

Separating Advertisements and DJ Chatter from Music

Bryan Klingner

11 December 2006

Abstract

Music on the radio is often interrupted by advertisements or DJ chatter that the listener would prefer not to hear. In this paper, we discuss a straightforward application of perceptually-based audio feature extraction and classification using a support vector machine to automatically differentiate between music and “non-music” audio, so that when non-music is detected the radio might automatically seek out more music on other stations.

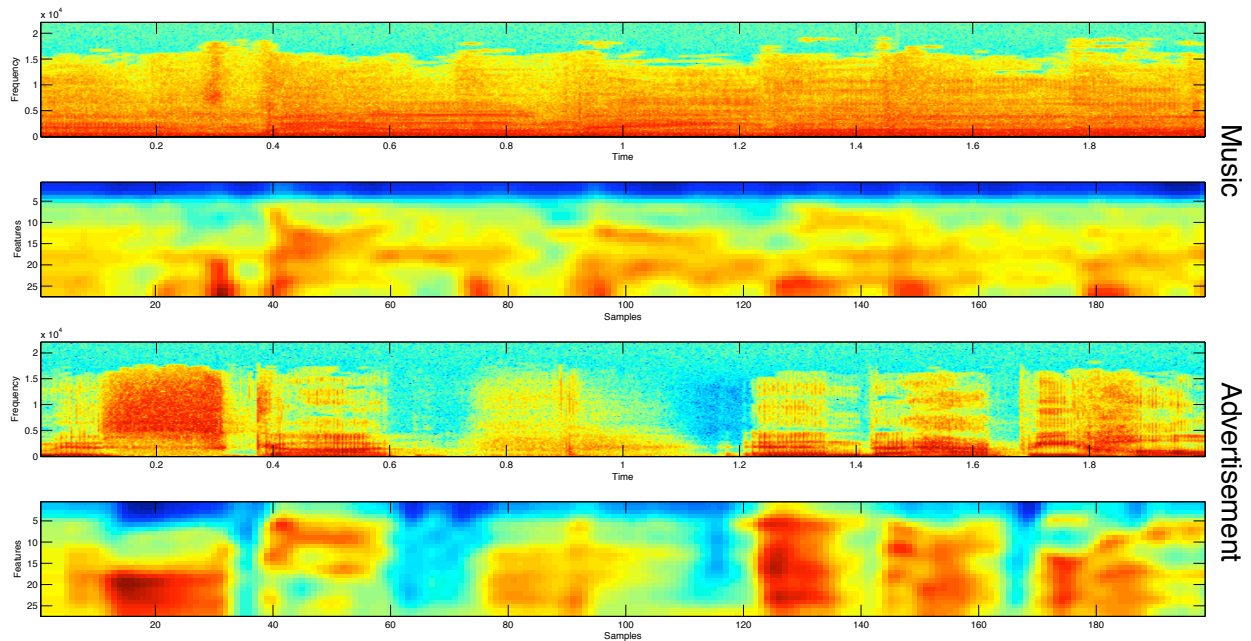


Figure 1: Top: the spectral intensity and extracted features for a two-second clip of music. Bottom: the same data from a two-second clip of a radio advertisement.

1 Introduction

Broadcast radio offers an appealing way to enjoy music. Devices used to access FM radio are ubiquitous in modern life: cars, portable music players, and home stereos almost universally contain the hardware necessary to tune in. Usually, there are at least a few and sometimes dozens of stations available, offering different genres of music and talk.

Because of the funding model of radio, advertisements appear frequently between periods of music. Listeners often find themselves scanning through channels looking for one that's actually playing music. Sometimes, this effort is undertaken while driving and can be dangerous. What if this job could be performed by the radio instead?

In this paper, we attempt automatically separate audio from radio into two classes: music and non-music. Non-music audio includes advertisements, DJ chatter, and talk radio and news such as NPR. With a robust method for differentiating these two classes of audio, one could imagine a completely automatic system built into the radio of a car that would always keep the radio tuned to a station that was playing music.

We perform this classification of audio in two steps. First, we extract salient features from clips of music and non-music audio (for an example of what these features can look like, see Figure 1). Then, we train a support vector machine to classify novel audio clips as it receives them.

