Preliminary Exam Report

Review of Hierarchical Dirichlet Process and Infinite HMMs

Amir Harati

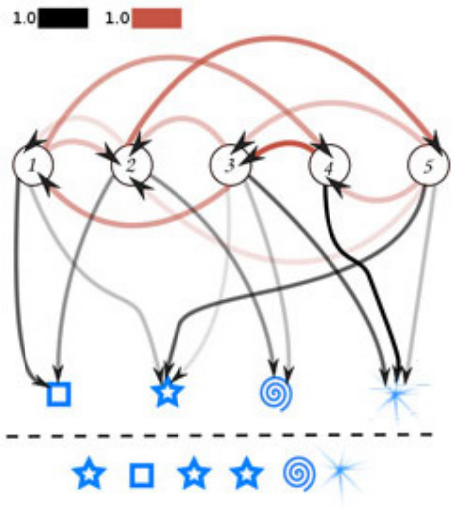
Amir.harati@gmail.com

Institute for Signal and Information Processing,

Department of Electrical Engineering,

Temple University

Feb 2012



EXECUTIVE SUMMARY

In this report, I review three research papers concerning hierarchical Dirichlet processes (HDP), infinite hidden Markov models (HDP-HMM), and several inference algorithms for these models. I also use several other sources to support the material discussed in this report.

The three primary references used in this exam are "Hierarchical Bayesian Nonparametric Models with Applications," "On-Line Learning for the Infinite Hidden Markov Model," and "Sticky HDP-HMM with Application to Speaker Diarization". The basic materials in all of these papers involve HDP and HDP-HMMs. The first paper is a more general overview and contains several other models which are not related directly to the main theme of this report. Therefore this report is not organized based on any one of these references but contains most of the relevant material from those papers. In many cases I have corrected several errors ranging from typos to simple algorithmic/computational errors and have also used a unified notation through the report that I believe makes the presentations easier.

To make the report more self-contained I have added a background review of Dirichlet processes (DP), but this review is very short and readers may need to review some background papers before reading this report. After reviewing DPs, I start by introducing HDP and the reason that they are needed. Several properties of HDP are derived (both in the main report and the appendix) and some of its properties are justified. Two basic inference algorithms for HDPs are presented in detail.

HDP-HMM is introduced based on the general framework of hierarchical Dirichlet processes. The differences between this new model and HDP are emphasized and several of their properties are derived and explained. Three important inference algorithms are reviewed and presented in detail.

In writing this review, one of my primary intentions is to produce a self-sufficient document that can be used as a reference to implement some of the inference algorithms for HDP-HMM. Moreover, an interested reader can easily start from these and derive more general or application specific algorithms.

Table of Contents

1 INTRODUCTION

Nonparametric Bayesian methods provide a consistent framework to infer the model complexity from the data. Moreover, Bayesian methods make hierarchical modeling easier and therefore open doors for more interesting and complex applications. In this report, we review hierarchical Dirichlet processes (HDP) and its applications to derive infinite hidden Markov models (HMM) or HDP-HMM. We also review three inference algorithms for the so called HDP-HMM in details.

This report is organized into five sections and two appendixes. Section two is a quick review of Dirichlet process. Section three is devoted to HDP and its inference algorithms and section four is focused on HDP-HMM and its inference algorithms. For the sake of readability, some of the mathematical details are presented in the appendix sections.

2 BACKGROUND

A Dirichlet process (DP) is a distribution over distributions, or more precisely over discrete distributions. Formally, a Dirichlet process $DP(\alpha, G_0)$ is "defined to be the distribution of a random probability measure G over Θ such that for any finite measurable partition $(A_1, A_2, ..., A_r)$ of Θ the random distribution $(G(A_1), ..., G(A_r))$ is distributed as finite dimensional Dirichlet distribution" (Teh Y. , Jordan, Beal, & Blei, 2006) :

$$(G(A_1), ..., G(A_r)) \sim Dir(\alpha G_0(A_1), ..., \alpha G_0(A_r)) \tag{2.1}$$

A constructive definition for Dirichlet process is given by Sethuraman (Sethuraman, 1994) which is known as stick-breaking construction. This construction explicitly shows that draws from a DP are discrete with probability one.

$$\begin{aligned} &v_k \mid \alpha, G_0 \sim Beta(1, \alpha), \quad \theta_k \mid \alpha, G_0 \sim G_0 \\ &\beta_k = v_k \prod_{l=1}^{k-1} (1 - \beta_l), \qquad G = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k} \end{aligned} \tag{2.2}$$

β can be interpreted as a random probability measure over positive integers and is denoted by $\beta \sim GEM(\alpha)$. In both of these definitions G_0, or base distribution, is the mean of the DP, and α is the concentration parameter which can be understood as the inverse of variance.

Another way to look at the DP is through the Polya urn scheme. In this approach, we have to consider i.i.d. draws from a DP and consider the predictive distribution over these draws (Teh Y. , Jordan, Beal, & Blei, 2006):

$$\theta_i \mid \theta_1, ..., \theta_{i-1}, \alpha, G_0 \sim \sum_{k=1}^{i-1} \frac{1}{N - 1 + \alpha} \delta_{\theta_k} + \frac{\alpha}{N - 1 + \alpha} G_0 \tag{2.3}$$

In the urn interpretation of equation (2.3), we have an urn with several balls of different colors in it. We draw a ball and put it back in the urn and add another ball of the same color to the urn. With probability

proportional to α we draw a ball with a new color. To make the clustering property more clear, we should introduce a new set of variables that represent distinct values of the atoms. Let $\theta_1^*,...,\theta_K^*$ to be the distinct values and m_k be the number of θ_i associated with θ_k^*. We would now have:

$$\theta_i \mid \theta_1,...,\theta_{i-1},\alpha,G_0 \sim \sum_{k=1}^{K} \frac{m_k}{N-1+\alpha}\delta_{\theta_k^*} + \frac{\alpha}{N-1+\alpha}G_0 \tag{2.4}$$

Another useful interpretation of (2.4) is the Chinese restaurant process (CRF). In CRF we have a Chinese restaurant with infinite number of tables. A new customer θ_i comes into the restaurant and can either sit around one of the occupied tables with probability proportional to the number of people already sitting there or start a new table with probability proportional to α. In this metaphor, each customer is a data point and each table is a cluster.

3 HIERARCHICAL DIRICHLET PROCESS

A Hierarchical Dirichlet Process (HDP) is the natural extension of a Dirichlet process for problems with multiple groups of data. Usually, data is split into J groups a priori. For example, consider a collection of documents. If words are considered as data points, each document would be a group. We want to model data inside a group using a mixture model. However, we are also interested to tie groups to each other, i.e. to share clusters across all groups. Let's assume that we have an indexed collection of DPs with a common base distribution $\{G_j\} \sim DP(\alpha,G_0)$. Unfortunately this simple model cannot solve the problem since for continues G_0 different G_j necessary have no atoms in common. The solution is to use a discrete G_0 with broad support. In other words, G_0 is itself a draw from a Dirichlet process. HDP is defined by (Teh & Jordan, 2010) equation (3.1).

$$\begin{aligned} &G_0 \mid \gamma,H \sim DP(\gamma,H) \\ &G_j \mid \alpha,G_0 \sim DP(\alpha,G_0) \\ &\theta_{ji} \mid G_j \sim G_j \\ &x_{ji} \mid \theta_{ji} \sim F(\theta_{ji}) \qquad for \quad j \in J \end{aligned} \tag{3.1}$$

In this definition H provides prior distribution for factor θ_{ji}. γ governs the variability of G_0 around H and α controls the variability of G_j around G_0. H, γ and α are hyper-parameters of HDP.

3.1 Stick-Breaking Construction

Because G_0 is a Dirichlet distribution it has a stick-breaking representation:

$$G_0 = \sum_{k=1}^{\infty} \beta_k \delta_{\theta_k^{**}}, \tag{3.2}$$

Where $\theta_k^{**} \sim H$ and $\beta = (\beta_k)_{k=1}^{\infty} \sim GEM(\gamma)$. Since support of G_j is contained in within the support of G_0

we can write a similar equation to (3.2) for G_j :

$$G_j = \sum_{k=1}^{\infty} \pi_{jk} \delta_{\theta_k^{**}} \tag{3.3}$$

Then we have:

$$\pi_j \sim DP(\alpha, \beta) \tag{3.4}$$

$$\begin{aligned} v_{jk} &\sim Beta\left(\alpha_0 \beta_k, \alpha_0 \left(1 - \sum_{l=1}^{k} \beta_l \right) \right) \\ \pi_{jk} &= v_{jk} \prod_{l=1}^{k-1} (1 - v_{jl}), \qquad for \quad k = 1, \ldots, \infty \end{aligned} \tag{3.5}$$

We also have:

$$\begin{aligned} E[\beta_k] &= \gamma^{k-1} (1+\gamma)^{-k} \\ E[\pi_{jk}] &= E[\beta_k] \\ Var[\pi_{jk}] &= E\left[\frac{\beta_k (1-\beta_k)}{1+\alpha} \right] + Var[\beta_k] > Var[\beta_k] \end{aligned} \tag{3.6}$$

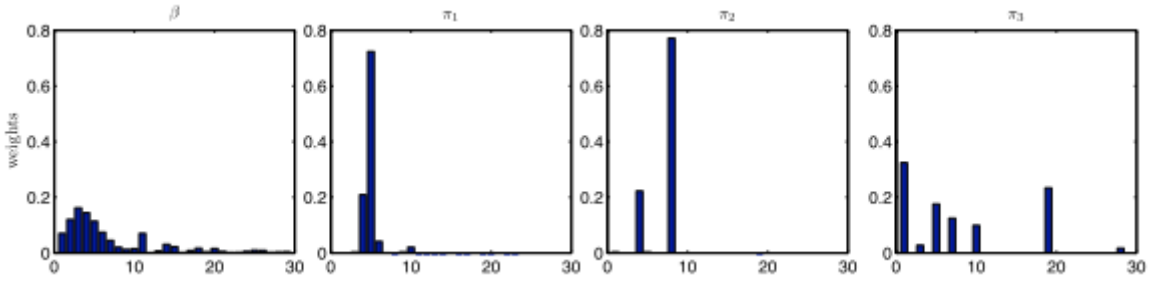


Figure 1 - Stick Breaking Construction for HDP: The left panel shows a draw of β, while the right three show independent draws conditioned on β (Teh & Jordan, 2010).

Stick-breaking construction for HDP and (3.6) are derived in A.1. Figure 1 demonstrates stick-breaking and cluster sharing of HDP.

3.2 Different Representations

Definition (3.1) shows the first representation of HDP. Another representation can be obtained by introducing an indicator variable as shown in equation (3.7).

Figure 2 shows the graphical models of both of these representations.

$$\begin{aligned}
&\beta \mid \gamma \sim GEM(\gamma) \\
&\pi_j \mid \alpha, \beta \sim DP(\alpha, \beta) \\
&\theta_k \mid H, \lambda \sim H(\lambda) \\
&z_{ji} \mid \pi_j \sim \pi_j \\
&x_{ji} \mid \{\theta_k\}_{k=1}^{\infty}, z_{ji} \sim F\left(\theta_{ji}\right)
\end{aligned} \tag{3.7}$$

3.3 Chinese Restaurant Franchise

The Chinese restaurant franchise (CRF) is the natural extension of Chinese restaurant process for HDPs. In CRF, we have a franchise with several restaurants and a franchise wide menu. The first customer in restaurant j sits at one of the tables and orders an item from the menu. Other customers either sit at one of the occupied tables and eat the food served at that table or sit at a new table and order their own food from the menu. Moreover, the probability of sitting at a table is proportional to the number of customers already seated at that table. In this metaphor, restaurants correspond to groups and customer i in restaurant j corresponds to θ_{ji} (customers are distributed according to G_j). Tables are i.i.d. variables θ_{jt}^{*} distributed according to G_0 and finally foods are i.i.d. variables θ_k^{**} distributed according to H. If customer i at restaurant j sits at table t_{ji} and that table serves dish k_{jt}, we will have $\theta_{ji} = \theta_{jt_{ji}}^{*} = \theta_{k_{t_{ji}}}^{**}$. In another way, each restaurant represents a simple DP and therefore a cluster over data points. At the franchise level we have another DP but this time clustering is over tables.

