CONTENT-BASED MUSICAL SIMILARITY COMPUTATION USING THE HIERARCHICAL DIRICHLET PROCESS

Matthew Hoffman
Princeton University
Dept. of Computer Science

David Blei
Princeton University
Dept. of Computer Science

Perry Cook
Princeton University
Dept. of Computer Science
Dept. of Music

ABSTRACT

We develop a method for discovering the latent structure in MFCC feature data using the Hierarchical Dirichlet Process (HDP). Based on this structure, we compute timbral similarity between recorded songs. The HDP is a nonparametric Bayesian model. Like the Gaussian Mixture Model (GMM), it represents each song as a mixture of some number of multivariate Gaussian distributions However, the number of mixture components is not fixed in the HDP, but is determined as part of the posterior inference process. Moreover, in the HDP the same set of Gaussians is used to model all songs, with only the mixture weights varying from song to song. We compute the similarity of songs based on these weights, which is faster than previous approaches that compare single Gaussian distributions directly. Experimental results on a genre-based retrieval task illustrate that our HDP-based method is both faster and produces better retrieval quality than such previous approaches.

1 INTRODUCTION

We develop a new method for estimating the timbral similarity between recorded songs. Our technique is based on the hierarchical Dirichlet process, a flexible Bayesian model for uncovering latent structure in high-dimensional data.

One approach to computing the timbral similarity of two songs is to train a single Gaussian or a Gaussian Mixture Model (GMM) on the Mel-Frequency Cepstral Coefficient (MFCC) feature vectors for each song and compute (for the single Gaussian) or approximate (for the GMM) the Kullback-Leibler (K-L) divergence between the two models [1]. The basic single Gaussian approach with full covariance matrix ("G1" [2]) has been successful, forming the core of the top-ranked entries to the MIREX similarity evaluation task two years running [3, 4].

Although MFCC data are not normally distributed within songs, using a richer model such as the GMM to more accurately represent their true distribution provides little or no improvement in numerous studies [2, 5, 1]. This suggests that a "glass ceiling" has been reached for this type of representation. Moreover, the computational cost of the Monte Carlo estimation procedure involved in comparing two GMMs is orders of magnitude more than that incurred by computing the K-L divergence between two single Gaussians exactly. This is a very significant issue if we want to compute similarity matrices for large sets of songs, since the number of comparisons between models that must be done grows quadratically with the number of songs.

Another approach [6] produced results statistically indistinguishable from the other top algorithms in MIREX 2007 by using a mid-level semantic feature representation to compute similarity. Using painstakingly human-labeled data, Barrington et al. trained GMMs to estimate the posterior likelihood that a song was best characterized by each of 146 words. These models then produced a vector for each test song defining a multinomial distribution over the 146 semantic concepts. To compute the dissimilarity of two songs, the K-L divergence between these multinomial distributions for the songs was computed.

The success of this method suggests that alternative statistical representations of songs are worth exploring. Rather than take a supervised approach requiring expensive hand-labeled data, we make use of the Hierarchical Dirichlet Process (HDP), which automatically discovers latent structure within and across groups of data (songs, in our case). This latent structure generates a compact alternative representation of each song, and the model provides a natural and efficient way of comparing songs using K-L divergence.

2 HDP-BASED SIMILARITY USING LATENT FEATURES

The hierarchical Dirichlet process (HDP) is an extension of the Dirichlet process (DP), a nonparametric Bayesian model of mixtures of an unknown number of simple densities. We first outline the DP and then describe how we model songs with an HDP.

