4YP

Rebecca Dawes

October 20, 2013

0.0.1 Literature Review

Currently just a load of notes from reading the papers...

A Hierarchical Bayesian Language Model Based on Pitman Yor Processes - *Teh* 2006

n-gram models approximate the distribution over sentences using the conditional distribution of each work given a context consisting of only the previous *n* - 1 words:

$$P(sentence) \approx \prod_{i=1}^{T} P(word_i | word_{i-n+1}^{i-1}) \tag{1}$$

Even for small *n*, the number of parameters is huge due to the large vocabulary size, therefore direct maximum likelihood parameter fitting severely over fits to the training data.

This paper proposes a novel language model based on a hierarchical Bayesian model where each hidden variable is distributed according to a Pitman Yor process (a nonparametric generalisation of the Dirichlet distribution).

Pitman Yor Process: Let W be a fixed and finite vocabulary of V words. For each word $w \in W$ let $G(w)$ be the (to be estimated) probability of w, and let $G = [G(w)]_{w \in W}$ be the vector of word probabilities. We place a Pitman Yor process prior on G: $G \sim PY(d, \theta, G_0)$ where discount parameter $0 \leq d < 1$, strength parameter $\theta > -d$ and a mean vector $G_0 = [G_0(w)]_{w \in W}$. $G_0(w)$ is the a priori probability of word w (usually $G_0(w) = 1/V$). θ and d control the amount of variability around G_0. When $d = 0$, the Pitman Yor process reduces to a Dirichlet distribution with parameters θG_0.

Let $x_1, x_2, ...$ be a sequence of words drawn independently and identically (i.i.d.) from G. The first word x_1 is assigned a value of the first draw y_1 from G_0. Let t be the current number of draws from G_0 (currently $t = 1$), c_k be the number of words assigned the value of draw y_k (currently $c_1 = 1$), and $c_\cdot = \sum_{k=1}^{t} c_k$ be the current number of draws from G. For each subsequent word $x_{c_\cdot+1}$, we either assign it the value of a previous draw y_k with probability $\frac{c_k - d}{\theta + c_\cdot}$ (increment c_k; set $x_{c_\cdot+1} \leftarrow y_k$), or we assign it the value of a new draw from G_0 with probability $\frac{\theta + dt}{\theta + c_\cdot}$ (increment t; set $c_t = 1$; draw $y_t \sim G_0$; set $x_{c_\cdot+1} < y_t$).

The more words that have been assigned to a draw from G_0, the more likely subsequent words will be assigned to the draw. Also, the more we draw from G_0, the more likely a new word will be assigned to a new draw from G_0. These effects together produce a power-law distribution where many unique words are observed, most of them rarely. θ controls the overall numbers of unique words, while d controls the asymptotic growth of the number of unique words. This procedure for generating words drawn from G is

the Chinese Restaurant Process. (Consider a sequence of customers (corresponding to the words drawn from G) visiting a Chinese restaurant with an unbounded number of tables (corresponding to the draws from G_0), each of which can accommodate an unbounded number of customers. The first customer sits at the first table, and each subsequent customer either joins an already occupied table (assign the word to the corresponding draw from G_0), or sits at a new table (assign the word to a new draw from G_0).)

Hierarchical Pitman Yor Language Models: Given a context $\mathbf{u}$, let $G_{\mathbf{u}}(w)$ be the probability of the current word taking on value w. We use a Pitman Yor process as the prior for $G_{\mathbf{u}}[G_{\mathbf{u}}(w)]_{w \in W}$, in particular,

$$G_{\mathbf{u}} \sim PY(d_{|\mathbf{u}|}, \theta_{|\mathbf{u}|}, G_{\pi(\mathbf{u})}) \tag{2}$$

