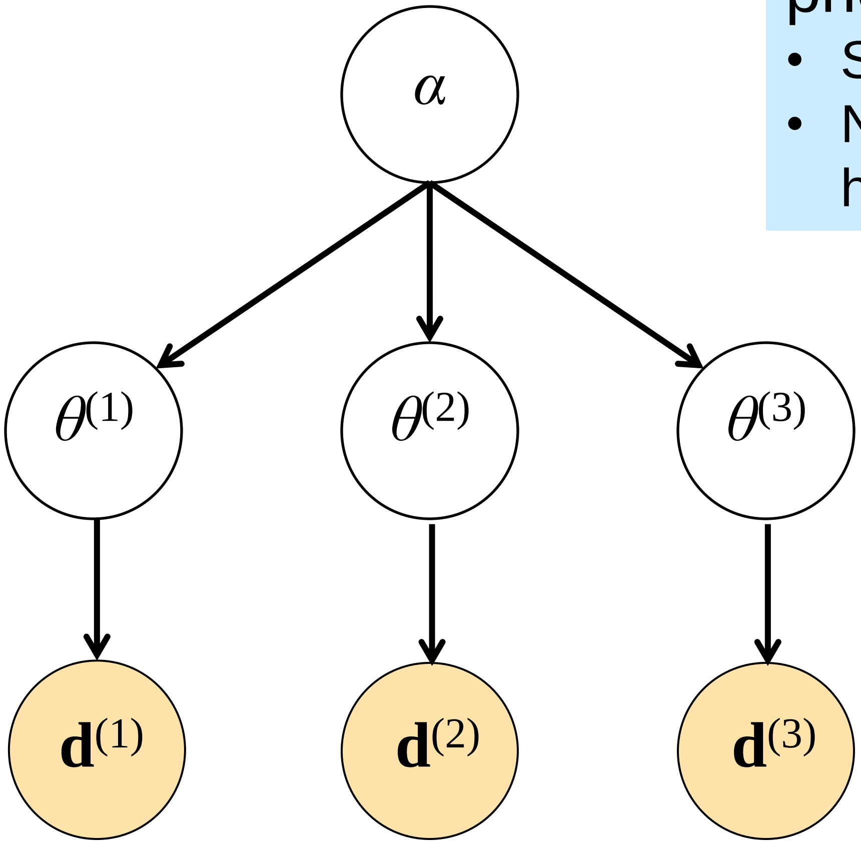
  - Replacing Dirichlet with fancier priors (e.g., Pitman-Yor process) improves the model and yields better results in practice (also deals with unbounded vocabulary size).*

See Goldwater et al. (2007; in press), Teh (2007).

Hierarchical models

- Two ways Bayesian models can improve predictions when data is limited:
  - Use prior and average over model parameters → Dirichlet-multinomial model yields add-λ smoothing.
  - Hierarchical models can share information across similar cases → hierarchical Dirichlet-multinomial model yields (something similar to) interpolation.