Now let introduce several variables that will be used throughout this paper. n_{jkt} is the number of

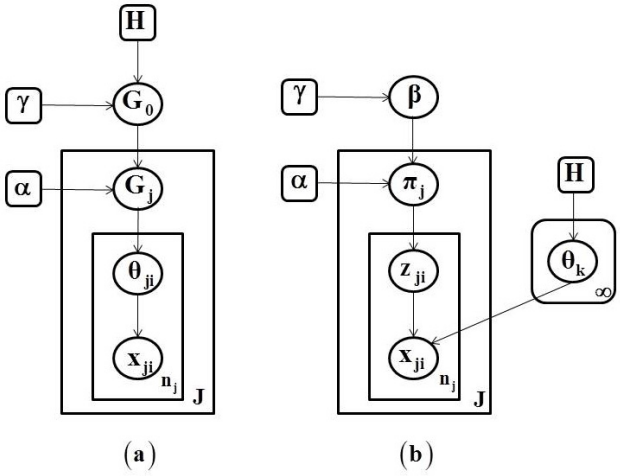


Figure 2-(a) HDP representation of (3.1) (b) Alternative indicator variable representation (3.7) (Teh, Jordan, Beal, & Blei, 2004)

customers in restaurant j , seated around table t ,and who eat dish k . m_{jk} is the number of tables in restaurant j serving dish k and K is the number of unique dishes served in the entire franchise. Marginal counts are denoted with dots. For example, $n_{j\bullet k}$ is the number of customers in restaurant j eating dish k .

3.3.1 Posterior and Conditional Distributions

CRF can be characterized by its state which consists of the dish labels $\boldsymbol{\theta}^{**} = \left\{\theta_k^{**}\right\}_{k=1,\ldots,K}$, the tables $\left\{t_{ji}\right\}_{\substack{j=1,\ldots,J \\ i=1,\ldots,n_{\bullet\bullet}}}$ and dishes $\left\{k_{jt_{ji}}\right\}_{\substack{j=1,\ldots,J \\ i=1,\ldots,n_{\bullet\bullet}}}$. As a function of the state of the CRF, we also have the number of customers $\mathbf{n} = \left\{n_{jtk}\right\}$, the number of tables $\mathbf{m} = \left\{m_{jk}\right\}$, customer labels $\boldsymbol{\theta} = \left\{\theta_{ji}\right\}$ and table labels $\boldsymbol{\theta}^{*} = \left\{\theta_{jt}^{*}\right\}$ (Teh & Jordan, 2010). The posterior distribution of G_0 is given by:

$$G_0 \mid \gamma, H, \boldsymbol{\theta}^{*} \sim DP\left(\gamma + m_{\bullet\bullet}, \frac{\gamma H + \sum_{k=1}^{K} m_{\bullet k}\delta_{\theta_k^{**}}}{\gamma + m_{\bullet\bullet}}\right) \tag{3.8}$$

Where $m_{\bullet\bullet}$ is the total number of tables in the franchise and $m_{\bullet k}$ is the total number of tables serving dish k . Equation (3.9) shows the posterior for G_j . $n_{j\bullet\bullet}$ is the total number of customers in restaurant j and $n_{j\bullet k}$ is the total number of customers in restaurant j eating dish k .

$$G_j \mid \alpha, G_0, \boldsymbol{\theta}_{\mathbf{j}} \sim DP\left(\alpha + n_{j\bullet\bullet}, \frac{\alpha G_0 + \sum_{k=1}^{K} n_{j\bullet k}\delta_{\theta_k^{**}}}{\alpha + n_{j\bullet\bullet}}\right) \tag{3.9}$$

Conditional distributions can be obtained by integrating out G_j and G_0 respectively. By integrating out G_j from (3.9) we obtain:

$$\theta_{ji} \mid \theta_{j1}, \ldots, \theta_{j,i-1}, \alpha, G_0 \sim \sum_{t=1}^{m_{j\bullet}} \frac{n_{jt\bullet}}{\alpha + n_{j\bullet\bullet}}\delta_{\theta_{jt}^{*}} + \frac{\alpha}{\alpha + n_{j\bullet\bullet}}G_0 \tag{3.10}$$

And by integrating out G_0 from (3.8) we obtain:

$$\theta_{jt}^{*} \mid \theta_{j1}^{*}, \ldots, \theta_{j,t-1}^{*}, \gamma, H \sim \sum_{k=1}^{K} \frac{m_{\bullet k}}{\gamma + m_{\bullet k}}\delta_{\theta_k^{**}} + \frac{\gamma}{\gamma + m_{\bullet\bullet}}H \tag{3.11}$$

A draw from (3.8) can be obtained using (3.12) and a draw from (3.9) can be obtained using (3.13).

$$\begin{aligned} &\beta_0, \beta_1, \ldots, \beta_K \mid \gamma, G_0, \boldsymbol{\theta}^{*} \sim Dir\left(\gamma, m_{\bullet 1}, \ldots, m_{\bullet K}\right) \\ &G_0' \mid \gamma, H \sim DP\left(\gamma, H\right) \\ &G_0 = \beta_0 G_0' + \sum_{k=1}^{K} \beta_k \delta_{\theta_k^{**}} \end{aligned} \tag{3.12}$$

$$\begin{aligned}
&\pi_{j0},\pi_{j1},...,\pi_{jK} \mid \alpha,\boldsymbol{\theta}_{\mathbf{j}} \sim Dir(\alpha\beta_0,\alpha\beta_1+n_{j\bullet 1},...,\alpha\beta_K+n_{j\bullet K}) \\
&G_j' \mid \alpha, G_0 \sim DP(\alpha\beta_0, G_0') \\
&G_j = \pi_{j0}G_j' + \sum_{k=1}^{K}\pi_{jk}\delta_{\theta_k^{**}}
\end{aligned} \tag{3.13}$$

From (3.12) and (3.13) we see that the posterior of G_0 is a mixture of atoms corresponding to dishes and an independent draw from $DP(\gamma,H)$ and G_j is a mixture of atoms at θ_k^{**} and an independent draw from $DP(\alpha\beta_0,G_0')$ (Teh & Jordan, 2010).

In an HDP each restaurant represents a DP and so $m_{j\bullet} \in O\left(\alpha \log \frac{n_{j\bullet\bullet}}{\alpha}\right)$ since the number of clusters scales logarithmically. On the other hand, equation (3.8) shows G_0 is a DP over tables and so $K \in O\left(\gamma \log \sum_j \frac{m_{j\bullet}}{\gamma}\right) = O\left(\gamma \log\left(\frac{\alpha}{\gamma}\sum_j \log \frac{n_{j\bullet\bullet}}{\alpha}\right)\right)$. This shows that HDP represents a prior belief that the number of clusters grows very slowly (double logarithmically) when increasing the number of data points but faster (logarithmically) in the number of groups (Teh & Jordan, 2010).

All of the above relationships are derived in A.2.

3.4 Inference Algorithms

First, let introduce several notations. $\mathbf{x}=\{x_{ji}\}$, $\mathbf{x_{jt}}=\{x_{ji} \mid t_{ji}=t\}$, $\mathbf{t}=\{t_{ji}\}$, $\mathbf{k}=\{k_{jt}\}$ and $\mathbf{z}=\{z_{ji}\}$. Where $z_{ji}=k_{jt_{ji}}$ denotes the mixture component associated with the observation x_{ji}. To indicate the removal of a variable from a set of variables or a count, we use a superscript $-ji$; for example $x^{-ji}=\mathbf{x}\setminus x_{ji}$ or $n_{jt\bullet}^{-ji}$ is the number of customers (observations) in restaurant (group) j seated at table t excluding x_{ji}. We also assume $F(\theta)$ has the density $f(\bullet\mid\theta)$ and H has the density $h(\bullet)$ and is conjugate to F. The conditional density of x_{ji} under mixture component k giving all data excluding x_{ji} is defined as:

$$\begin{aligned}
f_k^{-x_{ji}}(x_{ji}) &= \frac{\int h(\theta\mid\lambda)\prod_{j'i'\in D_k\cup x_{ji}} f_\theta(x_{j'i'})d\theta}{\int h(\theta\mid\lambda)\prod_{j'i'\in D_k\setminus x_{ji}} f_\theta(x_{j'i'})d\theta}, k=1,...,,K \\
f_{k^{new}}^{-x_{ji}}(x_{ji}) &= \int h(\theta\mid\lambda) f_\theta(x_{ji})\,d\theta, k=k^{new}
\end{aligned} \tag{3.14}$$

where $D_k=\{j'i': k_{j't_{j'i'}}=k\}$ denotes the set of indices of the data item currently associated with dish k. For the conjugate case, we could obtain a closed form for this likelihood function. Particularly if emission distributions are Gaussian with unknown mean and covariance, the conjugate prior is a normal-inverse-

Wishart distribution (Sudderth, 2006) denoted by $NIW(\lambda)_{\lambda=\{\zeta,\vartheta,\nu,\Delta\}}$ and $\theta_k=\{\mu_k,\Sigma_k\}$. Given some observations for component, k of the mixture $\{x^{(l)}\}_{l=1}^{L}$ (in this case $\{x^{(l)}\}_{l=1}^{L}=\{x^{-ji}\mid k_{j't'_{jt}}=k, j'=1,...,J, i'=1,...,n_{j\cdot\cdot}\}$) from a multivariate Gaussian distribution, the posterior still remains in the normal-inverse-Wishart family and its parameters are updated using:

$$\begin{aligned}\bar{\zeta}_k&=\zeta+L\\ \bar{\nu}_k&=\nu+L\\ \bar{\zeta}_k\bar{\vartheta}_k&=\zeta\vartheta+\sum_{l=1}^{L}x^{(l)}\\ \bar{\nu}_k\bar{\Delta}_k&=\nu\Delta+\sum_{l=1}^{L}x^{(l)}x^{(l)^T}+\zeta\vartheta\vartheta^T-\bar{\zeta}\bar{\vartheta}\bar{\vartheta}^T\end{aligned}\tag{3.15}$$

In practice there are some efficient ways (using Cholesky decomposition) to update these equations and allows fast likelihood evaluation for each data point. Finally, marginalizing θ_k induces a multivariate t-student distribution with $\bar{\nu}_k-d+1$ degree of freedom (d is the dimension of data points):

$$\begin{aligned}f_k^{-x_{ji}}(x_{ji})&=t_{\bar{\nu}_k-d-1}\left(x_{ji};\bar{\vartheta}_k,\frac{(\bar{\zeta}_k+1)\bar{\nu}_k}{\bar{\zeta}_k(\bar{\nu}_k-d-1)}\bar{\Delta}_k\right),k=1,...,K\\ f_{k^{new}}^{-x_{ji}}(x_{ji})&=t_{\nu-d-1}\left(x_{ji};\vartheta,\frac{(\zeta+1)\nu}{\zeta(\nu-d-1)}\Delta\right),k=k^{new}\end{aligned}\tag{3.16}$$

Assuming $\bar{\nu}_k>(d+1)$ (3.16) can be approximated by moment-match Gaussian (Sudderth, 2006).

3.4.1 Posterior Sampling in CRF

Equations (3.10) and (3.11) show how we can produce samples from the prior over θ_{ji} and θ^*_{jt}, using the proper likelihood function and how this framework can provide us with necessary tools to sample from the posterior given the observations $\mathbf{x}$. In this first algorithm, we sample index t_{ji} and k_{jt} using a simple Gibbs sampler.

In this algorithm we assumed that emission distributions are Gaussian but using other conjugate pairs is the same. The following list shows a single iteration of the algorithm, but to obtain reliable samples we have to run this many times. Notice that the most computational costly operation is the calculation of the conditional densities. Moreover, the number of events that could happen for each iteration of the algorithm is $K+1$.

1. Given the pervious state assignment K, $\mathbf{k}$ and $\mathbf{t}$:

Sample t:

2. For all $j=1,...,J, i=1,....,n_{j\cdot\cdot}$ do the follow sequentially

3. Compute the conditional density for x_{ji} using (3.16) for $k = 1,...,K,k^{new}$.
4. Calculate the likelihood for $t_{ji} = t^{new}$ using:

$$p\left(x_{ji} \mid \mathbf{t}^{-ji}, t_{ji} = t^{new}, \mathbf{k}\right) = \sum_{k=1}^{K} \frac{m_{\bullet k}^{-ji}}{m_{\bullet\bullet} + \gamma} f_k^{-x_{ji}}\left(x_{ji}\right) + \frac{\gamma}{m_{\bullet\bullet} + \gamma} f_{k^{new}}^{-x_{ji}}\left(x_{ji}\right) \tag{3.17}$$

5. Sample t_{ji} from the multinomial probability

$$p\left(t_{ji} = t \mid \mathbf{t}^{-ji}, \mathbf{k}\right) \propto \begin{cases} n_{jt\bullet}^{-ji} f_{k_{jt}}^{-x_{ji}}\left(x_{ji}\right), & \text{if t previously used} \\ \alpha p\left(x_{ji} \mid \mathbf{t}^{-ji}, t_{ji} = t^{new}, \mathbf{k}\right), & \text{if } t = t^{new} \end{cases} \tag{3.18}$$

6. If the sampled value of t_{ji} is t^{new}, obtain a sample of $k_{jt^{new}}$ by:

$$p\left(k_{jt^{new}} = k \mid \mathbf{t}, \mathbf{k}^{-jt^{new}}\right) \propto \begin{cases} m_{\bullet k}^{-x_{ji}} f_k^{-x_{ji}}\left(x_{ji}\right), & \text{If k previously used} \\ \gamma f_{k^{new}}^{-x_{ji}}\left(x_{ji}\right), & \text{If k =k}^{new} \end{cases} \tag{3.19}$$

7. If $k = k^{new}$ then increment K.
8. Update the cached statistics.
9. If a table t becomes unoccupied delete the corresponding k_{jt} and, if as a result some mixture component becomes unallocated, delete that mixture component too.