2.1 Dirichlet Process Mixture Models

The Gaussian Mixture Model (GMM) is a generative process that assumes that each of our feature vectors was generated by one of K multivariate Gaussian distributions. To

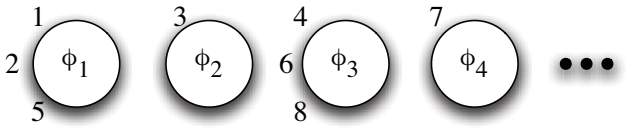

Figure 1. Four tables and eight customers in a Chinese Restaurant Process (CRP). In this example, the 1st, 3rd, 4th, and 7th customers all sat at an empty table, whereas the 2nd, 5th, 6th, and 8th sat at existing tables. The 9th customer will sit at table 1, 2, 3, or 4 with probabilities $\frac{3}{8+\alpha}$, $\frac{1}{8+\alpha}$, $\frac{3}{8+\alpha}$, and $\frac{1}{8+\alpha}$ respectively, or will sit at a new table with probability $\frac{\alpha}{8+\alpha}$

draw a new vector y_t, the process first chooses a mixture component index $z_t \in 1...K$ with probability π_{z_t} (where π is a vector of mixture probabilities summing to one), then draws the vector from the z_tth Gaussian distribution. Given K and a set of vectors assumed to have been generated by a GMM, algorithms such as Expectation-Maximization (EM) can find a maximum-likelihood estimate of the mixture probabilities $\pi_{1...K}$, the parameters defining the K Gaussians $\mu_{1...K}$ and $\Sigma_{1...K}$, and which mixture component z_t generated each vector y_t.

A nagging issue in mixture modeling is model selection, i.e., choosing the number of components K with which to explain the data. Recent work in nonparametric Bayesian statistics has produced models such as the Dirichlet Process Mixture Model (DPMM) that sidestep this issue. Where the GMM assumes the existence of K mixture components, the DPMM [7] assumes the existence of a countably infinite set of mixture components, only a finite subset of which are used to explain the observations.

A traditional metaphor for the way a DP generates data is called the Chinese Restaurant Process (CRP). In the CRP, we imagine a Chinese restaurant with an infinite number of communal tables and a positive scalar hyperparameter α. The restaurant is initially empty. The first customer sits at the first table and orders a dish. The second customer enters and decides either to sit at the first table with probability $\frac{1}{1+\alpha}$ or a new table with probability $\frac{\alpha}{1+\alpha}$. When sitting at a new table the customer orders a new dish. This process continues for each new customer, with the tth customer choosing either to sit at a new table with probability $\frac{\alpha}{\alpha+t-1}$ or at the kth existing table with probability $\frac{n_k}{\alpha+t-1}$, where n_k is the number of other customers already sitting at table k. Notice that popular tables become more popular, and that as more customers come in they become less and less likely to sit down at a new table.

We obtain a DPMM from a CRP as follows. The "dishes" in the CRP correspond to probability density functions, and the process of "ordering" a dish k corresponds to drawing the parameters ϕ_k to a PDF from a prior distribution H over those parameters. (For example, each dish ϕ_k can be a Gaussian with parameters $\{\mu_k, \Sigma_k\} = \phi_k \sim H$.) The process of a customer t choosing a table z_t corresponds to choosing a distribution ϕ_{z_t} from which to draw an observation y_t (in our case, a feature vector). Since customers in the CRP tend to sit at tables with many other customers, the DPMM tends to draw points from the same mixture components again and again even though it has an infinite number of mixture components to choose from.

Analysis under a DPMM involves inferring the posterior distribution over its latent parameters conditioned on the data. This provides a partition of the data (feature vectors) into an unknown number of clusters (the number of tables) and the identities of the parameters ϕ (the means and covariances of the Gaussian mixture components). The posterior distribution $P(\phi, z_{1...T}|y_{1...T})$ of the set of mixture component parameters ϕ and the cluster labels for each feature vector $z_{1...T}$ to a DPMM conditioned on the data $y_{1...T}$ can be inferred using Markov Chain Monte Carlo (MCMC) methods such as Gibbs sampling [7]. For simple data, there will be relatively few unique cluster labels in z, but more clusters will be necessary to explain more complex data.

