Content-Based Genre Classification and Sample Recognition Using Topic Models

Cora Johnson-Roberson
Department of Computer Science
Brown University
Providence, RI 02912
caj8@cs.brown.edu

Erik Sudderth (Advisor)
Department of Computer Science
Brown University
Providence, RI 02912
sudderth@cs.brown.edu

Abstract

Audio-related research benefits from analysis methods applicable to multiple tasks, as hand-crafting features and models for each task is time-consuming and difficult. In this paper we investigate the use of probabilistic topic modeling for two distinct audio tasks: genre classification and sample identification. Topic models transform frame-level audio features into a single track-level low-dimensional summary. We find that this topic-space representation produces a usable track-level feature for genre classification, but is not as effective for sample identification.

1 Introduction

The need to draw inferences from audio data arises in a variety of applications: for example, when recommending new music for online streaming services [1], or analyzing rhythm and timing in computational ethnomusicology [2]. Numerous techniques for such tasks have been developed in the field of music information retrieval (MIR); there is great interest in models that can perform well on multiple tasks.

Genre recognition is one of the most well-studied problems in MIR. This work typically uses classification accuracy on a handful of common datasets as a performance metric, although some authors have criticized the experimental design of such studies [3]. In particular, success with genre recognition on specific datasets may not generalize well, as genre is complex and context-dependent. Nonetheless, this task comes up in real-world situations like categorizing untagged (or inaccurately tagged) music.

Sampling is a musical practice used extensively in hip-hop and other contemporary popular music, in which sections of one or more original tracks are combined with new audio. Sample identification, then, is the process of identifying which tracks have been sampled to create the new track. Although it would be very useful to identify samples automatically, either for copyright protection or for musicological research, this task has not been as extensively studied as genre classification and many other tasks in MIR. The approach to sample identification in [4] uses the form of audio fingerprinting developed in [5]: namely, the use of "landmarks" in the audio signal to generate a hash that can be compared with the hash of an audio query in order to identify a track. Whereas most audio fingerprinting approaches assume an exact copy of a track is being played (perhaps with background noise), samples in hip-hop and other genres may be looped, pitch-shifted, and/or time-stretched, making retrieval more difficult.

These two tasks represent relatively different areas of MIR. A system that performs well on both might prove useful for an even wider variety of tasks. Thus we wish to explore whether topic models can serve as a valuable piece of such a system, given their broad utility in modeling text [6, 7, 8], images [9, 10], and audio [11, 12, 13].

In this paper, we will investigate the use of probabilistic topic models in genre classification and sample identification. In section 2, we briefly describe the formulation of these models with regard to real-valued data. In section 3, we describe the data and methods of our study. Section 4 reports our results for each combination of task, feature, and model; Section 5 contains discussion and future directions.

2 Topic Models with a Gaussian Observation Model

Topic models, such as latent Dirichlet allocation [6], are typically unsupervised models that assume documents are made up of a mixture of "topics," with each topic being a distribution over observed features. Intuitively, the per-document topic distributions reflect the idea that each document in a corpus tends to emphasize a small handful of themes, and that not every document will contain these themes in the same proportions.

LDA is perhaps the simplest model of this type; the number of topics to learn must be set in advance. The hierarchical Dirichlet process [14], a non-parametric extension of LDA, has the advantage that the number of topics is not fixed and will be adjusted according to the complexity of the data.

LDA has a Dirichlet prior and a multinomial posterior, allowing only word counts as input. Gaussian-LDA, described in [15], uses instead a Gaussian-Wishart prior and Gaussian posterior, permitting real-valued features as input. Gaussian-LDA operates similarly to a Gaussian mixture model, but unlike the GMM which presumes the same proportions for all documents, it draws the observations from a document-specific distribution of the available Gaussians.

The Python package BNPY [16] provides scalable inference for a variety of models, including both parametric and non-parametric topic models. BNPY conceptually divides each model into an allocation model and an observation model. This allows users to switch easily between finite models like LDA and infinite models like HDP, as well as discrete observations such as word counts (via the multinomial distribution) and continuous ones like most image and audio features (through a multivariate Gaussian). We make use of this flexibility in our experiments.

3 Data and Method[1]

3.1 Data

Genre classification experiments were run on the Homburg dataset [17], which contains 1886 samples of 10 s each, comprising 9 different genres. We used stratified 10-fold cross validation, preserving the class proportions in training and test sets.

Sample identification used the Van Balen dataset [4], consisting of 143 tracks of varying lengths: 68 original, or "candidate" tracks, and 75 tracks that sampled them, or "queries." Once again, stratified 10-fold cross validation is used to form training and test sets (considering original tracks as one class and sampling tracks as another).

