Unsupervised Learning of Structural Representation of Percussive Audio Using a Hierarchical Dirichlet Process Hidden Markov Model

Antich, Jose Luis Diez; Paterna, Mattia ; Marxer, Richard; Purwins, Hendrik

Published in:
MML 2016 9th International Workshop on Machine Learning and Music

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Diez Antich, Jose Luis, Paterna, Mattia, Marxer, Ricard, and Purwins, Hendrik. "Unsupervised Learning of Structural Representation of Percussive Audio Using a Hierarchical Dirichlet Process Hidden Markov Model." In: International Workshop on Machine Learning and Music (MML) at European Conference on Machine Learning and Principles and Practice of Knowledge Discovery (ECML-PKDD), p.6–10, 2016.

Unsupervised Learning of Structural Representation of Percussive Audio Using a Hierarchical Dirichlet Process Hidden Markov Model

Jose Luis Diez Antich[1], Mattia Paterna[1], Ricard Marxer[2], and Hendrik Purwins[1]

[1] Sound and Music Computing Group and Audio Analysis Group, Aalborg Universitet København,{jdieza15, mpater15}@student.aau.dk, hpu@create.aau.dk
[2] The Speech and Hearing Group,
Department of Computer Science, University of Sheffield, ricardmp@gmail.com

Abstract. A method is proposed that extracts a structural representation of percussive audio in an unsupervised manner. It consists of two parts: 1) The input signal is segmented into blocks of approximately even duration, aligned to a metrical grid, using onset and timbre feature extraction, agglomerative single-linkage clustering, metrical regularity calculation and beat detection. 2) The approx. equal length blocks are clustered into k clusters and the resulting cluster sequence is modelled by transition probabilities between clusters. The Hierarchical Dirichlet Process Hidden Markov Model is employed to jointly estimate the optimal number of sound clusters, to cluster the blocks, and to estimate the transition probabilities between clusters. The result is a segmentation of the input into a sequence of symbols (typically corresponding to hits of hi-hat, snare, bass, cymbal, etc.) that can be evaluated using the Adjusted Random Index (ARI). As a proof-of-concept, the system segmentation has been tested using two simple Disco-style drum loops, yielding a an ARI of 56% for the best stable HDP-HMM parameter setting.

Keywords: unsupervised learning, Hierarchical Dirichlet Process, Hidden Markov Model, clustering, musical structure.

1 Introduction

Unsupervised learning of music representation may provide a new paradigm for music analysis, generative music, music information retrieval, and intelligent musical human-computer interaction. Based on two cognitively plausible principles (unsupervised and statistical learning) such an approach may spare excessive musical annotation efforts, with - at the same time- a high degree of flexibility to learn representations for a multitude of musical styles.

Previous work on music analysis based on Markov models have been presented by Conklin & Witten [1], Pachet [12], and Hazan et al. [8]. In this paper the preprocessing by Marchini and Purwins [10] and the use of the Hierarchical

Dirichlet Process Hidden Markov Model (HDP-HMM) in [11] is combined. Cf. [2] for details. The Authors would like to thank *Marco Marchini* and *Emily B. Fox* for providing Matlab toolboxes used here and for useful suggestions and comments.

2 Analysis Chain: Pre-segmentation and HDP-HMM

The first step of the proposed method consists in pre-segmenting the audio signal into a sequence of blocks, in such a way that each bar contains the same number (e.g. 8) of blocks of approximately the same duration each. This way we ensure that blocks on a particular metrical position reoccur after a fixed number of blocks. This representation helps the subsequent structural analysis (by HDP-HMM) to account for the metrical structure of the input signal.

Following [10], the *pre-segmentation* consists of the following stages: 1) the audio input signal is segmented into blocks at onset positions, 2) from the salient part (the first 200 ms) of each event, the mean of the MFCCs weighted by the RMS energy of each frame (512 FFT size, 256 window size) is calculated. 3) The resulting 13 mean MFCCs are clustered using single linkage clustering. 4) The clustering threshold yielding the number of event clusters is the one with highest regularity, where the regularity is the strength of the first side peak of the autocorrelation of the inter onset histogram ([10] p. 210). 5) based on the most regular symbol subsequence (skeleton subsequence) the tempo is determined via a score voting criterion [3] ([10] p. 210-2). 6) The detected tempo pulse is aligned to the skeleton subsequence, by finding the phase matching the most onsets ([10] p. 212). 7) Dual subdivision between consecutive pulses are iteratively aligned to remaining onsets in the input. If no onset is found within a tolerance window around a pulse the prior symbol will be repeated ([10] p. 212-3).