Sample K:

10. For all $j = 1,...,J, t = 1,...,m_{j\bullet}$ do the following sequentially:
11. Compute the conditional density for $\mathbf{x_{jt}}$ using (3.16) for $k = 1,...,K,k^{new}$.
12. Sample k_{jt} from a multinomial distribution:

$$p\left(k_{jt} \mid \mathbf{t}, \mathbf{k}^{-\mathbf{jt}}\right) \propto \begin{cases} m_{\bullet k}^{-\mathbf{x_{jt}}} f_k^{-\mathbf{x_{jt}}}\left(\mathbf{x_{jt}}\right), & \text{If k previously used} \\ \gamma f_{k^{new}}^{-\mathbf{x_{jt}}}\left(\mathbf{x_{jt}}\right), & \text{If k =k}^{new} \end{cases} \tag{3.20}$$

13. If $k = k^{new}$ then increment K.
14. Update the cached statistics.
15. If a mixture component becomes unallocated delete that mixture component

Equation (3.17) is obtained from (3.11). From (3.10) we see the prior probability that t_{ji} takes a previously used value is proportional to $n_{jt\bullet}^{-ji}$ and the prior probability of taking a new value is proportional to α.The likelihood for x_{ji} given $t_{ji}=t$ for some previously used t is $f_k^{-x_{ji}}\left(x_{ji}\right)$ and the likelihood for $t_{ji}=t^{new}$ is given by (3.17). By multiplying these priors and likelihoods we can obtain the posterior distribution (3.18). In the same way, (3.19) and (3.20) can be obtained by multiplying the likelihoods and priors given by (3.11).

3.4.2 Augmented Posterior Representation Sampler

In the previous algorithm, the sampling for all groups is coupled which makes the derivation of the CRF sampler for certain models difficult (Teh Y. , Jordan, Beal, & Blei, 2006). This happens because G_0 was integrated out. An alternative approach is to sample G_0 . More specifically, we will use (3.12) and (3.13) to sample from G_0 and G_j respectively. This algorithm contains two main steps. First we sample cluster indices $\left\{z_{ji}\right\}$ (instead of tables and dishes) and then we sample β and $\left\{\pi_j\right\}_{j\in J}$. Equation (3.12) shows in order to sample from β we should first sample $\left\{m_{jk}\right\}$. This completes the second algorithm.

1. Given the pervious state assignment for $\mathbf{z}$, β and $\left\{\pi_j\right\}$ from previous step

Sample Z

2. For all $j=1,...,J, i=1,....,n_{j\bullet}$ do the follow sequentially
3. Compute the conditional density for x_{ji} using (3.16) for $k=1,...,K,k^{new}$.
4. Sample z_{ji} using:

$$p(z_{ji}\mid \mathbf{z}^{-ji},\beta)\propto\begin{cases}\pi_{jk}f_k^{-x_{ji}}\left(x_{ji}\right) & \text{If k previously used}\\ \pi_{j0}f_{k^{new}}^{-x_{ji}}\left(x_{ji}\right) & \text{If k=k}^{new}\end{cases} \tag{3.21}$$

5. If a new component k^{new} is chosen, the corresponding atom is initiated using (3.22) and set $z_{ji}=K+1$ and $K=K+1$.

$$\begin{aligned}&v_0\mid\gamma\sim Beta\left(\gamma,1\right)\\&\left(\beta_0^{new},\beta_{K+1}^{new}\right)=\left(\beta_0 v_0,\beta_0(1-v_0)\right)\\&v_j\mid\alpha,\beta_0,v_0\sim Beta\left(\alpha\beta_0 v_0,\alpha\beta_0(1-v_0)\right)\\&\left(\pi_{j0}^{new},\pi_{jK+1}^{new}\right)\sim\left(\pi_{j0}v_j,\pi_{j0}(1-v_j)\right)\end{aligned} \tag{3.22}$$

6. Update the cached statistics.

Sampling m

7. Sample $\{m_{jk}\}$ using (3.23) where $s(n,m)$ is the Stirling number of the first kind. Alternatively, we can sample $\{m_{jk}\}$ by simulating a CRF, which is more efficient for large $n_{j.k}$.

$$p(m_{jk} = m \mid \mathbf{z}, \beta, \alpha, n_{j\cdot k}) = \frac{\Gamma(\alpha\beta_k)}{\Gamma(\alpha\beta_k + n_{j\cdot k})} s(n_{j\cdot k}, m)(\alpha\beta_k)^m \tag{3.23}$$

Sampling β and π

8. Sample β and π using(3.12) and (3.13) respectively.

In this algorithm we have used alternative representation given by (3.7). Particularly, we can see $z_{ji} \sim \pi_j$. By combining this with (3.13) we can obtain a conditional prior probability function for z_{ji} and by multiplying it with the likelihoods we can obtain (3.21). (3.22) is the stick-breaking step for the new atom and follows the steps described in section 3.1. In particular the third line in (3.22) is obtained by replacing the remaining stick β_0 with a unit stick. The validity of this approach is shown in (Pitman, 1996).For a proof of (3.23)look at (Antoniak, 1974). It should be noted that computing this equation is generally very costly and we can alternatively simulate a CRF to sample $\{m_{jk}\}$.

3.5 Applications

Among the several applications of HDP, we will only review two of them in this section. It should be noted that the following section, HDP-HMM, is by itself an application of the general HDP framework.

One of the most cited applications of HDP is in the field of information retrieval (IR) (Teh & Jordan, 2010). A state of the art but heuristic algorithm which is very popular in IR applications is the "term-frequency inverse document frequency" (tf-idf) algorithm. The intuition behind this algorithm is that the relevance of a term to a document is proportional to the number of times that term occurred in the document. However, terms that occur in many documents should be down weighted. It has been shown that HDP provides a justification for this intuition (Teh & Jordan, 2010) and an algorithm based on HDP outperforms all state of the art algorithms.

Another application cited extensively in literature is topic modeling (Teh & Jordan, 2010). In topic modeling, we want to model documents with a mixture model. Topics are defined as probability distributions across a set of words while documents are defined as probability distributions across different topics. At the same time we want to share topics among documents within a corpus. So each document is a group with its own mixing proportions but components (topics) are shared across all documents using an HDP model.

4 HDP-HMM

Hidden Markov models (HMMs) are a class of doubly stochastic processes in which discrete state sequences are modeled as a Markov chain (Rabiner, 1989). In the following discussion we will denote the state of the Markov chain at time t with z_t and the state-specific transition distribution for state j by π_j .The Markovian structure means $z_t \sim \pi_{z_{t-1}}$. Observations are conditionally independent given the state of the HMM and are denoted by $x_t \sim F\left(\theta_{z_t}\right)$.

HDP-HMM is an extension of HMM in which the number of states can be infinite. The idea is relatively simple; at each state z_t we should be able to go to an infinite number of states so the transition distribution should be a draw from a DP. On the other hand, we want reachable states from one state to be shared among all states so these DPs should be linked together. The result is an HDP. In an HDP-HMM each state corresponds to a group (restaurant) and therefore, unlike HDP in which an association of data to groups is assumed to be known a priori, we are interested to infer this association. The major problem with original HDP-HMM is the state persistence. HDP-HMM has a tendency to make many redundant states and switch rapidly among them (Teh Y. , Jordan, Beal, & Blei, 2006). This problem is solved by introducing a sticky parameter to the definition of HDP-HMM (Fox E. , Sudderth, Jordan, & Willsky, 2011). Equation (4.1) shows the definition of a sticky HDP-HMM with unimodal emissions. κ is a sticky hyper-parameter and generally can be learned from data. Original HDP-HMM is a special case with $\kappa = 0$. From this equation we can see for each state (group) we have a simple unimodal emission distribution. This limitation can be addressed using a more general model defined in (4.2). In this model, a DP is associated with each state and a model with augmented state (z_t, s_t) is obtained. Figure 3 shows a graphical representation.

$$
\begin{aligned}
&\beta \mid \gamma \sim GEM(\gamma) \\
&\pi_j \mid \alpha, \beta \sim DP(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa}) \\
&\theta_j^{**} \mid H, \lambda \sim H(\lambda) \\
&z_t \mid z_{t-1}, \left\{\pi_j\right\}_{j=1}^{\infty} \sim \pi_{z_{t-1}} \\
&x_t \mid \left\{\theta_j^{**}\right\}_{j=1}^{\infty}, z_t \sim F\left(\theta_{z_t}\right)
\end{aligned}
\tag{4.1}
$$

$$
\begin{aligned}
&\beta \mid \gamma \sim GEM(\gamma) \\
&\pi_j \mid \alpha, \beta \sim DP(\alpha + \kappa, \frac{\alpha\beta + \kappa\delta_j}{\alpha + \kappa}) \\
&\psi_j \mid \sigma \sim GEM(\sigma) \\
&\theta_{kj}^{**} \mid H, \lambda \sim H(\lambda) \\
&z_t \mid z_{t-1}, \left\{\pi_j\right\}_{j=1}^{\infty} \sim \pi_{z_{t-1}} \\
&s_t \mid \left\{\psi_j\right\}_{j=1}^{\infty}, z_t \sim \psi_{z_t} \\
&x_t \mid \left\{\theta_{kj}^{**}\right\}_{k,j=1}^{\infty}, z_t \sim F\left(\theta_{z_t s_t}\right)
\end{aligned}
\tag{4.2}
$$

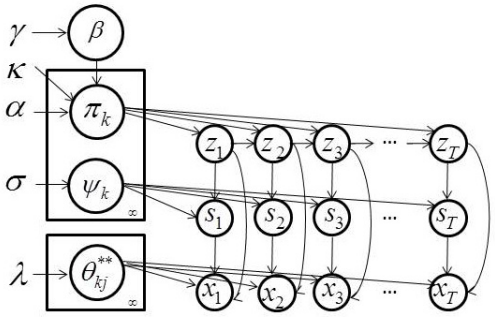


Figure 3-Graphical model of HDP-HMM (Fox E. , Sudderth, Jordan, & Willsky, 2011)

4.1 CRF with Loyal Customers

The metaphor for the Chinese restaurant franchise for sticky HDP-HMM is a franchise with loyal customers. In this case each restaurant has a special dish which is also served in other restaurants. If a customer x_t is going to restaurant j then it is more likely that he eats the specialty dish $z_t = j$ there. His children x_{t+1} also go to the same restaurant and eat the same dish. However, if x_t eats another dish ($z_t \neq j$) then his children go to the restaurant indexed by z_t and more likely eat their specialty dish. Thus customers are actually loyal to dishes and tend to go to restaurants where their favorite dish is the specialty.

4.2 Inference Algorithms

4.2.1 Direct Assignment Sampler

This sampler is adapted from (Fox E. , Sudderth, Jordan, & Willsky, 2011) and (Fox E. , Sudderth, Jordan, & Willsky, 2010). In this section we present the sampler for HDP-HMM with DP emission. This algorithm is very similar to the second inference algorithm for HDP presented in 3.4.2. The algorithm is divided into two steps: the first step is to sample the augmented state (z_t, s_t) and the second is to sample β .

