FINDING LATENT SOURCES IN RECORDED MUSIC WITH A SHIFT-INVARIANT HDP

Matthew D. Hoffman
Dept. of Computer Science
Princeton University
Princeton, New Jersey, USA
mdhoffma@cs.princeton.edu

*David M. Blei**
Dept. of Computer Science
Princeton University
Princeton, New Jersey, USA
blei@cs.princeton.edu

Perry R. Cook
Dept. of Computer Science
Dept. of Music
Princeton University
Princeton, New Jersey, USA
prc@cs.princeton.edu

ABSTRACT

We present the Shift-Invariant Hierarchical Dirichlet Process (SIHDP), a nonparametric Bayesian model for modeling multiple songs in terms of a shared vocabulary of latent sound sources. The SIHDP is an extension of the Hierarchical Dirichlet Process (HDP) that explicitly models the times at which each latent component appears in each song. This extension allows us to model how sound sources evolve over time, which is critical to the human ability to recognize and interpret sounds. To make inference on large datasets possible, we develop an exact distributed Gibbs sampling algorithm to do posterior inference. We evaluate the SIHDP's ability to model audio using a dataset of real popular music, and measure its ability to accurately find patterns in music using a set of synthesized drum loops. Ultimately, our model produces a rich representation of a set of songs consisting of a set of short sound sources and when they appear in each song.

1. INTRODUCTION

Much interesting work has been done in recent years on analyzing and finding good representations of music audio data for tasks such as content-based recommendation, audio fingerprinting, and automatic metadata generation. A common approach is to adapt the bag-of-words exchangeability assumption from text modeling and build a statistical model of a song or class of songs that learns the statistical properties of feature vectors describing short (commonly 10-500 ms) frames of audio. Although this approach makes modeling simpler, it fails to take into account the way that sounds evolve over time, and can only model the qualities of the mixed audio signal, not of individual sounds that occur simultaneously. In this paper, we will present the Shift-Invariant Hierarchical Dirichlet Process (SIHDP), a generative model that moves beyond this approach and allows us to represent songs in terms of the instruments and other sounds that generated them.

The same instruments tend to appear in multiple recordings in different combinations, and without hand-generated metadata there is no way of knowing a priori how many or which sources will appear in a given recording. This suggests that a model based on the Hierarchical Dirichlet Process (HDP) would be ideally suited to modeling groups of songs, since it represents groups of observations (such as songs) as being generated by an initially unspecified number of shared latent components [1].

However, the HDP requires that our observations be directly comparable, which is not the case for audio data. Human listeners need to hear how a sound evolves over time to recognize and interpret that sound, but computers cannot directly observe when events in audio signals begin and end. We therefore modified the HDP to make it invariant to shifts in time by explicitly modeling when in each song latent sources appear.

This allows us to discover a shared vocabulary of latent sources that describe different events in our set of songs, and to produce a rough transcription of each song in terms of that shared vocabulary. This transcription provides a rich representation of our songs with which we can compare and analyze our songs.

We perform posterior inference on the SIHDP using Gibbs sampling. To make it feasible to do inference on large data sets in a reasonable amount of time, we also develop an exact parallel Gibbs sampler for the SIHDP that can also be applied to the original HDP.

2. A SHIFT-INVARIANT NONPARAMETRIC BAYESIAN MODEL

We define a probabilistic generative model for recorded songs. We assume that a song is generated by repeatedly selecting a sonic component from a set of available components and then selecting both a time at which it occurs and an amplitude with which it is manifested. Such a component might be, for example, a snare drum or the note middle C on a piano. Components may overlap in time. (This resembles the process by which a sample-based sequencer produces audio.)

Below, we will present a probabilistic generative model that corresponds to this process. Rather than operate directly on a time-domain representation of audio, we use a quantized time-

* David M. Blei is supported by ONR 175-6343, NSF CAREER 0745520, and grants from Google and Microsoft.

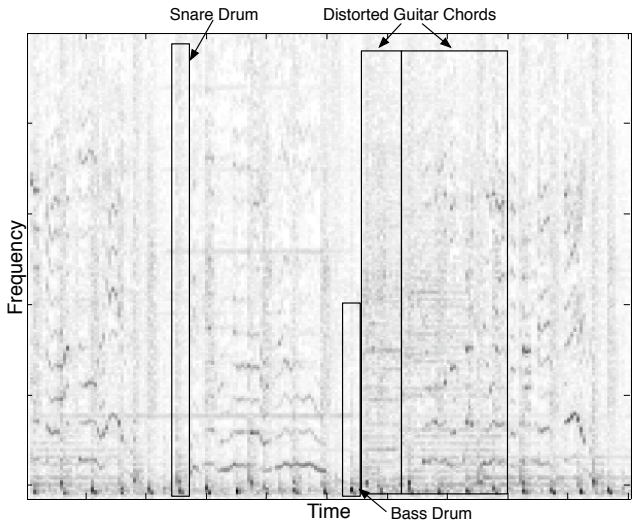

Figure 1: Spectrogram of 4.64 seconds (200 512-sample windows) of the AC/DC song "Dirty Deeds Done Dirt Cheap," annotated with the locations in time and frequency of a few instrument sounds.

frequency representation that is robust to imperceptible changes in phase. Further, this representation allows us to adapt existing models designed to handle counts data.

2.1. Data Representation

We represent each song using a quantized time-frequency spectrogram representation derived from the Short-Time Fourier Transform (STFT). First, we divide the song into a series of W short non-overlapping frames of S samples each. We multiply each frame by the Hanning window and compute the magnitude spectrum of the Discrete Fourier Transform (DFT) for that window of samples. This yields $B = \frac{S}{2} + 1$ coefficients giving the amplitude in that window of evenly spaced frequencies from 0 Hz to one half of the sampling rate.

