The effect of non-tightness on Bayesian estimation of PCFGs

Shay B. Cohen
Department of Computer Science
Columbia University
scohen@cs.columbia.edu

Mark Johnson
Department of Computing
Macquarie University
mark.johnson@mq.edu.au

Abstract

Probabilistic context-free grammars have the unusual property of not always defining tight distributions (i.e., the sum of the "probabilities" of the trees the grammar generates can be less than one). This paper reviews how this non-tightness can arise and discusses its impact on Bayesian estimation of PCFGs. We begin by presenting the notion of "almost everywhere tight grammars" and show that linear CFGs follow it. We then propose three different ways of reinterpreting non-tight PCFGs to make them tight, show that the Bayesian estimators in Johnson et al. (2007) are correct under one of them, and provide MCMC samplers for the other two. We conclude with a discussion of the impact of tightness empirically.

1 Introduction

Probabilistic Context-Free Grammars (PCFGs) play a special role in computational linguistics because they are perhaps the simplest probabilistic models of hierarchical structures. Their simplicity enables us to mathematically analyze their properties to a detail that would be difficult with linguistically more accurate models. Such analysis is useful because it is reasonable to expect more complex models to exhibit similar properties as well.

The problem of inferring PCFG rule probabilities from training data consisting of yields or strings alone is interesting from both cognitive and engineering perspectives. Cognitively it is implausible that children can perceive the parse trees of the language they are learning, but it is more reasonable to assume that they can obtain the terminal strings or yield of these trees. Unsupervised methods for learning a grammar from terminal strings alone is also interesting from an engineering perspective because such training data is cheap and plentiful, while the manually parsed data required by supervised methods are expensive to produce and relatively rare.

Cohen and Smith (2012) show that inferring PCFG rule probabilities from strings alone is computationally intractable, so we should not expect to find an efficient, general-purpose algorithm for the unsupervised problem. Instead, approximation algorithms are standardly used. For example, the Inside-Outside (IO) algorithm efficiently implements the Expectation-Maximization (EM) procedure for approximating a Maximum Likelihood estimator (Lari and Young, 1990). Bayesian estimators for PCFG rule probabilities have also been attracting attention because they provide a theoretically-principled way of incorporating prior information. Kurihara and Sato (2006) proposed a Variational Bayes estimator based on a mean-field approximation, and Johnson et al. (2007) proposed MCMC samplers for the posterior distribution over rule probabilities and the parse trees of the training data strings.

PCFGs have the interesting property (which we expect most linguistically more realistic models to also possess) that the distributions they define are not always properly normalized or "tight". In a non-tight PCFG the partition function (i.e., sum of the "probabilities" of all the trees generated by the PCFG) is less than one. (Booth and Thompson, 1973, called such non-tight PCFGs "inconsistent", but we follow Chi and Geman (1998) in calling them "non-tight" to avoid confusion with the consistency of statistical estimators). Chi (1999) showed that renormalized nontight PCFGs (which he called "Gibbs CFGs") define the same class of distributions over trees as do tight PCFGs with the same rules, and provided an algorithm for mapping any PCFG to a tight PCFG with the same rules that defines the same distribution over trees.

An obvious question is then: how does tightness

affect the inference of PCFGs? Chi and Geman (1998) studied the question for Maximum Likelihood (ML) estimation, and showed that ML estimates are always tight for both the supervised case (where the input consists of parse trees) and the unsupervised case (where the input consists of yields or terminal strings). This means that ML estimators can simply ignore issues of tightness, and rest assured that the PCFGs they estimate are in fact tight.

The situation is more subtle with Bayesian estimators. We show that for the special case of linear PCFGs (which include HMMs) with non-degenerate priors the posterior puts zero mass on non-tight PCFGs, so tightness is not an issue with Bayesian estimation of such grammars. However, because all of the commonly used priors (such as the Dirichlet or the logistic normal) assign non-zero probability across the whole probability simplex, in general the posterior may assign non-zero probability to non-tight PCFGs. We discuss three different possible approaches to this in this paper:

1. the *only-tight* approach, where we modify the prior so it only assigns non-zero probability to tight PCFGs,

2. the *renormalization* approach, where we renormalize non-tight PCFGs so they define a probability distribution over trees, and

3. the *sink-element* approach, where we reinterpret non-tight PCFGs as assigning non-zero probability to a "sink element", so both tight and non-tight PCFGs are properly normalized.

We show how to modify the Gibbs sampler described by Johnson et al. (2007) so it produces samples from the posterior distributions defined by the only-tight and renormalization approaches. Perhaps surprisingly, we show that Gibbs sampler as defined by Johnson et al. actually produces samples from the posterior distributions defined by the sink-element approach.

We conclude by studying the effect of requiring tightness on the estimation of some simple PCFGs. Because the Bayesian posterior converges around the (tight) ML estimate as the size of the data grows, requiring tightness only seems to make a difference with highly biased priors or with very small training corpora.

2 PCFGs and tightness

Let $G = (T, N, S, R)$ be a Context-Free Grammar in Chomsky normal form with no useless productions, where T is a finite set of *terminal symbols*, N is a finite set of *nonterminal symbols* (disjoint from T), $S \in N$ is a distinguished nonterminal called the *start symbol*, and R is a finite set of *productions* of the form $A \to B\,C$ or $A \to w$, where $A, B, C \in N$ and $w \in T$. In what follows we use β as a variable ranging over $(N \times N) \cup T$.