2.2 The Hierarchical Dirichlet Process

The Hierarchical Dirichlet Process (HDP) [8] is a model of *grouped data*, which is more appropriate than the DPMM for modeling songs represented as a collection of MFCCs. Rather than associate each song with a single table in the restaurant, each song is represented as a group of features which sit at a song-specific "local" restaurant. The dishes for this restaurant, however, are drawn from a "global" set of dishes. Thus, each song is represented as a distribution over latent components, but the population of latent components is shared across songs. Similarity between songs can be defined according to the similarity between their corresponding distributions over components.

The generative process underlying the HDP can be understood with the Chinese Restaurant Franchise (CRF). The CRF takes two hyperparameters α and γ. Each song j has its own CRP, and each feature vector $y_{j,t}$ chooses a table from CRP(α). If it sits down at a new table, then it chooses a dish for that table from a global CRP (with hyperparameter γ) shared by all songs – that is, it either chooses a dish that is already being served at some number of other tables m with probability proportional to m, or it chooses a new dish with probability proportional to γ.

Although we have defined the CRP as a sequential process, in fact data under a CRP are exchangeable – the probability of a seating plan under the CRP is the same regardless of the order in which the customers sat down. This allows us to think of the CRP as defining an implicit prior on infinite multinomial distributions over mixture components. In the DPMM, the infinite-dimensional vector of probabilities $\bar{\pi}$ defining such an infinite multinomial distribution is analo-

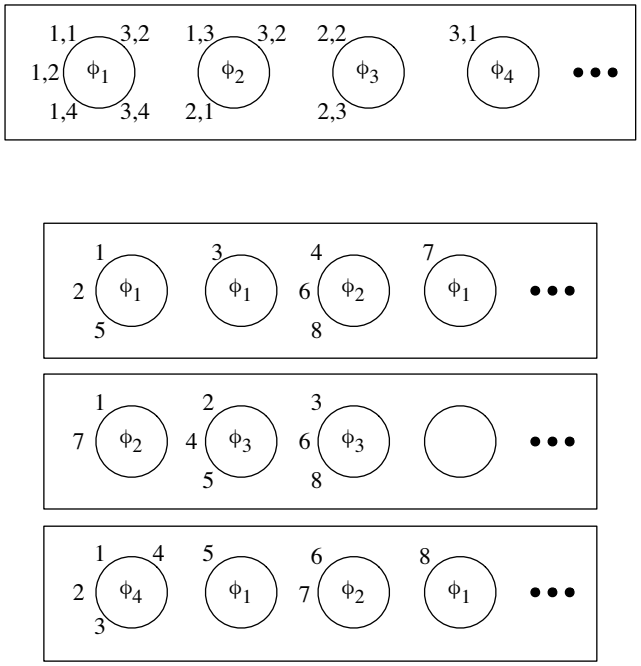

Figure 2. Chinese Restaurant Franchise (CRF) for three songs with eight observations. Below are three CRPs (corresponding to the three songs), and above is the global CRP from which the CRPs get their dishes. Each customer j, i sitting at a table in the global CRP corresponds to table i in restaurant j, and customer j, i's table membership in the global CRP determines the dish that is served at table i in restaurant j. If a new customer coming into a restaurant j sits down at a new table, then the dish for that table will be ϕ_1, ϕ_2, ϕ_3, or ϕ_4 with probability $\frac{5}{\gamma+11}$, $\frac{3}{\gamma+11}$, $\frac{2}{\gamma+11}$, or $\frac{1}{\gamma+11}$ respectively, or a new dish with probability $\frac{\gamma}{\gamma+11}$.

gous to the K-dimensional vector π in the GMM. The HDP has J such vectors $\bar{\pi}_{1...J}$, each of which defines a different distribution over the same mixture components.

