A Probabilistic Topic Model for Music Analysis

Diane J. Hu and Lawrence K. Saul
Department of Computer Science
University of California, San Diego
{dhu,saul}@cs.ucsd.edu

Abstract

We describe a probabilistic model for learning musical key-profiles from symbolic and audio files of polyphonic, classical music. Our model is based on Latent Dirichlet Allocation (LDA), a statistical approach for discovering hidden topics in large corpora of text. In our adaptation of LDA, music files play the role of text documents, groups of musical notes play the role of words, and musical key-profiles play the role of topics. We show how these learnt key-profiles can be used to determine the key of a musical piece and track its harmonic modulations.

1 Introduction

In western tonal music, composers generally work within a theoretical framework that is highly structured and organized. As such, musical pieces are commonly studied by analyzing their melodic and harmonic structures. Two important concepts in any such analysis are the *key* and *tonic*. The key of a musical piece identifies a principal set of pitches that the composer uses to build its melodies and harmonies; the tonic is the most stable pitch in this set. Each musical piece is characterized by one overall key. However, the key can be shifted within a piece by a compositional technique known as *modulation*. Notwithstanding the infinite number of variations possible in music, most pieces can be analyzed in these terms.

In this paper, we review our recent work on modeling a corpus of symbolic music pieces in terms of its harmonic structure [1]. We then describe work-in-progress that extends this framework for musical pieces in the form of raw audio. Both frameworks are based on Latent Dirichlet Allocation (LDA) [2], a popular probabilistic model for discovering latent semantic topics in large collections of text documents. In our variant of LDA, we model each musical piece as a random mixture of keys. Each key is then modeled as a "key-profile" – a distribution over a set of pitch classes. Using this representation, we show how to accomplish the tasks of key-finding and modulation-tracking, the first steps to any kind of harmonic analysis.

Our approach to such tasks is unlike that of previous studies, which depend on extensive prior music knowledge or supervision by domain experts [3, 4]. Based on a model of *unsupervised* learning, our approach bypasses the need for manually key-annotated musical pieces, a process that is both expensive and prone to error. As an additional benefit, it can also discover correlations in the data of which the designers of rule-based approaches are unaware. Since we do not rely on extensive prior knowledge, our model can also be applied in a straightforward way to other, non-western genres of music with different tonal systems.

2 Modeling Symbolic Music

This section describes our probabilistic topic model for modeling symbolic files of polyphonic, classical music [1]. We chose to initially work with symbolic music files so that note count information

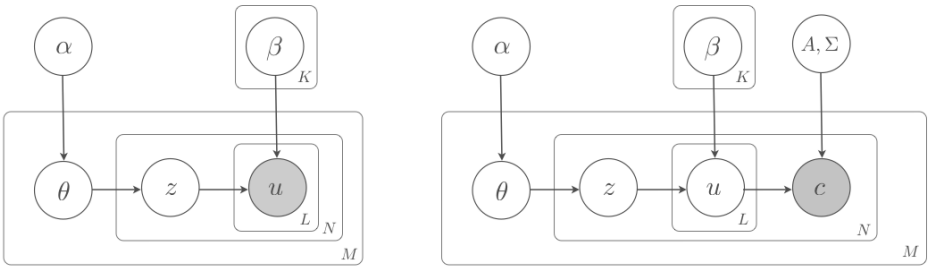

Figure 1: Graphical representation of our model for (left) symbolic music and (right) audio music.

could be unambiguously extracted from musical pieces. We use the following terms to describe our probabilistic topic model:

1. A *note* $u \in \{\text{A}, \text{A}\sharp, \text{B}, \ldots, \text{G}\sharp\}$ is the most basic unit of data. It is an element from the set of 12 neutral pitch-classes. We refer to $V = 12$ as the vocabulary size of our model.
2. A *segment* is a basic unit of time in a song (e.g., a measure). We denote the notes in the nth segment by $\mathbf{u}_n = \{u_{n1}, \ldots, u_{nL}\}$, where $u_{n\ell}$ is the ℓth note in the segment.
3. A *song* s is a sequence of notes in N segments: $s = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$.
4. A music *corpus* is a collection of M songs denoted by $\mathcal{S} = \{s_1, \ldots, s_M\}$.
5. A *key* $z \in \{1, \ldots, K\}$ is a probability distribution over the vocabulary of $V = 12$ pitch-classes. We will refer to each of the K distributions as a "key-profile" that describes the importance and stability of each pitch class relative to the key. Intuitively, a key will model groups of notes that frequently occur together within song segments, and is thus analogous to a "topic" in the original LDA model.