In order to sample (z_t, s_t) we need to have the posterior. By inspecting Figure 3 and using the chain rule we can write the following relationship for this posterior.

$$\begin{aligned}
&p(z_t = k, s_t = j \mid z_{\setminus t}, s_{\setminus t}, x_{1:T}, \beta, \alpha, \sigma, \kappa, \lambda) = \\
&p(s_t = j \mid z_t = k, z_{\setminus t}, s_{\setminus t}, x_{1:T}, \sigma, \lambda)\, p(z_t = k \mid z_{\setminus t}, s_{\setminus t}, x_{1:T}, \beta, \alpha, \kappa, \lambda) \propto \\
&p(s_t = j \mid \{s_\tau \mid z_\tau = k, \tau \neq t\}, \sigma)\, p(x_t \mid \{x_\tau \mid z_\tau = k, s_t = j, \tau \neq t\}) \\
&p(z_t = k \mid z_{\setminus t}, \beta, \alpha, \kappa) \sum_{s_t} \left(p(x_t \mid \{x_\tau \mid z_\tau = k, s_t, \tau \neq t\})\, p(s_t \mid \{s_\tau \mid z_\tau = k, \tau \neq t\}, \sigma) \right)
\end{aligned} \tag{4.3}$$

The reason that we have summed over s_t in the last line is because we are interested to calculate the likelihood for each state. This equation also tells us that we should first sample the state z_t and then conditioned on the current state, sample the mixture component for that state. In B.1. we will derive the following relationships for the component of (4.3). (4.6) is written for Gaussian emissions but we can always use the general relationship (3.14) for an arbitrary emission distribution.

$$p\left(z_t = k \mid z_{\setminus t}, \beta, \alpha, \kappa\right) \propto \begin{cases} \left(\alpha\beta_k + n_{z_{t-1}k}^{-t} + \kappa\delta\left(z_{t-1}, k\right)\right)\left(\dfrac{\alpha\beta_{z_{t+1}} + n_{kz_{t+1}}^{-t} + \kappa\delta\left(k, z_{t+1}\right) + \delta\left(z_{t-1}, k\right)\delta\left(k, z_{t+1}\right)}{\alpha + n_{k\bullet}^{-t} + \kappa + \delta\left(z_{t-1}, k\right)}\right) & k \in \{1, \ldots, K\} \\ \dfrac{\alpha^2 \beta_{\bar{k}} \beta_{z_t+1}}{\alpha + \kappa}, & k = K+1 \end{cases} \tag{4.4}$$

$$p\left(s_t = j \mid \left\{s_\tau \mid z_\tau = k, \tau \neq t\right\}, \sigma\right) = \begin{cases} \dfrac{n_{kj}'^{-t}}{\sigma + n_{k\bullet}'^{-t}}, & j \in \{1, \ldots, K_k'\} \\ \dfrac{\sigma}{\sigma + n_{k\bullet}'^{-t}}, & j = K_k' + 1 \end{cases} \tag{4.5}$$

$$\begin{aligned} p\left(x_t \mid \left\{x_\tau \mid z_\tau = k, s_t = j, \tau \neq t\right\}\right) &= t_{\bar{v}_{kj}-d-1}\left(x_t; \bar{\vartheta}_{kj}, \frac{\left(\bar{\zeta}_{kj}+1\right)\bar{v}_{kj}}{\bar{\zeta}_{kj}\left(\bar{v}_{kj}-d-1\right)}\bar{\Delta}_{kj}\right), k = 1, \ldots, K, j = 1, \ldots, K_k' \\ p\left(x_t \mid \left\{x_\tau \mid z_\tau = k, s_t = j^{new}, \tau \neq t\right\}\right) &= t_{\bar{v}_{kj^{new}}-d-1}\left(x_t; \vartheta, \frac{\left(\zeta+1\right)v}{\zeta\left(v-d-1\right)}\Delta\right), k = 1, \ldots, K, j = j^{new} \\ p\left(x_t \mid \left\{x_\tau \mid z_\tau = k^{new}, \tau \neq t\right\}\right) &= t_{v_{k^{new}}-d-1}\left(x_t; \vartheta, \frac{\left(\zeta+1\right)v}{\zeta\left(v-d-1\right)}\Delta\right), k = k^{new} \end{aligned} \tag{4.6}$$

After sampling $\left(z_t, s_t\right)$ we have to sample β but from (3.12) we see that we need to know the distribution of the number of tables considering dish k ($\{\bar{m}_{\bullet k}\}_{k=1}^{K}$). The approach is to first find the distribution of tables serving dish k ($\{m_{\bullet k}\}_{k=1}^{K}$). In this algorithm, instead of using the approach based on Stirling numbers, we can obtain this distribution by a simulation of the CRF, and then adjust this distribution to obtain the real distribution of considered dishes by tables. To review the reason that this adjustment is necessary, we should notice that κ introduces a non-informative bias to each restaurant so customers are more likely to select the specialty dish of the restaurant. In order to obtain the considered dish distribution we should reduce this bias from the distribution of the served dish. This can be done using an override variable. Suppose m_{jk} tables are serving dish k in restaurant j. If $k \neq j$ then $m_{jk} = \bar{m}_{jk}$ since the served dish is not the house specialty but if $k = j$ then there is probability that tables are overridden by the house specialty. Suppose that ω_{jt} is the override variable with prior $p\left(\omega_{jt} \mid \rho\right) \sim \rho$, we can write:

$$p\left(\omega_{jt} \mid k_{jt}=j,\beta,\rho\right) \propto p\left(k_{jt}=j \mid \omega_{jt},\beta,\rho\right) p\left(\omega_{jt} \mid \rho\right) \propto \begin{cases} \beta_j(1-\rho) & \omega_{jt}=0 \\ \rho & \omega_{jt}=1 \end{cases} \tag{4.7}$$

The sum of these Bernoulli random variables is a binomial random variable and finally we can calculate the number of tables that considered ordering dish k by (4.17).

1. Given a previous set of $\left(z_{1:T}^{(n-1)}, s_{1:T}^{(n-1)}\right)$ and $\beta^{(n-1)}$
2. For all $t \in \{1,2,...,T\}$.
3. For each of the K currently instantiated states compute:

- The predictive conditional distributions for each of the K_k' currently instantiated mixture components for this state, and also for a new component and for a new state.

$$f'_{k,j}(x_t) = \left(\frac{n'^{-t}_{kj}}{\sigma + n'^{-t}_{k\bullet}}\right) p\left(x_t \mid \{x_\tau \mid z_\tau = k, s_t = j, \tau \neq t\}\right) \tag{4.8}$$

$$f_{k,K_k'+1}(x_t) = \frac{\sigma}{\sigma + n'^{-t}_{k\bullet}} p\left(x_t \mid \left\{x_\tau \mid z_\tau = k, s_t = j^{new}, \tau \neq t\right\}\right) \tag{4.9}$$

$$f'_{k^{new},0}(x_t) = \left(\frac{\sigma}{\sigma + n'^{-t}_{\bullet\bullet}}\right) p\left(x_t \mid \left\{x_\tau \mid z_\tau = k^{new}, \tau \neq t\right\}\right) \tag{4.10}$$

- The predictive conditional distribution of the HDP-HMM state without knowledge of the current mixture component.

$$f_k(x_t) = \left(\alpha\beta_k + n^{-t}_{z_{t-1}} + \kappa\delta(z_{t-1},k)\right)\left(\frac{\alpha\beta_{z_{t+1}} + n^{-t}_{kz_{t+1}} + \kappa\delta(k,z_{t+1}) + \delta(z_{t-1},k)\delta(k,z_{t+1})}{\alpha + n^{-t}_{k\bullet} + \kappa + \delta(z_{t-1},k)}\right) \left(\sum_{j=1}^{K_k'} f'_{k,j}(x_t) + f_{k,K_k'+1}(x_t)\right), k = 1,...,K \tag{4.11}$$

$$f_{K+1}(x_t) = \frac{\alpha^2 \beta_{\bar{k}} \beta_{z_t+1}}{\alpha + \kappa} f'_{k^{new},0}(x_t), k = K+1$$

4. Sample z_t:

$$z_t \sim \sum_{k=1}^{K} f_k(x_t)\delta(z_t,k) + f_{K+1}(x_t)\delta(z_t,K+1) \tag{4.12}$$

5. Sample s_t conditioned on z_t :

$$s_t \sim \sum_{j=1}^{K'_k} f'_{k,j}(x_t)\delta(s_t, j) + f_{k,K'_k+1}(x_t)\delta(s_t, K'_k + 1) \tag{4.13}$$

6. If $k = K + 1$ increase the K and transform β as

$$\begin{aligned} {}_0|\,\gamma &\sim Beta(\gamma, 1) \\ \left(\beta_0^{new}, \beta_{K+1}^{new}\right) &= \left(\beta_0 v_0, \beta_0(1 - v_0)\right) \end{aligned} \tag{4.14}$$

7. If $s_t = K'_k + 1$ increment K'_k .
8. Update the cache. If there is a state with $n_{k\bullet} = 0$ or $n_{\bullet k} = 0$ remove k and decrease K . If $n'_{kj} = 0$ remove the component j and decrease K'_k .
9. Sample auxiliary variables by simulating a CRF:
10. For each $(j,k) \in \{1,...,K\}^2$ set $m_{jk} = 0$ and $n = 0$. For each customer in restaurant j eating dish k ($i = 1,...,n_{jk}$), sample:

$$x \sim Ber\left(\frac{\alpha\beta_k + \kappa\delta(j,k)}{n + \alpha\beta_k + \kappa\delta(j,k)}\right) \tag{4.15}$$

11. Increment n and if $x = 1$ increment m_{jk} .
12. For each $j \in \{1,...,K\}$,sample the override variables in restaurant j :

$$\omega_{j\bullet} \sim Binomial\left(m_{jj}, \frac{\rho}{\rho + \beta_j(1-\rho)}\right), \rho = \frac{\kappa}{\alpha + \kappa} \tag{4.16}$$

13. Set the number of informative tables in restaurant j :

$$\bar{m}_{jk} = \begin{cases} m_{jk} & j \neq k \\ m_{jj} - \omega_{j\bullet} & j = k \end{cases} \tag{4.17}$$

14. Sample β :

$$\beta^{(n)} \sim Dir\left(\gamma, \bar{m}_{\bullet 1}, ..., \bar{m}_{\bullet K}\right) \tag{4.18}$$

15. Optionally sample hyper-parameters σ, γ, α and κ .

4.2.2 Block Sampler

The problem with the direct assignment sampler mentioned in the previous section is the slow convergence rate since we sample states sequentially. The sampler can also group two temporal sets of observations related to one underlying state into two separate states. However, in the last sampling scheme we have not used the Markovian structure to improve the performance. In this section a variant of forward-backward procedure is incorporated in the sampling algorithm that enables us to sample the state sequence $z_{1:T}$ at once. However, to achieve this goal, a fixed truncation level L should be accepted which in a sense reduces the model into a parametric model (Fox E. , Sudderth, Jordan, & Willsky, 2011). However, it should be noted that the result is different from a classical parametric Bayesian HMM since the truncated HDP priors induce a shared sparse subset of the L possible states (Fox E. , Sudderth, Jordan, & Willsky, 2011). In short, we obtain an approximation to the nonparametric Bayesian HDP-HMM with maximum number of possible states set to L. However, for almost all applications this should not cause any problem if we set L reasonably high. The approximation used in this algorithm is the degree L weak limit approximation to the DP (Ishwaran & Zarepour, 2002) which is defined as:

$$GEM_L(\alpha) \triangleq Dir(\alpha / L, ..., \alpha / L) \tag{4.19}$$

Using (4.19) β is approximated as (Fox, Sudderth, Jordan, & Willsky, Supplement to " A Sticky HDP-HMM with Application to Speaker Diarization", 2010):

$$\beta \mid \gamma \sim Dir(\gamma / L, ..., \gamma / L) \tag{4.20}$$

Similar to (3.4) we can write:

$$\pi_j \mid \alpha, \kappa, \beta \sim Dir(\alpha\beta_1, ..., \alpha\beta_j + \kappa, ...\alpha\beta_L) \tag{4.21}$$

And posteriors are (similar to (3.13)):

$$\begin{aligned} &\beta \mid \bar{\mathbf{m}}, \gamma \sim Dir(\gamma / L + \bar{m}_{\cdot 1}, ..., \gamma / L + \bar{m}_{\cdot L}) \\ &\pi_j \mid z_{1:T}, \alpha, \beta \sim Dir(\alpha\beta_1 + n_{j1}, ..., \alpha\beta_j + \kappa + n_{jj}, ..., \alpha\beta_L + n_{jL}) \end{aligned} \tag{4.22}$$

In (4.22) n_{jk} is the number of transitions from state j to state k and $\bar{m}_{jk}$ is the same as (4.17).