We use Gibbs sampling to approximate the posterior distribution over the latent variables conditioned on observed data. The distribution is over the cluster partition assigning feature vectors to clusters and a truncated vector π_j defining the mixture proportions for each song over the finite subset of K mixture components that are actually assigned to observations. We let $\pi_{j,1...K} = \bar{\pi}_{j,1...K}$, and $\pi_{j,K+1} = 1 - \sum_{k=1}^{K} \bar{\pi}_{j,k}$, where $\pi_{j,K+1}$ is the probability of drawing an observation from a mixture component that has not been used to explain any feature vector in any song.

For a complete exposition of the HDP, including how to infer the posteriors for its parameters conditioned on data, see [8].

2.3 Representing Songs Using the HDP

The mixture components parameterized by $\phi_{1...K}$ capture the latent structure in the feature data, and the mixture proportion vectors $\pi_{1...J}$ express the feature data for songs $1...J$ in terms of that latent structure. ϕ and π_j together can describe the empirical distribution of feature vectors for a song j as richly as a GMM can, but the HDP does not require that we choose a fixed value of K, and represents the songs in a more compact way.

To compute the distance between two songs i and j, we can compute the symmetrized KL divergence between the posterior distributions $P(\pi_i|\beta, m)$ and $P(\pi_j|\beta, m)$ which are of the form

$$P(\pi_j|\beta, m) = \text{Dir}(\beta_1 + m_{j,1}, ..., \beta_K + m_{j,K}, \beta_{K+1}) \quad (1)$$

where $m_{j,k}$ is the number of tables in restaurant j serving dish k, β_k is the likelihood of choosing a dish k from the global CRP, and β_{K+1} is $1 - \sum_{k=1}^{K} \beta_k$, the likelihood of choosing a previously unseen dish in the global CRP.

This allows us to compare two songs in terms of the latent structure of their feature data, rather than directly comparing their distributions over the low-level features as the G1 algorithm and GMM-based algorithms do. The KL divergence between these two posteriors can be efficiently computed. The KL divergence between two Dirichlet distributions with parameters v and w each of length K is [9]:

$$D(\text{Dir}(v)||\text{Dir}(w)) = \log \frac{\Gamma(\sum v)}{\Gamma(\sum w)} + \sum_{s=1}^{K} \frac{\log(\Gamma(w_s))}{\log(\Gamma(v_s))} + \sum_{s=1}^{K} ((v_s - w_s)(\Psi(v_s) - \Psi(\sum v))$$

where $\Gamma(x)$ is the gamma function, $\Psi(x)$ is the digamma function (the first derivative of the log gamma function), and $\sum v$ and $\sum w$ denote the sum of the K elements of v and w respectively.

This is less expensive to compute than the KL divergence between two high-dimensional multivariate Gaussian densities. It can be sped up further by computing the gamma and digamma terms offline for each song.

2.4 Generalizing to New Songs

It is important that our approach be scalable to new songs not seen during training. Once we have inferred the global dish likelihoods β and the mixture component parameters $\phi_{1...K}$, we can infer the posterior distribution over the mixture proportions π_{J+1} for a new song $J + 1$ conditioned on β, ϕ, and the new data y_{J+1} using the same Gibbs sampling techniques originally used to train the model, holding all other parameters constant.

3 EVALUATION

In this section we describe the experiments we performed to evaluate our approach against G1, GK (the analogous algorithm for K-component GMMs), and an approach based on Vector Quantization (VQ).

3.1 South by Southwest Dataset

We test our approach on a dataset that we compiled from the South by Southwest (SXSW) 2007 and 2008 festivals' freely distributed "artist showcase" mp3s [10]. We selected a set of up to twenty mp3s (all by different artists to avoid biasing the results) for seven genres: country, electronic, hip-hop, jazz, metal, punk, and rock. Songs that we felt were unrepresentative of their genre were removed or replaced prior to any quantitative evaluations. There were fewer than 20 usable songs available for country (12), jazz (14), and metal (15), so those genres are slightly underrepresented. There are a total of 121 songs in the dataset.