3.2 Features

We extracted audio features from the data using the Python packages ESSENTIA [18] and LIBROSA [19]. All audio data was converted to mono with a 22050 Hz sample rate before further processing.

The same set of features are used in both genre classification and sample identification tasks, to determine which features are most helpful for each task. In the following experiments, we chose two relatively low-level features dealing with timbre and rhythm, the combination thereof (formed by concatenating the feature vectors), and a feature drawn from the pre-trained convolutional neural network described in [20].

3.2.1 Timbre

Audio has several distinct perceptual dimensions that can be usefully quantified. Timbre, for example, is the "character" of a sound, as opposed to its pitch (fundamental frequencies) or rhythm (temporal

[1] Code to reproduce these experiments can be found at `https://bitbucket.org/corajr/masters/src`.

organization). Timbre is largely a function of the instruments used in a piece, which provides an important signal for genre recognition.

Most MIR work uses mel-frequency cepstral coefficients (MFCCs) as a measure of timbre. These features were originally used in speech-related tasks, but have proven broadly useful as a succinct general-purpose audio descriptor in both speech and music. In this paper, however, we use gammatone-frequency cepstral coefficients (GFCCs) [21] in lieu of MFCCs. GFCCs have been shown to be more robust to noise than MFCCs in speech applications, and have also begun to be used in MIR [22, 23].

GFCCs are calculated similarly to MFCCs, but use a gammatone filterbank instead of a mel filterbank. The spectrum of each frame (1024 samples, or 46 ms) is passed through the filterbank; the log-power spectrum of the filter bands is fed into the discrete cosine transform (DCT) to decorrelate output. We use only dimensions 1 through 13 of this vector, discarding the DC and high-frequency components.

Many audio features, like MFCCs and GFCCs, are frame-based (calculated e.g. every 512 or 1024 samples), raising the question of how to extract higher-level features that can summarize an entire track. One common approach is to perform vector quantization on MFCCs, typically by K-means [24], counting the resulting sonic "words" to produce a bag-of-frames representation for each track. Such an approach is simple and quick to implement, but has the disadvantage that document membership information is discarded when learning features. We include this K-means approach as another feature, with $K = 2000$, to compare our results with this common method.

3.2.2 Rhythm

Rhythm is quantified using the scale transform, using a method developed in [25]. The onset strength signal (OSS) is computed in 8 second rectangular windows over the audio signal, with a hop size of 4 s. The autocorrelation of the OSS is then passed to the scale transform with 512 bins (computed using the fast Mellin transform (FMT)), creating a 257-dimensional complex vector. Following [26], we take the absolute value and compute the DCT to decorrelate the output, taking coefficients 1 through 60. This process creates a real-valued descriptor of rhythm which is invariant to time-stretching or changes in tempo; the hop size helps to mitigate the shift-dependence of the scale transform.

Because of the comparatively lengthy windows and long hop size, there are far fewer observations of the rhythmic feature on each track; for training purposes, the scale transform outputs are repeated until reaching the same number of observations as the GFCCs (roughly 86 copies of each observation).

3.2.3 Convnet Feature

Finally, along with these lower-level features we include two variants of a higher-level feature, computed by the pre-trained convolutional neural network of [20]. This network comprises 5 convolutional layers with 32 feature maps each; it takes a mel-spectrogram as input, then average pools the activations from each layer to produce successively higher-level summaries of the audio. The convnet was originally trained for genre classification on a different dataset, but has demonstrated effectiveness in transfer learning across a range of datasets and tasks.

The first variant of the feature, just as with the original work in [20], is produced by concatenating the feature map activations of each layer; this yields a single 160-D vector that summarizes an entire track at multiple levels of detail. However, because there is just one "word" per document in this case, this feature is unsuitable to use as input to a topic model.

The variant we used for the topic model was created by extracting hypercolumns [27] over each "pixel" of the mel-spectrogram, then taking the hypercolumns' maximum across all feature maps. The resulting 96-D feature vectors (1 per frame) are essentially a selective amplification of the input, where parts of the spectrum that activated the neural network are given higher values while the rest tend toward zero. (Using the full hypercolumn ($160 * 96 = 15360$D) was impractical for our case; other variations, such as taking the mean of the hypercolumns across all frames or using only the feature maps from the highest-level layer, produced substantially worse results in preliminary experiments and are omitted here.)

3.3 Models

All topic models were trained using BNPY. For each real-valued feature, LDA with a full-covariance Gaussian observation model was trained at $K = 10$ and $K = 50$, and HDP with full-covariance Gaussian was trained with $K_{init} = 100$.[2] We also trained standard (count-based) LDA with 50 topics on the K-means bag-of-frames vector, to compare with a more typical feature in this setting.