Finally an order L' weak limit approximation is used for the DP prior on the emission parameters:

$$\psi_k \mid z_{1:T}, s_{1:T}, \sigma \sim Dir(\sigma / L' + n'_{k1}, ..., \sigma / L' + n'_{kL'}) \tag{4.23}$$

The forward-backward algorithm for the joint sample $z_{1:T}$ and $s_{1:T}$ given $x_{1:T}$ can be obtained by:

$$\begin{aligned} &p(z_t, s_t \mid x_{1:T}, \mathbf{z}, \boldsymbol{\pi}, \boldsymbol{\psi}, \boldsymbol{\theta}) \\ &\propto p(z_t \mid z_{t-1}, x_{1:T}, \boldsymbol{\pi}, \boldsymbol{\theta})\, p(s_t \mid \psi_{z_t})\, p(x_t \mid \theta_{z_t, s_t})\, p(x_{1:t-1} \mid z_t, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\psi})\, p(x_{t+1:T} \mid z_t, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\psi}) \end{aligned} \tag{4.24}$$

The right side of equation (4.24) has two parts: forward and backward probabilities (Rabiner, 1989).The forward probability includes $p(z_t \mid z_{t-1}, x_{1:T}, \boldsymbol{\pi}, \boldsymbol{\theta}) p(s_t \mid \psi_{z_t}) f(x_t \mid \theta_{z_t, s_t}) p(x_{1:t-1} \mid z_t, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\psi})$ and backward probability includes $p(x_{t+1:T} \mid z_t, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\psi})$. It seems that the authors in this work approximate the forward probabilities with $p(z_t \mid z_{t-1}, x_{1:T}, \boldsymbol{\pi}, \boldsymbol{\theta}) p(s_t \mid \psi_{z_t}) f(x_t \mid \theta_{z_t, s_t})$, and for backward probabilities we have:

$$
\begin{aligned}
& p(x_{t+1:T} \mid z_t, \boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\psi}) \propto m_{t,t-1}(z_{t-1}) \\
& \propto \begin{cases} \sum_{z_t} \sum_{s_t} p(z_t \mid \pi_{z_t-1}) p(s_t \mid \psi_{z_t}) f(x_t \mid \theta_{z_t, s_t}) m_{t+1,t}(z_t) & t \le T \\ 1 & t = T+1 \end{cases} \\
& \Rightarrow m_{t,t-1}(k) \propto \begin{cases} \sum_{i=1}^{L} \sum_{l=1}^{L} \pi_{ki} \psi_{il} f(x_t \mid \theta_{z_t, s_t}) m_{t+1,t}(z_t) & t \le T \quad k = 1, \ldots L \\ 1 & t = T+1 \end{cases}
\end{aligned}
\tag{4.25}
$$

As a result we would have (Fox E. , Sudderth, Jordan, & Willsky, 2010) :

$$
\begin{aligned}
& p(z_{\ t} = k, s_t = j \mid x_{1:T}, \mathbf{z}, \boldsymbol{\pi}, \boldsymbol{\psi}, \boldsymbol{\theta}) \\
& \propto \pi_{z_{t-1} k} \psi_{kj} f(x_t \mid \theta_{z_t, s_t}) m_{t+1,t}(z_t)
\end{aligned}
\tag{4.26}
$$

where for Gaussian emission for components are given by $f(x_t \mid \theta_{z_t, s_t}) = \mathrm{N}(x_t; \mu_{kj}, \Sigma_{kj})$

The algorithm is as follows (Fox E. , Sudderth, Jordan, & Willsky, 2010):

1. Given the previous $\boldsymbol{\pi}^{(n-1)}, \boldsymbol{\psi}^{(n-1)}, \boldsymbol{\beta}^{(n-1)}$ and $\boldsymbol{\theta}^{(n-1)}$.
2. For $k \in \{1, \ldots, L\}$, initialize messages to $m_{T+1,T}(k) = 1$
3. For $t \in \{T-1, \ldots, 1\}$ and $k \in \{1, \ldots, L\}$ compute

$$
m_{t,t-1}(k) = \sum_{i=1}^{L} \sum_{l=1}^{L} \pi_{ki} \psi_{il} N(x_{t+1}; \mu_{il}, \Sigma_{il}) m_{t+1,t}(i) \tag{4.27}
$$

4. Sample the augmented state (z_t, s_t) sequentially and start from $t = 1$:

Set $n_{ik} = 0, n'_{kj} = 0$ and $\Upsilon_{kj} = \varnothing$ for $(i,k) \in \{1, \ldots, L\}^2$ and $(k,j) \in \{1, \ldots, L\} \times \{1, \ldots, L'\}$

For all $(k,j) \in \{1, \ldots, L\} \times \{1, \ldots, L'\}$ compute:

$$
f_{k,j}(x_t) = \pi_{z_{t-1}, k} \psi_{k,j} N(x_t; \mu_{k,j}, \Sigma_{k,j}) m_{t+1,t}(k) \tag{4.28}
$$

5. Sample augmented state (z_t, s_t):

$$\left(z_t, s_t\right) \sim \sum_{k=1}^{L} \sum_{j=1}^{L'} f_{k,j}\left(x_t\right)\delta\left(z_t, k\right)\delta\left(s_t, j\right) \tag{4.29}$$

6. Increase $n_{z_{t-1}z_t}$ and $n'_{z_t s_t}$ and add x_t to the cached statistics.

$$\Upsilon_{k,j} \leftarrow \Upsilon_{k,j} \oplus x_t \tag{4.30}$$

7. Sample $\mathbf{m}, \boldsymbol{\omega}, \bar{\mathbf{m}}$ similar to the previous algorithm
8. Update β:

$$\beta \sim Dir\left(\gamma / L + \bar{m}_{\bullet 1}, ..., \gamma / L + \bar{m}_{\bullet L}\right) \tag{4.31}$$

9. For $k \in \{1, ..., L\}$:

- Sample π_k and ψ_k:

$$\begin{aligned} \pi_k &\sim Dir\left(\alpha\beta_1 + n_{k1}, ..., \alpha\beta_k + \kappa + n_{kk}, ..., \alpha\beta_L + n_{kL}\right) \\ \psi_k &\sim Dir\left(\sigma / L' + n'_{k1}, ..., \sigma / L' + n'_{kL'}\right) \end{aligned} \tag{4.32}$$

- For $j \in \{1, ..., L'\}$ sample:

$$\theta_{k,j} \sim p\left(\theta \mid \lambda, \Upsilon_{k,j}\right) \tag{4.33}$$

10. Set $\boldsymbol{\pi}^{(n)} = \boldsymbol{\pi}, \boldsymbol{\psi}^{(n)} = \boldsymbol{\psi}, \beta^{(n)} = \beta$ and $\boldsymbol{\theta}^{(n)} = \boldsymbol{\theta}$
11. Optionally sample hyper-parameters σ, γ, α and κ.

4.2.3 Learning Hyper-parameters

Hyper-parameters including α, κ, γ and σ can also be inferred like other parameters of the model (Fox, Sudderth, Jordan, & Willsky, 2010).

4.2.3.1 Posterior for $(\boldsymbol{\alpha} + \boldsymbol{\kappa})$

Consider the probability of data x_{ji} to sit behind table t:

$$p\left(t_{ji} = t \mid \mathbf{t}^{-ji}, n_{jt\bullet}^{-ji}, \alpha, \kappa\right) \propto \begin{cases} n_{jt\bullet}^{-ji} & t \in \{t_1, ..., m_{j\bullet}\} \\ \alpha + \kappa & t = t^{new} \end{cases} \tag{4.34}$$

This equation can be written by considering equation (3.10) and (4.1). From this equation we can say customer table assignment follows a DP with concentration parameter $\alpha+\kappa$. Antoniak (Antoniak, 1974) has shown that if $\beta\sim GEM(\gamma), z_i\sim\beta$ then the distribution of the number of unique values of z_i resulting from N draws from β has the following form:

$$p(K\mid N,\gamma)=\frac{\Gamma(\gamma)}{\Gamma(\gamma+N)}s(N,K)\gamma^K \tag{4.35}$$

Where $s(N,K)$ is the Stirling number of the first kind. Using these two equations the distribution of the number of tables in the restaurant *j* is as follows:

$$p(m_{j\cdot}\mid\alpha+\kappa,n_{j\cdot})=s(n_{j\cdot\cdot},m_{j\cdot})(\alpha+\kappa)^{m_{j\cdot}}\frac{\Gamma(\alpha+\kappa)}{\Gamma(\alpha+\kappa+n_{j\cdot})} \tag{4.36}$$

The posterior over $\alpha+\kappa$ is as follows:

$$\begin{aligned}p(\alpha+\kappa\mid m_{1\cdot},...,m_{J\cdot},n_{1\cdot},...,n_{J\cdot}) &\propto p(\alpha+\kappa)\,p(m_{1\cdot},...,m_{J\cdot}\mid\alpha+\kappa,n_{1\cdot},...,n_{J\cdot})\\ &\propto p(\alpha+\kappa)\prod_{j=1}^{J}p(m_{j\cdot}\mid\alpha+\kappa,n_{j\cdot})\\ &\propto p(\alpha+\kappa)\prod_{j=1}^{J}s(n_{j\cdot},m_{j\cdot})(\alpha+\kappa)^{m_{j\cdot}}\frac{\Gamma(\alpha+\kappa)}{\Gamma(\alpha+\kappa+n_{j\cdot})}\\ &\propto p(\alpha+\kappa)(\alpha+\kappa)^{m_{\cdot\cdot}}\prod_{j=1}^{J}\frac{\Gamma(\alpha+\kappa)}{\Gamma(\alpha+\kappa+n_{j\cdot})}\end{aligned} \tag{4.37}$$

The reason for the last line is that $\prod_{j=1}^{J}s(n_{j\cdot},m_{j\cdot})$ is not a function of $\alpha+\kappa$ and therefore can be ignored.

By substitution of $\beta(x,y)=\frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)}=\int_0^1 t^{x-1}(1-t)^{y-1}\,dt$ and also by considering that $\Gamma(x+1)=x\Gamma(x)$ we obtain:

$$p(\alpha+\kappa\mid m_{1\cdot},...,m_{J\cdot},n_{1\cdot},...,n_{J\cdot})\propto p(\alpha+\kappa)(\alpha+\kappa)^{m_{\cdot\cdot}}\prod_{j=1}^{J}\left(1+\frac{n_{j\cdot}}{\alpha+\kappa}\right)\int_0^1 r_j^{\alpha+\kappa}(1-r_j)^{n_{j\cdot}-1}\,dr_j \tag{4.38}$$

Finally by considering the fact that we have placed a $Gamma(a,b)$ prior on $\alpha+\kappa$ we can write:

$$p(\alpha+\kappa,r,s\mid m_{1\cdot},...,m_{J\cdot},n_{1\cdot},...,n_{J\cdot})\propto(\alpha+\kappa)^{a+m_{\cdot\cdot}-1}e^{-(\alpha+\kappa)b}\prod_{j=1}^{J}\left(\frac{n_{j\cdot}}{\alpha+\kappa}\right)^{s_j}r_j^{\alpha+\kappa}(1-r_j)^{n_{j\cdot}-1} \tag{4.39}$$

Where s_j can be either one or zero. For marginal probabilities we obtain:

$$p\left(\alpha+\kappa \mid r,s,m_{1\bullet},...,m_{J\bullet},n_{1\bullet},...,n_{J\bullet}\right) \propto (\alpha+\kappa)^{a+m_{\bullet\bullet}-1-\sum_{j=1}^{J}s_j} e^{-(\alpha+\kappa)\left(b-\sum_{j=1}^{J}\log r_j\right)} = Gamma\left(\alpha+m_{\bullet\bullet}-\sum_{j=1}^{J}s_j, b-\sum_{j=1}^{J}\log r_j\right) \tag{4.40}$$

$$p\left(r_j \mid \alpha+\kappa, r_{\setminus j}, s, m_{1\bullet},...,m_{J\bullet},n_{1\bullet},...,n_{J\bullet}\right) \propto r_j^{\alpha+\kappa}\left(1-r_j\right)^{n_{j\bullet}-1} = Beta\left(\alpha+\kappa+1, n_{j\bullet}\right) \tag{4.41}$$

$$p\left(s_j \mid \alpha+\kappa, r, s_{\setminus j} m_{1\bullet},...,m_{J\bullet},n_{1\bullet},...,n_{J\bullet}\right)\alpha\left(\frac{n_{j\bullet}}{\alpha+\kappa}\right)^{s_j} = Ber\left(\frac{n_{j\bullet}}{n_{j\bullet}+\alpha+\kappa}\right) \tag{4.42}$$

4.2.3.2 Posterior of γ

Similar to the discussion for (4.35) if we want to find the distribution of the unique number of dishes served in the whole franchise we would have $p\left(K \mid \gamma, \bar{m}_{\bullet\bullet}\right) = s\left(\bar{m}_{\bullet\bullet}, K\right)\frac{\Gamma(\gamma)}{\Gamma\left(\gamma+\bar{m}_{\bullet\bullet}\right)}$. Therefore for the posterior distribution of γ we can write:

$$\begin{aligned} p\left(\gamma \mid K, \bar{m}_{\bullet\bullet}\right) &\propto p(\gamma)\, p\left(K \mid \gamma, \bar{m}_{\bullet\bullet}\right) \\ &\propto p(\gamma)\gamma^K \frac{\beta\left(\gamma+1, \bar{m}_{\bullet\bullet}\right)}{\gamma\Gamma\left(\bar{m}_{\bullet\bullet}\right)} \\ &\propto p(\gamma)\gamma^K \left(\gamma+\bar{m}_{\bullet\bullet}\right)\int_0^1 \eta^{\gamma}(1-\eta)^{\bar{m}_{\bullet\bullet}-1}\, d\eta \end{aligned} \tag{4.43}$$

By considering the fact that that prior over γ is $Gamma\left(a,b\right)$ we can finally write:

$$p\left(\gamma,\eta,\varsigma \mid K, \bar{m}_{\bullet\bullet}\right) \propto \gamma^{\alpha+K-1}\left(\frac{\bar{m}_{\bullet\bullet}}{\gamma}\right)^{\varsigma} e^{-\gamma(b-\log\eta)}\left(1-\eta\right)^{\bar{m}_{\bullet\bullet}-1} \tag{4.44}$$

And finally for the marginal distributions we have:

$$p\left(\gamma \mid \eta,\varsigma,K,\bar{m}_{\bullet\bullet}\right) \propto \gamma^{\alpha+K-1-\varsigma} e^{-\gamma(b-\log\eta)} = Gamma\left(\alpha+K-\varsigma, b-\log\eta\right) \tag{4.45}$$

$$p\left(\eta \mid \gamma,\varsigma,K,\bar{m}_{\bullet\bullet}\right) \propto \eta^{\gamma}\left(1-\eta\right)^{\bar{m}_{\bullet\bullet}-1} = Beta\left(\gamma+1, \bar{m}_{\bullet\bullet}\right) \tag{4.46}$$

$$p\left(\varsigma \mid \gamma,\eta,K,\bar{m}_{\bullet\bullet}\right) \propto \left(\frac{\bar{m}_{\bullet\bullet}}{\gamma}\right)^{\varsigma} = Ber\left(\frac{\bar{m}_{\bullet\bullet}}{\bar{m}_{\bullet\bullet}+\gamma}\right) \tag{4.47}$$

4.2.3.3 Posterior of σ

The posterior for σ is obtained in a similar way to $\alpha+\kappa$. We use two auxiliary variables r' and s' and the final marginalized distributions are:

$$p\left(\sigma \mid r', s', K'_{1\bullet}, ..., K'_{J\bullet}, n_{1\bullet}, ..., n_{J\bullet}\right) \propto \left(\sigma\right)^{\alpha + K'_{\bullet\bullet} - 1 - \sum_{j=1}^{J} s'_j} e^{-\sigma\left(b - \sum_{j=1}^{J} \log r'_j\right)} \tag{4.48}$$

$$p\left(r'_j \mid \sigma, r'_{\setminus j}, s', K'_{1\bullet}, ..., K'_{J\bullet}, n_{1\bullet}, ..., n_{J\bullet}\right) \propto r_j'^{\sigma} \left(1 - r_j'^{\sigma}\right)^{n_{j\bullet} - 1} \tag{4.49}$$

$$p\left(s'_j \mid \sigma, r', s'_{\setminus j}, K'_{1\bullet}, ..., K'_{J\bullet}, n_{1\bullet}, ..., n_{J\bullet}\right) \propto \left(\frac{n_{j\bullet}}{\sigma}\right)^{s'_j} \tag{4.50}$$

It should be noted that in cases where we use auxiliary variables we prefer to iterate several times before moving to the next iteration of the main algorithm.