where $\pi(\mathbf{u})$ is the suffix of $\mathbf{u}$ consisting of all but the earliest word. As we don't know $G_{\pi(\mathbf{u})}$, we recursively place a prior over $G_{\pi(\mathbf{u})}$ using Equation 2, but now with parameters $\theta_{|\pi(\mathbf{u})|}, d_{|\pi(\mathbf{u})|}$ and mean vector $G_{\pi(\pi(\mathbf{u}))}$ instead. This is repeated until we get to $G_\emptyset$, the vector of probabilities over the current word given the empty context $\emptyset$. Finally we place a prior on $G_\emptyset$:

$$G_\emptyset \sim PY(d_0, \theta_0, G_0) \tag{3}$$

where G_0 is the global mean vector, given a uniform value of $G_0(w) = 1/V$ for all $w \in W$. Finally, we place a uniform prior on the discount parameters and a Gamma(1,1) prior on the strength parameters. The total number of parameters in the model is $2n$.

The prior is structured as a suffix tree of depth n, where each node corresponds to a context consisting of up to $n-1$ words, and each child corresponds to adding a different word to the beginning of the context (words appearing earlier in a context have (a priori) the least importance in modelling the probability of the current word).

Hierarchical Chinese Restaurant Process: We may treat each $G_{\mathbf{u}}$ as a distribution over the current word. Since $G_{\mathbf{u}}$ is Pitman Yor process distributed, we can draw words from it using the Chinese Restaurant process. The same applies for $G_{\pi(\mathbf{u})}$. This is recursively applied until we need draws from the global mean distribution G_0, which is easy since it is just uniformally distributed.

For each context $\mathbf{u}$, there is a sequence of words $x_{\mathbf{u}1}, x_{\mathbf{u}2}, ...$ drawn i.i.d. from $G_{\mathbf{u}}$ and another sequence of words $y_{\mathbf{u}1}, y_{\mathbf{u}2}, ...$ drawn i.i.d. from the parent distribution $G_{\pi(\mathbf{u})}$. We use l to index draws from $G_{\mathbf{u}}$ and k to index draws from $G_{\pi(\mathbf{u})}$. $t_{\mathbf{u}wk} = 1$ if $y_{\mathbf{u}k}$ takes on value w, and $t_{\mathbf{u}wk} = 0$ otherwise. Each word

$x_{\mathbf{u}l}$ is assigned to one of the draws $y_{\mathbf{u}k}$ from $G_{\pi(\mathbf{u})}$. If $y_{\mathbf{u}k}$ takes on value w define $c_{\mathbf{u}wk}$ as the number of words $x_{\mathbf{u}l}$ drawn from $G_{\mathbf{u}}$ assigned to $y_{\mathbf{u}k}$, otherwise let $c_{\mathbf{u}wk} = 0$. Finally, we denote marginal counts by dots, e.g. $c_{\mathbf{u}.k}$ is the number of $x_{\mathbf{u}l}$s assigned with the value of $y_{\mathbf{u}k}$, $c_{\mathbf{u}w.}$ is the number of $x_{\mathbf{u}l}$s with value w, and $t_{\mathbf{u}..}$ is the current number of draws $y_{\mathbf{u}k}$ from $G_{\pi(\mathbf{u})}$.

$$\begin{cases} t_{\mathbf{u}w.} = 0 & if\ c_{\mathbf{u}w.} = 0 \\ 1 \leq t_{\mathbf{u}w.} \leq c_{\mathbf{u}w.} & if\ c_{\mathbf{u}w.} > 0 \end{cases} \tag{4}$$

$$c_{\mathbf{u}w.} = \sum_{\mathbf{u}':\pi(\mathbf{u}')=\mathbf{u}} t_{\mathbf{u}'w.} \tag{5}$$

The more a word w has been drawn in context **u**, the more likely it will be drawn again in context **u**. Word w will be reinforced for other contexts that share a common suffix with **u**, with the probability of drawing w increasing as the length of the common suffix increases (w will be more likely under the context of the common suffix as well).