Hierarchical model: coin factory



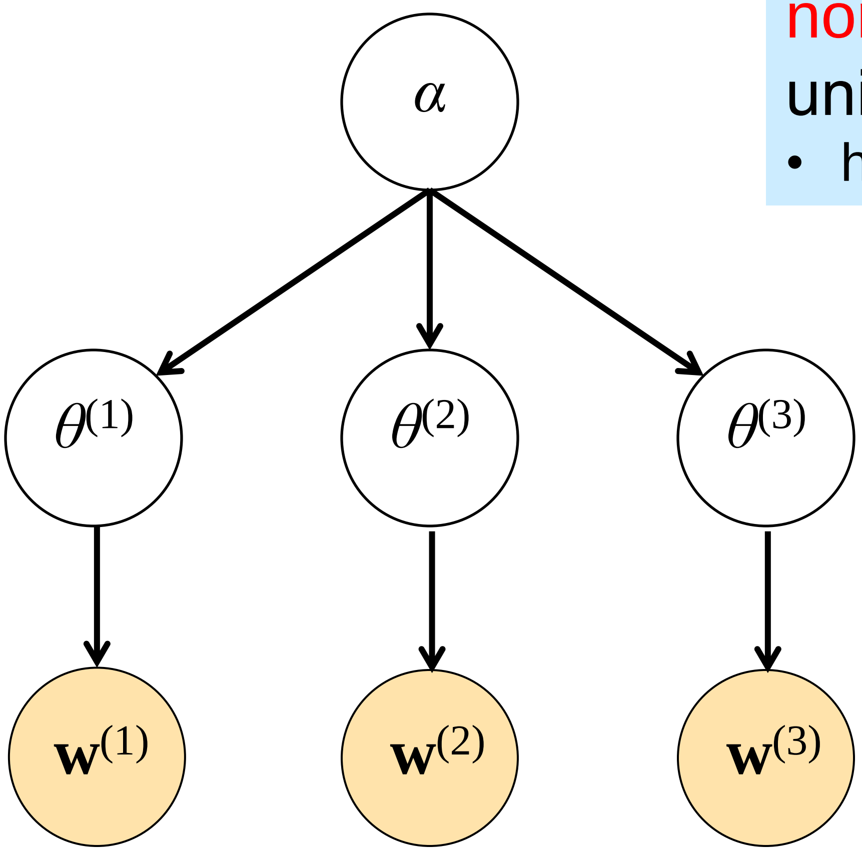
prior beliefs about factory

- Symmetric: Are all coins fair or not?
- Non-symmetric: what proportion heads is typical?

distribution for each coin

data from each coin

Hierarchical language model



non-symmetric prior: α_k is a unigram 'backoff'

- how probable is w_k in new context?

distribution for each context (bigram)

words in each context

Predictive distribution

- Under this model, the Bayesian prediction is

$$P(w_{N+1} = k \mid w_N = k', w_1 \ldots w_{N-1}) = \frac{n_{k',k} + \alpha_k}{n_{k'} + \sum_k \alpha_k}$$

- α_k is the prior probability of seeing w_k in a context where it hasn’t been seen before.
  - But what is its value?

Predictive distribution

- Optimal α_k is not the unigram relative frequency of w_k.
  - Consider this example (MacKay and Peto, 1995):

Imagine, you see, that the language, you see, has, you see, a frequently occurring couplet, 'you see', you see, in which the second word of the couplet, see, follows the first word, you, with very high probability, you see. Then the marginal statistics, you see, are going to become hugely dominated, you see, by the words you and see, with equal frequency, you see.

- P(see) and P(you) are both high, but see nearly always follows you. So P(see|novel) should be much lower than P(you|novel).
- Bayesian prediction: α_k is related to the number of distinct contexts where w_k appears.
  - The best frequentist smoothing methods also use the number of contexts as a key component.

Summary

- Features of linguistic data:
  - Discrete, hierarchical. Cognitive questions at all levels; models often deal with only one level, which may or may not be hierarchical itself.
  - Generative, sparse. Models must be able to generalize to novel instances, regardless of how much data is seen.
- N-gram models:
  - Non-hierarchical (sequence) model. Not cognitively plausible for syntax, but very useful approximation.
  - Bayesian models can ‘smooth’ to deal with sparse data.
  - So far, smoothing methods assume fixed vocab size. Next week, nonparametric models to deal with unbounded vocab.

References

- Sharon Goldwater, Thomas L. Griffiths, and Mark Johnson. In press. Producing power-law distributions and damping word frequencies with two-stage language models. *Journal of Machine Learning Research*.
- Thomas L. Griffiths and Mark Steyvers and Joshua B. Tenenbaum. 2007. Topics in semantic representation. *Psychological Review*, 114, 211-244.
- David MacKay and Linda Bauman Peto. 1995. A hierarchical Dirichlet language model. *Natural language engineering*, 1(3), 289-308.
- Yee Whye Teh. 2006. A Hierarchical Bayesian Language Model based on Pitman-Yor Processes. *Proceedings of Coling/ACL*.