After doing this for each frame, we have a $B \times W$ matrix $\hat{\boldsymbol{y}}_j$ of non-negative real numbers, with $\hat{y}_{jbw}$ giving the amplitude of DFT frequency bin b at time step w. Since the overall amplitude scale of digital audio is arbitrary, we can normalize our spectrograms $\hat{\boldsymbol{y}}_j$ so that $\sum_{b=0}^{B} \sum_{w=1}^{W} \hat{y}_{jbw} = 1$, and $\hat{\boldsymbol{y}}_j$ defines a multinomial probability distribution over times w and frequencies b.

Finally, we transform each normalized $\hat{\boldsymbol{y}}_j$ into quantized counts data whose empirical distribution approximates $\hat{\boldsymbol{y}}_j$. We multiply the normalized $\hat{\boldsymbol{y}}_j$ by a constant $\nu \times W \times B$ and round the result to get the number of observed magnitude "quanta" $\bar{y}_{jbw}$ in bin b at time w of song j:

$$\bar{y}_{jbw} = \text{round}(\nu W B \hat{y}_{jbw}) \tag{1}$$

$$N_j = \sum_{b=1}^{B} \sum_{w=1}^{W} \bar{y}_{jbw} \tag{2}$$

ν is roughly the average number of quanta per time/bin pair, and N_j is the total number of observed quanta in song j. We use the notation $y_{ji} = \{w_{ji}, b_{ji}\}$ to refer to the ith (exchangeable) quantum of energy in song j as occurring at time w_{ji} and bin b_{ji}, for $i \in \{1, \ldots, N_j\}$.

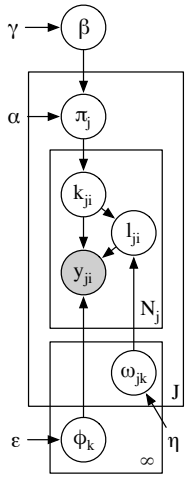

Figure 2: The graphical model for our shift-invariant HDP, described in section 2.2.

Although we only discuss DFT spectrogram data in this paper, our model could also be applied to other time-frequency representations such as the constant-Q spectrogram or the output of a wavelet transform.

2.2. Generative Process

We present the Shift-Invariant Hierarchical Dirichlet Process (SI-HDP), a generative model for our quantized spectrogram data that is an extension of the Hierarchical Dirichlet Process (HDP) [1] with discrete observations. The HDP assumes an infinite number of multinomial mixture components $\boldsymbol{\phi}_k$ drawn independently from a Dirichlet prior with parameter ϵ:

$$\boldsymbol{\phi}_k \sim \text{Dir}(\epsilon, \ldots, \epsilon)$$

An infinite vector $\boldsymbol{\beta}$ defining the global proportions of these components is drawn from a stick-breaking process with concentration parameter γ (denoted GEM(γ)). Each group j of observations draws a group-level set of proportions $\boldsymbol{\pi}_j$ from a Dirichlet Process (DP) with concentration parameter α and the multinomial defined by $\boldsymbol{\beta}$ as its base distribution:

$$\boldsymbol{\beta} \sim \text{GEM}(\gamma); \quad \boldsymbol{\pi}_j \sim \text{DP}(\alpha, \text{Mult}(\boldsymbol{\beta}))$$

The ith observation in group j is drawn by first choosing a component k_{ji} from the group-level proportion distribution $\boldsymbol{\pi}_j$, and then drawing the observation y_{ji} from $\boldsymbol{\phi}_{k_{ji}}$:

$$k_{ji} \sim \text{Mult}(\boldsymbol{\pi}_j); \quad y_{ji} \sim \boldsymbol{\phi}_{k_{ji}}$$

We will use a variant of the HDP to analyze groups of songs. Our analysis will find:

1. The set of components used to generate those songs.
2. When and how prominently those components appear in each song.

In order to do this, we need to explicitly model how prominent each component is at any given time in each song.

The SIHDP extends the HDP by modeling each observed time w_{ji} as a sum of two terms: a base time $c_{ji} \in \{1, \ldots, C\}$ and a discrete time offset $l_{ji} \in \{-C+1, \ldots, W-1\}$. We define $L = W + C - 1$ to be the size of this set of possible time offsets. We set C to be the length of the latent components that we wish to model (which will be short relative to the song). The time offsets l can take on any range of values such that there is some c for which $l + c \in \{1, \ldots, W\}$.

As in the HDP, we begin by drawing a set of latent components $\boldsymbol{\phi}$ from a symmetric Dirichlet prior with parameter ϵ, but $\boldsymbol{\phi}$ is now a two-dimensional joint distribution over base times c and frequency bins b. Each $\boldsymbol{\phi}$ can be interpreted as a normalized spectrogram of a short audio source. The global component proportion vector $\boldsymbol{\beta}$ is again drawn from a stick-breaking process with concentration parameter γ, and the song-level component proportion vector $\boldsymbol{\pi}_j$ for each song j is drawn from a DP with concentration parameter α and base distribution Mult($\boldsymbol{\beta}$).

Each component k in song j in the SIHDP has a set of multinomial distributions $\boldsymbol{\omega}_{jk}$ over time offsets drawn from a symmetric Dirichlet prior with parameter η.

Each observed quantum of energy y_{ji} consists of a time w_{ji} and a frequency bin b_{ji} at which the quantum appears. To generate y_{ji}, we first select a component k_{ji} to generate the quantum. We draw k_{ji} from Mult($\boldsymbol{\pi}_j$), the song-level distribution over components. We then draw a base time c_{ji} and frequency b_{ji} jointly from $\boldsymbol{\phi}_{k_{ji}}$, and draw a time offset l_{ji} from the distribution over time offsets $\boldsymbol{\omega}_{jk_{ji}}$ for component k_{ji} in song j.