Markov Chain Monte Carlo sampling based inference scheme for the hierarchical Pitman Yor language model: Training data D consists of the number of occurrences $c_{\mathbf{u}w.}$ of each word w after each context **u** of length exactly $n-1$. This corresponds to observing word w drawn $c_{\mathbf{u}w.}$ times from $G_{\mathbf{u}}$. Given D, we are interested in the posterior distribution over the latent vectors $\mathbf{G} = \{G_{\mathbf{v}} \colon all\ contexts\ \mathbf{v}\}$ and parameters $\mathbf{\Theta} = \{\theta_m, d_m \colon 0 \leq m \leq n-1\}$:

$$p(\mathbf{G}, \mathbf{\Theta}|D) = p(\mathbf{G}, \mathbf{\Theta}, D)/p(D) \tag{6}$$

The hierarchical Chinese Restaurant process marginalises out each $G_{\mathbf{u}}$, replacing it with the seating arrangement in the corresponding restaurant, $S_{\mathbf{u}}$. Let $\mathbf{S} = \{S_{\mathbf{v}} : all\ context\ \mathbf{v}\}$. Therefore we are instead interested in the posterior over seating arrangements:

$$p(\mathbf{S}, \mathbf{\Theta}|D) = p(\mathbf{S}, \mathbf{\Theta}, D)/p(D) \tag{7}$$

The probability of a test word w after a context **u** is given by:

$$p(w|\mathbf{u}, D) = \int p(w|\mathbf{u}, \mathbf{S}, \mathbf{\Theta}) p(\mathbf{S}, \mathbf{\Theta}|D) d(\mathbf{S}, \mathbf{\Theta}) \tag{8}$$

where the first probability on the right is the predictive probability under a particular setting of seating

arrangements **S** and parameters $\mathbf{\Theta}$, and the overall predictive probability is obtained by averaging this with respect to the posterior over **S** and $\mathbf{\Theta}$ (second probability on right).

A Hierarchical, Hierarchical Pitman Yor Process Language Model - *Wood, Teh*

Normal Pitman Yor process language modelling: Distribution over words following a particular context:

$$w_t|w_{t-1}, w_{t-2} \sim G^0_{\{w_{t-2},w_{t-1}\}} \tag{9}$$

(context length 2) is a random distribution:

$$G^0_{\{w_{t-2},w_{t-1}\}} \sim PY(d_2, \alpha_2, H) \tag{10}$$

where $PY(d, \alpha, H)$ is a Pitman Yor process with a distributed count d, concentration α and base distribution H. When the base distribution is the distribution over words following the same context with one fewer antecedents, $H = G^0_{\{w_{t-1}\}}$, and $G^0_{\{w_{t-1}\}}$ is a random distribution which is distributed according to a Pitman Yor process with another more general base distribution, this is a *Hierarchical Pitman Yor Process (HPYP)*. This "recursion" continues until the set of antecedent words is empty - "root" PY process is typically given a base distribution which is uniform over the corpus vocabulary.

Graphical Pitman Yor Process: Assume we have corpora from domains D_1, D_2 and we take the approach common to Bayesain domain adaptation, specifying a hierarchical model that allows statistical sharing between the models of each sorpus. The model has the same form as the HPYP, except that the base distribution of every PY process in the hierarchy is different:

$$G^{D_i}_{\{w_{t-2},w_{t-1}\}} \sim PY(d_j, \theta_j, \pi G^{D_i}_{\{w_{t-1}\}} + (1-\pi) G^0_{\{w_{t-2},w_{t-1}\}}) \tag{11}$$

The distribution over words in a particular context in a particular domain could reasonably back off to a distribution over words given a shorted context in the same domain or a distribution over words given the whole context in a general domain. Here π is the parameter that controls how closely the base distribution is tied to the domain specific model or the general model.

Dirichlet Process - *Teh* 2010

The Dirichlet Process is a stochastic process used in Bayesian nonparametric models of data. It is a distribution over distributions and has Dirichlet distributed finite dimensional marginal distributions. Distributions drawn from a Dirichlet process are discrete, but cannot be described using a finite number of parameters (nonparametric).