3.2 Features

All models were trained on the same sets of feature vectors, which for each frame consisted of 13 MFCCs (extracted using jAudio [11]) combined with 26 delta features computed by subtracting the MFCCs for frame t from those at frame $t-1$ and $t-2$, for a total of 39 dimensions. Each frame was approximately 23 ms long, or 512 samples at the files' sampling rate of 22050 Hz, with a hop size of 512 samples (no overlap). 1000 feature vectors were extracted from the middle of each song.

3.3 Models Evaluated

3.3.1 G1

As described above, G1 models each song's distribution over feature vectors with a single multivariate Gaussian distribution with full covariance matrix. Models are compared using the symmetrized KL divergence.

3.3.2 K-component GMMs

We train K-component GMMs for each song using the E-M algorithm. The symmetrized KL divergence between models is approximated by drawing 1000 synthetic feature vectors from the trained models and evaluating their log likelihoods under both models [1]. This approach is evaluated for $K = 5$, 10, 20, and 30.

3.3.3 VQ Codebook

This algorithm is meant to be a simple approximation to the HDP method we outlined above. First, we cluster all of the feature vectors for all songs into K groups using the k-means algorithm, renormalizing the data so that all dimensions have unit standard deviation. This defines a codebook of K cluster centers that identifies every feature vector with the cluster center to which it is closest in Euclidean space. For each song j, we compute the vector $\pi_{j,1\ldots K}$ of the relative frequencies of each cluster label. Each $\pi_{j,1\ldots K}$ defines a multinomial distribution over clusters, and we compute the distance between songs as the symmetrized KL divergence between these multinomial distributions (smoothed by a factor of 10^{-5} to prevent numerical issues).

This algorithm, like our HDP-based method, represents each song as a multinomial distribution over latent cluster identities discovered using an unsupervised algorithm, and lets us see how a much simpler algorithm that uses similar ideas performs compared with the HDP.

3.3.4 HDP

We train an HDP on all of the data using the direct assignment method [8], inferring the posterior distributions over the π_j's for each song j and computing the distance between two songs i and j as the KL divergence between the posteriors over π_i and π_j. We place vague gamma priors on α and γ [8]:

$$\alpha \sim \text{gamma}(1, 0.1), \quad \gamma \sim \text{gamma}(1, 0.1) \qquad (2)$$

and learn them during inference. For the prior H over ϕ, we use the normal-inverse-Wishart distribution [12] with parameters $\kappa_0 = 2$, $\nu_0 = 41$ (the number of dimensions plus two), and $\mu_0 = \bar{y}$ (the mean of all feature vectors across songs). The normal-inverse-Wishart matrix parameter Λ_0 was chosen by averaging the covariance matrices from 100 clusters of feature vectors, each of which was obtained by choosing a feature vector at random and choosing the 24,200 feature vectors closest to it under a Euclidean distance metric. (The number 24,200 was chosen because it was $1/5$ of the total number of points.) The goal of this process is to choose a matrix Λ_0 that resembles the covariance matrix of fairly large cluster of points, encouraging the model to find similarly shaped clusters. Using smaller (larger) clusters to choose Λ_0 would result in the model creating more (fewer) latent topics to explain the data.

3.4 Experiments

Since human-labeled ground truth similarity data is inherently expensive and difficult to acquire, we follow previous researchers [1, 2] in using genre as a proxy for similarity. We assume that all songs labeled with the same genre are "similar," which allows us to use evaluation metrics from the information retrieval literature. We first compute a full 121x121 distance matrix between all songs using each algorithm. For each query song s_q, each other song s_i is

G1	G5	G10	G20	G30	VQ5	VQ10	VQ30	VQ50	VQ100	HDP
13.24	829	1487	2786	4072	0.58	0.59	0.63	0.686	0.85	**0.25**

Table 1. Time in seconds required to compute a 121x121 distance matrix for G1, GMM-based ($K = 5, 10, 20, 30$), VQ-based ($K = 5, 10, 30, 50, 100$), and HDP-based algorithms.