We use audio features produced by RASTA filtering [Ell05], which separates and emphasizes the frequency bands of audio important to human perception. RASTA filtering also has the advantage of allowing relatively large sample windows which capture some of the temporal aspects of audio. We selected a support vector machine method for classification because of the highly non-linear discrimination between audio features.

2 Background

Two main bodies of previous work were tapped for this application: extraction of salient features from audio, and classification of data using support vector machines.

2.1 Audio Feature Extraction

In the early years of audio classification research, the main focus of researchers was on ways to classify human speech for automatic speech recognition systems. The goals of these systems were diverse, but usually centered on recognizing human speech alone, and possibly in the presence of background noise. To this end, the Mel frequency cepstral coefficients (MFCC) were developed [You95]. These coefficients improved on a simple linear spectral amplitude analysis by performing two transforms on the data in order to emphasize properties that were thought to be important in the recognition of human speech.

Later, the problem of automatically classifying other types of audio data emerged as a focus of much research. Because of the large body of work analyzing the effectiveness of MFCC, this set of features was applied to music with significant success [Foo97]. Still, it

seems like an incomplete solution to use MFCC without consideration of what other factors might be important for music. Indeed, recent research into what features best distinguish different types of music and general classes of audio indicate that MFCC is not ideal and that better performance can be achieved when using a metric which exploits perceptual properties of the human auditory system [MB03].

In particular, another method of audio processing based on human perception, Relative Spectral Transform - Perceptual Linear Prediction (RASTA), which was also developed to improve speech recognition but showed promise as a method of feature extraction for music and other audio [HM94]. RASTA is a technique that applies a band-pass filter to the energy in each frequency section to smooth over short-term noise variations and to remove any constant offset resulting from static spectral coloration in the speech channel. In practice, as discussed later, it serves as an excellent method for audio feature extraction when attempting to classify music from non-music. We used a freely available implementation of RASTA processing made available on the web and written for the `MATLAB` environment [Eli05].

2.2 Support Vector Machine Classification

Many methods for discriminating between classes of examples based on collections of features have been used over the last few decades. Recently, support vector machines (SVMs) have emerged as an effective, theoretically rigorous, and easy-to-implement tool for attacking problems of classification in many different fields [BC00]. Audio classification in particular is resistant to older methods of classification such as linear or quadratic discriminant analysis because of the large number of features considered and because it is in general quite unlikely that any plane or simple parabola can be found to differentiate between the important qualities in these large feature sets.

Many implementations of SVM learners and classifiers have been made available to the research community. We selected the `LIBSVM` implementation [CL01]. `LIBSVM` provides an easy-to-use implementation with several kernels available and also includes a tool for parameter optimization of your selected kernel.

3 Methods

Our first task in building an automatic classifier for music and non-music audio was to select the audio that would be used for examples. Then, we extracted the features of these clips. Finally, we used these features to train an SVM classifier and to predict audio class.

3.1 Example Audio Clips

The space of audio signals that might be classified as “music” is huge. Even with a giant space of musical examples, we could not begin to cover all of the variation of styles and genres that exist. Also, some kinds of music (such as rhythmic poetry or even types of rap or hip hop) are so similar at times to simple, unaccompanied human speech that even a human

Name	Class	Length	Description
deathcabforcutie[1,2].wav	music	2 sec	Male vocals, electric.
decemberists[1,2].wav	music	2 sec	Mixed vocals, acoustic.
franzferdinand[1,2].wav	music	2 sec	Male vocals, guitars, drums.
loscampesinos[1,2].wav	music	2 sec	Punk with male vocals.
annuals[1,2].wav	music	2 sec	Electronic and rock, mixed vocals.
whitestripes[1,2].wav	music	2 sec	Male vocals, electric.
yolatengo[1,2].wav	music	2 sec	Male vocals, electric.
adairamerica[1,2].wav	non-music	2 sec	Ad with background music.
adbkgnd[1,2].wav	non-music	2 sec	Ad with loud background music.
adpeacecorps[1,2].wav	non-music	4 sec	Ad with background music.
adnpr[1,2].wav	non-music	2 sec	Ad with background music.
djlowvoice[1,2].wav	non-music	2 sec	DJ chatter with music.
djnobackground[1,2].wav	non-music	2 sec	DJ chatter, no music.
nprnews[1,2].wav	non-music	4 sec	News report, no music.