The observed quantum appears at time $w_{ji} = c_{ji} + l_{ji}$ and frequency b_{ji}.

The full generative process for the SIHDP is:

$$\begin{aligned} \boldsymbol{\phi}_k &\sim \text{Dir}(\epsilon, \ldots, \epsilon) & \boldsymbol{\omega}_{jk} &\sim \text{Dir}(\eta, \ldots, \eta) \\ \boldsymbol{\beta} &\sim \text{GEM}(\gamma) & \boldsymbol{\pi}_j &\sim \text{DP}(\alpha, \text{Mult}(\boldsymbol{\beta})) \\ k_{ji} &\sim \text{Mult}(\boldsymbol{\pi}_j) & l_{ji} &\sim \text{Mult}(\boldsymbol{\omega}_{jk_{ji}}) \\ c_{ji}, b_{ji} &\sim \text{Mult}(\boldsymbol{\phi}_{k_{ji}}) & w_{ji} &= c_{ji} + l_{ji} \\ y_{ji} &= \{w_{ji}, b_{ji}\} \end{aligned} \tag{3}$$

The SIHDP is a hierarchical nonparametric Bayesian version of Shift-Invariant Probabilistic Latent Component Analysis (SI-PLCA) [2]. It improves on SI-PLCA by allowing components to be shared across multiple songs, and by automatically determining the number of latent components that are needed to explain the data.

This SIHDP is also related to the Transformed Dirichlet Process (TDP) in that draws from a countably infinite global set of mixture components undergo transformations to generate observations [3]. In the TDP, however, transformations are associated with observations indirectly through table assignments in the Chinese Restaurant Franchise (CRF). This means that the concentration paramater α influences both the group-level component proportions $\boldsymbol{\pi}$ and the number of transformations that a group of observations (such as a song or an image) can take on; a high α simultaneously makes each $\boldsymbol{\pi}_j$ less likely to diverge from the global component proportions $\boldsymbol{\beta}$ and makes a large number of transformations in each group j more likely.

Our model takes a simpler approach, directly associating each observation y_{ji} with a transformation l_{ji} that depends only on the cluster assignment k_{ji} and a group-level multinomial distribution over transformations $\boldsymbol{\omega}_{jk_{ji}}$. We do not need to use the CRF machinery of the HDP to generate transformations, since our set of transformations is discrete. We note that this decoupled approach can be generalized to continuous transformations by using a set of DP's for each group to discretize a space of continuous transformations. This may be worth exploring in other models.

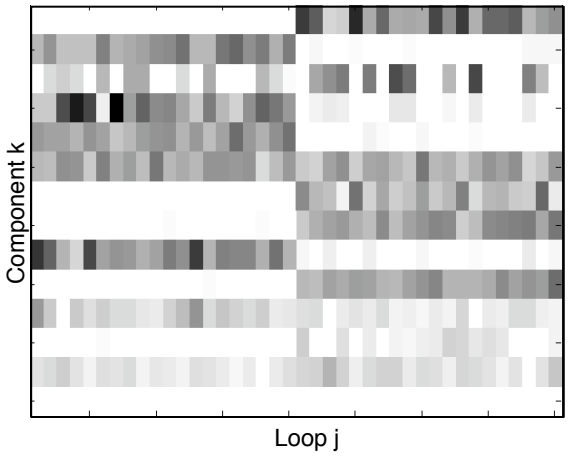

Figure 3: A graphical representation of our model's MAP estimate of $\boldsymbol{\pi}$ for the 40 synthetic drum loops. A darker pixel in row k of column j indicates a higher relative proportion π_{jk} of latent component k in song j.

Although this paper is focused on the applications of the SIHDP to music, it could equally well be applied to any of the other application areas described in [2], such as images or video. Likewise, although we only discuss shift-invariance in time, analogous models can be constructed that are invariant to shifts in frequency at extra computational expense. A log-frequency representation such as the constant-Q transform would be a more appropriate input for such a model than the linear frequency spectrogram.

To learn the posterior distribution over the model parameters conditioned on the observed spectrograms, we adapt the direct assignment Gibbs sampler from [1]. This Gibbs sampler gives us a set of samples from the posterior distribution over a set of the variables in our model, which we then use to compute a Maximum A Posteriori (MAP) estimate of the remaining parameters. Full details of the inference procedure can be found in appendix A.

3. EVALUATION

We conducted several experiments to test the SIHDP on music audio data—one using synthetic drum loops and three using songs taken from the CAL500 dataset, which consists of 500 songs of various genres of Western popular music each recorded by a different artist within the last 50 years [5]. In all experiments, we placed a gamma$(1, 0.0001)$ prior on both α and γ, and set $\epsilon = 0.02$ and $\eta = 0.01$.

3.1. Drum Loop Transcription

We synthesized a set of 40 randomly generated 32-beat drum loops lasting 6 seconds each. We studied the SIHDP's ability to discover

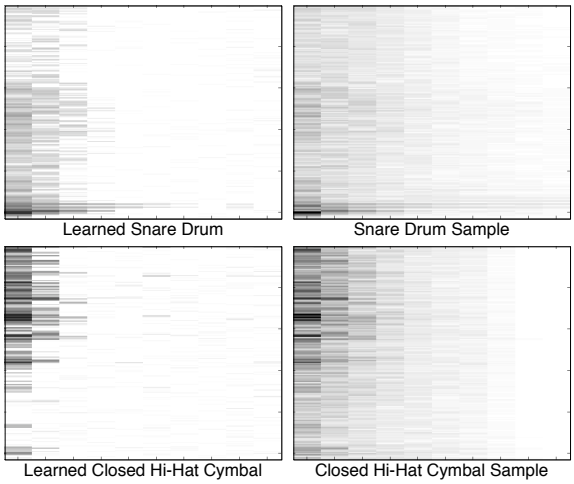

Figure 4: Left: Two latent distributions ϕ_k discovered by our model. Right: Spectrograms of two drum samples closely matching the latent components at left.