DP is a stochastic process whose sample paths are probability measures with probability 1. Stochastic processes are distributions over function spaces, with sample paths being random functions drawn from the distribution. DP is a distribution over probability measures, which are functions with certain special properties which allow them to be interpreted as distributions over some probability space Θ. Therefore draws from a DP can be interpreted as random distributions. DP is an infinite dimensional generalisation of Dirichlet distributions.

The Sequence Memoizer - *Wood, Gasthaus, Archambeau, James, Teh* 2011

"The Sequence Memoizer is a new hierarchical Bayesian model for discrete sequence data that captures long range dependencies and power-law characteristics, while remaining computationally attractive."

Let Σ be the set of symbols that can occur in some sequence. Suppose that we are given a sequence $\mathbf{x} = x_1, x_2, ..., x_T$ of symbols from Σ and want to estimate the probability that the next symbol takes a particular value $s \in \Sigma$.

Using Relative Frequency: i.e. if s occurs frequently in **x** we expect its probability of appearing next to be high as well. $N(s)$ is number os occurrences of s in **x**. Probability of s being next symbol is $G(s) = N(s)/T = N(s)/\sum_{s' \in \Sigma} N(s')$. G is a discrete distribution over the elements of Σ: it assigns a non-negative number $G(s)$ to each symbol s signifying the probability of observing s with the numbers summing to 1 over Σ. This approach is reasonable only if the process generating **x** has no history dependence. **Taking into account context:** If the last symbol in **x** is **u**, then we can estimate the probability of the next symbol being s by counting the number of times s occurs after **u** in **x**:

$$G_{\mathbf{u}}(s) = \frac{N(\mathbf{u}s)}{\sum_{s' \in \Sigma} N(\mathbf{u}s')} \tag{12}$$

is the estimated probability go s occurring after **u**, where $N(\mathbf{u}s)$ is the number os occurrences of the subsequence $\mathbf{u}s$ in **x**. $G_{\mathbf{u}}$ is a discrete distribution over the symbols in Σ, but it is a conditional distribution as the probability assigned to each symbol s depends on the context **u**.

Maximum Likelihood: An optimistic procedure - assumes **x** is an accurate reflection of the true underlying process that generated it, to the ML parameters will be an accurate estimate of the true parameters - leads to overfitting. if **u** is long, the chance that it nevers occurs in **x** is high, therefore the denominator in Equation 12 is 0. If **u** did occur in **x**, a high probability will be assigned to the symbols that followed **u** and zero probability to all other symbols. The amount of data in **x** is insufficient to characterise the conditional distribution $G_{\mathbf{u}}$. Avoid this by making a fixed-order Markov assumption and restricting to estimating collections of distributions conditioned on short contexts.

Bayesian Modelling: conservative compared to ML approach. Uncertainty in estimation is taken into account by treating the parameters Θ as random, with a prior distribution $p(\Theta)$ reflecting the prior knowledge we have about the true data generating process. $p(\Theta|\mathbf{x}) = p(\Theta)p(\mathbf{x}|\Theta)/p(\mathbf{x})$. Computations such as prediction are then done taking into account the a posteriori uncertainty about the underlying parameters.

Natural sequence data often exhibits power-law properties. Conditional distributions of similar contexts tend to be similar themselves, particularly in the sense that recency matters.

Power-law scaling: There are a small number of words that occur disproportionately frequently and a very large number of rare words that, although each occurs rarely, when taken together make up a large proportion of the language. ML estimates the probabilities of the frequently occurring symbols well, since they are based on many observations of the symbols. The estimates of the rare symbols will be bad.If a symbol did not occur in our sequence, our estimate of its probability is 0, while the estimate of a rare symbol that occurred by chance will be too high.