Starting from the pre-segmented sequence of onsets with roughly equally long inter-onset times, we train a time series model that on one hand estimates the number of distinct labels and, on the other hand, estimates the transition probabilities between labels. An HMM is defined as a fixed set of states (in our case percussion sound clusters), initial and observation probabilities to observe e.g. a particular set of MFCCs given a particular sound (e.g. hi-hat), and transition probabilities (e.g. from hi-hat to snare). However, in the HMM we assume a fixed number of hidden states (i.e. percussive sounds). The HDP-HMM [14, 6] is an extension of the HMM in which the number of different states does not need to be known beforehand and is estimated as well. The Dirichlet Process (DP) is a model of clustering with an unbound and a priori unknown number of clusters [5], where a large parameter α favours a higher number of clusters and large γ favours a greater variability of segmentation results. We use the HDP-HMM Matlab implementation by Fox [4].

3 Experiment

Audio Drum Loops We performed a preliminary evaluation of the unsupervised segmentation of our model based on two audio drum loop files from the ENST

drum data set [7]. Both audio excerpts follow a simple duple meter and follow a regular rhythmic structure, composed of a repetition of bass drum (*bd*) and snare drum (*sd*) on strong metrical positions, alternated with open / closed hi-hats (*ohh / chh*) on weak metrical positions. Both audio files are provided with annotations, i.e. pairs of onset times/percussion sound labels (bd, sd, ohh, chh). We gather annotations occurring within a maximal time tolerance to joint labels (e.g.bd+chh+sd, bd+ohh) used as the ground truth for evaluation.

Adjusted Rand index Whereas either accuracy nor f-measure are valid methods to evaluate classification outcomes, an evaluation of clustering needs to account for the problem of assigning annotation labels to clusters found by the clustering and for situations such as when two annotated labels are contained in one single cluster. The Rand index [13] is the quotient of item pairs clustered consistently w.r.t. the annotations divided by the number of all item pairs. The adjusted Rand index (ARI) [9] normalizes the Rand index R with respect to the expected Rand index $E(R)$ of a random clustering/annotation configuration and the maximal possible rand index $max(R)$: $ARI = \frac{R-E(R)}{max(R)-E(R)}$ with $ARI \leq 1$ [15].

Evaluation Procedure We use Gibbs sampling to estimate the structural segmentation. For each parameter setting, we perform 10 trials each of 1200 iterations of the HDP-HMM inference sampler. Each iteration results in a state sequence (structural segmentation). Of the 1200 iterations we discard the first 200 iterations (burnout stage). From the remaining 1000 state sequences, we select the *most frequent* state sequence. The ARI of this latter sequence and the ground truth annotation is then calculated. Then the ARI for the 10 runs for a particular parameter setting is averaged providing this particular configuration of the HDP-HMM with a performance value.

4 Results

We will compare the performance of the system with respect to the parameters α and γ of the HDP-HMM and with respect to an additional processing step of dimension reduction using Principle Component Analysis (PCA). We evaluate the performance (via ARI) when representing each segment either by the first two MFCCs or by the scores on the two first principal components after applying PCA to the first 13 MFCCs. We observed that a high γ yield more variable segmentation results for different runs. Figure 1 displays ARI results depending on the parameter value pairs (α, γ)=(0.1,1.0),(1.5,1.5),(3.0,3.0),(10.0,1.0) and MFCCs with/without PCA. Although $\alpha = \beta = 1.5$ yields the best average ARI=0.58, when using PCA the standard deviation is high (0.16), whereas the results for $\alpha = \beta = 3.0$ are slightly lower (ARI=0.56), the ARI is the same for MFCC and for PCA with 0 standard deviation, yielding more stable results across different runs and configurations (with/without PCA).

α	γ	MFCC mean(std)	PCA mean(std)
0.1	1.0	0.04(0.08)	0.02 (0.05)
1.5	1.5	0.55(0.14)	0.58 (0.16)
3.0	3.0	0.56(0.00)	0.56 (0.00)
10.0	1.0	0.31(0.08)	0.34 (0.06)

Fig. 1: Average ARI results using different initial parameters α and β for the HDP-HMM inference algorithm, with and without PCA following MFCC. Higher values of ARI represent more accurate clusterings.

Example: Simple Disco Drum Loop The pre-segmentation process results in 33 almost equal length blocks, 8 for each bar consisting of twice the sequence *bass, hi-hat, snare, hi-hat*. In the experiment, we observe an increase in number of clusters when increasing α and γ. In Fig. 2, for events no. 26, 28, and 33, clustering and annotation do not match. The cluster indices, predicted by the HDP-HMM for those events are a continuation of the previous bass, hi-hat, snare, hi-hat pattern. It appears that the high learned transition probabilities between states overrun the low probability of observing such a change. In that sense, the HPD-HMM corrects artifacts from the previous pre-segmentation process. For parameter settings $\alpha = \gamma = 3$, the HDP-HMM yields 4 clusters with a maximum ARI= 65%, where the hi-hat cluster from the $\alpha = 1, \gamma = 0.1$ setting splits into two clusters, corresponding to the two alternating metrical positions 1+, 3+, vs 2+, 4+ of the hi-hat occurrences.