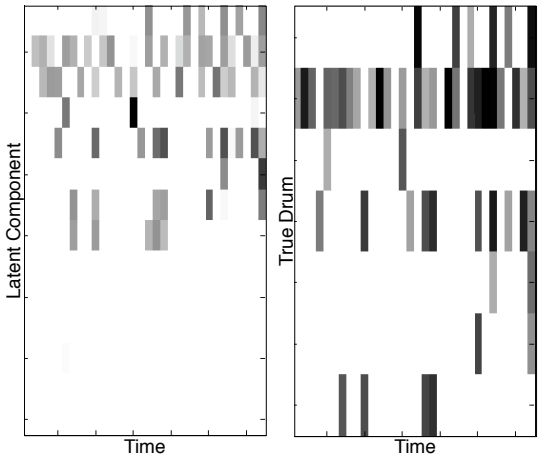

Figure 5: Left: An unsupervised transcription $\hat{\boldsymbol{\omega}}$ generated by our model of a drum loop. Right: The actual times and amplitudes of the drum loop. Darker pixels correspond to higher amplitudes.

the drum sounds that were used to create the files, and when in each file they appear. We used a simple algorithm to generate the loops: for each drum s at each beat i in song j, we draw a Bernoulli variable $r_{sij} \sim \text{Bern}(p_s)$ that indicates whether or not drum s is present at beat i. If it is, then we draw its amplitude $a_{sij} \sim \text{Unif}(0.3, 0.9)$, otherwise we set $a_{sij} = 0$. Audio was synthesized using the ChucK music programming language [6] and two sets of drum samples from Apple's GarageBand.

Our objective was to recover from the audio alone an estimate of $\boldsymbol{a}$ for each drum in each song, without any prior knowledge of what the drum samples sound like (aside from their maximum length), how many there are, or how frequently they appear.

We ran our Gibbs sampler until the posterior likelihood failed to increase for 20 iterations on the 40 synthesized files, choosing $C = 10$ and $\nu = 0.25$. We then computed a MAP estimate of the time offset distribution $\boldsymbol{\omega}|\boldsymbol{k}, \boldsymbol{l}$, and calculated a distribution $\boldsymbol{\omega}'$ quantized to 32 beats, so that ω'_{jki} is the probability that a quantum of energy generated by component k in song j will fall anywhere in beat i. $\hat{\omega}_{jki} = \pi_{jk}\omega'_{jki}$ is then the relative prominence of component k at beat i in loop j.

Figure 3 graphically represents the distribution $\boldsymbol{\pi}$ discovered by our model. Note that most of the latent components tend to appear in either the first 20 loops or the last 20, but not both. This is because the first 20 loops were generated using a different set of drum samples than the last 20, and our model was able to distinguish between the two synthetic drum kits.

We evaluated the Bhattacharyya distance for each song between the joint distribution over components and times defined by $\hat{\boldsymbol{\omega}}_j$ and the joint distribution over drums and times defined by our ground truth $\boldsymbol{a}_j$ (normalized so that it can be treated as a multinomial distribution). The Bhattacharyya distance between two probability distributions p and q over the same domain $\boldsymbol{X}$ is a symmetric measure of the dissimilarity of those two distributions, and is defined as

$$D_B(p, q) = -\log\left(\sum_{x \in \boldsymbol{X}} \sqrt{p(x)q(x)}\right) \tag{4}$$

We assume (naïvely) that the component k in song j that corresponds to drum s is the one that maximizes $\text{Corr}(\hat{\boldsymbol{\omega}}_{jk}, \boldsymbol{a}_{js})$. The average Bhattacharyya distance between our transcription and the ground truth was 0.4236 with a standard error of 0.0204. The average Bhattacharyya distance obtained by repeating the experiment with a a normalized matrix of numbers drawn uniformly at random substituted for $\hat{\boldsymbol{\omega}}_j$ was 1.1923 with a standard error of 0.0131. The SIHDP did dramatically better than chance at transcribing the drum tracks.

Figure 4 compares the spectrograms of two of the drum samples used to generate our data with the discovered latent components they most closely match. Figure 5 compares the ground truth transcription $\boldsymbol{a}$ with the SIHDP's transcription $\hat{\boldsymbol{\omega}}$ for a single drum loop. The rows of $\hat{\boldsymbol{\omega}}$ have been manually ordered to make their correspondance to the rows of $\boldsymbol{a}$ clearer. The second drum seems to have been split across two components, but most of the drums have a clear one-to-one mapping to latent components, which confirms our quantitative results. The empty rows correspond to latent components not used to model this loop.

3.2. Experiments on Recorded Popular Music

We ran our distributed Gibbs sampler on a training set of 48 songs from the CAL500 dataset to get MAP estimates of the global component proportions $\boldsymbol{\beta}$, the global components $\boldsymbol{\phi}$, the song-level component proportions $\boldsymbol{\pi}$, and the song-level offset distributions $\boldsymbol{\omega}$. We set the length C of the latent components to 20 windows. We used 2000 512-sample windows (46 seconds of audio) from each song with ν set to 1, for an average of 514,000 observations per song. The Gibbs sampler took about a day to converge running on 48 processors, and discovered 575 components.