A *Probabilistic Context-Free Grammar* (G, Θ) is a pair consisting of a context-free grammar G and a real-valued vector Θ of length $|R|$ indexed by productions, where $\theta_{A\to\beta}$ is the *production probability* associated with the production $A \to \beta \in R$. We require that $\theta_{A\to\beta} \geq 0$ and that for all nonterminals $A \in N$, $\sum_{A\to\beta\in R_A} \theta_{A\to\beta} = 1$, where R_A is the subset of rules R expanding the nonterminal A.

A PCFG (G, Θ) defines a measure μ_Θ over trees t as follows:

$$\mu_\Theta(t) = \prod_{r\in R} \theta_r^{f_r(t)}$$

where $f_r(t)$ is the number of times the production $r = A \to \beta \in R$ is used in the derivation of t.

The *partition function* Z or measure of all possible trees is:

$$Z(\Theta) = \sum_{t'\in\mathcal{T}} \prod_{r\in R} \theta_r^{f_r(t')}$$

where $\mathcal{T}$ is the set of all (finite) trees generated by G. A PCFG is *tight* iff the partition function $Z(\Theta) = 1$. In this paper we use $\boldsymbol{\Theta}^\perp$ to denote the set of rule probability vectors Θ for which G is non-tight. Nederhof and Satta (2008) survey several algorithms for computing $Z(\Theta)$, and hence for determining whether a PCFG is tight.[1]

Non-tightness can arise in very simple PCFGs, such as the "Catalan" PCFG $S \to S\,S \,|\, a$. This grammar produces binary trees where all internal

[1] We found out that finding whether a PCFG is tight by directly inspecting the partition function value is less stable than using the method in Wetherell (1980). For this reason, we used Wetherell's approach, which is based on finding the principal eigenvalue of the matrix M.

nodes are labeled as S and the yield of these trees is a sequence of as. If the probability of the rule $S \rightarrow S\,S$ is greater than 0.5 then this PCFG is non-tight.

Perhaps the most straight-forward way to understand this non-tightness is to view this grammar as defining a branching process where an S can either "reproduce" with probability $\theta_{S \rightarrow S\,S}$ or "die out" with probability $\theta_{S \rightarrow a}$. When $\theta_{S \rightarrow S\,S} > \theta_{S \rightarrow a}$ the S nodes reproduce at a faster rate than they die out, so the derivation has a non-zero probability of endlessly rewriting (Atherya and Ney, 1972).

3 Bayesian inference for PCFGs

The goal of Bayesian inference for PCFGs is to infer a posterior distribution over the rule probability vectors Θ given observed data D. This posterior distribution is obtained by combining the likelihood $\mathrm{P}(D \mid \Theta)$ with a prior distribution $\mathrm{P}(\Theta)$ over Θ using Bayes Rule.

$$\mathrm{P}(\Theta \mid D) \propto \mathrm{P}(D \mid \Theta)\,\mathrm{P}(\Theta)$$

We now formally define the three approaches to handling non-tightness mentioned earlier:

the only-tight approach: we only permit priors where $\mathrm{P}(\boldsymbol{\Theta}^{\perp}) = 0$, i.e., we insist that the prior assign zero mass to non-tight rule probability vectors, so $Z = 1$. This means we can define:

$$\mathrm{P}(t \mid \Theta) = \mu_{\Theta}(t)$$

the renormalization approach: we renormalize non-tight PCFGs by dividing by the partition function:

$$\mathrm{P}(t \mid \Theta) = \frac{1}{Z(\Theta)}\,\mu_{\Theta}(t) \qquad (1)$$

the sink-element approach: we redefine our probability distribution so its domain is a set $\mathcal{T}' = \mathcal{T} \cup \{\perp\}$, where $\mathcal{T}$ is the set of (finite) trees generated by G and $\perp \notin \mathcal{T}$ is a new element that serves as a "sink state" to which the "missing mass" $1 - Z(\Theta)$ is assigned. Then we define:[2]

$$\mathrm{P}(t \mid \Theta) = \begin{cases} \mu_{\Theta}(t) & \text{if } t \in \mathcal{T} \\ 1 - Z(\Theta) & \text{if } t = \perp \end{cases}$$

With this in hand, we can now define the likelihood term. We consider two types of data D here. In the *supervised setting* the data D consists of a corpus of parse trees $D = (t_1, \ldots, t_n)$ where each tree t_i is generated by the PCFG G, so

$$\mathrm{P}(D \mid \Theta) = \prod_{i=1}^{n} \mathrm{P}(t_i \mid \Theta)$$

In the *unsupervised setting* the data D consists of a corpus of strings $D = (w_1, \ldots, w_n)$ where each string w_i is the yield of one or more trees generated by G. In this setting

$$\mathrm{P}(D \mid \Theta) = \prod_{i=1}^{n} \mathrm{P}(w_i \mid \Theta), \text{ where:}$$

$$\mathrm{P}(w \mid \Theta) = \sum_{t \in \mathcal{T} : \mathrm{yield}(t) = w} \mathrm{P}(t \mid \Theta)$$

4 The special case of linear PCFGs

One way to handle the issue of tightness is to identify a family of CFGs for which practically any parameter setting will yield a tight PCFG. This is the focus of this section, in which we identify a subset of CFGs, which are "almost everywhere" tight. This family of CFGs includes many of the CFGs used in NLP applications.