Using Pitman Yor:

$$p(\mathbf{x}_{T+1} = s|\mathbf{x}) = \int (\mathbf{x}_{T+1} = s|G)p(G|\mathbf{x})dG = \mathbb{E}[G(s)] \tag{13}$$

where $\mathbb{E}$ stands for expectation wrt posterior distribution $p(G|\mathbf{x})$. This equation doesn't always have an analytic solution therefore use numerical integration approaches, including sampling and Monte Carlo integration.

In addition to the counts $\{N(s')\}_{s'\in\Sigma}$ assume that there is another set of random "counts" $\{M(s')\}_{s'\in\Sigma}$ satisfying $1/leqM(s')/leqN(s')$ if $N(s') >)$ and $M(s') > 0$ otherwise. Probability of a symbol s occurring next:

$$\mathbb{E}[G(s)] = \mathbb{E}\left[\frac{N(s) - \alpha M(s) + \sum_{s' \in \Sigma} \alpha M(s') G_0(s)}{\sum_{s' \in \Sigma} N(s')}\right] \tag{14}$$

Each $M(s)$ reduces the count $N)s)$ by $\alpha M(s)$. The total amount subtracted is redistributed across all symbols in Σ proportionally according to the symbols' probability under the base distribution G_0. Therefore non-zero counts are usually reduced, with larger counts typically reduced by a larger amount. This mitigates the overestimation of probabilities of rare symbols that happen to appear by chance. For symbols that did not appear at all, the estimates of their probabilities are pulled upward from zero, mitigating underestimation.

Context trees: If two contexts are similar, then the corresponding conditional distributions over the symbols that follow those context will tend to be similar as well. Similarity is defined by overlapping contextual suffixes.

Using a hierarchical Bayesian model and considering only fixed, finite length contexts, we are making an n^{th} order Markov assumption - each symbol depends only on the last *n* observed symbols. This assumption dictates that distributions are not only similar, but equal among context whose suffixes overlap in their last *n* symbols.

To construct a context tree: Arrange the context $\mathbf{u}$ (and the associated distributions $G_{\mathbf{u}}$) in a tree where the parent of a node $\mathbf{u}$, denoted by $\sigma(\mathbf{u})$, is given by its longest proper suffix (i.e. $\mathbf{u}$ with its first symbol from the left removed). Since we are making an n^{th} order Markov assumption, it is sufficient to consider only the contexts $\mathbf{u} \in \Sigma^*_{\mathbf{u}} = \{u' \in \Sigma^* : |\mathbf{U}'| \leq n\}$ of length at most *n*. The resulting context tree has height *n* and the total number of nodes in the tree grows exponentially in *n*. The memory complexity of models built on such context trees usually grows too large and too quickly for reasonable values of *n* and $|\Sigma|$.

HPYP:

$$G_\varepsilon \sim PY(\alpha_0, G_0) \tag{15}$$

$$G_{\mathbf{u}} | G_{\sigma(\mathbf{u})} \sim PY(\alpha_{|\mathbf{u}|}, G_{\sigma(\mathbf{u})}) \quad for\ all\ \mathbf{u} \in \Sigma^*_n / \varepsilon \tag{16}$$

$$x_i | \mathbf{x}_{i-n:i-1} = \mathbf{u}, G_{\mathbf{u}} \sim G_{\mathbf{u}} \quad for\ i = 1, ..., T \tag{17}$$

Equation 16 says that a priori the conditional distribution $G_{\mathbf{u}}$ should be similar to $G_{\sigma(\mathbf{u})}$, its parent in the context tree. The variation of $G_{\mathbf{u}}$ around its mean $G_{\sigma(\mathbf{u})}$ is described by a PYP with a context length-dependent discount parameter $\alpha_{|\mathbf{u}|}$. At the top of the tree, the distribution G_ε for the empty context ε is similar to an overall base distribution G_0 which specifies our prior belief that each symbol s will appear with the probability $G_0(s)$. Equation 17 describes the *n*th order Markov model for **x** - the distribution over each symbol x_i in **x**, given that its context consisting of the previous *n* symbols $x_{i-n:i-1}$ is **u**, is simply $G_{\mathbf{u}}$. This is the hierarchical PY process.