Figure 6 shows several latent components discovered by the SIHDP from the 48 training songs. Qualitatively, these sound like (clockwise from bottom left) a bass drum, a male voice singing "aah," a snare drum, and a high-pitched whistle. While the first three components clearly correspond to real-world sound sources, it seems more likely that the fourth component is being used to model fine details of the data that are cannot be captured by the

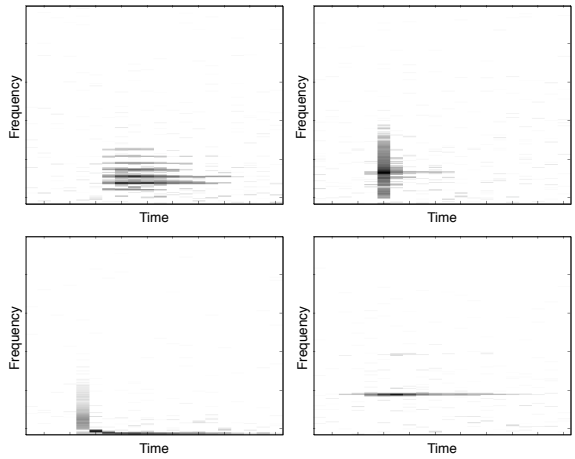

Figure 6: Four latent components discovered from 48 songs taken from the CAL500 corpus of popular music.

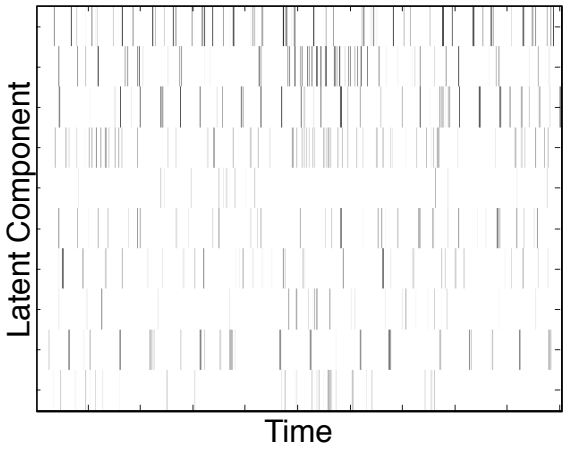

Figure 7: The 10 most prominent components of the unsupervised transcription $\hat{\boldsymbol{\omega}}$ inferred from 11 seconds of the AC/DC song "Dirty Deeds Done Dirt Cheap." Some components are relatively weak here, but become more prominent elsewhere in the song.

more complex components. Preliminary experiments with a more flexible model suggest that these simple "detail" components are less common if details of the components are allowed to vary from song to song.

Figure 7 shows the intensities $\hat{\omega}_{jkl} = \omega_{jkl}\pi_{jk}$ with which the 10 most prominent components k appear at each time offset l in the song "Dirty Deeds Done Dirt Cheap" by AC/DC. Different, but related rhythmic patterns for each component are clearly visible. Exploiting the rhythmic information in this representation may prove valuable for music information retrieval tasks.

3.2.1. Perplexity

After obtaining MAP estimates of the global component proportions $\boldsymbol{\beta}$ and the global components $\boldsymbol{\phi}$, we ran our Gibbs sampler on 400 held-out songs from the same dataset holding the global component proportions $\boldsymbol{\beta}$ and the component distributions $\boldsymbol{\phi}$ fixed at the MAP estimates from the training set, and estimated the perplexity on the held out data using the harmonic mean of the likelihoods of the data under samples from the posterior of the hidden parameters[1]. For comparison, we also built a simple DP model that also assumes an infinite set of latent components, but has each song choose a single latent component that it uses to generate every observed quantum. We estimated the perplexity of this model on the same held out data. The DP's perplexity was 1265.2, and our SIHDP model's perplexity was 62.1. This dramatic reduction in perplexity illustrates the value of modeling songs as mixtures of latent components.

4. CONCLUSION

In this paper, we presented the Shift-Invariant Hierarchical Dirichlet Process (SIHDP), a model that can discover a rich representation of groups of songs in terms of the instruments and vocal sounds that generated those songs. We developed an exact parallel Gibbs sampler that enabled us to run experiments on a significant number of songs, and showed that the SIHDP can discover latent audio sources that are shared across multiple songs, as well as when those sources occur in each song. In the future, we hope extend our model to capture the temporal structure of songs in terms of these latent sources, to allow fine details of sources to change from song to song, and to model pitch more explicitly.

[1]While this is a widely used method (cf. e.g. [7]) there is some debate in the statistics community as to its effectiveness compared with alternative estimation methods such as [8].

5. REFERENCES

[1] Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2007.

[2] P. Smaragdis, B. Raj, and M. Shashanka, "Sparse and shift-invariant feature extraction from non-negative data," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 2069–2072.

[3] E. Sudderth, A. Torralba, W. Freeman, and A. Willsky, "Describing visual scenes using transformed Dirichlet processes," in *Advances in Neural Information Processing Systems 18*, 2005.

[4] A. Asuncion, P. Smyth, and M. Welling, "Asynchronous Distributed Learning of Topic Models," in *Advances in Neural Information Processing Systems 20 (NIPS) 20*. 2008, MIT Press.

[5] Douglas Turnbull, Luke Barrington, David Torres, and Gert Lanckriet, "Towards musical query-by-semantic description using the CAL500 data set," in *ACM Special Interest Group on Information Retrieval Conference (SIGIR '07)*, 2007.

[6] Ge Wang and Perry R. Cook, "Chuck: A concurrent, on-the-fly, audio programming language," in *2003 International Computer Music Conference*, 2003.

[7] T. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National Academy of Science*, 2004.

[8] X. Meng and W. Wong, "Simulating ratios of normalizing constants via a simple identity: A theoretical exploration," *Statistica Sinica*, vol. 6, pp. 831–860, 1996.

A. APPENDIX: INFERENCE PROCEDURES

A.1. Direct Assignment Gibbs Sampler

To draw from the posterior, we adapt the direct assignment Gibbs sampler described in [1]. We integrate out all variables besides the component indicators $\boldsymbol{k}$, the time offsets $\boldsymbol{l}$, and the global component proportions $\boldsymbol{\beta}$, whose values comprise the state of the Markov chain.

