Bayesian Inference for PCFGs via Markov chain Monte Carlo

Mark Johnson, Thomas L. Griffiths and Sharon Goldwater

HLT/NAACL 2007

Outline

Introduction

Bayesian inference for PCFGs

Gibbs sampler for $(\mathbf{t}, \theta)$

Metropolis-Hastings collapsed sampler for $\mathbf{t}$

Application: Unsupervised morphological analysis of Sesotho verbs

Conclusion

Goal and motivation

- Develop algorithms for Bayesian inference for Probabilistic Context-Free Grammars (PCFGs)
- Bayesian inference combines *likelihood* with *prior* information
  - bias learner toward *sparse grammars*
- The techniques presented here generalize to other generative models with branching structure
  - more complex parsing models (e.g., “Adaptor Grammars”, a non-parametric extension of PCFGs)
  - Hidden Markov Models (a sampler that resamples all labels at once, cf. Neal 2006)

Probabilistic context-free grammars

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

$$\mathrm{P}(t|\theta) \quad = \quad \prod_{r \in R} \theta_r^{f_r(t)}$$

where $f_r(t)$ is the number of times rule r appears in tree t.

- The *probability of a string* w is the sum of probabilities of all trees with w as their yield

$$\mathrm{P}(w|\theta) \quad = \quad \sum_{t : y(t) = w} \mathrm{P}(t|\theta)$$

$$R = \left\{ \begin{array}{l} \mathrm{S} \to \mathrm{NP}\ \mathrm{VP} \\ \mathrm{NP} \to \mathrm{Al}, \\ \mathrm{NP} \to \mathrm{George} \\ \mathrm{VP} \to \mathrm{barks} \\ \mathrm{VP} \to \mathrm{snores} \end{array} \right\}, \begin{array}{lcl} \theta_{\mathrm{S} \to \mathrm{NP}\ \mathrm{VP}} & = & 1.0 \\ \theta_{\mathrm{NP} \to \mathrm{Al}} & = & 0.5 \\ \theta_{\mathrm{NP} \to \mathrm{George}} & = & 0.5 \\ \theta_{\mathrm{VP} \to \mathrm{barks}} & = & 0.2 \\ \theta_{\mathrm{VP} \to \mathrm{snores}} & = & 0.8 \end{array}, \ \mathrm{P}\left([_{\mathrm{S}}\ [_{\mathrm{NP}}\ \mathrm{Al}]\ [_{\mathrm{VP}}\ \mathrm{barks}]] \right) = 0.1$$

Unsupervised inference for PCFGs

- Given rules R and corpus of strings $\mathbf{w}$, infer:
  - rule probabilities θ
  - trees $\mathbf{t}$ for $\mathbf{w}$
- Maximum likelihood, e.g. Inside-Outside/EM (a *point estimate*)

$$\hat{\theta} = \operatorname*{argmax}_{\theta} \mathrm{P}(\mathbf{w}|\theta) \qquad \text{(EM)}$$

$$\hat{\mathbf{t}} = \operatorname*{argmax}_{\mathbf{t}} \mathrm{P}(\mathbf{t}|\mathbf{w},\hat{\theta}) \qquad \text{(Viterbi)}$$

- Bayesian inference incorporates *prior* $\mathrm{P}(\theta)$ and infers a *posterior distribution*

$$\underbrace{\mathrm{P}(\theta|\mathbf{w})}_{\text{Posterior}} \propto \underbrace{\mathrm{P}(\mathbf{w}|\theta)}_{\text{Likelihood}} \underbrace{\mathrm{P}(\theta)}_{\text{Prior}}$$

$$\mathrm{P}(\mathbf{t}|\mathbf{w}) \propto \int_{\Delta} \mathrm{P}(\mathbf{w},\mathbf{t}|\theta)\,\mathrm{P}(\theta)\,d\theta$$