To translate the topic model output into genre predictions, a linear SVM classifier was trained on the topic proportions for each document in the training set, and accuracy was then computed for the test set. The classifier was also trained on the document-level mean of each feature (or on the single document-level feature, in the CNN's case), to assess how much the underlying feature was responsible for performance.

Sample identification was undertaken by comparing the topic proportions of each document via cosine similarity to find the closest candidate tracks, after which the mean average precision (MAP) was computed. Similarly, the cosine similarity was also calculated for the document-level mean of each feature, the CNN output, and the GFCC bag-of-frames vector.

4 Results

Top performance on each task is given in **bold**; reported values are the mean percentages over the cross-validation splits (standard deviations are listed in parentheses). The following abbreviations are used in the table:

Random Baseline (100 repetitions)

Mean Track-level mean of the feature described

GaussLDA-K LDA of K components with Gaussian observation model

GaussHDP-K HDP of K initial components with Gaussian observation model

LDA-K LDA of K components with multinomial observation model

CNN Convolutional neural net

Cos Cosine similarity-based retrieval

SVM Linear SVM

4.1 Genre Classification

The original paper [17] used 49 features, achieving a top accuracy of 53.23% using a k-nearest neighbor method with an adaptive distance metric. State of the art performance is around 63.46% [28].

Table 1 shows our results for genre classification. The top result was 62.04%, from the document-level CNN feature fed directly into the linear SVM (with no topic model).

4.2 Sample Identification

In the original paper [4], an audio fingerprinting system was used to extract a hash from each track and then cosine similarity was used to retrieve candidates. In addition, "noise" tracks were added to challenge the system in the original work, but these were not available at the time of this writing. The original authors' system achieved a mean average precision (MAP) of 22.8% for the basic system, and 39.0% for a modified form including pitch-shifted tracks as candidates.

Table 2 shows our results for sample identification. The top result was 12.48%, achieved by the GFCC bag-of-frames feature and a linear SVM (with no topic model).

[2] With the HDP model, the hypercolumn CNN feature took 8 hours to train for a single cross-validation fold (1 out of 10) on available hardware; it was hence omitted as impractical.

Table 1: Genre Classification

Model	Feature(s)	Accuracy (std. dev.)
Random	—	11.29 (2.13)
Mean + SVM	GFCC	46.33 (2.30)
Mean + SVM	Scale	34.67 (1.90)
Mean + SVM	GFCC + Scale	46.29 (2.28)
GaussLDA-10 + SVM	GFCC	46.98 (1.48)
GaussLDA-10 + SVM	Scale	32.34 (2.22)
GaussLDA-10 + SVM	GFCC + Scale	29.88 (2.37)
GaussLDA-10 + SVM	CNN	47.55 (1.76)
GaussLDA-50 + SVM	GFCC	51.00 (2.44)
GaussLDA-50 + SVM	Scale	33.46 (2.54)
GaussLDA-50 + SVM	GFCC + Scale	28.57 (2.77)
GaussLDA-50 + SVM	CNN	50.89 (2.25)
GaussHDP-100 + SVM	GFCC	50.79 (1.56)
GaussHDP-100 + SVM	Scale	32.71 (2.47)
GaussHDP-100 + SVM	GFCC + Scale	28.04 (2.76)
SVM	GFCC K-means	26.72 (0.17)
LDA-50 + SVM	GFCC K-means	28.01 (2.03)
SVM	CNN	**62.04** (2.28)

Table 2: Sample Identification

Model	Feature(s)	MAP (std. dev.)
Random	—	4.07 (1.30)
Mean + Cos	GFCC	4.82 (0.00)
Mean + Cos	Scale	10.96 (0.00)
Mean + Cos	GFCC + Scale	4.82 (0.00)
GaussLDA-10 + Cos	GFCC	10.48 (2.01)
GaussLDA-10 + Cos	Scale	11.80 (2.68)
GaussLDA-10 + Cos	GFCC + Scale	11.76 (1.38)
GaussLDA-10 + Cos	CNN	10.86 (3.02)
GaussLDA-50 + Cos	GFCC	11.16 (1.23)
GaussLDA-50 + Cos	Scale	9.66 (1.82)
GaussLDA-50 + Cos	GFCC + Scale	11.28 (1.53)
GaussLDA-50 + Cos	CNN	12.22 (1.15)
GaussHDP-100 + Cos	GFCC	9.66 (0.44)
GaussHDP-100 + Cos	Scale	10.33 (2.23)
GaussHDP-100 + Cos	GFCC + Scale	10.11 (2.15)
SVM	GFCC K-means	**12.48** (1.38)
LDA-50 + SVM	GFCC K-means	11.48 (1.19)
Cos	CNN	12.04 (0.00)