	G1	G5	G10	G20	G30	VQ5	VQ10	VQ30	VQ50	VQ100	HDP
RP	0.3254	0.3190	0.3287	0.3144	0.3146	0.2659	0.2997	0.3191	0.340	0.3313	**0.3495**
AP	0.3850	0.3761	0.3746	0.3721	0.3706	0.3171	0.3546	0.3850	0.3989	0.3910	**0.3995**
AUC	0.6723	0.6712	0.6687	0.6679	0.6661	0.6513	0.6675	0.6846	0.6893	0.6758	**0.7002**

Table 2. Three measures of retrieval quality: mean R-Precision (RP), mean Average Precision (AP), and mean Area Under ROC Curve (AUC) for G1, GMM-based ($K = 5, 10, 20, 30$), VQ-based ($K = 5, 10, 30, 50, 100$), and HDP-based algorithms on the large SXSW dataset.

	G1	HDP
RP	0.5486	**0.6000**
AP	0.6807	**0.7154**
AUC	0.8419	**0.8983**

Table 3. Mean R-Precision (RP), mean Average Precision (AP), and mean Area Under ROC Curve (AUC) for G1 and our HDP-based algorithm on the smaller dataset.

given a rank $r_{q,i}$ based on its similarity to s_q. The quality of this ranking, i.e. how well it does at ranking songs of the same genre as s_q more similar than songs of different genres, is summarized using R-Precision (RP), Average Precision (AP), and the Area Under the ROC Curve (AUC), which are standard metrics from the information retrieval literature [13]. All experiments were conducted on a MacBook Pro with a 2.0 GHz Intel Core Duo processor and 2 GB of RAM. All models were implemented in MATLAB.

3.4.1 Testing on Additional Data

To test our HDP-based method's ability to generalize to unseen data using the method in section 2.4, we use the HDP trained on the large SXSW set to compute a similarity matrix on a smaller set consisting of 5 artist-filtered songs per genre (35 in all) by artists not in the training set. The electronic, punk, rap, and rock songs came from the SXSW artist showcase collection, and the country, jazz, and metal songs came from a dataset previously used by George Tzanetakis [14]. We also compute a similarity matrix on this dataset using G1, and compare the RP, AP, and AUC metrics for retrieval quality obtained using both algorithms.

4 RESULTS

Tables 1, 2, and 3 summarize the results of our experiments. The best results in each row are in bold.

The amount of time required to compute the distance matrices for the GMMs was, as expected, enormous by comparison to the other models. The cost of computing the KL divergence for the VQ-based and HDP-based models was more than an order of magnitude lower even than the cost of computing the KL divergence between single Gaussians.

The HDP performed better than the other models for all three standard information retrieval metrics, although the VQ model with $K = 50$ was a very close second. None of the GMMs outperformed G1.

The results in table 3 show that the HDP-based approach does generalize well to new songs, showing that the algorithm can be scaled up efficiently to databases of many songs.

4.1 SIMILARITY HUBS

The G1 and GK approaches are known to produce "hubs" [1] – an undesirable phenomenon where certain songs are found to be similar to many other songs. The hub phenomenon is a potentially serious concern, since it can result in very bad matches being selected as similar to a query song.

Our HDP-based approach does not suffer from this problem. Figure 3 shows how often each song is ranked in the top five of another song's similarity list for similarity matrices obtained from G1, the HDP, and choosing distances at random. The randomly generated histogram shows the sort of distribution of hubs one would expect to see due to chance in a dataset of this size. The HDP's histogram closely resembles the random one, indicating an absence of abnormal hubs. G1's histogram, by contrast, shows more severe and more numerous hubs than the other two histograms.