Resampling the component indicators $\boldsymbol{k}$ and time offsets $\boldsymbol{l}$: First, we jointly resample each pair of variables k_{ji}, l_{ji} indicating which component k_{ji} at what time offset l_{ji} generated observation i in song j, conditioned on the values of all other indicator variables $\boldsymbol{k}_{-ji}, \boldsymbol{l}_{-ji}$, the global component proportion weights $\boldsymbol{\beta}$, the observed data $\boldsymbol{y}$, the concentration parameter α, and the prior on the mixture components $\boldsymbol{\phi}$ defined by ϵ.

$$\begin{aligned} &P(k_{ji}, l_{ji} | \boldsymbol{k}_{-ji}, \boldsymbol{l}_{-ji}, \boldsymbol{\beta}, \alpha, \epsilon, \boldsymbol{y}) \propto \\ &P(y_{ji} | \boldsymbol{k}, \boldsymbol{l}, \boldsymbol{y}_{-ji}, \epsilon) P(k_{ji}, l_{ji} | \boldsymbol{k}_{-ji}, \boldsymbol{l}_{-ji}, \boldsymbol{\beta}, \alpha, \eta) = \\ &P(y_{ji} | \boldsymbol{k}, \boldsymbol{l}, \boldsymbol{y}_{-ji}, \epsilon) P(l_{ji} | k_{ji}, \boldsymbol{l}_{-ji}, \eta) P(k_{ji} | \boldsymbol{k}_{-ji}, \boldsymbol{\beta}, \alpha) \end{aligned} \tag{5}$$

Define n_{lkj} to be the number of observations in song j coming from component k with time offset l, excluding the observation we're currently resampling. Define o_{cbjk} to be the number of observations in song j coming from component k with base time c and frequency bin b, again excluding the current observation.

Given l_{ji} and $y_{ji} = \{w_{ji}, b_{ji}\}$, we can calculate the base offset $c_{ji} = w_{ji} - l_{ji}$, and so the first term becomes:

$$\begin{aligned} &P(y_{ji} | \boldsymbol{k}, \boldsymbol{l}, \boldsymbol{y}_{-ji}, \epsilon) = P(c_{ji}, b_{ji} | \boldsymbol{k}, \boldsymbol{l}, \boldsymbol{y}_{-ji}, \epsilon) \\ &= \textstyle\int_\phi P(c_{ji}, b_{ji} | \phi) P(\phi | \boldsymbol{c}_{-ji}, \boldsymbol{b}_{-ji}, \boldsymbol{k}, \boldsymbol{l}, \epsilon) d\phi \\ &= (o_{c_{ji} b_{ji} j k_{ji}} + \epsilon) / (o_{\cdot\cdot j k_{ji}} + CB\epsilon) \end{aligned} \tag{6}$$

For a new component k, the predictive likelihood is a constant $\frac{1}{CB}$, since the prior on $\boldsymbol{\phi}$ is symmetric.

The marginal likelihood of the component indicator k_{ji} conditioned on the other $\boldsymbol{k}_{j,-i}$ in the same song j and on the global component proportions $\boldsymbol{\beta}$ is given by the Chinese restaurant franchise:

$$P(k_{ji} | \boldsymbol{k}_{j,-i}, \boldsymbol{\beta}, \alpha) = \begin{cases} \frac{n_{\cdot kj} + \alpha \beta_k}{N_j + \alpha} & \text{if } k \in \{1, \ldots, K\} \\ \frac{\alpha \beta^{\text{new}}}{N_j + \alpha} & \text{if } k = k^{\text{new}} \end{cases} \tag{7}$$

Where β^{new} is the global likelihood of choosing a component not currently associated with any observations:

$$\beta^{\text{new}} = 1 - \sum_{k=1}^{K} \beta_k \tag{8}$$

The likelihood of time offset l_{ji} conditioned on the other $\boldsymbol{l}_{j,-i}$ and on k_{ji} if $k_{ji} \in \{1, \ldots, K\}$ is given by:

$$\begin{aligned} P(l_{ji} | k_{ji}, l_{-ji}, \eta) &= \int_\omega P(l_{ji} | \omega) P(\omega | \boldsymbol{l}_{j,-i}, \eta) d\omega \\ &= \frac{n_{l_{ji} k_{ji} j} + \eta}{n_{\cdot k_{ji} j} + \eta L} \end{aligned} \tag{9}$$

The predictive likelihood for a new component k^{new} is a constant $\frac{1}{L}$, since the prior on $\boldsymbol{\omega}$ is symmetric.

Therefore, the joint posterior likelihood of k_{ji} and l_{ji} for a given observation $y_{ji} = \{c_{ji} + l_{ji}, b_{ji}\}$ conditioned on $\boldsymbol{k}_{-ji}, \boldsymbol{l}_{-ji}$, $\boldsymbol{y}_{-ji}$, $\boldsymbol{\beta}$, α, and ϵ is:

$$\begin{aligned} &P(k_{ji} = k, l_{ji} = l | \boldsymbol{k}_{-ji}, \boldsymbol{l}_{-ji}, \boldsymbol{\beta}, \epsilon, \alpha, \boldsymbol{y}) \\ &\propto \frac{(o_{c_{ji} b_{ji} j k} + \epsilon)(n_{\cdot kj} + \alpha \beta_k)(n_{lkj} + \eta)}{(o_{\cdot\cdot jk} + CB\epsilon)(n_{\cdot\cdot j} + \alpha)(n_{\cdot kj} + \eta L)} \end{aligned} \tag{10}$$

for $k \in \{1, \ldots, K\}$. For $k = k^{\text{new}}$,

$$P(k_{ji} = k^{\text{new}}, l_{ji} = l | \boldsymbol{\beta}, \alpha, N_j) \propto \frac{\alpha \beta^{\text{new}}}{CBL(N_j - 1 + \alpha)} \tag{11}$$

If $n_{\cdot k \cdot} = 0$ for some component k at some point during resampling, then that component may be eliminated from future considerations.