2.1 Generative process

We begin by describing the process for generating a song in the corpus. First, we draw a "key weight vector" that determines which keys are likely to appear in the song. The key weight vector is modeled as a Dirichlet random variable. Next, for each segment of the song, we sample from the key weight vector to determine the key (e.g., A minor) of that segment. Finally, we repeatedly draw notes from the key's key-profile until we have generated all the notes in the segment. More formally, we can describe this generative process as follows:

1. For each song in the corpus, choose a K-dimensional key weight vector $\theta \sim \text{Dirichlet}(\alpha)$.
2. For each segment $\mathbf{u}_n$ in a song, choose a key $z_n \sim \text{Multinomial}(\theta)$.
3. For each note u_{nl} in the nth measure, choose a pitch class from the $p(u_{n\ell} = i | z_n = j, \beta) = \beta_{ij}$. The β parameter is a $V \times K$ matrix that encodes each key as a distribution over $V = 12$ neutral pitch-classes.

Figure 1 (left) depicts the graphical representation for this model, using plate notation to represent independently, identically distributed random variables within the model.

2.2 Incorporating Prior Knowledge

To achieve our results, we incorporate two pieces of prior information; this is the full extent to which our approach uses prior music knowledge. First, we set $K = 24$ to look specifically for key-profiles corresponding to the major and minor scales of western tonal music. Second, we assume that all learnt key-profiles within the major or minor mode are related by simple transposition. This assumption adds a simple constraint to our learning procedure: instead of learning $V \times K$ independent elements in the β matrix, we tie diagonal elements across different keys of the same mode (major or minor). Enforcing this constraint, the learnt K distributions can be unambiguously

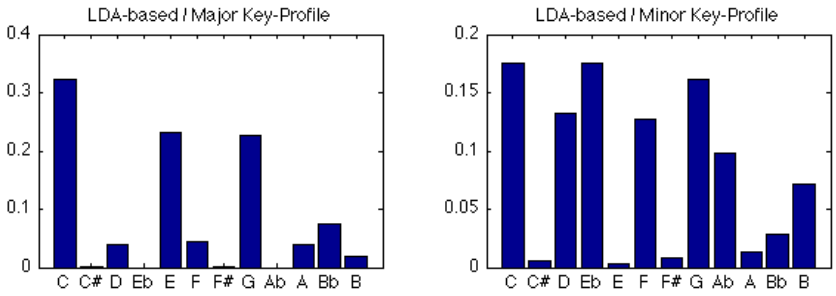

Figure 2: The C major and C minor key-profiles learned by our model, as encoded by the β matrix. Resulting key-profiles are obtained by transposition.

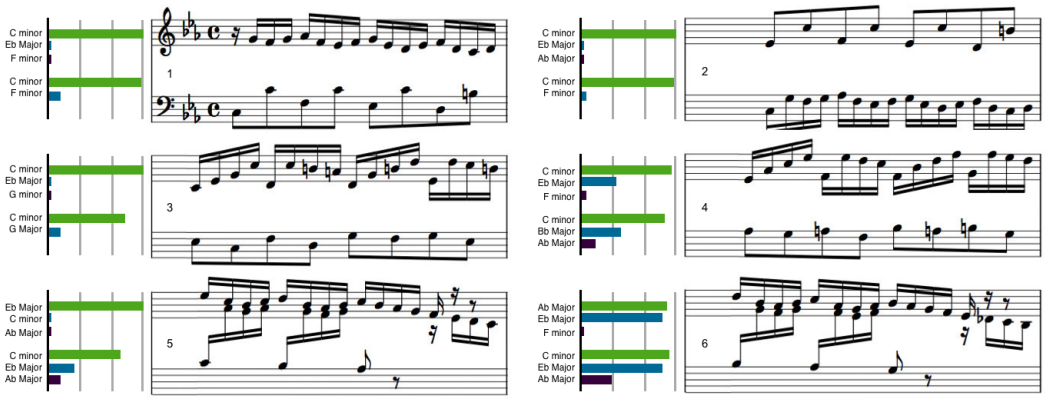

Figure 3: Key judgments for the first 6 measures of Bach's Prelude in C minor, WTC-II. Annotations for each measure show the top three keys (and relative strengths) chosen for each measure. The top set of three annotations are judgments from our LDA-based model; the bottom set of three are from human expert judgments [3].

identified with the 24 major and minor modes of classical western music. We note that our approach is still regarded as unsupervised because we do not learn from labeled or annotated data.