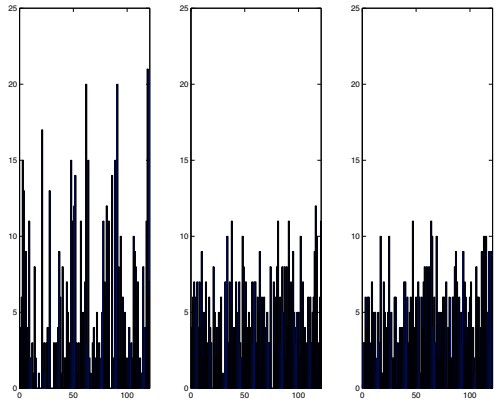

Figure 3. Histograms of how often each song is ranked in the top five of another song's similarity list for similarity matrices obtained using G1 (left), the HDP (center), and by choosing distances at random (right).

5 CONCLUSION

We developed a new method for assessing the similarity between songs. Our HDP-based approach outperformed the G1 algorithm, can compute large distance matrices efficiently, and does not suffer from the "hub" problem where some songs are found to be similar to all other songs. Since our approach does not have access to any information about temporal structure beyond that provided by the MFCC deltas (about 69 ms in total), we expect that combining the distances it provides with fluctuation patterns or some similar feature set would provide an improvement in similarity performance, as it does for the G1C algorithm [2].

One area of future work involves relaxing the bag-of-feature-vectors assumption. For example, we might learn distributions over texture patches of feature vectors instead of individual feature vectors. Hidden Markov models can also be fit into the HDP framework [8], and may yield improved results.

6 REFERENCES

[1] Aucouturier, J-J and Pachet, F. "Improving Timbre Similarity: How High's the Sky?," *Journal of Negative Results in Speech and Audio Sciences* 1 (2004), no. 1, http://journal.speech.cs.cmu.edu/articles/2004/3.

[2] Pampalk, E. "Computational Models of Music Similarity and Their Application to Music Information Retrieval." Ph.D. Dissertation, Vienna Inst. of Tech., Austria, 2006.

[3] Pampalk, E. "Audio-Based Music Similarity and Retrieval: Combining a Spectral Similarity Model with Information Extracted from Fluctuation Patterns," *Proceedings of the International Symposium on Music Information Retrieval*, Victoria, BC, Canada, 2006.

[4] Pohle, T. and Schnitzer, D. "Striving for an Improved Audio Similarity Measure," *Proceedings of the International Symposium on Music Information Retrieval*, Vienna, Austria, 2007.

[5] Jensen, J., Ellis, D.P.W., Christensen, M., and Jensen, S. "Evaluation of Distance Measures Between Gaussian Mixture Models of MFCCs," *Proceedings of the International Symposium on Music Information Retrieval*, Vienna, Austria, 2007.

[6] Barrington, L., Turnbull, D., Torres, D., and Lanckriet, G. "Semantic Similarity for Music Retrieval," *Proceedings of the International Symposium on Music Information Retrieval*, Vienna, Austria, 2007.

[7] Neal, R. "Markov chain sampling methods for Dirichlet process mixture models," *Journal of Computational and Graphical Statistics*, 9(2):249-265, 2000.

[8] Teh, Y., Jordan, M., Beal, M., and Blei, D. "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, 101(476):1566-1581, 2007.

[9] Penny, W.D. "Kullback-Liebler Divergences of Normal, Gamma, Dirichlet and Wishart Densities." Technical report, Wellcome Department of Cognitive Neurology, 2001.

[10] http://2008.sxsw.com/music/showcases/alpha/0.html

[11] McEnnis, D., McKay, C., Fujinaga, I., and Depalle, P. "jAudio: A Feature Extraction Library," *Proceedings of the International Symposium on Music Information Retrieval*, London, UK, 2005.

[12] Gelman, A., Carlin, J., Stern, H., and Rubin, B. *Bayesian Data Analysis* CRC Press, New York, 2004.

[13] Manning, C., Raghavan, P., and Schütze, H. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK, 2008.

[14] Tzanetakis, G. "Manipulation, Analysis and Retrieval Systems for Audio Signals." PhD thesis, Computer Science Department, Princeton University, 2002.