We cannot expect that a CFG will yield a tight PCFG for *any* assignment to the rule probabilities (i.e. that $\boldsymbol{\Theta}^{\perp} = \emptyset$). Even in simple cases, such as the grammar $S \rightarrow S|a$, the assignment of probability 1 to $S \rightarrow S$ and 0 to the other rule renders the S nonterminal useless, and places all of the probability

[2] This definition of a distribution over trees can be induced by a tight PCFG with a special $\perp$ symbol in its vocabulary. Given G, the first step is to create a tight grammar G_0 using the renormalization approach. Then, a new start symbol is added to G_0, S_0, and also rules $S_0 \rightarrow S$ (where S is the old start symbol in G_0) and $S_0 \rightarrow \perp$. The first rule is given probability $Z(\Theta)$ and the second rule is given probability $1 - Z(\Theta)$. It can be then readily shown that the new tight PCFG G_0 induces a distribution over trees just like in Eq. 3, only with additional S_0 on top of all trees.

mass on infinite structures of the form $S \rightarrow S \rightarrow S \rightarrow \ldots$.

However, we can weaken our requirement so that the cases in which parameter assignment yields a non-tight PCFG are rare, or have measure zero. To put it more formally, we say that a prior $\mathrm{P}(\Theta)$ is "tight almost everywhere for G" if

$$\mathrm{P}(\boldsymbol{\Theta}^{\perp}) = \int_{\Theta \in \boldsymbol{\Theta}^{\perp}} \mathrm{P}(\Theta)\, d\Theta = 0.$$

We now provide a sufficient condition (linearity) for CFGs under which they are tight almost everywhere with any continuous prior.

For a nonterminal $A \in N$ and $\beta \in (N \cup T)^*$, we use $A \Rightarrow^k \beta$ to denote that A can be re-written using a sequence of rules from R to the sentential form β in k derivation steps. We use $A \Rightarrow^+ \beta$ to denote that there exists a $k > 0$ such that $A \Rightarrow^k \beta$.

Definition 1 *A context-free grammar G is* linear *if there are no $A \in N$ such that*[3]

$$A \Rightarrow^+ \ldots A \ldots A \ldots .$$

Let $L(A) = \{w | A \Rightarrow^* w, w \in T^*\}$. Define $G(A)$ to be the grammar G where S is replaced by A. We assume G has no useless nonterminals, i.e. each nonterminal A participates in some complete tree derivation (but it could potentially have probability 0). Useless nonterminals can always be removed from a grammar without changing the language generated by the grammar.

Definition 2 *A nonterminal $A \in N$ in a probabilistic context-free grammar G with parameters Θ is* nonterminating *if:*

- *A is recursive: there is a β such that $A \Rightarrow^+ \beta$ and A appears in β.*
- $\mathrm{P}_{G(A)}(L(A)) = \sum_{w \in L(A)} \mathrm{P}_{G(A)}(w) = 0$.

Lemma 1 *A linear PCFG G with parameters Θ which does not have any nonterminating nonterminals is tight.*

[3]Note that this definition of linear CFGs deviates from the traditional definition, which states that a PCFG is linear if the right handside of each rule includes at most one nonterminal. The traditional definition implies Definition 1.

Proof: Our proof relies on the properties of a certain $|N| \times |N|$ matrix M where:

$$M_{AB} = \sum_{A \rightarrow \beta \in R_A} n(\beta, B)\ \theta_{A \rightarrow \beta}$$

where $n(\beta, B)$ is the number of appearances of the nonterminal B in the sequence β. M_{AB} is the expected number of B nonterminals generated from an A nonterminal in one single derivational step, so $[M^k]_{AB}$ is the expected number of B nonterminals generated from an A nonterminal in a k-step derivation (Wetherell, 1980).

Since M is a non-negative matrix, under some regularity conditions, the Frobenius-Perron theorem states that the largest eigenvalue of this matrix (in absolute value) is a real number. Let this eigenvalue be denoted by λ.

A PCFG is called "subcritical" if $\lambda < 1$ and supercritical if $\lambda > 1$. Then, in turn, a PCFG is tight if it is subcritical. It is not tight if it is supercritical. The case of $\lambda = 1$ is a borderline case that does not give sufficient information to know whether the PCFG is tight or not. In the Bayesian case, for a continuous prior such as the Dirichlet prior, this borderline case will have measure zero under the prior.

Now let $A \in N$. Since the grammar is linear, there is no derivation $A \Rightarrow^+ \ldots A \ldots A \ldots$. Therefore, any derivation of the form $A \Rightarrow^+ \ldots A \ldots$ includes A on the right hand-side exactly once. Because the grammar has no nonterminating nonterminals, the probability of such a derivation is strictly smaller than 1.