2.3 Results & Applications

Our learnt key-profiles are shown in Figure 2. We note that these key-profiles are consistent with music theory principals: In both major and minor modes, weights are given in descending order to degrees of the triad, diatonic, and finally chromatic scales. Intuitively, these key-profiles represent the underlying distributions that are used to characterize all the songs in the corpus.

We also show how to do key-finding and modulation-tracking using the representations learned by our model. The goal of key-finding is to determine the overall key of a musical piece, given the notes of the composition. Since the key weight vector θ represents the most likely keys present in each song, we classify each song as the key that is given the largest weight in θ. A related task is modulation-tracking, which identifies where the modulations occur within a piece. We achieve this by determining the key of each segment from the most probable values of its topic latent variable z.

We estimated our model from a collection of 235 MIDI files compiled from `classicalmusicmidipage.com`. The collection included works by Bach, Vivaldi, Mozart, Beethoven, Chopin, and Rachmaninoff. These composers were chosen to span the baroque through romantic periods of western, classical music. Our results for key-finding achieved an accuracy of 86%, out-performing several other key-finding algorithms, including the popular KS model [3]. We also show in Figure 3 that our annotations for modulation-tracking are comparable to those given by music theory experts. More results can be found in our paper [1].

3 Modeling Audio Music

We extend our framework for modeling symbolic files of music for use with real audio. In the first section, we describe our method for acquiring and preprocessing audio files of classical, polyphonic music. In the second section, we describe the generative process for this new model.

3.1 Data Representation

To evaluate the performance of our model, we need a large quantity of annotated audio music files to compare with. We address this problem by generating our own time-aligned audio files from the dataset used in our symbolic music experiments (section 2.3). In this way, all annotations available for each symbolic song are also available for its audio counterpart. We synthesize all audio files using Timidity++, a free software synthesizer that converts MIDI files into audio files in a WAVE format. Soundfonts Fluid3 and Sinfon36 were used to render the MIDI files to achieve harmonically rich audio resembling that of real recordings.

To simulate note counts, we use 12-bin chroma vectors to represent the prevalence of each pitch class in a segment of real audio. A tuning algorithm [5] is applied to the chromagram calculation to more accurately map frequencies to bins. Chromagram values are then averaged across chunks of time that are equal to the segments in its corresponding symbolic file. In this way, the dimensions for the symbolic and audio form of each song are identical.

3.2 Generative Process

Our new probabilistic model represents each input song as a collection of chroma vectors $c \in \mathcal{R}^{12}$. We denote the nth chroma vector c_n to be computed from the same window of time as the nth segment $\mathbf{u}_n$ in its corresponding symbolic music file.

In this new model, previously observed notes u are now hidden random variables that must be inferred from input chroma vectors. This adds an additional step to our generative process: once all notes $\{u_{n1}, \ldots, u_{nL}\}$ have been generated for a segment $\mathbf{u}_n$, a chroma vector c_n is drawn from the probability distribution

$$p(c_n|\mathbf{u}_n, A) = \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(c_n - A\mathbf{u}_n)^T\Sigma^{-1}(c_n - A\mathbf{u}_n)\right\} \quad (1)$$

where A and Σ are additional model parameters to learn. The graphical representation is shown in Figure 1 (right); an additional node models the chroma vectors as observed random variables.

3.3 Discussion

We use this new model to accomplish the same tasks of key-finding and modulation-tracking for real audio music files. Future work also aims to provide features for the application of song segmentation in real audio. Results of current experiments seem encouraging. However, as expected, we see a $\sim$20% degradation in accuracy rates when going from MIDI to WAVE inputs. To counter this, we are working on three extensions: 1) model topic dependencies, as certain key transitions are more likely 2) incorporate additional information about our input data (e.g. timbre, beat), and 3) try a semi-supervised framework in which a small number of key labels are provided.

References

[1] D. J. Hu and L. K. Saul. A probabilistic topic model for unsupervised learning of musical key-profiles. *International Conference on Music Information Retrieval*, 2009.

[2] D. M. Blei, A. Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] C. Krumhansl. In *Cognitive Foundations of Musical Pitch*, Oxford, 1990. Oxford University Press.

[4] D. Temperley. A bayesian approach to key-finding. *Lecture Notes in Computer Science*, 2445:195–206, 2002.

[5] C.A. Harte and M.B. Sandler. Automatic chord identification using a quantised chromagram. *Proc. Audio Eng. Soc.*, 2005.