Outline

Introduction

Bayesian inference for PCFGs

Gibbs sampler for $(\mathbf{t}, \theta)$

Metropolis-Hastings collapsed sampler for $\mathbf{t}$

Application: Unsupervised morphological analysis of Sesotho verbs

Conclusion

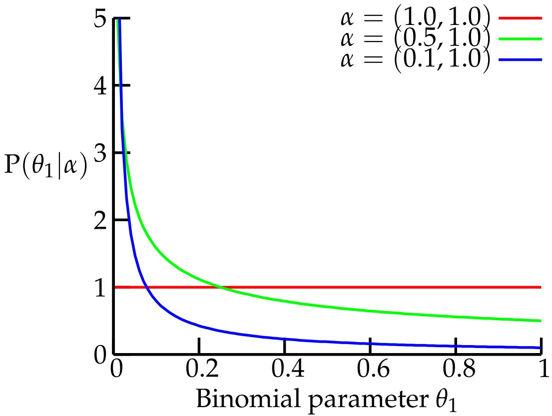
Bayesian priors

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

- Hypothesis = rule probabilities θ, Data = strings $\mathbf{w}$
- Prior can incorporate linguistic insights ("universal grammar")
- Math/computation vastly simplified if *prior is conjugate to likelihood*
  - posterior belongs to the same model family as prior
- PCFGs are *products of multinomials*, one for each nonterminal A
  - model has a parameter $\theta_{A \to \beta}$ for each rule $A \to \beta \in R$

$\Rightarrow$ Conjugate prior is *product of Dirichlets*, one for each nonterminal A
  - prior has a hyper-parameter $\alpha_{A \to \beta}$ for each rule $A \to \beta \in R$

Dirichlet priors for multinomials



- Outcomes $1, \ldots, m$
- Multinomial $P(X = i) = \theta_i$
  $\theta = (\theta_1, \ldots, \theta_m)$
- Dirichlet prior parameters
  $\alpha = (\alpha_1, \ldots, \alpha_m)$

$$P_D(\theta|\alpha) = \frac{1}{Z(\alpha)} \prod_{i=1}^{m} \theta_i^{\alpha_i - 1}$$

$$Z(\alpha) = \frac{\prod_{i=1}^{m} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{m} \alpha_i)}$$

- As α_1 approaches 0, $P(\theta_1|\alpha)$ concentrates around 0
- PCFG prior is product of Dirichlets (one for each $A \in N$)
- Dirichlet for A in PCFG prior has hyper-parameter vector α_A
- Dirichlet prior can prefer *sparse grammars* in which $\theta_r = 0$

Dirichlet priors for PCFGs

- Let R_A be the rules expanding A in R, and θ_A, α_A be the subvectors of θ, α corresponding to R_A
- Conjugacy makes the posterior simple to compute given trees $\mathbf{t}$:

$$\begin{aligned}
\mathrm{P}_D(\theta|\alpha) &= \prod_{A \in N} \mathrm{P}_D(\theta_A|\alpha_A) \;\propto\; \prod_{r \in R} \theta^{\alpha_r} \\
\mathrm{P}(\theta|\mathbf{t},\alpha) &\propto \mathrm{P}(\mathbf{t}|\theta)\mathrm{P}_D(\theta|\alpha) \\
&\propto \left(\prod_{r \in R} \theta_r^{f_r(\mathbf{t})}\right) \left(\prod_{r \in R} \theta_r^{\alpha_r - 1}\right) \\
&= \prod_{r \in R} \theta_r^{f_r(\mathbf{t}) + \alpha_r - 1}, \quad \text{so} \\
\mathrm{P}(\theta|\mathbf{t},\alpha) &= \mathrm{P}_D(\theta|\mathbf{f}(\mathbf{t}) + \alpha)
\end{aligned}$$

- So when trees $\mathbf{t}$ are observed, posterior is product of Dirichlets
- But what if trees $\mathbf{t}$ are hidden, and only strings $\mathbf{w}$ are observed?

Algorithms for Bayesian inference

- Posterior is *computationally intractable*

$$\mathrm{P}(\mathbf{t}, \theta | \mathbf{w}) \quad \propto \quad \mathrm{P}(\mathbf{w}, \mathbf{t} | \theta)\, \mathrm{P}(\theta)$$

- Maximum A Posteriori (MAP) estimation finds the posterior mode

$$\theta^{\star} \quad = \quad \underset{\theta}{\operatorname{argmax}}\, \mathrm{P}(\mathbf{w} | \theta)\, \mathrm{P}(\theta)$$

- Variational Bayes assumes posterior approximately factorizes

$$\mathrm{P}(\mathbf{w}, \mathbf{t}, \theta) \quad \approx \quad Q(\mathbf{t}) Q(\theta)$$

  EM-like iterations using Inside-Outside (Kurihara and Sato 2006)
- Markov Chain Monte Carlo methods construct a Markov chain whose states are samples from $\mathrm{P}(\mathbf{t}, \theta | \mathbf{w})$