For each $A \in N$, define:

$$p_A = \sum_{\beta = \ldots A \ldots} \mathrm{P}(A \Rightarrow^{|N|} \beta | \Theta).$$

Since A is not useless, then $p_A < 1$. Therefore $q = \max_A p_A < 1$. Since any derivation of length k of the form $A \Rightarrow \ldots A \ldots$ can be decomposed to at least $\dfrac{k}{2|N|}$ cycles that start at a terminal $B \in N$ and end in the same nonterminal $B \in N$, it holds that:

$$[M^k]_{AA} \leq q^{\frac{k}{2|N|}} \overset{k \rightarrow \infty}{\rightarrow} 0.$$

This means that $\mathrm{trace}(M^k) \overset{k \rightarrow \infty}{\rightarrow} 0$. This means that the eigenvalue of M is strictly smaller than 1 (linear algebra), and therefore the PCFG is tight. ■

Proposition 1 *Any continuous prior* $\mathrm{P}(\Theta)$ *on a linear grammar* G *is tight almost everywhere for* G*.*

Proof: Let G be a linear grammar. With a continuous prior, the probability of G getting parameters from the prior which yield a useless non-terminal is 0 – it would require setting at least one rule in the grammar with rule probability which is exactly 1. Therefore, with probability 1, the parameters taken from the prior yield a PCFG which is linear and does not have nonterminating nonterminals. According to Lemma 1, this means the PCFG is tight. ■

Deciding whether a grammar G is linear can be done in polynomial time using the construction from Bar-Hillel et al. (1964). We can first eliminate the differences between nonterminals and terminal symbols by adding a rule $A \rightarrow c_A$ for each nonterminal $A \in N$, after extending the set of terminal symbols A with $\{c_A | A \in N\}$. Let G_A be the grammar G with the start symbol being replaced with A. We can then intersect the grammar G_A with the regular language $T^* c_A T^* c_A T^*$ (for each nonterminal $A \in N$). If for any nonterminal A the intersection is not the empty set (with respect to the language that the intersection generates), then the grammar is not linear. Checking whether the intersection is the empty set or not can be done in polynomial time.

We conclude this section by remarking that many of the models used in computational linguistics are in fact equivalent to linear PCFGs, so continuous Bayesian priors are almost everywhere tight. For example, HMMs and many kinds of "stacked" finite-state machines are equivalent to linear PCFGs, as are the example PCFGs given in Johnson et al. (2007) to motivate the MCMC estimation procedures.

5 Dirichlet priors

The first step in Bayesian inference is to specify a prior on Θ. In the rest of this paper we take $\mathrm{P}(\Theta)$ to be a product of Dirichlet distributions, with one distribution for each non-terminal $A \in N$, as this turns out to simplify the computations considerably. The prior is parameterized by a positive real valued vector α indexed by productions R, so each production probability $\theta_{A \rightarrow \beta}$ has a corresponding Dirichlet parameter $\alpha_{A \rightarrow \beta}$. As before, let R_A be the set of productions in R with left-hand side A, and let θ_A and α_A refer to the component subvectors of θ and α respectively indexed by productions in R_A. The Dirichlet prior $\mathrm{P}(\Theta \mid \alpha)$ is:

$$\mathrm{P}(\Theta \mid \alpha) = \prod_{A \in N} \mathrm{P}_D(\Theta_A \mid \alpha_A),$$

where

$$\mathrm{P}_D(\Theta_A \mid \alpha_A) = \frac{1}{C(\alpha_A)} \prod_{r \in R_A} \theta_r^{\alpha_r - 1} \quad \text{and}$$

$$C(\alpha_A) = \frac{\prod_{r \in R_A} \Gamma(\alpha_r)}{\Gamma(\sum_{r \in R_A} \alpha_r)}$$

where Γ is the generalized factorial function and $C(\alpha)$ is a normalization constant that does not depend on Θ_A.

Dirichlet priors are useful because they are *conjugate* to the multinomial distribution, which is the building block of PCFGs. Ignoring issues of tightness for the moment and setting $\mathrm{P}(t \mid \Theta) = \mu_\Theta(t)$, this means that in the supervised setting the posterior distribution $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ given a set of parse trees $\mathbf{t} = (t_1, \ldots, t_n)$ is also a product of Dirichlets distribution.

$$\begin{aligned}
\mathrm{P}(\Theta \mid \mathbf{t}, \alpha) &\propto \mathrm{P}(\mathbf{t} \mid \Theta)\, \mathrm{P}(\Theta \mid \alpha) \\
&\propto \left(\prod_{r \in R} \theta_r^{f_r(\mathbf{t})} \right) \left(\prod_{r \in R} \theta_r^{\alpha_r - 1} \right) \\
&= \prod_{r \in R} \theta_r^{f_r(\mathbf{t}) + \alpha_r - 1}
\end{aligned}$$

which is a product of Dirichlet distributions with parameters $\mathbf{f}(\mathbf{t}) + \alpha$, where $\mathbf{f}(\mathbf{t})$ is the vector of rule counts in $\mathbf{t}$ indexed by $r \in R$. We can thus write:

$$\mathrm{P}(\Theta \mid \mathbf{t}, \alpha) = \mathrm{P}(\Theta \mid \mathbf{f}(\mathbf{t}) + \alpha)$$

which makes it clear that the rule counts are directly added to the parameters of the prior to produce the parameters of the posterior.