Creating a new mixture component: If $k_{ji} = k^{\text{new}}$, then a new mixture component needs to be created. When this happens, we draw a stick-breaking weight $s \sim \text{beta}(1, \gamma)$, set $\beta_{k^{\text{new}}} = s\beta^{\text{new}}$ and then update $\beta^{\text{new}} := (1 - s)\beta^{\text{new}}$, as in the direct assignment sampler for the HDP [1]. We choose the time offset l_{ji} uniformly at random from the set of offsets $\{w_{ji} - C + 1, \ldots, w_{ji}\}$ that are consistent with an observation at time w_{ji}.

Resampling the global mixture proportions $\boldsymbol{\beta}$: After the component indicators $\boldsymbol{k}$ and time offsets $\boldsymbol{l}$ have been resampled, we resample the global component proportions $\boldsymbol{\beta} | \boldsymbol{k}, \alpha, \gamma$ by simulating the Chinese Restaurant Franchise. Let m_{jk} be the number of tables in restaurant j eating dish k. Then

$$\boldsymbol{\beta} | m, \gamma \sim \text{Dir}(m_{\cdot 1}, \ldots, m_{\cdot K}, \gamma) \tag{12}$$

For each restaurant j and dish k, draw $m_{jk} | \alpha, \beta, n_{\cdot kj}$ as follows[2]:

1. Set $m_{jk} = 0$
2. For i $\in \{0, \ldots, n_{\cdot kj} - 1\}$:
   (a) Increment m_{jk} by $t_i \sim \text{Bernoulli}(\frac{\alpha \beta_k}{\alpha \beta_k + i})$

Once $\boldsymbol{m}$ has been drawn for all j, k, redraw $\boldsymbol{\beta}$ according to equation 12.

Sampling the components $\boldsymbol{\phi}$: We can also sample the latent components $\boldsymbol{\phi}$ instead of integrating them out—this slows convergence, but makes the distributed inference algorithm presented in the following section possible, which allows us to apply our model to larger datasets.

If we instantiate $\boldsymbol{\phi}$ rather than integrating it out, equation 10 simplifies to:

$$\begin{aligned} &P(k_{ji} = k, l_{ji} = l | \boldsymbol{k}_{-ji}, \boldsymbol{l}_{-ji}, \boldsymbol{\beta}, \epsilon, \alpha, \boldsymbol{y}) \\ &\propto \phi_{c_{ji} b_{ji} k} \frac{(n_{\cdot kj} + \alpha \beta_k)(n_{lkj} + \eta)}{(n_{\cdot\cdot j} + \alpha)(n_{\cdot kj} + \eta L)} \end{aligned} \tag{13}$$

All other updates are the same as before.

To update $\boldsymbol{\phi}_k$, we can simply draw from its posterior conditioned on the indicator variables $\boldsymbol{k}, \boldsymbol{l}$, the observations $\boldsymbol{y}$, and the prior parameter ϵ:

$$\boldsymbol{\phi}_k | \boldsymbol{k}, \boldsymbol{l}, \boldsymbol{y}, \epsilon \sim \text{Dir}(o_{1,1,\cdot,k} + \epsilon, \ldots, o_{C,B,\cdot,k} + \epsilon) \tag{14}$$

Resampling the hyperparameters α and γ: We can resample the hyperparameters α and γ in the same way as in the HDP.

[2] Note that n and o here include all observations in all songs, unlike when we were redrawing k and l.

A.2. Distributed Inference

Resampling the component indicators $\boldsymbol{k}$ and time offsets $\boldsymbol{l}$ for each observation requires $O(CKN.)$ operations per iteration. Say that our songs are all 2000 512-sample frames long (corresponding to 46 seconds at a sampling rate of 22050 Hz) and we choose $\nu = 1.0$ and $C = 20$ (corresponding to components lasting 460 ms). Then if our model discovers 200 latent components, resampling $\boldsymbol{k}$ and $\boldsymbol{l}$ will require billions of floating-point operations and memory accesses per song. This may lead to unacceptably long run times even for small datasets, particularly if the songs are heterogeneous and a larger number of components is needed to model them.

A solution to this problem is to split the work of resampling $\boldsymbol{k}$ and $\boldsymbol{l}$ across multiple processors, assigning one processor to deal with each song j. These indicator variables for each song are conditionally independent of those in all other songs given the global component proportions $\boldsymbol{\beta}$ and the components $\boldsymbol{\phi}$, so the only situation in which we have to do anything differently from the single-processor Gibbs sampler described in the previous section is when creating or eliminating components, since these actions affect the state of the global variables $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$.

We can put off eliminating components until after all $\boldsymbol{k}$ and $\boldsymbol{l}$ have been resampled. At that point, if a component k has no observations associated with it then $P(\beta_k > 0|\boldsymbol{k}, \alpha, \gamma)$ will be 0 and the component can be eliminated.

Creating a new component k^{new} is more complicated, since creating a new component involves sampling $\beta_{k^{\text{new}}}$ and $\boldsymbol{\phi}_{k^{\text{new}}}$, which alters the global state of the Markov chain in ways that affect other groups. In [4], Asuncion et al. propose an approximate Gibbs sampler for the HDP that allows each process to create new components as usual, and then merges the component ID's across processors arbitrarily. This approach is not guaranteed to converge, and they report experimental results in which it converges to a final number of topics much more slowly than a single-threaded exact Gibbs sampler does.