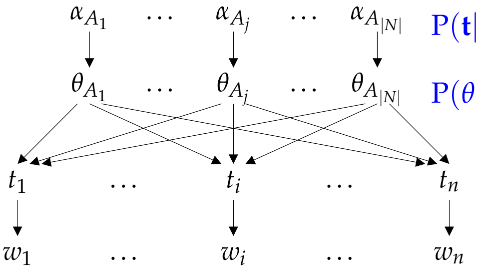
Markov chain Monte Carlo

- MCMC algorithms define a Markov chain where:
  - the states s are the objects we wish to sample; e.g., $s = (\mathbf{t}, \theta)$
    - the state space is astronomically large
  - transition probabilities $\mathrm{P}(s'|s)$ are chosen so that chain converges on desired distribution $\pi(s)$
    - many standard recipes for defining $\mathrm{P}(s'|s)$ from $\pi(s)$ (e.g., Gibbs, Metropolis-Hastings)
- "Run" the chain by:
  - pick a start state s_0
  - pick state s_{t+1} by sampling from $\mathrm{P}(s'|s_t)$
- To estimate the *expected value* of any function f of state s (e.g., rule probabilities θ):
  - discard first few "burn-in" samples from chain
  - average $f(s)$ over the remaining samples from chain

Outline

A Gibbs sampler for $\mathbf{t}$ and θ

- Gibbs samplers require states factor into components $s = (\mathbf{t}, \theta)$
- Update each component in turn by resampling, conditioned on values for other components
  - Resample trees $\mathbf{t}$ given strings $\mathbf{w}$ and rule probabilities θ
  - Resample rule probabilities θ given trees $\mathbf{t}$ and priors α



$$
\begin{aligned}
\mathrm{P}(\mathbf{t}|\theta, \mathbf{w}, \alpha) &= \prod_{i=1}^{n} \mathrm{P}(t_i|w_i, \theta) \\
\mathrm{P}(\theta|\mathbf{t}, \mathbf{w}, \alpha) &= \mathrm{P}_D(\theta|\mathbf{f}(\mathbf{t}) + \alpha) \\
&= \prod_{A \in N} \mathrm{P}_D(\theta|\mathbf{f}_A(\mathbf{t}) + \alpha_A)
\end{aligned}
$$

- There are standard algorithms for sampling from these distributions
- Trees $\mathbf{t}$ are *independent given rule probabilities θ*

  $\Rightarrow$ each t_i can be sampled in parallel

  $\Rightarrow$ t_i only influences t_j via θ ("mixes slowly", "poor mobility")

Outline

Marginalizing out the rule probabilities θ

- Define MCMC sampler whose states are the vectors of trees $\mathbf{t}$
- Integrate out the rule probabilities θ, collapsing dependencies and coupling trees

$$\mathrm{P}(\mathbf{t}|\alpha) \quad = \quad \int_{\Delta} \mathrm{P}(\mathbf{t}|\theta)\, \mathrm{P}(\theta|\alpha)\, d\theta \;=\; \prod_{A \in N} \frac{Z(\mathbf{f}_A(\mathbf{t}) + \alpha_A)}{Z(\alpha_A)}$$

- Components of state are the trees t_i for strings w_i
  - resample t_i given trees $\mathbf{t}_{-i}$ for other strings $\mathbf{w}_i$

$$\mathrm{P}(t_i|\mathbf{t}_{-i},\alpha) \quad = \quad \frac{\mathrm{P}(\mathbf{t}|\alpha)}{\mathrm{P}(\mathbf{t}_{-i}|\alpha)} \;=\; \prod_{A \in N} \frac{Z(\mathbf{f}_A(\mathbf{t}) + \alpha_A)}{Z(\mathbf{f}_A(\mathbf{t}_{-i}) + \alpha_A)}$$

- (Sample θ from $\mathrm{P}(\theta|\mathbf{t},\alpha)$ if required).
- If we could sample from

$$\mathrm{P}(t_i|w_i,\mathbf{t}_{-i},\alpha) \quad = \quad \frac{\mathrm{P}(w_i|t_i)\mathrm{P}(t_i|\mathbf{t}_{-i},\alpha)}{\mathrm{P}(w_i|\mathbf{t}_{-i},\alpha)}$$

we could build a Gibbs sampler whose states are trees $\mathbf{t}$

Why Metropolis-Hastings?