4.2.3.4 Posterior of ρ

By definition $\rho = \frac{\kappa}{\alpha + \kappa}$ and by considering the fact that the prior on ρ is $Beta\left(c, d\right)$ and $\omega_{jt} \sim Ber\left(\rho\right)$ we can write:

$$\begin{aligned} p\left(\rho \mid \omega\right) &\propto p\left(\omega \mid \rho\right) p\left(\rho\right) \\ &\propto Binomial\left(\sum_j \omega_{j\bullet}; m_{\bullet\bullet}, \rho\right) Beta\left(c, d\right) \\ &\propto Beta\left(\sum_j \omega_{j\bullet} + c, m_{\bullet\bullet} - \sum_j \omega_{j\bullet} + d\right) \end{aligned} \tag{4.51}$$

4.2.4 Online learning

The last two approaches are based on batch learning methodology. One problem with these methods is the need to run the whole algorithm for the whole data set when new data points become available. More than that, for large datasets we might face some practical constraints such as memory size. Another alternative approach is to use sequential learning techniques which essentially let us update models once a new data point becomes available. The algorithm that we are describing here is adapted from (Rodriguez, 2011), but the main idea for a general case is published in (Carvalho, Johannes, Lopes, & Polson, 2010) and (Carvalho, Lopes, Polson, & Taddy, 2010). For Bayesian problems different versions of particle filters are used to replace batch MCMC methods. For further information about particle filters refer to (Cappe, Godsill, & Moulines, 2007). It should be noted that this algorithm is developed for the non-sticky ($\kappa = 0$) HDP-HMM with one mixture per state but generalization to sticky HDP-HMM with DP emissions is straightforward.

4.2.4.1 Particle learning (PL) Framework for mixtures

PL is proposed in (Carvalho, Johannes, Lopes, & Polson, 2010) and (Carvalho, Lopes, Polson, & Taddy, 2010) and is a special formulation of augmented particle filters. A general mixture model that PL is supposed to infer can be represented by:

$$\begin{aligned} y_{t+1} &= f\left(x_{t+1}, \theta\right) \\ x_{t+1} &= g\left(x^t, \theta\right) \end{aligned} \tag{4.52}$$

In this set of equations, the first line is the observation equation and the second line is the state evolution which, in case of mixtures, indicates which component is assigned to the observations and $x^t = \left(x_1, ..., x_t\right)$

In order to estimate states and parameters we should define an "essential state vector" $z_t = \left(x_t, s_t, \theta\right)$ where $s_t = \left(s_t^x, s_t^\theta\right)$ and s_t^x are the state sufficient statistics and s_t^θ is the parameter sufficient statistics (Carvalho, Johannes, Lopes, & Polson, 2010). After observing y_{t+1}, particles should be updated based on:

$$p\left(z_t \mid y^{t+1}\right) = \frac{p\left(y_{t+1} \mid z_t\right) p\left(z_t \mid y^t\right)}{p\left(y_{t+1} \mid y^t\right)} \tag{4.53}$$

$$p\left(x_{t+1} \mid y^{t+1}\right) = \int p\left(x_{t+1} \mid z_t, y_{t+1}\right) p\left(z_t \mid y^{t+1}\right) dz_t \tag{4.54}$$

A particle approximation to (4.53) is:

$$p^N\left(z_t \mid y^{t+1}\right) = \sum_{i=1}^{N} \omega_t^{(i)} \delta_{z_t^{(i)}}(z_t), \quad \omega_t^{(i)} = \frac{p\left(y_{t+1} \mid z_t^{(i)}\right)}{\sum_{j=1}^{N} p\left(y_{t+1} \mid z_t^{(j)}\right)} \tag{4.55}$$

Using this approximation we can generate propagated samples from the posterior $p\left(x_{t+1} \mid z_t, y_{t+1}\right)$ to approximate $p\left(x_{t+1} \mid y^{t+1}\right)$. Sufficient statistics can be updated using a deterministic mapping and finally parameters should be updated using sufficient statistics. The main condition for this algorithm to be possible is the tractability of $p\left(y_{t+1} \mid z_t^{(i)}\right)$ and $p\left(x_{t+1} \mid z_t + y_{t+1}\right)$. The PL algorithm is as follows:

1. Resample particles z_t with weights $\omega_t^{(i)} \propto p\left(y_{t+1} \mid z_t^{(i)}\right)$
2. Propagate new states x_{t+1} using $p\left(x_{t+1} \mid z_t + y_{t+1}\right)$
3. Update state and parameter sufficient statistics deterministically $s_{t+1} = S\left(s_t, x_{t+1}, y_{t+1}\right)$
4. Sample θ from $p\left(\theta \mid s_{t+1}\right)$

After one sequential run through the data we can use smoothing algorithms to obtain $p\left(x^T \mid y^T\right)$. The algorithm is repeated $b = 1, ..., B$ to generate B sample paths.

1. Sample $\left(x_T^{(b)}, s_T^{(b)}, \theta^{(b)}\right)$ from output of particle filter $\sum_{i=1}^{N} \frac{1}{N} \delta_{z_T^{(i)}}\left(z_T\right)$

For $t = T-1:1$

2. Sample $\left(x_T^{(b)}, s_T^{(b)}\right) \sim \sum_{s=1}^{N} \bar{\omega}_t^{(i)} \delta_{z_t^{(s)}}$, $\bar{\omega}_t^{(i)} = \frac{q_t^{(i)}}{\sum_{s=1}^{N} q_t^{(s)}}$, $q_t^{(i)} = p\left(x_{t+1}^{(b)}, r_{t+1}^{(b)} \mid z_t^{(i)}\right)$

4.2.4.2 PL for HDP-HMM

In this algorithm we use $Z_t^{(i)} = \left(z_t^{(i)}, L_t^{(i)}, \left\{n_{ljt}^{(i)}\right\}, \left\{S_{lt}^{(i)}\right\}, \alpha_t^{(i)}, \left\{\beta_{lt}^{(i)}\right\}, \gamma_t^{(i)}, \left\{m_{ljt}^{(i)}\right\}, \phi_t^{(i)}, \left\{u_{lt}^{(i)}\right\}, \left\{\varsigma_{lt}^{(i)}\right\}\right)$ where z_t is the state, L_t is the number of states at time t, n_{ljt} is the number of transitions from state l to state j at time t, S_{lt} is the sufficient statistics for state l at time t and other variables have the same definitions as previous sections (the last three are auxiliary variables which are used to infer hyper-parameters.).

From (4.4) and by setting $\kappa = 0$ we can write:

$$p\left(z_{t+1} = k \mid z_1, ..., z_t, z_{t+2} = k', \beta, \alpha\right) \propto \begin{cases} \left(\alpha\beta_k + n_{z_t kt}\right)\left(\frac{\alpha\beta_{k'} + n_{kk't} + \delta\left(z_t, k\right)\delta\left(k, k'\right)}{\alpha + n_{k\bullet t} + \delta\left(z_t, k\right)}\right), k \in \left\{1, .., L_t\right\} \\ \alpha\beta_{L_t+1}\beta_k \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad , k = L_t + 1 \end{cases} \quad (4.56)$$

In this equation k' is the next state z_{t+2} that we have not seen yet. Because of this we should integrate this out by summing over all possibilities. The result is:

$$\begin{aligned} p\left(z_{t+1} = k \mid z_1, ..., z_t, \beta, \alpha\right) &\propto \begin{cases} \left(\alpha\beta_k + n_{z_t kt}\right)\left(\frac{\alpha + n_{k\bullet t} + \delta\left(z_t, k\right)}{\alpha + n_{k\bullet t} + \delta\left(z_t, k\right)}\right), k \in \left\{1, .., L_t\right\} \\ \alpha\beta_{L_t+1} \qquad\qquad\qquad\qquad\qquad , k = L_t + 1 \end{cases} \\ &\propto \begin{cases} \left(\alpha\beta_k + n_{z_t kt}\right), k \in \left\{1, .., L_t\right\} \\ \alpha\beta_{L_t+1} \qquad , k = L_t + 1 \end{cases} \end{aligned} \quad (4.57)$$

After normalization we can write:

$$p\left(z_{t+1} \mid z_1, ..., z_t, \beta, \alpha\right) = \sum_{k=1}^{L_t} \frac{\left(\alpha\beta_k + n_{z_t kt}\right)}{n_{z_t\bullet} + \alpha}\delta_k + \frac{\alpha\beta_{L_t+1}}{n_{z_t\bullet} + \alpha}\delta_{L_t+1} \quad (4.58)$$

$$p\left(x_{t+1} \mid Z_t\right) = \int p\left(x_{t+1} \mid z_{t+1}\right) p\left(z_{t+1} \mid Z_t\right) dz_{t+1} = \sum_{k=1}^{L_t} \frac{\left(\alpha\beta_k + n_{z_t kt}\right)}{n_{z_t\bullet} + \alpha} f_k^{-x_{t+1}}\left(x_{t+1}\right) + \frac{\alpha\beta_{L_t+1}}{n_{z_t\bullet} + \alpha} f_{L_t+1}^{-x_{t+1}}\left(x_{t+1}\right) \quad (4.59)$$

The algorithm is as follow:

1. Compute $\omega_t^{(i)} = \frac{v_t^{(i)}}{\sum_{j=1}^{N} v_t^{(j)}}$ where $v_t^{(i)} = \sum_{l=1}^{L_t^{(i)}} q_l^{(i)}\left(Z_t^{(i)}, x_{t+1}\right)$ and we have:

$$q_l^{(i)}\left(Z_t^{(i)}, x_{t+1}\right) = \begin{cases} \frac{n_{z_t^{(i)}lt}^{(i)} + \alpha_t^{(i)}\beta_{lt}^{(i)}}{n_{z_t^{(i)}\bullet t}^{(i)} + \alpha_t^{(i)}} f_l^{-x_{t+1}}\left(x_{t+1}\right), l = 1,...,L_t^{(i)} \\ \frac{\alpha_t^{(i)}\beta_{L_T^{(i)}+1,t}^{(i)}}{n_{z_t^{(i)}\bullet t}^{(i)} + \alpha_t^{(i)}} f_{L_t^{(i)}+1}^{-x_{t+1}}\left(x_{t+1}\right) \quad , l = L_t^{(i)} + 1 \end{cases} \tag{4.60}$$

Where $f_l^{-x_{t+1}}(\bullet)$ is similar to (4.6).