The Sequence Memoizer Model: Instead of limiting context lengths to *n*, the model is extended to include the set of distributions in all contexts of any (finite) length - the distribution over each symbol is now conditioned on all previous symbols, not just the previous *n*. The model is the hierarchical PY model defined in Equations 15, **??** and 17, but with 2 changes: the contexts range over all finite nonempty strings, $\mathbf{u} \in \Sigma^*/\varepsilon$; Equation 17 becomes $x_i|\mathbf{x}_{1:i-1} = \mathbf{u}, G_{\mathbf{u}} \sim G_{\mathbf{u}}$ (conditioning on all previous symbols). The Model can be interpreted as the limit of an HPYP model as the Markov order *n* tends to infinity.

Compacting the context tree: Given a finite length sequence of symbols **x** we need access to only a finite number of conditional distributions. We need only $G_{x_{1:i}}$ where $i = 0, ..., T$ and all the ancestors of each $G_{x_{1:i}}$ in the context tree. The ancestors are needed because each $G_{\mathbf{u}}$ has a prior that depends on its parents $G_{\sigma(\mathbf{u})}$. The resulting set of conditional distributions that the sequence **x** actually depends on consists of $G_{\mathbf{u}}$ where **u** ranges over all continuous substrings of **x**, a finite set of $O(T^2)$ contexts. This subtree is denoted $T(\mathbf{x})$.

Many of the contexts that appear in $T(\mathbf{x})$ appear only in non-branching chains, i.e. each node on the chain has only one child in $T(\mathbf{x})$. If we can directly express the prior of the longer context in terms of the shortest non-branching suffix, then we can effectively ignore those in between and marginalise them out from the model (coagulation: $G_{11}|G_1 \sim PY(\alpha_2, G_1)$ and $G_{011}|G_{11} \sim PY(\alpha_3, G_{11})$ gives $G_{011}|G_1 \sim PY(\alpha_2\alpha_3, G_1)$ where G_{11} has been marginalised out - i.e. the prior for G_{011} is another PYP whose discount parameter is simply the product of the discount parameters along the chaing leading into it on the tree $T(\mathbf{x})$, while the base distribution is simply the head of the G_1.

We can apply this marginalization procedure to all non-branching chains of $T(\mathbf{x})$ to give a compact context tree $\hat{T}(\mathbf{x})$ where all internal nodes have at least two children. The number of nodes in $\hat{T}(\mathbf{x})$ is at most twice the length of the sequence **x** (independent of $|\Sigma|$). The structure of the compact context tree is given by the suffix structure for the reverse sequence $x_T, x_{T-1}, ..., x_1$.

Inference in the full SM model with an infinite number of parameters is equivalent to inference in the

compact context tree with a linear number of parameters. The prior over the conditional distributions on $\hat{T}(\mathbf{x})$ still retains the form of an HPYP - each node has a PYP prior with its parent as the base distribution.

$$\mathbb{E}[G_{\mathbf{u}}(s)] = \mathbb{E}\left[\frac{N(\mathbf{u}s) - \alpha_{\mathbf{u}}M(\mathbf{u}s) + \sum_{s' \in \Sigma} \alpha_{\mathbf{u}}M(\mathbf{u}s')G_{\sigma(\mathbf{u})}(s)}{\sum_{s' \in \Sigma} N(\mathbf{u}s')}\right] \tag{18}$$

A Hierarchical Nonparametric Bayesian Approach to Statistical Language Model Domain Adaptation - *Wood, Teh* 2009

A Stochastic Memoizer for Sequence Data - *Wood, Archambeau, Gasthaus, James, Teh* 2009

A Nonparametric Bayesian Alternative to Spike Sorting - *Wood, Black* 2006

Deplump for Streaming Data - *Bartlett, Wood* 2011

Other Notes

n-gram model is a contiguous sequence of *n* items from a given sequence of text.