6 Inference in the supervised setting

We first discuss Bayesian inference in the supervised setting, as inference in the unsupervised setting is based on inference for the supervised setting. For each of the three approaches to non-tightness we provide an algorithm that characterizes the posterior $\mathrm{P}(\Theta \mid \mathbf{t})$, where $\mathbf{t} = (t_1, \ldots, t_n)$ is a sequence of trees, by generating samples from that posterior. Our MCMC algorithms for the unsupervised setting build on these samplers for the supervised setting.

Input: Grammar G, vector of trees $\mathbf{t}$, vector of hyperparameters α, previous parameters Θ_0.
Result: A vector of parameters Θ
repeat
  draw θ from products of Dirichlet with hyperparameters $\alpha + \mathbf{f}(\mathbf{t})$
until Θ *is tight for* G;
return Θ

Algorithm 1: An algorithm for generating samples from $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ for the only-tight approach.

Input: Grammar G, vector of trees $\mathbf{t}$, vector of hyperparameters α, previous rule parameters Θ_0.
Result: A vector of parameters Θ
draw a proposal Θ^* from a product of Dirichlets with parameters $\alpha + \mathbf{f}(\mathbf{t})$.
draw a uniform number u from $[0, 1]$.
if $u < \min\{1, \left(Z(\Theta^{(i-1)})/Z(\Theta^*)\right)^n\}$ return Θ^*.
return Θ_0.

Algorithm 2: One step of Metropolis-Hastings algorithm for generating samples from $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ for the renormalization approach.

6.1 The only-tight approach

The "only-tight" approach requires that the prior assign zero mass to non-tight rule probability vectors $\boldsymbol{\Theta}^{\perp}$. One way to define such a distribution is to restrict the domain of an existing prior distribution with the set of tight Θ and renormalize. In more detail, if $\mathrm{P}(\Theta)$ is a prior over rule probabilities, then its renormalization is the prior P' defined as:

$$\mathrm{P}'(\Theta) = \frac{\mathrm{P}(\Theta) I(\Theta \notin \boldsymbol{\Theta}^{\perp})}{Z(\boldsymbol{\Theta}^{\perp})}. \tag{2}$$

where $Z(\boldsymbol{\Theta}^{\perp}) = \int_{\Theta} \mathrm{P}(\Theta) I(\Theta \notin \boldsymbol{\Theta}^{\perp}) d\Theta$.

Perhaps surprisingly, it turns out that if $\mathrm{P}(\Theta)$ belongs to a family of conjugate priors, then $\mathrm{P}'(\Theta)$ also belongs to a (different) family of conjugate priors as well.

Proposition 2 *Let* $\mathrm{P}(\Theta|\alpha)$ *be a prior with hyperparameters* α *over the parameters of* G *such that* P *is conjugate to the grammar likelihood. Then* P'*, defined in Eq.* 2*, is conjugate to the grammar likelihood as well.*

Proof: Assume that trees $\mathbf{t}$ are observed, and the prior over the grammar parameters is the prior defined in Eq. 2. Therefore, the posterior is:

$$\begin{aligned}\mathrm{P}(\Theta|\mathbf{t}, \alpha) &\propto \mathrm{P}'(\Theta|\alpha) p(\mathbf{t}|\Theta) \\ &= \frac{\mathrm{P}(\Theta|\alpha) p(\mathbf{t}|\Theta) I(\Theta \notin \boldsymbol{\Theta}^{\perp})}{Z(\boldsymbol{\Theta}^{\perp})} \\ &\propto \frac{\mathrm{P}(\Theta|\mathbf{t}, \alpha) I(\Theta \notin \boldsymbol{\Theta}^{\perp})}{Z(\boldsymbol{\Theta}^{\perp})}.\end{aligned}$$

Since $\mathrm{P}(\Theta|\alpha)$ is a conjugate prior to the PCFG likelihood, then there exists $\alpha' = \alpha'(\mathbf{t})$ such that $\mathrm{P}(\Theta|\mathbf{t}, \alpha) = \mathrm{P}'(\Theta|\alpha')$. Therefore:

$$\mathrm{P}(\Theta|\mathbf{t}, \alpha) \propto \frac{\mathrm{P}(\Theta|\alpha') I(\Theta \notin \boldsymbol{\Theta}^{\perp})}{Z(\boldsymbol{\Theta}^{\perp})}.$$

which exactly equals $\mathrm{P}'(\Theta|\alpha')$. ■

Sampling from the posterior over the parameters given a set of trees $\mathbf{t}$ is therefore quite simple when assuming the base prior being renormalized is a product of Dirichlets. Algorithm 1 samples from a product of Dirichlets distribution with hyperparameters $\alpha + \mathbf{f}(\mathbf{t})$ repeatedly, each time checking and rejecting the sample until we obtain a tight PCFG.

The more mass the Dirichlet distribution with hyperparameters $\alpha + \mathbf{f}(\mathbf{t})$ puts on non-tight PCFGs, the more rejections will happen. In general, if the probability mass on non-tight PCFGs is $q_{\perp}$, then it would require, on average $1/(1 - q_{\perp})$ samples from this distribution in order to obtain a tight PCFG.