2. Sample $Z_t^{(1)}, Z_t^{(2)}, ..., Z_t^{(N)}$ from

$$p^N\left(Z_t \mid y^{t+1}\right) = \sum_{i=1}^{N} \omega_t^{(i)} \delta_{Z_t^{(i)}}\left(Z_t\right) \tag{4.61}$$

3. Propagate the particles to generate $Z_{t+1}^{(i)}$:
4. Sample $z_{t+1}^{(i)}$:

$$z_{t+1}^{(i)} \sim p\left(z_{t+1}^{(i)} \mid Z_t^{(i)}, x_{t+1}\right) \propto \sum_{l=1}^{L_t^{(i)}+1} \frac{q_l^{(i)}\left(Z_t^{(i)}, x_{t+1}\right)}{\sum_{j=1}^{L_t^{(i)}+1} q_l^{(j)}\left(Z_t^{(j)}, x_{t+1}\right)} \delta_l\left(z_{t+1}^{(i)}\right) \tag{4.62}$$

5. Update the number of states:

$$L_{t+1}^{(i)} = \begin{cases} L_t^{(i)} + 1, & z_{t+1}^{(i)} = L_t^{(i)} + 1 \\ L_{t+1}^{(i)} = L_t^{(i)} & otherwise \end{cases} \tag{4.63}$$

6. Update the sufficient statistics:

$$n_{z_t^{(i)}, z_{t+1}^{(i)}, t+1}^{(i)} = n_{z_t^{(i)}, z_{t+1}^{(i)}, t}^{(i)} + 1 \quad S_{z_{t+1}^{(i)}, t+1} = S_{z_{t+1}^{(i)}, t} + s\left(x_t\right) \tag{4.64}$$

7. If $z_{t+1}^{(i)} > L_t^{(i)}$:

$$\begin{aligned}
&\beta_{t+1}^{(i)} = \beta_t^{(i)} \\
&v_0 \mid \gamma \sim Beta\left(\gamma_t^{(i)}, 1\right) \\
&\left(\beta_{0,t+1}^{(i)}, \beta_{L_t^{(i)}+1,t}^{(i)}\right) = \left(\beta_{0,t}^{(i)} v_0, \beta_{0,t}^{(i)}(1-v_0)\right)
\end{aligned} \tag{4.65}$$

8. Hyper-parameters update:
9. Sample $m_{l,j,t+1}^{(i)}$:

$$p\left(m_{l,j,t+1}^{(i)} = m\right) \propto s\left(n_{l,j,t+1}^{(i)}, m\right)\left(\alpha_t^{(i)} \beta_{j,t+1}^{(i)}\right)^m \tag{4.66}$$

Alternatively we can simulate a CRF instead of computing Stirling numbers.

10. Sample $\gamma_{t+1}^{(i)}$ by first sample $\phi_{t+1}^{(i)} \sim Beta\left(\gamma_t^{(i)} + 1, m_{\bullet\bullet t+1}^{(i)}\right)$:

$$\begin{aligned}
&\gamma_{t+1}^{(i)} \sim \varepsilon Gama\left(a_\gamma + L_{t+1}^{(i)}, b_\gamma - \log\left(\phi_{t+1}^{(i)}\right)\right) + (1-\varepsilon) Gama\left(a_\gamma + L_{t+1}^{(i)} - 1, b_\gamma - \log\left(\phi_{t+1}^{(i)}\right)\right) \\
&\frac{\varepsilon}{1-\varepsilon} = \frac{a_\gamma + L_{t+1}^{(i)} - 1}{m_{\bullet\bullet t+1}^{(i)}\left(b_\gamma - \log\left(\phi_{t+1}^{(i)}\right)\right)}
\end{aligned} \tag{4.67}$$

11. Sampling $\alpha_{t+1}^{(i)}$ using auxiliary variables:

$$\begin{aligned}
&\varsigma_{l,t+1}^{(i)} \sim Beta\left(\alpha_t^{(i)} + 1, n_{l\bullet t+1}^{(i)}\right) \\
&u_{l,t+1}^{(i)} \sim Ber\left(\frac{n_{l\bullet t+1}^{(i)}}{\left(\alpha_t^{(i)} + n_{l\bullet t+1}^{(i)}\right)}\right) \\
&\alpha_{t+1}^{(i)} \sim Gam\left(a_a + m_{\bullet\bullet t+1}^{(i)} - u_{\bullet t+1}^{(i)}, b_\alpha - \sum_{l=1}^{L_{t+1}^{(i)}} \log \varsigma_{l,t+1}^{(i)}\right)
\end{aligned} \tag{4.68}$$

12. Resample β:

$$\beta_{t+1}^{(i)} \sim Dir\left(m_{\bullet,1,t+1}, \ldots, m_{\bullet,L_{t+1}^{(i)},t+1}, \gamma_{t+1}^{(i)}\right) \tag{4.69}$$

After finishing this inference step, we can optionally smooth the states using an algorithm similar to the one discussed in 4.2.4.1. However one drawback of this algorithm is the fact that paths for $z_1, \ldots, z_T$ are coupled together since we integrate out π_l for the inference algorithm. To improve particle diversity we can sample transition probability and emission parameters explicitly. The smoothing algorithm would be

as follows:

1. Sample $Z_T^{(b)} \sim \sum_{i=1}^{N} \frac{1}{N} \delta_{Z_T^{(i)}}$
2. Sample $\left\{\pi_l^{(b)}\right\}$

$$\left\{\pi_l^{(b)}\right\} \sim Dir\left(\alpha^{(b)} + n_{l\bullet T}^{(b)}, \beta_1^{(b)} + n_{l1T}^{(b)}, ..., \beta_{L_T^{(b)}}^{(b)} + n_{lL_T^{(b)}T}^{(b)}\right) \tag{4.70}$$

3. Sample $\left\{\theta_l^{(b)}\right\}$ from the $NIW\left(\theta_l \mid \lambda\right)_{\lambda=\{S_{lT}, n_{\bullet lT}, \zeta, \vartheta, \nu, \Delta\}}$.
4. Sample $z_1^{(b)}, ..., z_T^{(b)}$ using a single run of Forward-Backward algorithm applications. Use $\left\{\pi_l^{(b)}\right\}$ and $\left\{\theta_l^{(b)}\right\}$ as parameters for the Forward-Backward algorithm.

4.3 Applications

One of the applications of HDP-HMM, which is extensively discussed in (Fox E. , Sudderth, Jordan, & Willsky, 2011), is speaker diarization. In this application, we are interested to segment an audio file into time intervals associated with different speakers. If the number of speakers is known a priori a classic HMM can be used and each speaker can be modeled as different states of HMM. However, in real world applications the number of speakers is not known and therefore nonparametric models are a natural solution. It has been shown in (Fox E. , Sudderth, Jordan, & Willsky, 2011) that HDP-HMM can produce results comparable to other state of the art systems.

Another application which is cited as an application of HDP-HMM is word segmentation (Teh & Jordan, 2010). In this problem, we have an utterance and we are interested to segment it into words. Each word can be represented as a state in a HDP-HMM and transition distributions can define a grammar over words.

5 CONCLUSION

In this report, we have investigated hierarchical Dirichlet processes and its application to extend HMMs into infinite HMMs. We also reviewed two inference algorithms for HDP and three inference algorithms for HDP-HMM.

HDP-HMM seems to be a good candidate for many applications which traditionally use HMMs. Using a nonparametric Bayesian approach could help us to automatically learn the complexity of the models from the data instead of relying on heuristic tuning methods. Moreover, the framework can provide a generic and simple approach to organize all models (i.e. different HMMs in a speech recognizer) in a well-defined hierarchy and tie parameters of different models using Bayesian hierarchical methods. The definition of HDP-HMM (with DP emission) can also be altered to include another HDP that links DP emissions of different states together (to link different components of mixture models together.) Another area of work is in inference algorithms. We have presented three algorithms based on Gibbs sampling. It seems block and sequential samplers have some interesting properties that make them reasonable candidates for big

datasets. Particularly it seems easy to build a parallel implementation of sequential sampler which can be an important factor for large scale problems. Studying other kinds of inference methods like variational methods or parallel implementation of these algorithms can be a subject of further research.

6 REFERENCE

Antoniak, C. (1974). Mixtures of Dirichlet Process with Applications to Bayesian Nonparametric Problems. *The Annals of Statistics*, 1152-1174.

Cappe, O., Godsill, S. J., & Moulines, E. (2007, May). An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo. *Proceedings of the IEEE, 95*(5), 899-924.

Carvalho, C. M., Johannes, M., Lopes, H. F., & Polson, N. (2010). Particel Learning and Smoothing. *Statistical Science*, 88-106.

Carvalho, C. M., Lopes, H. F., Polson, N. G., & Taddy, M. A. (2010). Particle Learning for General Mixtures. *Bayesian Analysis*, 709-740.

Fox, E., Sudderth, E., Jordan, M., & Willsky, A. (2010). *Supplement to " A Sticky HDP-HMM with Application to Speaker Diarization".* doi:10.1214/10-AOAS395SUPP

Fox, E., Sudderth, E., Jordan, M., & Willsky, A. (2011). A Sticky HDP-HMM with Application to Speaker Diarization. *The Annalas of Applied Statistics, 5*, 1020-1056.

Ishwaran, H., & Zarepour, M. (2002, June). Exact and approximate sum representations for the Dirichlet process. *Canadian Journal of Statistics*, 269-283.

Pitman, J. (1996). Random Discrete Distributions Invariant under Size-Biased Permutation. *Advances in Applied ProbabilityApplied Probability Trust*, 525--539.

Rabiner, L. R. (1989, February). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE, 77*(2), 257-286.

Rodriguez, A. (2011, July). On-Line Learning for the Infinite Hidden Markov. *Communications in Statistics: Simulation and Computation, 40*(6), 879-893.

Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, 639-650.

Sudderth, E. B. (2006). chapter 2 of "Graphical Models for Visual Object Recognition and Tracking". Cambridge, MA: Massachusetts Institute of Technology.

Teh, Y., & Jordan, M. (2010). Hierarchical Bayesian Nonparametric Models with Applications. In N. Hjort, C. Holmes, P. Mueller, & S. Walker, *Bayesian Nonparametrics: Principles and Practice.* Cambridge, UK: Cambridge University Press.

Teh, Y., Jordan, M., Beal, M., & Blei, D. (2004). *Hierarchical Dirichlet Processes.* Technical Report 653 UC Berkeley.

Teh, Y., Jordan, M., Beal, M., & Blei, D. (2006). Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 1566-1581.

APPENDIX A: DERIVATION OF HDP RELATIONSHIPS

A.1. Stick-Breaking Construction (Teh Y. , Jordan, Beal, & Blei, 2006)

Lemma A.1.1

In this lemma we show $\pi_k = v_k \prod_{l=1}^{k-1}(1-v_l)$

We know:

$$\pi_k = v_k\left(1-\sum_{l=1}^{K-1}\pi_l\right)$$

Using this equation we can write:

$$1-\sum_{l=1}^{k-1}\pi_l = 1-v_1-v_2(1-v_1)-v_3(1-v_1)(1-v_2)....=(1-v_1)\left(1-v_2-v_3(1-v_2)....\right)=\prod_{l=1}^{K}(1-v_l)$$

$$\Rightarrow \pi_k = v_k\prod_{l=1}^{k-1}(1-v_l)$$

•

We know $G_j = \sum_{k=1}^{\infty}\pi_{jk}\delta_{\theta_k^{**}}$ .Let $(A_1,...,A_r)$ be a random partition on, define $K_l = \left\{k:\theta_k^{**}\in A_l\right\}, l=1,...,r$ from(3.1) and the definition of DP we have:

$$\left(G_j(A_1),...,G_j(A_r)\right)\sim Dir\left(\alpha G_0(A_1),...,\alpha G_0(A_r)\right) \tag{A1.1}$$

Using (A1.1), (3.2) and(3.3) we obtain:

$$\left(\sum_{k\in K_1}\pi_{jk},...,\sum_{k\in K_r}\pi_{jk}\right)\sim Dir\left(\alpha\sum_{k\in K_1}\beta_k,...,\alpha\sum_{k\in K_r}\beta_k\right) \tag{A1.2}$$

Since this is correct for every finite partition of positive integers we conclude that $\pi_j \sim DP(\alpha,\beta)$

Now for a partition $\left(\{1,...,k-1\},\{k\},\{k+1,...\}\right)$ and aggregation property of Dirichlet distribution we obtain:

$$\left(\sum_{l=1}^{k-1}\pi_{jl},\pi_{jk},\sum_{l=k+1}^{\infty}\pi_{jl}\right)\sim Dir\left(\alpha\sum_{l=1}^{k-1}\beta_l,\alpha\beta_k,\alpha\sum_{k=k+1}^{\infty}\beta_l\right) \tag{A1.3}$$

Using the neutrality property of a Dirichlet distribution:

$$\frac{1}{1-\sum_{l=1}^{k-1}\pi_{jl}}\left(\pi_{jk},\sum_{l=k+1}^{\infty}\pi_{jl}\right)\sim Dir\left(\alpha\beta_k,\alpha\sum_{l=k+1}^{\infty}\beta_l\right) \tag{A1.4}$$

Notice that $\sum_{l=k+1}^{\infty}\beta_l=1-\sum_{l=1}^{k}\beta_l$ and also defining $v_{jk}=\frac{\pi_{jk}}{1-\sum_{l=1}^{k-1}\pi_{jl}}$ we have:

$$v_{jk}\sim Beta\left(\alpha\beta_k,\alpha\left(1-\sum_{l=1}^{k}\beta_l\right)\right) \tag{A1.5}$$

Using Lemma A.1.1

$$\pi_{jk}=v_{jk}\left(1-\sum_{l=1}^{k-1}\pi_{jl}\right)=v_{jk}\prod_{l=1}^{k-1}\left(1-v_{jl}\right) \tag{A1.6}$$