$$P(t_i|\mathbf{t}_{-i},\alpha) \quad = \quad \prod_{A \in N} \frac{Z(\mathbf{f}_A(\mathbf{t}) + \alpha_A)}{Z(\mathbf{f}_A(\mathbf{t}_{-i}) + \alpha_A)}$$

- What makes $P(t_i|\mathbf{t}_{-i},\alpha)$ so hard to sample?
  - Probability of choosing rule r used n_r times before $\propto n_r + \alpha_r$
  - Previous occurences of r "prime" the rule r
  - *Rule probabilities can change on the fly* inside a sentence
  - Breaks dynamic programming sampling algorithms, which require "context-freeness"
- Metropolis-Hastings algorithms don't need samples from $P(t_i|\mathbf{t}_{-i},\alpha)$
  - sample from a user-specified *proposal distribution* Q
  - use *acceptance-rejection procedure* to convert stream of samples from Q into stream of samples from $P(\mathbf{t})$
- Proposal distribution Q can be any strictly positive distribution
  - more efficient (fewer rejections) if Q close to $P(\mathbf{t})$
  - our proposal distribution $Q_i(t_i)$ is *PCFG approximation* $E[\theta|\mathbf{t}_{-i},\alpha]$

Metropolis-Hastings collapsed PCFG sampler

- Sampler state: vector of trees $\mathbf{t}$, t_i is a parse of w_i
- Repeat until convergence:
  - randomly choose index i of tree to resample
  - compute PCFG probabilities to be used as proposal distribution

$$\tilde{\theta}_{A\to\beta} \;=\; E[\theta_{A\to\beta}|\mathbf{t}_{-i},\alpha] \;=\; \frac{f_{A\to\beta}(\mathbf{t}_{-i})+\alpha_{A\to\beta}}{\sum_{A\to\beta'\in R_A} f_{A\to\beta'}(\mathbf{t}_{-i})+\alpha_{A\to\beta'}}$$

  - sample a proposal tree t'_i from $\mathrm{P}(t_i|w_i,\tilde{\theta})$
  - compute acceptance probability $A(t_i,t'_i)$ for t'_i

$$A(t_i,t'_i) \;=\; \min\left\{1, \frac{\mathrm{P}(t'_i|\mathbf{t}_{-i},\alpha)\mathrm{P}(t_i|w_i,\tilde{\theta})}{\mathrm{P}(t_i|\mathbf{t}_{-i},\alpha)\mathrm{P}(t'_i|w_i,\tilde{\theta})}\right\}$$

(easy to compute since t'_i is fixed)

  - choose a random number $x \in U[0,1]$
    - if $x < A(t_i,t'_i)$ then *accept t'_i*, i.e., replace t_i with t'_i
    - if $x > A(t_i,t'_i)$ then *reject t'_i*, i.e., keep t_i unchanged

Outline

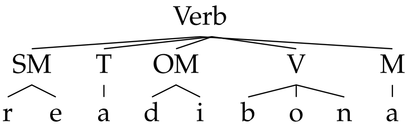
Sesotho verbal morphology

- Sesotho is a Bantu language with complex morphology, not "messed up" much by phonology

re	*a*	*di*	*bon*	*a*
SM	T	OM	V	M

"We see them"

- Demuth's Sesotho corpus contains morphological parses for 2,283 distinct verb types; can we *learn them automatically*?
- Morphological structure reasonably well described by a CFG

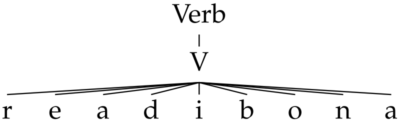


Verb → V
Verb → V M
Verb → SM V M
Verb → SM T V M
Verb → SM T OM V M

- We added 81,755 productions expanding each preterminal to each of the 16,350 *contiguous substrings* of any verb in corpus

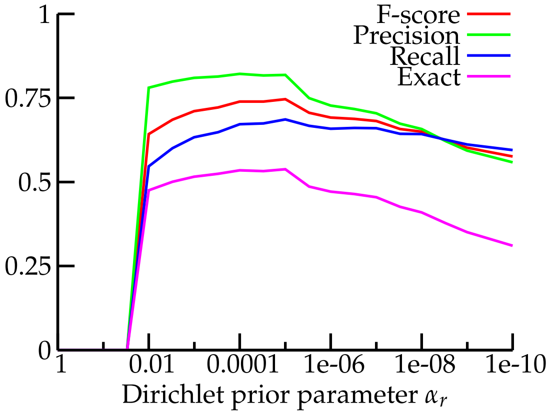
Maximum likelihood finds trivial "saturated" grammar