Table 1: A listing of example audio clips. The [1,2] in the file names indicate two audio clips of equal length, taken from disjoint sections of the same recording. The first set of audio clips (ending in 1) were used for training, while the second (ending in 2) were used for testing.

might not be able to differentiate between the two. Because of this vast space, we decided to focus our efforts on a proof-of-concept classifier that handles music that universally includes instrumental aspects. In particular, we selected contemporary and indie rock clips with both male and female singles, with varying musical texture and complexity.

For our examples of advertisement, DJ chatter, and other talk, we were more capable of covering the space. We sampled clips of DJs talking between songs, clips of advertisements, and clips of news reports. Some of these clips feature unaccompanied human voice, but most of them also included background music layered under the speaker. Because actual radio advertisement and talk frequently have background music, we felt that to demonstrate the utility of the classifier we needed examples that also had background music. A listing of all our example audio clips can be found in Table 1.

3.2 Audio Feature Extraction

We considered using both the popular MFCC audio features and the more recent RASTA features. Because literature indicates that the RASTA features may be a better method for differentiation of types of music, we decided to use it for our features.

Feature extraction proceeds as follows:

1. WAV files in PCM (pulse code modulation) format are read into numeric arrays in the MATLAB environment.
2. The audio data is separated into short chunks.

3. A bank of 27 fourth-order band-pass filters tuned to single and emphasize frequencies that the human auditory is sensitive to filters the audio.
4. A fast Fourier transform is applied to the audio, moving it to frequency space.
5. The amplitude of the 27 bands is stored as the feature vector for each sample.

Because we used a total of 14 seconds of music and 18 seconds of ads and other non-music examples, this feature extraction process netted about 2000 examples for music and 2400 for non-music audio. An example comparison of the features extracted by MFCC and RASTA can be found in Figure 2.