We instead propose a method for allowing multiple processes to create new components without sacrificing consistency. Before resampling the indicator variables $\boldsymbol{k}$ and $\boldsymbol{l}$, we draw a set of A global auxiliary components $\boldsymbol{\phi}_{K+1,\ldots,K+A}$ from their prior:

$$\boldsymbol{\phi}_{K+a} \sim \text{Dir}(\epsilon, \ldots, \epsilon) \tag{15}$$

We also augment the global component proportions $\boldsymbol{\beta}$ with a series of A additional weights partitioning the probability mass in β^{new} using the stick-breaking process:

$$\begin{aligned} s_a \sim \text{beta}(1, \gamma); \quad \beta_{K+1} &= s_1 \beta^{\text{new}} \\ \beta_{K+a} &= s_a(1 - s_{a-1})\beta_{K+a-1}; \\ \hat{\beta}^{\text{new}} &= (1 - s_A)\beta_A \end{aligned} \tag{16}$$

Effectively we have sampled from the prior an extra A latent components not associated with any observations, and assigned them weights in $\boldsymbol{\beta}$ according to the stick-breaking process. If we include these auxiliary components when resampling $\boldsymbol{k}$ and $\boldsymbol{l}$, then the model has a set of A new components to which it can assign observations without having to change any global variables. There is still a chance that a song will choose a component $\hat{k}^{\text{new}}$ for which we have not sampled a component $\boldsymbol{\phi}$, however:

$$P(k_{ji} = \hat{k}^{\text{new}}) \propto \frac{\alpha \hat{\beta}^{\text{new}}}{LB(N_j - 1 + \alpha)} \tag{17}$$

If the number of auxiliary components A is chosen to be sufficiently large, $\hat{\beta}^{\text{new}}$ will be dramatically smaller than β^{new}, and so this will be a much less likely event than choosing a component $k \in \{K+1, \ldots, K+A\}$. It is important to choose a value for A large enough that $\hat{k}^{\text{new}}$ is never chosen, since it is difficult to deal with this event in a principled way. We could simply abort this round of resampling $\boldsymbol{k}$ and $\boldsymbol{l}$, increase A, draw a new set of auxiliary variables and try again, but this could potentially introduce a bias that is hard to account for. In our experiments we chose a sufficiently large value for A that $\hat{k}^{\text{new}}$ was never chosen.

If A is large, a naïve approach introduces significant extra computation. Since there are no observations associated with the auxiliary components, however, we can sidestep this extra computation by efficiently precalculating the marginal probability of associating an observation with *any* component not yet associated with any observations. Denote this set as $\boldsymbol{k}^{\text{new}} = \{K+1, \ldots, K+A, \hat{k}^{\text{new}}\}$.

$$\begin{aligned} & P(k_{ji} \in \boldsymbol{k}^{\text{new}}|y_{ji}, \boldsymbol{\beta}, \hat{\beta}^{\text{new}}, \boldsymbol{\phi}, \alpha, N_j) \\ \propto \quad & P(y_{ji}|k_{ji} \in \boldsymbol{k}^{\text{new}}, \boldsymbol{\phi}, \boldsymbol{\beta}, \hat{\beta}^{\text{new}}) \times \\ & P(k_{ji} \in \boldsymbol{k}^{\text{new}}|\beta^{\text{new}}, \alpha, N_j) \end{aligned} \tag{18}$$

The first term can be summarized as a weighted average of the auxiliary components $\boldsymbol{\phi}$ and the likelihood of an observation drawn from a $\boldsymbol{\phi}_{\hat{k}^{\text{new}}}$

$$\begin{aligned} & P(y_{ji}|k_{ji} \in \boldsymbol{k}^{\text{new}}, \boldsymbol{\phi}, \boldsymbol{\beta}, \hat{\beta}^{\text{new}}) \\ = \quad & \sum_{c=1}^{C} \left[P\left(c, b_{ji}|k_{ji} \in \boldsymbol{k}^{\text{new}}, \boldsymbol{\phi}, \boldsymbol{\beta}, \hat{\beta}^{\text{new}}\right) \times \right. \\ & \left. P\left(l_{ji} = w_{ji} - c|k_{ji} \in \boldsymbol{k}^{\text{new}}\right) \right] \\ = \quad & \frac{1}{L\beta^{\text{new}}} \sum_{c=1}^{C} \left(\frac{\hat{\beta}^{\text{new}}}{CD} + \sum_{k=K+1}^{K+A} \phi_{cb_{ji}k} \beta_k \right) \end{aligned} \tag{19}$$

The second term is simply the prior likelihood of sitting at any empty table in the CRF:

$$P\left(k_{ji} \in \boldsymbol{k}^{\text{new}}|\beta^{\text{new}}, \alpha, N_j\right) = \frac{\alpha \beta^{\text{new}}}{N_j - 1 + \alpha} \tag{20}$$

Neither of these terms depend on the component indicators $\boldsymbol{k}$ or the time offsets $\boldsymbol{l}$, so they only need to be computed once for each possible frequency bin b before resampling $\boldsymbol{k}$ and $\boldsymbol{l}$. Then, when resampling k_{ji} we can efficiently sample whether or not $k_{ji} \in \boldsymbol{k}^{\text{new}}$. If $k_{ji} \notin \boldsymbol{k}^{\text{new}}$ (as will usually be the case) we can safely ignore all auxiliary variables.

A simpler version of this auxiliary variable method can also be applied to the original HDP formulation, as long as the global component proportions $\boldsymbol{\beta}$ and latent components $\boldsymbol{\phi}$ are sampled rather than integrated out, and the space of possible observations is discrete. Although this case is known to converge slowly, for some very large datasets this might be outweighed by the ability to deploy more computational resources.