

# MUSIC GENRE CLASSIFICATION WITH TAXONOMY

Tao Li

School of Computer Science  
Florida International University  
Miami, FL, 33199

Mitsunori Ogihara

Department of Computer Science  
University of Rochester  
Rochester, NY, 14627

## ABSTRACT

Automatic music genre classification is a fundamental component of music information retrieval systems and has been gaining importance and enjoying a growing amount of attention with the emergence of digital music on the Internet. Although considerable research has been conducted in automatic music genre classification, little has been done on hierarchical classification with taxonomies.

The underlying hierarchical taxonomy identifies the relationships of dependence between different genres and provides valuable sources of information for genre classification. This paper investigates the use of taxonomy for music genre classification. Our empirical experiments on two datasets show that using taxonomy improves the classification performance. We also propose an approach for automatically generating genre taxonomies based on the confusion matrix via linear discriminant projection. Our work also provides some insights for future research.

## 1. INTRODUCTION

Music is not only for entertainment and for pleasure, but has been used for a wide range of purposes due to its social and physiological effects. At the beginning of the 21st century the world is facing ever-increasing growth of the on-line music information, empowered by the permeation of Internet into daily life. Efficient and accurate automatic music information processing (accessing and retrieval, in particular) will be an extremely important issue, and it has been enjoying a growing amount of attention.

A currently popular topic in automatic music information retrieval is the problem of automatic music genre classification. By automatic musical genre classification we mean here the most strict form of the problem, i.e., classification of music signals into a single unique class based computational analysis of music feature representations. The process of genre classification in music is divided into two steps: feature extraction and multi-class classification. In the first step, we extract from the music signals information representing the music. In the second step, we build a mechanism (an algorithm and/or a mathematical model) for identifying the labels from the representation of the music sounds with respect to their features.

There has been a considerable amount of work on extracting descriptive features for music signals and on music genre classification. Most studies in this area, however, are performed on datasets with a relatively small number of classes. In addition, they mainly focus on *flat classification*, in which the music genres are treated individually and equally so that no structures exist to define relationships among them [1]. Limitations to the flat

classification approach exists in the fact that, as the music industry grows, the number of possible genres increases and the borderlines among them are blurred. In reality, juke box programs and music stores use much larger number of classes. For example, Music-match Jukebox has a flat-level classification consisting of 34 genres. Amazon uses 23 main genres and further divides some of them according to music styles, media types, and artist groups. Barnes & Noble offer much more detailed classification. They use a three-tier genre and style classification, with eighteen top-level classes and with 1,000 bottom-level classes. To manage such a large number of genres and styles, hierarchical approaches (using taxonomy) will be effective. In such a hierarchical structure, the genres become more specific as we go down in the taxonomy. However, even with such an approach, building a sound-based classifier that is able to distinguish among 1,000 different classes seems a daunting task. (Also, it is true that not a large fraction of music listeners listen to 1,000 different genres and styles.)

The classification problem of a large number of styles and genres can be addressed by studying how minute classification can be done using a hierarchical taxonomy and a state-of-the-art classification algorithm. Rauber, Pampalk, and Merkl [2] use self-organizing map to divide music collection into nine groups. Their experiments show hierarchical approaches are promising. In this paper, we investigate the use of hierarchical taxonomy in music genre classification. Specifically, we discuss the reasons for incorporating the hierarchical taxonomy, experimentally evaluate the effect of using taxonomy and propose an automatic approach for generating the taxonomy.

The rest of the paper is organized as follows: Section 2 elaborates the reasons for incorporating the taxonomy; Section 3 introduces the feature extraction methods used in our experiments; Section 4 presents our experimental evaluation on the effect of using taxonomy in music genre classification; Section 5 proposes an approach for automatic taxonomy generation; Section 6 reviews related work and Finally Section 7 discusses our future work and concludes.

## 2. WHY USE TAXONOMY?

There are several reasons that the taxonomy is very useful for music genre classification. First, rather than issuing general queries, many users prefer to look for music information by browsing hierarchical catalogs and by issuing queries that are corresponding to specific types. Experiments have shown that using taxonomies improves usability, search success rate and user satisfaction.