6.2 The renormalization approach

The renormalization approach modifies the likelihood function instead of the prior. Here we use a product of Dirichlets prior $\mathrm{P}(\Theta \mid \alpha)$ on rule probability vectors Θ, but the presence of the partition function $Z(\Theta)$ in Eq. 1 means that the likelihood is no longer conjugate to the prior. Instead we have:

Input: Grammar G, vector of trees $\mathbf{t}$, vector of hyperparameters α, previous parameters Θ_0.
Result: A vector of parameters Θ
draw Θ from products of Dirichlet with hyperparameters $\alpha + \mathbf{f}(\mathbf{t})$
return Θ

Algorithm 3: An algorithm for generating samples from $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ for the sink-state approach.

$$\mathrm{P}(\Theta \mid \mathbf{t}) = \prod_{i=1}^{n} \frac{\mu_\Theta(t_i)}{Z(\Theta)} \mathrm{P}(\Theta \mid \alpha)$$
$$\propto \frac{1}{Z(\Theta)^n} \mathrm{P}(\Theta \mid \alpha + \mathbf{f}(\mathbf{t})). \quad (3)$$

Note that the factor $Z(\Theta)$ depends on Θ, and therefore cannot be absorbed into the constant. Algorithm 2 describes a Metropolis-Hastings sampler for sampling from the posterior in Eq. 3 that uses a product of Dirichlets with parameters $\alpha + \mathbf{f}(\mathbf{t})$ as a proposal distribution.

In our experiments, we use the algorithm from Nederhof and Satta (2008) to compute the partition function which is needed in Algorithm 2.

6.3 The "sink element" approach

The "sink element" approach does not affect the likelihood (since the probability of a tree t is just the product of the probabilities of the rules used to generate it), nor does it require a change to the prior. (The sink element $\perp$ is not a member of the set of trees $\mathcal{T}$, so it cannot appear in the data $\mathbf{t}$).

This means that the conjugacy argument given at the bottom of section 5 holds in this approach, so the posterior $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ is a product of Dirichlets with parameters $\mathbf{f}(\mathbf{t}) + \alpha$. Algorithm 3 gives a sampler for $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ for the sink element approach.

7 Inference in the unsupervised setting

Johnson et al. (2007) provide two Markov chain Monte Carlo algorithms for Bayesian inference for PCFG rule probabilities in the unsupervised setting (i.e., where the data consists of a corpus of strings $\mathbf{w} = (w_1, \ldots, w_n)$ alone). The algorithms we give here are based on their Gibbs sampler, which in each iteration first samples parse trees $\mathbf{t} = (t_1, \ldots, t_n)$, where each t_i is a parse for w_i, from $\mathrm{P}(\mathbf{t} \mid \mathbf{w}, \Theta)$, and then samples Θ from $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$.

Notice that the conditional distribution $\mathrm{P}(t \mid w, \Theta)$ is unaffected in each of our three approaches (the partition functions cancel in the renormalization approach), so the algorithm for sampling from $\mathrm{P}(\mathbf{t} \mid \mathbf{w}, \Theta)$ given by Johnson et al. applies in each of our three approaches as well.

Johnson et al. ignored tightness and assumed that $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ is a product of Dirichlets with parameters $\mathbf{f}(\mathbf{t}) + \alpha$. As we noted in section 6.3, this assumption holds for the sink-state approach to non-tightness, so their sampler is in fact correct for the sink-state approach.

Input: Grammar G, vector of hyperparameters α, vector of strings $\mathbf{w} = (w_1, \ldots, w_n)$, previous rule parameters Θ_0.
Result: A vector of parameters Θ
for $i \leftarrow 1$ **to** n **do**
  draw t_i from $\mathrm{P}(t_i | w_i, \Theta_0)$
end
use Algorithm 2 to sample Θ given G, $\mathbf{t}$, α and Θ_0
return Θ

Algorithm 4: One step of the Metropolis-within-Gibbs sampler for the renormalization approach.

In fact, we obtain samplers for the unsupervised setting for each of our approaches by "plugging in" the corresponding sampling algorithm (Eq. 1–3) for $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ into the generic Gibbs sampler framework of Johnson et al.

The one complication is that because we use a Metropolis-Hastings procedure to generate samples from $\mathrm{P}(\Theta \mid \mathbf{t}, \alpha)$ in the renormalization approach, we use the Metropolis-within-Gibbs procedure given in Algorithm 4 (Robert and Casella, 2004).

8 The expressive power of the three approaches

Probably the most important question to ask with respect to the three different approaches to non-tightness is whether they differ in terms of expressive power. Clearly the three approaches differ in terms of the grammars they admit (the only-tight approach requires the prior to only assign non-zero probability to tight PCFGs, while the other two approaches permit the prior to assign non-zero probability to non-tight PCFGs as well). However, if we regard a grammar as merely a device for defining a distribution over trees and a prior as defining a distribution over distributions over trees, it is reasonable to ask whether the class of distributions over distributions of trees that each of these approaches define are the same or differ. We believe, but have not proved, that all three approaches define the same class of distributions over distributions of trees in the

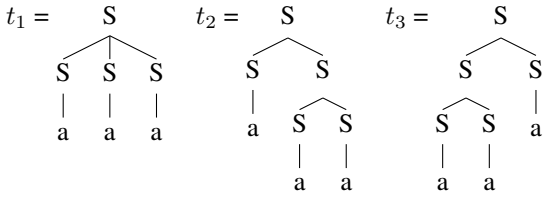
following sense: any prior used with one of the approaches can be transformed into a different prior that can be used with one of the other approaches, and yield identical posterior over trees conditioned on a string, marginalizing out the parameters.