- Grammar has more productions (81,000) than training strings (2,283)
- Maximum likelihood (e.g., Inside/Outside, EM) tries to make predicted probabilities match empirical probabilities
- *"Saturated" grammar:* every word type has its own production



  - exactly matches empirical probabilities
  - this is what Inside-Outside EM finds
  - none of these analyses are correct

Bayesian estimates with sparse prior find nontrivial structure



- Dirichlet prior for all rules set to same value α
- Dirichlet prior prefers sparse grammars when $\alpha \ll 1$
- Non-trivial structure emerges when $\alpha < 0.01$
- Exact word match accuracy ≈ 0.54 at $\alpha = 10^{-5}$

Outline

Introduction

Bayesian inference for PCFGs

Gibbs sampler for $(\mathbf{t}, \theta)$

Metropolis-Hastings collapsed sampler for $\mathbf{t}$

Application: Unsupervised morphological analysis of Sesotho verbs

Conclusion

Conclusion and future work

- Bayesian estimates incorporates *prior* as well as likelihood
  - product of Dirichlets is conjugate prior for PCFGs
  - can be used to prefer *sparse grammars*
- Even though the full Bayesian posterior is mathematically and computationally intractible, it can be approximated using MCMC
  - Gibbs sampler alternates sampling from $\mathrm{P}(\mathbf{t}|\theta)$ and $\mathrm{P}(\theta|\mathbf{t})$
  - Metropolis-Hastings collapsed sampler integrates out θ and samples $\mathrm{P}(t_i|\mathbf{t}_{-i})$
  - C++ implementations available on my Brown web site
- Need to compare these methods with Variational Bayes
- MCMC methods are usually more flexible than other approaches
  - should generalize well to more complex models

Bayesian MAP EM

- EM re-estimation of θ uses ML estimate in M-step

$$\theta_r^{(t+1)} \quad \propto \quad \mathrm{E}[f_r|\mathbf{w},\theta^{(t)}]$$

- Use Bayesian MAP estimate for θ instead of ML estimate

$$\theta_r^{(t+1)} \quad \propto \quad \max(0, \mathrm{E}[f_r|\mathbf{w},\theta^{(t)}] + \alpha_r - 1)$$

- If $\mathrm{E}[f_r|\mathbf{w},\theta^{(t)}] \approx 0$ and $\alpha_r \ll 1$ then
  - $\theta_r^{(t+1)} \; = \; 0$
  - if $\theta_r = 0$ for sufficiently many rules r, then some input strings may fail to parse
  - this occurs in Sesotho example when α_r is small enough to find non-trivial structure
- *Variational Bayes* is the right way to do this!

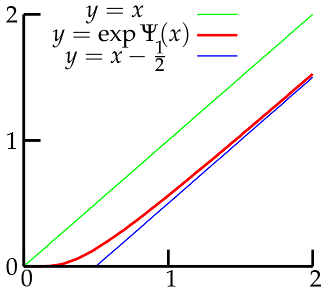
Variational Bayes for PCFGs

- Variational Bayes seeks a factorized representation Q

$$Q(\mathbf{t})Q(\theta) \quad \approx \quad \mathrm{P}(\mathbf{w}, \mathbf{t}, \theta | \alpha)$$

  that maximizes a lower bound on the log likelihood w.r.t. Q
- With Dirichlet prior, yields EM-like updates for variational parameters $\tilde{\theta}$

$$\tilde{\theta}^{(t+1)}_{A\to\beta} \quad = \quad \frac{\exp\Psi(\ \mathrm{E}[f_{A\to\beta}|\mathbf{w}, \tilde{\theta}^{(t)}] + \alpha_{A\to\beta}\ )}{\exp\Psi(\ \sum_{A\to\beta'\in R_A}\ \mathrm{E}[f_{A\to\beta'}|\mathbf{w}, \tilde{\theta}^{(t)}] + \alpha_{A\to\beta'}\ )}$$



- Ψ is the *digamma function*
- $\exp\Psi(x) > 0$ for all $x > 0$, so Bayesian MAP estimator problem never arises with Variational Bayes