A.2. Deriving Posterior and Predictive Distributions

Derivation of equation(3.8):

Since G_0 is a random distribution, we can draw from it. Let assume $\boldsymbol{\theta}^{**}$ are i.i.d. draws from G_0. $\boldsymbol{\theta}^{**}$ takes value in Θ since G_0 is a distribution over Θ. Let $A_1, A_2, ..., A_k$ be a finite measureable partition of Θ and $m_{\bullet r}=\#\left\{i:\theta_i^{**}\in A_r\right\}$. By using the conjugacy and also definition of DP we can write:

$$G_0(A_1),...,G_0(A_k)\,|\,\theta_1^{**},...,\theta_n^{**}\sim Dir\left(\gamma H\left(A_1\right)+m_{\bullet 1},...,\gamma H\left(A_k\right)+m_{\bullet k}\right) \tag{A1.7}$$

This shows the posterior of G_0 is a DP with concentration parameter equal to $\gamma+\sum_{r=}^{k}m_{\bullet r}=\gamma+m_{\bullet\bullet}$ and mean of $\frac{\gamma H+\sum_{i=1}^{n}\delta_{\theta_i^{**}}}{\gamma+m_{\bullet\bullet}}$. We also know $\sum_{i=1}^{n}\delta_{\theta_i^{**}}=\sum_{k=1}^{K}m_{\bullet k}\delta_{\theta_k^{**}}$ so we can write:

$$G_0 \mid \gamma, H, \mathbf{\theta}^{**} \sim DP\left(\gamma + m_{\bullet\bullet}, \frac{\gamma H + \sum_{k=1}^{K} m_{\bullet k}\delta_{\theta_k^{**}}}{\gamma + m_{\bullet\bullet}}\right) \tag{A1.8}$$

Derivation of equation(3.9):

Similar to the last proof, let $A_1, A_2, ..., A_k$ be a finite measureable partition of Θ and $n_{j\bullet k} = \#\left\{i : \theta_{ji} \in A_r\right\}$

$$G_j(A_1), ..., G_j(A_k) \mid \theta_{j1}, ..., \theta_{jn} \sim Dir\left(\alpha G_0\left(A_1\right) + n_{j\bullet 1}, ..., \alpha G_0\left(A_k\right) + n_{j\bullet k}\right) \tag{A1.9}$$

So G_j is a DP with concentration $\alpha + \sum_{k=1}^{K} n_{j\bullet k} = \alpha + n_{j\bullet\bullet}$ and mean $\frac{\alpha G_0 + \sum_{i=1}^{n} \delta_{\theta_{ji}}}{\alpha + n_{j\bullet\bullet}}$ and finally we can write:

$$G_j \mid \alpha, G_0, \mathbf{\theta} \sim DP\left(\alpha + n_{j\bullet\bullet}, \frac{\alpha G_0 + \sum_{k=1}^{K} n_{j\bullet k}\delta_{\theta_k^{**}}}{\alpha + n_{j\bullet\bullet}}\right) \tag{A1.10}$$

Derivation of equation(3.10):

For $A \subset \Theta$:

$$\begin{aligned} p\left(\theta_{ji} \in A \mid \theta_{j1} ... \theta_{ji-1}\right) &= E\left[G_j\left(A\right) \mid \theta_{j1} ... \theta_{ji-1}\right] \\ &= \frac{1}{\alpha + n_{j\bullet\bullet}}\left(\alpha G_0(A) + \sum_{k=1}^{K} n_{j\bullet k}\delta_{\theta_k^{**}}\right) \end{aligned} \tag{A1.11}$$

$$\begin{aligned} & n_{j\bullet k} = \sum_{\left\{t:\theta_{jt}^{*} = \theta_k^{**}\right\}} n_{jt\bullet}\delta_{\theta_{jt}^{*}} \\ & \Rightarrow \sum_{k=1}^{K} n_{j\bullet k}\delta_{\theta_k^{**}} = \sum_{k=1}^{K} \sum_{\left\{t:\theta_{jt}^{*} = \theta_k^{**}\right\}} n_{jt\bullet}\delta_{\theta_{jt}^{*}} = \sum_{t=1}^{m_{j\bullet}} n_{jt\bullet}\delta_{\theta_{jt}^{*}} \end{aligned} \tag{A1.12}$$

From(A1.11) and (A1.12):

$$\theta_{ji} \mid \theta_{j1} ... \theta_{ji-1}, \alpha, G_0 \sim \frac{1}{\alpha + n_{j\bullet\bullet}}\left(\alpha G_0 + \sum_{t=1}^{m_{j\bullet}} n_{jt\bullet}\delta_{\theta_{jt}^{*}}\right) \tag{A1.13}$$

Derivation of Equation(3.11) is very similar to the above lines and we just need to calculate the expectation of G_0 instead of G_j .

Deration of Equation(3.12)

We know that $G_0 \mid \gamma, H, \boldsymbol{\theta}^{**} \sim DP\left(\gamma + m_{\bullet\bullet}, \dfrac{\gamma H + \sum_{k=1}^{K} m_{\bullet k}\delta_{\theta_k^{**}}}{\gamma + m_{\bullet\bullet}}\right)$ and also we know that we can write G_0 has two parts; one is a draw form a DP and the other is a draw from a multinomial distribution:

$$G_0 = \sum_{k=1}^{\infty} \lambda_k \delta_{\theta_k} = \sum_{j=1}^{\infty} \beta_0 \alpha_j \delta_{\theta_j} + \sum_{k=1}^{K} \beta_k \delta_{\theta_k^{**}} \tag{A1.14}$$

Let $(A_0, A_1, ..., A_k)$ be a partition on Θ where A_0 contains the whole Θ except spikes located at θ_k^{**} and $A_j, j = 1, ..., K$ contains spikes. We can write:

$$\begin{gathered} \left(G_0(A_0), G_0(A_1), ..., G_0(A_k)\right) \sim Dir\left(\alpha_0 F(A_0), ..., \alpha_0 F(A_k)\right) \\ \alpha_0 = \gamma + m_{\bullet\bullet}, \quad F = \frac{\gamma H + \sum_{k=1}^{K} m_{\bullet k}\delta_k^{**}}{\gamma + m_{\bullet\bullet}} \end{gathered} \tag{A1.15}$$

And as a result we can write:

$$\left(\beta_0, \beta_1, ..., \beta_K\right) \sim Dir\left(\gamma, m_{\bullet 1}, ..., m_{\bullet K}\right) \tag{A1.16}$$

Derivation of equation (3.13) follows the same lines.

APPENDIX B: DERIVATION OF HDP-HMM RELATIONSHIPS

B.1. Derivation of the posterior distribution for $(\mathbf{z_t}, \mathbf{s_t})$

Lemma B.1.1

$$\int \prod_{k=1}^{K} \pi_k^{\alpha_k - 1} d\pi = \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)} \int \underbrace{\left(\frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \pi_k^{\alpha_k - 1} \right)}_{Dir(\alpha_k)} d\pi = \frac{\prod_{k=1}^{K} \Gamma(\alpha_k)}{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)} \tag{B1.1}$$

•

Derivation of $p(z_t = k \mid z_{\setminus t}, \beta, \alpha, \kappa)$:

By using the chain rule and graphical model of Figure 3 we can write:

$$\begin{aligned} p(z_t = k \mid z_{\setminus t}, \beta, \alpha, \kappa) &\propto \int_{\pi} \prod_i p(\pi_i \mid \alpha, \beta, \kappa) \prod_{\tau} p\left(z_\tau \mid \pi_{z_{\tau-1}}\right) d\pi \\ &\propto \int_{\pi} p(z_{t+1} \mid \pi_k)\, p\left(z_t = k \mid \pi_{z_t - 1}\right) \prod_i \left(p(\pi_i \mid \alpha, \beta, \kappa) \prod_{\tau \mid z_{\tau-1} = i, \tau \neq t, t+1} p(z_\tau \mid \pi_i) \right) d\pi \\ &\propto \int_{\pi} p(z_{t+1} \mid \pi_k)\, p\left(z_t = k \mid \pi_{z_t - 1}\right) \prod_i p\left(\pi_i \mid \{z_\tau \mid z_{\tau-1} = i, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) d\pi \end{aligned} \tag{B1.2}$$

For $z_{t-1} = j$ and $k \neq j$ we can write:

$$\begin{aligned} p(z_t = k \mid z_{\setminus t}, \beta, \alpha, \kappa) \propto & \int_{\pi_k} p(z_{t+1} \mid \pi_k) p\left(\pi_k \mid \{z_\tau \mid z_{\tau-1} = k, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) d\pi_k \\ & \int_{\pi_j} p\left(z_t = k \mid \pi_j\right) p\left(\pi_j \mid \{z_\tau \mid z_{\tau-1} = j, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) d\pi_j \\ & \propto p\left(z_{t+1} \mid \{z_\tau \mid z_{\tau-1} = k, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) \\ & \quad p\left(z_t = k \mid \{z_\tau \mid z_{\tau-1} = j, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) \end{aligned} \tag{B1.3}$$

For $k = j$

$$\begin{aligned} p(z_t = j \mid z_{\setminus t}, \beta, \alpha, \kappa) \propto & \int_{\pi_j} p\left(z_{t+1} \mid \pi_j\right) p\left(z_t = j \mid \pi_j\right) p\left(\pi_j \mid \{z_\tau \mid z_{\tau-1} = j, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) d\pi_j \\ & \propto p\left(z_t = j, z_{t+1} \mid \{z_\tau \mid z_{\tau-1} = j, \tau \neq t, t+1\}, \beta, \alpha, \kappa\right) \end{aligned} \tag{B1.4}$$

Using the fact that z_τ has a multinomial distribution, $\pi_j \mid \alpha, \beta \sim Dir\left(\alpha\beta_1, ..., \alpha\beta_j + \kappa, ..., \alpha\beta_K, \alpha\beta_{K+1}\right)$, and by using Lemma B.1.1:

$$
\begin{aligned}
p\left(\left\{z_\tau \mid z_{\tau-1}=i\right\} \mid \beta,\alpha,\kappa\right) &= \int_{\pi_i} p\left(\pi_i \mid \beta,\alpha,\kappa\right) p\left(\left\{z_\tau \mid z_{\tau-1}=i\right\} \mid \pi_i\right) d\pi_i \\
&= \int_{\pi_i} \frac{\Gamma\left(\sum_k \alpha\beta_k + \kappa\delta\left(k,i\right)\right)}{\prod_k \Gamma\left(\alpha\beta_k + \kappa\delta\left(k,i\right)\right)} \prod_{k=1}^{K+1} \pi_{ik}^{\alpha\beta_k + \kappa\delta(k,i)-1} \prod_{k=1}^{K+1} \pi_{ik}^{n_{ik}} d\pi_i \\
&= \frac{\Gamma\left(\sum_k \alpha\beta_k + \kappa\delta\left(k,i\right)\right)}{\prod_k \Gamma\left(\alpha\beta_k + \kappa\delta\left(k,i\right)\right)} \frac{\prod_k \Gamma\left(\alpha\beta_k + \kappa\delta\left(k,i\right) + n_{ik}\right)}{\Gamma\left(\sum_k \alpha\beta_k + \kappa\delta\left(k,i\right) + n_{ik}\right)} \\
&= \frac{\Gamma\left(\alpha+\kappa\right)}{\Gamma\left(\alpha+\kappa+n_{i\bullet}\right)} \prod_k \frac{\Gamma\left(\alpha\beta_k + \kappa\delta\left(k,i\right) + n_{ik}\right)}{\Gamma\left(\alpha\beta_k + \kappa\delta\left(k,i\right)\right)}
\end{aligned}
\tag{B1.5}
$$

In the equation above, n_{ik} denotes the number of transitions from state i to state j. Using (B1.5), (B1.3), (B1.4), and after some algebra we can obtain:

$$
p\left(z_t = k \mid z_{\setminus t}, \beta, \alpha, \kappa\right) \propto \begin{cases} \left(\alpha\beta_k + n_{z_{t-1}}^{-t} + \kappa\delta\left(z_{t-1},k\right)\right)\left(\dfrac{\alpha\beta_{z_{t+1}} + n_{kz_{t+1}}^{-t} + \kappa\delta\left(z_{t-1},k\right)\delta\left(k,z_{t+1}\right)}{\alpha + n_{k\bullet}^{-t} + \kappa + \delta\left(z_{t-1},k\right)}\right) & k \in \{1,\ldots,K\} \\ \dfrac{\alpha^2 \beta_{\bar{k}} \beta_{z_t+1}}{\alpha+\kappa}, & k = K+1 \end{cases}
\tag{B1.6}
$$

Equation (4.5) can be obtained similar to (3.10). Notice in this case that for each state we have a DP and therefore numbers of data points for DP are all data points associated with that state.

Equation (4.6) can be obtained similar to (3.16). The only difference is that we only consider observations assigned to state $z_t = k$ and $s_t = j$.