This does not mean that the three approaches are equivalent, however. In this section we provide a grammar such that with a uniform prior over rule probabilities, the conditional distribution over trees given a fixed string varies under each of the three different approaches.

The grammar we consider has three rules $S \rightarrow S\,S\,S|S\,S|a$ with probabilities θ_1, θ_2 and $1 - \theta_1 - \theta_2$, respectively. The Θ parameters are required to satisfy $\theta_1 + \theta_2 \leq 1$ and $\theta_i \geq 0$ for $i = 1, 2$.

We compute the posterior distribution over parse trees for the string $w = a\,a\,a$.

The grammar generates three parse trees for w_1, namely:



The partition function Z for this grammar is the smallest positive root of the cubic equation:

$$Z = \theta_1 Z^3 + \theta_2 Z^2 + (1 - \theta_1 - \theta_2)$$

We used Mathematica to find an analytic solution for Z in this equation, obtaining not only an expression for the partition function $Z(\Theta)$ but also identifying the non-tight region $\boldsymbol{\Theta}^{\perp}$.

In order to compute $\mathrm{P}(t_1|w)$, we used Mathematica to first compute the following quantities:

$$q_{\text{sinkElement}}(t_i) = \int_{\Theta} \mu_{\Theta}(t_i)\, d\Theta$$

$$q_{\text{tightOnly}}(t_i) = \int_{\Theta} \mu_{\Theta}(t_i)\, I(\Theta \notin \boldsymbol{\Theta}^{\perp})\, d\Theta$$

$$q_{\text{tightOnly}}(t_i) = \int_{\Theta} \mu_{\Theta}(t_i)/Z(\Theta)\, d\Theta$$

where $i \in \{1, 2, 3\}$. We used Mathematica to analytically compute $q(t_i)$ for each approach and each $i \in \{1, 2, 3\}$. Then it's easy to show that:

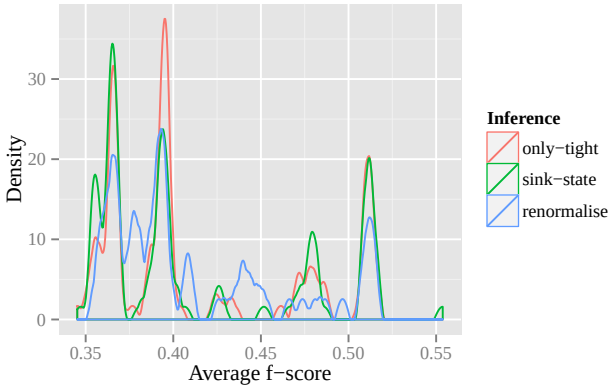


Figure 1: The density of the F_1-scores with the three approaches. The prior used is a symmetric Dirichlet with $\alpha = 0.1$.

$$\mathrm{P}(t_i \mid w) = \frac{q(t_i)}{\sum_{i'=1}^{3} q(t_{i'})}$$

where the q used is based on the approach to tightness desired. For the sink-element approach, $\mathrm{P}(t_1|w) = \frac{7}{11} \approx 0.636364$. For the only-tight approach $\mathrm{P}(t_1|w) = \frac{11179}{17221} \approx 0.649149$. For the renormalization approach the analytic expression is too complex to include in this paper, but it approximately equals 0.619893. A log of our Mathematica calculations is available at `http://www.cs.columbia.edu/~scohen/acl13tightness-mathematica.pdf`, and we confirmed these results to three decimal places using the samplers described above (which required 10^7 samples per approach).

While the differences between these conditional probabilites are not great, the conditional probabilities are clearly different, so the three approaches do in fact define different distributions over trees under a uniform prior on rule probabilities.

9 Empirical effects of the three approaches in unsupervised grammar induction

In this section we present experiments using the three samplers just described in an unsupervised grammar induction problem. Our goal here is not to improve the state-of-the-art in unsupervised grammar induction, but to try to measure empirical dif-

ferences in the estimates produced by the three different approaches to tightness just described. The bottom line of our experiments is that we could not detect any significant difference in the estimates produced by samplers for these three different approaches.

In our experiments we used the English Penn treebank (Marcus et al., 1993). We use the part-of-speech tag sequences of sentences shorter than 11 words in sections 2–21. The grammar we use is the PCFG version of the dependency model with valence (Klein and Manning, 2004), as it appears in Smith (2006).

We used a symmetric Dirichlet prior with hyperparameter $\alpha = 0.1$. For each of the three approaches for handling tightness, we ran 100 times the samplers in §7, each for 1,000 iterations. We discarded the first 900 sweeps of each run, and calculated the F_1-scores of the sampled trees every 10th sweep from the last 100 sweeps. For each run we calculated the average F_1-score over the 10 sweeps we evaluated. We thus have 100 average F_1-scores for each of the samplers.