5 Discussion and Conclusion

In the genre task, topic models managed to effectively capture latent factors in the dataset and achieve moderate performance, but were ultimately outperformed by the pre-trained convnet with a linear classifier. The GFCC (timbral) feature was the most effective input to the topic models at $K = 50$, producing a result of 51.00% accuracy with the CNN hypercolumn feature close behind. (This is somewhat lower than the 53% baseline of the original paper that used 49 features, but is fairly good performance for using a single feature.) Comparing results from feature means, the GFCC and concatenated GFCC/scale transform were comparable to each other, hovering around 46%. Using the CNN's document-level feature on its own produced the top result of 62.04%, which is close to state of the art for this dataset.

Sample identification proved substantially more difficult for all the models used. This may be due to the underlying assumption that only one topic is present at a given time point (violated when the samples are mixed together), or because of the trivial method of cosine similarity used to retrieve candidate tracks. Nevertheless, this is a difficult task and there have not yet been any particularly impressive results on this dataset. Comparing the sample results using the raw feature means, the scale transform (rhythmic feature) gave the best results. In our experiments, the top performance was given by running a linear SVM on the K-means bag-of-frames representation, although results were very close to the CNN feature (both the hypercolumn feature with 50-topic LDA and the document-level version).

On the sample task, the concatenated timbral and rhythmic feature showed improvement over timbre alone, while on the genre task, the combined feature generally did worse. This likely reflects a difference in the underlying datasets as well as the tasks; the sample ID dataset is mostly comprised of jazz and hip-hop, both of which tend to emphasize rhythmic organization, while the genre dataset spans many genres that are most readily differentiated not by rhythm but by the different musical instruments used (and hence, the different timbres present).

In both tasks, the use of HDP to fit the number of topics to the data did not show substantial improvement over using a fixed numbers of topics: the number of topics was reduced only by 10 or so from the initial 100, and performance was often worse than the LDA trained at $K = 50$. More tuning of hyperparameters might have helped to improve this result.

It is clear that the performance of the topic model is heavily dependent on the underlying feature(s). This is evident from the fact that the performance of simply taking the mean of the feature across the entire track was in each case similar to the result of first training the topic model on the hundreds or thousands of observations from each track. Further, timbral features appear to be more important for the success of genre classification, and rhythmic features for sample identification, perhaps stemming from the difference in datasets noted above.

Predictive performance of the convnet feature when fed directly to the SVM exceeded the CNN + topic model + SVM combination in nearly every case. This is unsurprising, as the convnet is optimized specifically for good discriminative performance in genre classification, while the topic model is optimizing an unrelated objective (the posterior probability of the documents). In general, using unsupervised topic modeling (or any technique) for dimensionality reduction often loses information that would aid discriminative tasks, as illustrated by the original LDA paper's experiments with text classification [6]. This loss could be mitigated by using a supervised topic model, such as sLDA [29], which models a response variable jointly with document content. Such a supervised model would learn a representation that has better predictive power for the task at hand.

This paper demonstrates that unsupervised topic models can provide a viable low-dimensional representation of audio without labeled data, although their ultimate performance depends strongly on input features. The convnet feature in particular is versatile, obtaining good performance on both tasks, and at present often works better on its own than as an input to the topic model. Future work could explore the use of a supervised topic model (ideally one that can accommodate real-valued input data), or a fully hybrid neural network/topic model with end-to-end discriminative training, as in [30]. Such a model would presumably improve performance by richly capturing the details of the underlying data via the neural net, while also leveraging the topic model's structure to capture a higher-level representation of features as they are differently distributed across documents.

References

[1] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. In *Advances in Neural Information Processing Systems*, pages 2643–2651, 2013.

[2] George Tzanetakis, Ajay Kapur, W. Andrew Schloss, and Matthew Wright. Computational Ethnomusicology. *Journal of Interdisciplinary Music Studies*, 1(2):1–24, 2007.

[3] Bob L. Sturm. Classification accuracy is not enough. *Journal of Intelligent Information Systems*, 41(3):371–406, 2013.

[4] Jan Van Balen, Joan Serrà, and Martín Haro. Sample Identification in Hip Hop Music. In *International Symposium on Computer Music Modeling and Retrieval*, pages 301–312. Springer, 2012.