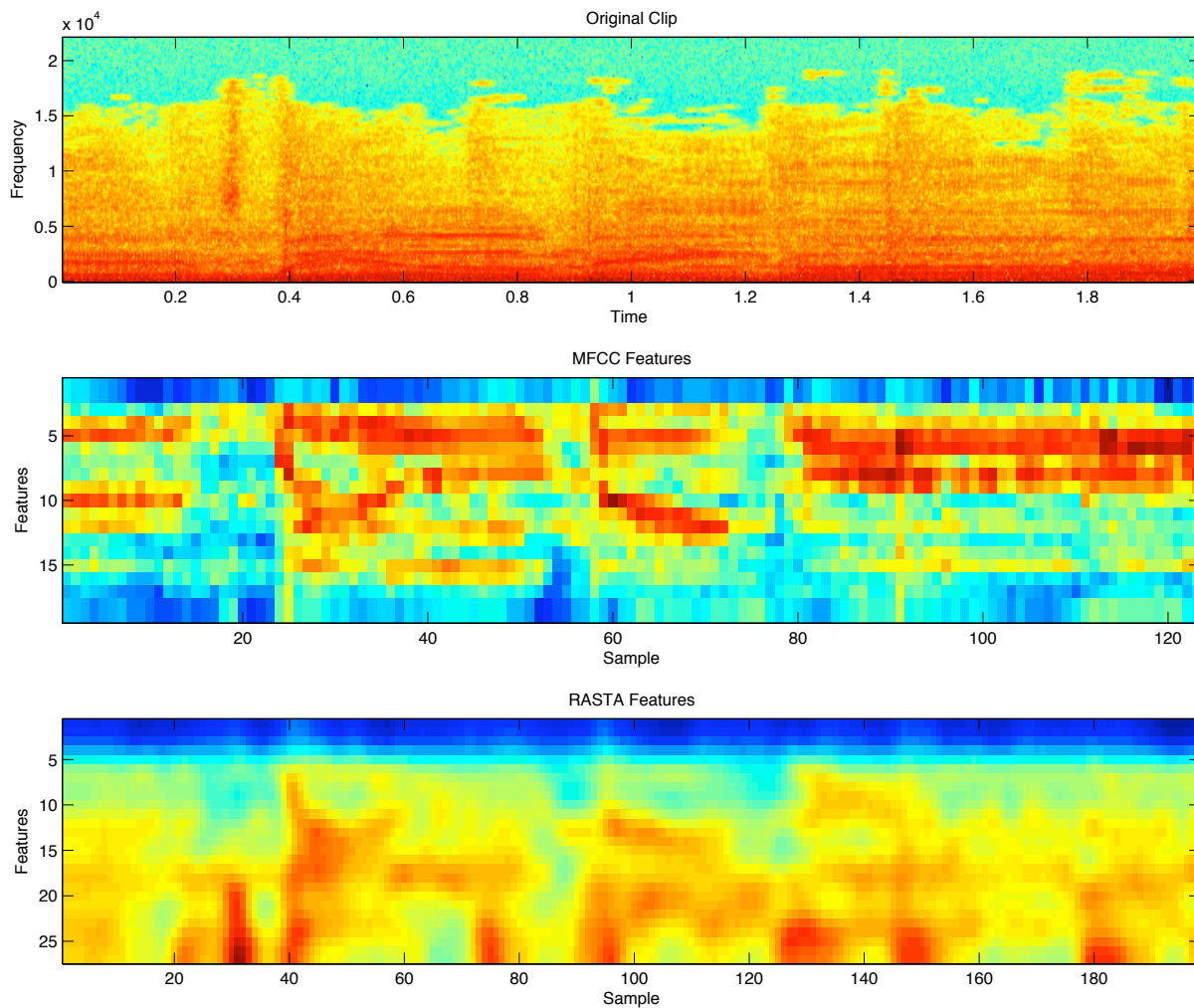


Figure 2: Top: the spectral intensity of the original sound clip. Middle: the extracted MFCC features. Bottom: the extracted RASTA features.