Figure 1 plots the density of F_1 scores (compared to the gold standard) resulting from the Gibbs sampler, using all three approaches. The mean value for each of the approaches is 0.41 with standard deviation 0.06 (only-tight), 0.41 with standard deviation 0.05 (renormalization) and 0.42 with standard deviation 0.06 (sink element). In addition, the only-tight approach results in an average of 437 (s.d., 142) rejected proposals in 1,000 samples, while the renormalization approach results in an average of 232 (s.d., 114) rejected proposals in 1,000 samples. (It's not surprising that the only-tight approach results in more rejections as it keeps proposing new Θ until a tight proposal is found, while the renormalization approach simply uses the old Θ).

We performed two-sample Kolmogorov-Smirnov tests (which are non-parametric tests designed to determine if two distributions are different; see DeGroot, 1991) on each of the three pairs of 100 F_1-scores. None of the tests were close to significant; the p-values were all above 0.5. Thus our experiments provided no evidence that the samplers produced different distributions over trees, although it's reasonable to expect that these distributions do indeed differ.

In terms of running time, our implementation of the renormalization approach was several times slower than our implementations of the other two approaches because we used the naive fixed-point algorithm to compute the partition function: perhaps this could be improved using one of the more sophisticated partition function algorithms described in Nederhof and Satta (2008).

10 Conclusion

In this paper we characterized the notion of an almost everywhere tight grammar in the Bayesian setting and showed it holds for linear CFGs. For non-linear CFGs, we described three different approaches to handle non-tightness. The "only-tight" approach restricts attention to tight PCFGs, and perhaps surprisingly, we showed that conjugacy still obtains when the domain of a product of Dirichlets prior is restricted to the subset of tight grammars. The renormalization approach involves renormalizing the PCFG measure μ over trees when the grammar is non-tight, which destroys conjugacy with a product of Dirichlets prior. Perhaps most surprisingly of all, the sink-element approach, which assigns the missing mass in non-tight PCFG to a sink element $\perp$, turns out to be equivalent to existing practice where tightness is ignored.

We studied the posterior distributions over trees induced by the three approaches under a uniform prior for a simple grammar and showed that they differ. We leave for future work the important question of whether the classes of distributions over distributions over trees that the three approaches define are the same or different.

We described samplers for the supervised and unsupervised settings for each of these approaches, and applied them to an unsupervised grammar induction problem. (The code for the unsupervised samplers is available from `http://web.science.mq.edu.au/~mjohnson`).

We could not detect any difference in the posterior distributions over trees produced by these samplers, despite devoting considerable computational resources to the problem. This suggests that for these kinds of problems at least, tightness is not of practical concern for Bayesian inference of PCFGs.

Acknowledgements

We thank the anonymous reviewers and Giorgio Satta for their valuable comments. Shay Cohen was supported by the National Science Foundation under Grant #1136996 to the Computing Research Association for the CIFellows Project, and Mark Johnson was supported by the Australian Research Council's Discovery Projects funding scheme (project numbers DP110102506 and DP110102593).

References

K. B. Atherya and P. E. Ney. 1972. *Branching Processes*. Dover Publications.

Y. Bar-Hillel, M. Perles, and E. Shamir. 1964. On formal properties of simple phrase structure grammars. *Language and Information: Selected Essays on Their Theory and Application*, pages 116–150.

T. L. Booth and R. A. Thompson. 1973. Applying probability measures to abstract languages. *IEEE Transactions on Computers*, C-22:442–450.

Z. Chi and S. Geman. 1998. Estimation of probabilistic context-free grammars. *Computational Linguistics*, 24(2):299–305.

Z. Chi. 1999. Statistical properties of probabilistic context-free grammars. *Computational Linguistics*, 25(1):131–160.

S. B. Cohen and N. A. Smith. 2012. Empirical risk minimization for probabilistic grammars: Sample complexity and hardness of learning. *Computational Linguistics*, 38(3):479–526.

M. H. DeGroot. 1991. *Probability and Statistics (3rd edition)*. Addison-Wesley.

M. Johnson, T. L. Griffiths, and S. Goldwater. 2007. Bayesian inference for PCFGs via Markov chain Monte Carlo. In *Proceedings of NAACL*.

D. Klein and C. D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of ACL*.

K. Kurihara and T. Sato. 2006. Variational Bayesian grammar induction for natural language. In *8th International Colloquium on Grammatical Inference*.

K. Lari and S.J. Young. 1990. The estimation of Stochastic Context-Free Grammars using the Inside-Outside algorithm. *Computer Speech and Language*, 4(35-56).

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19:313–330.

M.-J. Nederhof and G. Satta. 2008. Computing partition functions of PCFGs. *Research on Language and Computation*, 6(2):139–162.

C. P. Robert and G. Casella. 2004. *Monte Carlo Statistical Methods*. Springer-Verlag New York.

N. A. Smith. 2006. *Novel Estimation Methods for Unsupervised Discovery of Latent Structure in Natural Language Text*. Ph.D. thesis, Johns Hopkins University.

C. S. Wetherell. 1980. Probabilistic languages: A review and some open questions. *Computing Surveys*, 12:361–379.