5 Conclusion

In this paper, we gave a proof-of-concept of a method that extracts a structural representation of percussive audio in an unsupervised manner. The audio is represented with reference to a metrical grid. The number of sound classes is identified and musical style is modelled via transition probabilites between sound

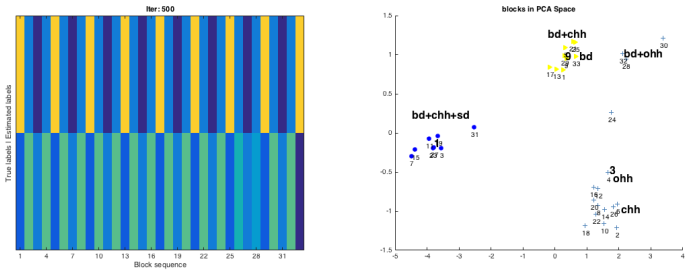

Fig. 2: Most frequently estimated block pattern (*left, top*) when using the 2 principle components and HDP-HMM parameters $\alpha = 1, \gamma = 1$. The event annotations are shown *bottom*, *left*. Different colors indicate different clusters (*top*) and annotations (*bottom*). The scatter plot on the *right* displays the events in the space spanned by the first two principal components of the MFCCs, where the differently colored shapes indicate different estimated clusters. Also, the centers of the 6 event groups each with a particular annotated joint label (ohh, bd+chh, bd+chh+sd, etc.) are indicated.

classes, using the HDP-HMM. We also demonstrated how a change in the parameterization of our model is able to generate reasonably scaled representations (e.g. gathering or segregating open /closed hi-hats). Our model currently does not work for rhythmically complex sequences or very inaccurate onset detection and it assumes an approximately constant tempo. In the future, testing on a larger database is neeed, metrical alignment should align to micro tempo variations (e.g. ritardando) and parameters (α, γ) should be learned during training.

References

[1] Darrell Conklin and Ian H Witten. "Multiple viewpoint systems for music prediction". In: *Journal of New Music Research* 24.1 (1995), pp. 51–73.

[2] JL. Diez Antich and Mattia Paterna. *Unsupervised structure analysis of an audio file using Hierarchical Dirichlet Process Hidden Markov Models*. Aalborg University, København, 2016.

[3] Simon Dixon. "Automatic extraction of tempo and beat from expressive performances". In: *Journal of New Music Research* 30.1 (2001), pp. 39–58.

[4] Emily B Fox. "Bayesian nonparametric learning of complex dynamical phenomena". PhD thesis. Massachusetts Institute of Technology, 2009.

[5] Emily B Fox et al. "A sticky HDP-HMM with application to speaker diarization". In: *The Annals of Applied Statistics* (2011), pp. 1020–1056.

[6] Andrew Gelman et al. *Bayesian data analysis*. Vol. 2. Taylor & Francis, 2014.

[7] Olivier Gillet and Gaël Richard. "ENST-Drums: an extensive audio-visual database for drum signals processing." In: *ISMIR*. 2006, pp. 156–159.

[8] Amaury Hazan et al. "What/when causal expectation modelling applied to audio signals". In: *Connection Science* 21.2-3 (2009), pp. 119–143.

[9] Lawrence Hubert and Phipps Arabie. "Comparing partitions". In: *Journal of classification* 2.1 (1985), pp. 193–218.

[10] Marco Marchini and Hendrik Purwins. "Unsupervised analysis and generation of audio percussion sequences". In: *Exploring Music Contents*. Springer, 2010, pp. 205–218.

[11] Ricard Marxer and Hendrik Purwins. "Unsupervised Incremental Online Learning and Prediction of Musical Audio Signals". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 24.5 (2016), pp. 863–874.

[12] Francois Pachet. "The continuator: Musical interaction with style". In: *Journal of New Music Research* 32.3 (2003), pp. 333–341.

[13] William M Rand. "Objective criteria for the evaluation of clustering methods". In: *Journal of the American Statistical association* 66.336 (1971), pp. 846–850.

[14] Yee Whye Teh. "Dirichlet process". In: *Encyclopedia of machine learning*. Springer, 2011, pp. 280–287.

[15] Ka Yee Yeung and Walter L Ruzzo. "An empirical study on principal component analysis for clustering gene expression data". In: *Bioinformatics* 17.9 (2001), pp. 763–774.