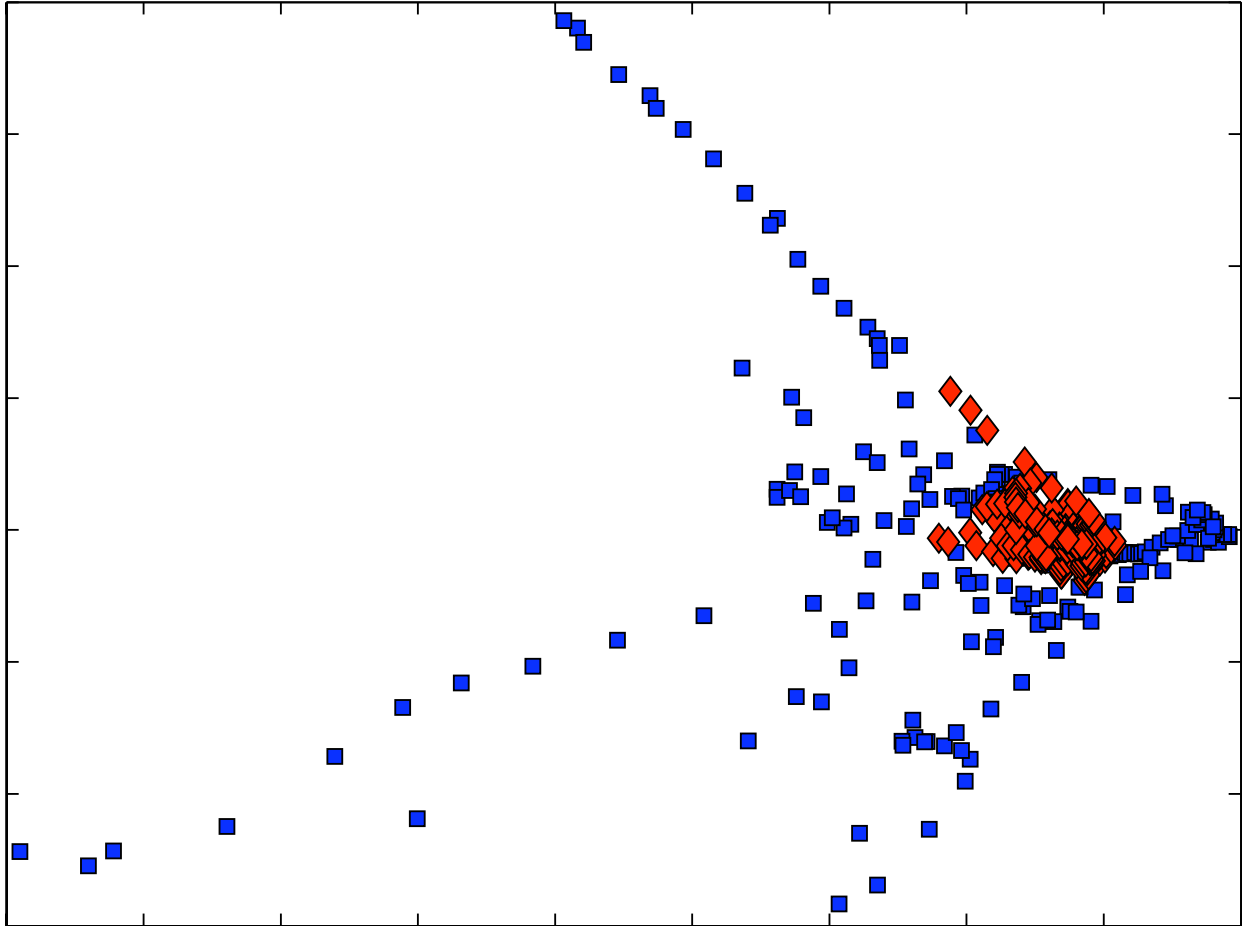


Figure 3: A plot of some examples projected down onto the two most significant dimensions of variation using PCA analysis. Music data is shown as red diamonds while non-music data is shown as blue squares. The arrangement seems promising for the application of a radial kernel.

3.3 Classification

We tried several classification methods in an effort to differentiate our classes of audio data. First, we tried using a linear discriminant analysis. This method failed miserably, performing barely better than chance. The reason for such a failure was that there was indeed no suitable plane that could be used to separate our data. For completeness, we also attempted quadratic linear discriminant analysis, which has been used successfully in the past for audio classification [MB03]. We got only marginally better results with this approach.

In order to further investigate the relationship of the different examples, we performed a principle component analysis to find the two most significant dimensions of variation in the data. We then projected the data down onto this reduced basis in order to visualize how music and non-music data might be separated using a kernel. A visualization of the results

is shown in Figure 3. The music data lies clustered in a tight ball, while the non-music data is spread more evenly through out the space. This is a hint that a non-linear, perhaps radial kernel might function better for this application.

After the failure of simpler techniques, we tried applying a support vector machine. We didn't spend any time trying a linear kernel because of the failure of the LDA and QDA methods. Instead, we directly applied a Gaussian-based radial basis function kernel. Parameters for such kernels are notoriously difficult to locate, so we resorted to a grid search for the C and γ parameters that define tolerance and shape of the fitting kernels. The grid search proceeded as follows:

1. Separate the training data into two equal-size sets.
2. For given values of C and γ , compute the cross-validation accuracy of the SVM against this pseudo-training set.
3. Repeat for many values of C and γ in a grid until the optimal values are located.

Figure 4 shows the results of the grid search, which yielded optimal values $C = 32$ and $\gamma = 2$. Once these values had been obtained, final training of the SVM using the training data was performed, and the novel test data was used to determine the accuracy of classification.

4 Results

Overall, our classifier was quite successful. For the training and testing sets used, it correctly classified music and non-music data 88.6% of the time. The optimization concluded within about 11,000 iterations and included 1365 total support vectors. It correctly predicted the class of 3690 of 4172 examples.

Running times were quite manageable. On a four-processor Power Mac G5 2.5GHz, the grid search for parameter optimization broadly dominated the running time at about 10.2 minutes. Once parameters were selected, the training of the SVM took 2.90 seconds, and classification of the 4172 examples took just 1.04 seconds.

5 Discussion and Future Work

We were pleased with the success of our simple audio classifier. It is clearly quite rudimentary, but it demonstrates that music / non-music classification is a tenable task easily met by available methods for feature extraction and classification.

One obvious area for improvement would be to massively increase the size of the example set to include hundreds or thousands of clips from many more genres of music and types of non-music data. We found, encouragingly, that the more examples we added to our original set of two clips for each class, the better the classifier became. This indicates that the differences between music and non-music data are in some sense intrinsic and are resistant to specific examples influencing large-scale classification behavior.

