measures. It is parameterized by a base measure and a concentration parameter. A draw from a DP generates a finite set of samples from the base measure (the concentration parameter controls the number of samples). A key advantage of DP’s is that the number of unique parameters (i.e., clusters) can grow and adapt to each data set depending on its size and characteristics. Under this new probability model, data points are generated in the following way:

1. Sample from the DP, $G \sim DP(\gamma, H_\alpha \times H_\lambda)$. $\gamma$ is the concentration parameter, and $H_\alpha$ and $H_\lambda$ are the base measures for $F_T(\phi)$ and $F_D(\theta)$ respectively.
2. For each data point, $x_i$, sample a data and transformation parameter pair, $(\theta_i, \varphi_i) \sim G$.
3. Sample a transformation and canonical data item from their distributions, $y_i \sim F_D(\theta_i)$ and $\rho_i \sim F_T(\phi_i)$.
4. Transform the canonical data item to generate the observed sample, $x_i = \tau(y_i, \rho_i)$.

Figure 6 depicts the generative process as described above (distributional form, right) and in the more traditional graphical representation with the cluster random variable, $z$, and mixture weights, $\pi$, made explicit (left).

Our model can thus be seen as an extension of the standard Bayesian infinite mixture model where we introduced an additional latent variable, $\rho_i$, for each data point to represent its transformation. Several existing alignment models [11, 12, 21, 23] can be viewed as similar extensions to other standard generative models. Sometimes the transformations are applied to other model parameters instead of data points as in the case of transformed Dirichlet processes (TDP) [29]. TDP is an extension of hierarchical Dirichlet processes where global mixture components are transformed before being reused in each group. The challenge in introducing additional latent variables is in designing efficient learning schemes that can accommodate this increase in model complexity.

3.1 Learning

We consider two different learning schemes for this model. The first is a blocked, Rao-Blackwellized Gibbs sampler, where we sample both the cluster assignment $z_i$, and transformation parameters $\rho_i$, simultaneously:

\[
(z_i^{(t)}, \rho_i^{(t)}) \sim p(z_i, \rho_i | z_{-i}^{(t)}, \rho_{-i}^{(t)}, x, \gamma, \alpha, \lambda) \\
\propto p(z_i | z_{-i}^{(t)}, \gamma)p(\rho_i | \rho_{-i}^{(t)}, \alpha)p(y_i | y_{-i}^{(t)}, \lambda).
\]

As with the BA model, we approximate $p(\rho_i | \rho_{-i}^{(t)}, \alpha)p(y_i | y_{-i}^{(t)}, \lambda)$ with a point estimate based on its mode. Consequently this learning scheme is a direct generalization of the one derived for the BA model. Note that $p(z_i | z_{-i}^{(t)}, \gamma)$ is the cluster predictive distribution based on the Chinese restaurant process (CRP) [2].

While this sampler is effective (it produced the positive result in Figure 1) it scales linearly with the number of clusters and computing the most likely transformation for a cluster is an expensive operation. We designed an alternative sampling scheme that does not require the expensive mode computation and whose running time is independent of the number of clusters.

The second sampler further integrates out the transformation parameter, and only samples the cluster assignment. We now derive an implementation for this sampler.

\[
\forall i=1:N \quad z_i^{(t)} \sim p(z_i | z_{-i}^{(t)}, \lambda) \\
\propto p(z_i, x_i | z_{-i}^{(t)}, x_{-i}, \gamma, \alpha, \lambda) \\
= p(z_i | z_{-i}^{(t)}, \gamma)p(x_i | z_{-i}^{(t)}, x_{-i}, \alpha, \lambda).
\]

\[
p(x_i | z, x_{-i}, \alpha, \lambda) \\
= \int_{\theta} \int_{\varphi} \int_{\rho_i} p(x_i, \rho_i, \theta, \varphi | z, x_{-i}, \alpha, \lambda) \ d\rho_i \ d\varphi \ d\theta \\
= \int_{\theta} \int_{\varphi} \left( \int_{\rho_i} p(x_i, \rho_i | z, \theta, \varphi, \alpha, \lambda) \ d\rho_i \right) \cdots \int_{\rho_i} p(\theta, \varphi | z_{-i}, x_{-i}, \alpha, \lambda) \ d\varphi \ d\theta \\
\approx \int_{\rho_i} p(x_i, \rho_i | z_i, \hat{\theta}, \hat{\varphi}, \alpha, \lambda) \ d\rho_i, \quad s.t. \quad (\hat{\theta}, \hat{\varphi}) = \arg\max_{(\theta, \varphi)} p(\theta, \varphi | z_{-i}, x_{-i}, \alpha, \lambda) \\
= \int_{\rho_i} p(\rho_i, \hat{\varphi} | z_i, \alpha) p(x_i | \rho_i, z_i, \hat{\theta}, \lambda) \ d\rho_i \\
= \int_{\rho_i} p(\rho_i, \hat{\varphi} | z_i, \alpha) p(x_i | \rho_i, \hat{\theta}_z, \lambda) \ d\rho_i.
\]