[5] Avery Wang. An Industrial Strength Audio Search Algorithm. In *Proceedings of the 4th International Conference on Music Information Retrieval*, pages 7–13. Washington, DC, 2003.

[6] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[7] David Hall, Daniel Jurafsky, and Christopher D. Manning. Studying the history of ideas using topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 363–371, Stroudsburg, PA, USA, 2008.

[8] Sean Gerrish and David M. Blei. A language-based approach to measuring scholarly impact. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 375–382, 2010.

[9] David M. Blei and Michael I. Jordan. Modeling annotated data. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 127–134. ACM, 2003.

[10] Erik B. Sudderth, Antonio Torralba, William T. Freeman, and Alan S. Willsky. Describing Visual Scenes Using Transformed Objects and Parts. *International Journal of Computer Vision*, 77(1-3):291–330, 2008.

[11] Matthew D. Hoffman, David M. Blei, and Perry R. Cook. Content-Based Musical Similarity Computation using the Hierarchical Dirichlet Process. In *Proceedings of the 9th International Conference on Music Information Retrieval*, pages 349–354, 2008.

[12] M. Hoffman, D. Blei, and Perry R. Cook. Finding latent sources in recorded music with a shift-invariant HDP. In *Proceedings of the 12th International Conference on Digital Audio Effects*, 2009.

[13] S. Kim, S. Narayanan, and S. Sundaram. Acoustic topic model for audio information retrieval. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA '09*, pages 37–40, 2009.

[14] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[15] Pengfei Hu, Wenju Liu, Wei Jiang, and Zhanlei Yang. Latent topic model based on Gaussian-LDA for audio retrieval. In *Chinese Conference on Pattern Recognition*, pages 556–563. Springer, 2012.

[16] Michael Hughes and Erik Sudderth. Bnpy: Reliable and scalable variational inference for Bayesian nonparametric models. In *NIPS Probabilistic Programimming Workshop*, 2014.

[17] Helge Homburg, Ingo Mierswa, Bülent Möller, Katharina Morik, and Michael Wurst. A Benchmark Dataset for Audio Classification and Clustering. In *Proceedings of the 6th International Symposium on Music Information Retrieval*, volume 2005, pages 528–31, 2005.

[18] Dmitry Bogdanov, Nicolas Wack, Emilia Gómez, Sankalp Gulati, Perfecto Herrera, Oscar Mayor, Gerard Roma, Justin Salamon, José R. Zapata, and Xavier Serra. Essentia: An Audio Analysis Library for Music Information Retrieval. In *Proceedings of the 14th International Society for Music Information Retrieval Conference*, pages 493–498, 2013.

[19] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. Librosa: Audio and music signal analysis in python. In *Proceedings of the 14th Python in Science Conference*, 2015.

[20] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Transfer learning for music classification and regression tasks. 2017.

[21] Yang Shao, Zhaozhang Jin, DeLiang Wang, and Soundararajan Srinivasan. An auditory-based feature for robust speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4625–4628. IEEE, 2009.

[22] Wei Cai, Qiang Li, and Xin Guan. Automatic singer identification based on auditory features. In *7th International Conference on Natural Computation*, volume 3, pages 1624–1628. IEEE, 2011.

[23] Mi Tian and Mark B. Sandler. Music Structural Segmentation Across Genres with Gammatone Features. In *Proceedings of the 17th ISMIR Conference*, pages 561–567, New York, NY, 2016.

[24] Jason Weston, Chong Wang, Ron Weiss, and Adam Berenzweig. Latent collaborative retrieval. *arXiv preprint arXiv:1206.4603*, 2012.

[25] Andre Holzapfel and Yannis Stylianou. A scale transform based method for rhythmic similarity of music. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 317–320, 2009.

[26] Matthew Prockup, Andreas F. Ehmann, Fabien Gouyon, Erik M. Schmidt, and Youngmoo E. Kim. Modeling musical rhythm at scale with the Music Genome project. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 1–5. IEEE, 2015.

[27] Bharath Hariharan, Pablo Arbeláez, Ross Girshick, and Jitendra Malik. Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 447–456, 2015.

[28] Yannis Panagakis, Constantine L. Kotropoulos, and Gonzalo R. Arce. Music Genre Classification via Joint Sparse Low-rank Representation of Audio Features. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 22(12):1905–1917, 2014.

[29] David M. Blei and Jon D. Mcauliffe. Supervised topic models. In *Advances in Neural Information Processing Systems 20*, pages 121–128, 2008.

[30] Li Wan, Leo Zhu, and Rob Fergus. A Hybrid Neural Network-Latent Topic Model. In *AISTATS*, volume 12, pages 1287–1294, 2012.