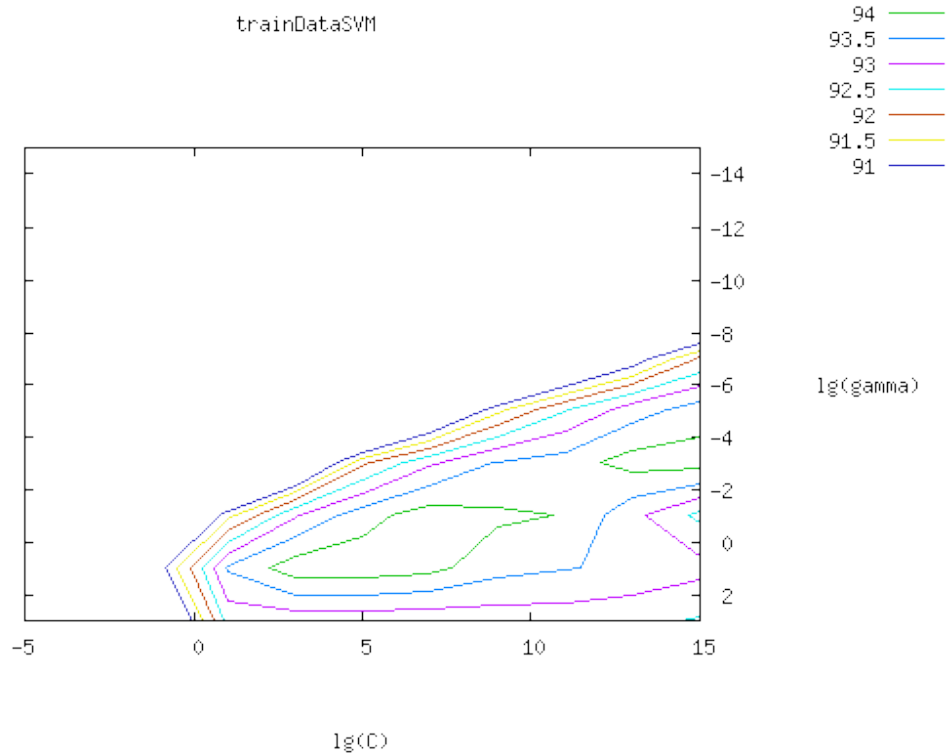


Figure 4: A plot of isocontours of cross-validation accuracy for different values of the RBF kernel parameters C and γ .

Also, although the RASTA sound features were quite effective in differentiating our two classes, it seems clear that in the future a set of perceptually-based audio features that more broadly capture the range of audio encountered by people could have a positive effect on the domain of automatic audio categorization. The features used today are still largely driven by speech recognition, which represents a tiny fraction of the palette of auditory stimulation we're exposed to every day.

Could this system be incorporated into radio receivers? Maybe. While the initial training of the SVM and the selection of its parameters took a considerable amount of computing power, the ultimate classification of results was quite speedy. Given the power of in-dash navigation computers that are commonly found in new vehicles, it doesn't seem unreasonable to imagine a pre-trained SVM making classification decisions on the fly and automatically seeking out radio stations that are playing whatever it is the listener wants to hear—whether that is music or not.

References

- [BC00] Kristin P. Bennett and Colin Campbell. Support vector machines: Hype or hallelujah? *SIGKDD Explorations*, 2(2):1–13, 2000.
- [CL01] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [Ell05] Daniel P. W. Ellis. PLP and RASTA (and MFCC, and inversion) in Matlab, 2005. online web resource.
- [Foo97] J. Foote. Content-based retrieval of music and audio, 1997.
- [HM94] H Hermansky and N Morgan. Rasta processing of speech. In *Proceedings of IEEE Transactions on Speech and Audio Processing*, 1994.
- [MB03] Martin F. McKinney and Jeroen Breebaart. Features for audio and music classification. In *Proceedings of the 4th International Conference on Music Information Retrieval*, Baltimore, MD, October 2003. Johns Hopkins University.
- [You95] S. Young. Large vocabulary continuous speech recognition: A review, 1995.