Second, taxonomy structures identify the relationships of dependence between the genres and provide a valuable information

source for many problems. Generally, the use of hierarchical structures allows for efficiencies in both learning and representation. Hierarchical structures enable the use of a divide-and-conquer approach and thus result in higher efficiency and accuracy. In practice, each classifier has to deal with a more easily separable problem, and can use an independently optimized feature set; this should lead to improvements in accuracy apart from the gain in training and testing speed.

Third, using taxonomy allows the classification errors to be more acceptable than in the case of flat classification [3]. Divide-and-conquer approach makes the errors concentrate within the given level of the hierarchy.

### 3. FEATURE EXTRACTION

In this section, we describe the feature extraction methods used in our later experiments. The extracted feature contains traditional sound features including MFCC and other timbral features and DWCHs.

#### 3.1. Mel-Frequency Cepstral Coefficients (MFCC)

MFCC is designed to capture short-term spectral-based features. After taking the logarithm of the amplitude spectrum based on short-term Fourier transform for each frame, the frequency bins are grouped and smoothed according to Mel-frequency scaling, which is design to agree with perception. MFCC features are generated by decorrelating the Mel-spectral vectors using discrete cosine transform.

#### 3.2. Other Timbral Features

*Spectral Centroid* is the centroid of the magnitude spectrum of short-term Fourier transform and is a measure of spectral brightness. *Spectral Rolloff* is the frequency below which 85% of the magnitude distribution is concentrated. It measures the spectral shape. *Spectral Flux* is the squared difference between the normalized magnitudes of successive spectral distributions. It measures the amount of local spectral change. *Zero Crossings* is the number of time domain zero crossings of the signal. It measures noisiness of the signal. *Low Energy* is the percentage of frames that have energy less than the average energy over the whole signal. It measures amplitude distribution of the signal.

#### 3.3. DWCH

There are many kinds of wavelet filters, including Daubechies wavelet filter, Gabor filter etc. Daubechies wavelet filters are the one commonly in image retrieval (more details on wavelet filter can be found in [4]). In our work, we use Daubechies wavelet filter Db8 with seven levels of decomposition. After the decomposition, we construct the histogram of the wavelet coefficients at each subband. The coefficient histogram provides a good approximation of the waveform variations at each subband. From probability theory, a probability distribution is uniquely characterized by its moments. Hence, if we interpret the waveform distribution as a probability distribution, then it can be characterized by its moments. To characterize the waveform distribution, the first three moments of a histogram is used [5]. The first three moments are the average, the variance and the skewness of each subband. In addition, we also compute the subband energy, defined as the mean of the absolute value of coefficients, for each subband. In addition, our final

DWCH feature set also includes the tradition timbral features for speech recognition.

Our DWCH feature set contains four features for each of seven frequency subbands along with nineteen traditional timbral features. However, we found that not all the frequency subbands are informative and we only use four subbands. The total number of features is 35. More details can be found in [6].

## 4. EXPERIMENTS ON HIERARCHICAL APPROACHES

This section presents our experimental evaluation on the effect of using taxonomy in music genre classification.

### 4.1. Datasets

We use two datasets for our experiments. The first dataset, Dataset A, contains 1000 songs over ten genres with 100 songs per genre. This dataset is used in [7]. The ten genres are *Blues*, *Classical*, *Country*, *Disco*, *Hip-hop*, *Jazz*, *Metal*, *Pop*, *Reggae*, and *Rock*. The excerpts of the dataset were taken from radio, compact disks, and MP3 compressed audio files. The second dataset, Dataset B, contains 756 sounds over five genres: *Ambient*, *Classical*, *Fusion*, *Jazz*, and *Rock*. This dataset was constructed for this paper from the CD collection of the second author. *Ambient* and *Fusion*, are thought of as the genre bridging between *Jazz* and *Classical* and as the genre bridging between *Jazz* and *Rock*, respectively. Because of this overlapping nature it is anticipated that the genre classification of this dataset is hard in general. The collection of 756 sound files was created from 189 music albums as follows: From each album the first four music tracks were chosen (three tracks from albums with only three music tracks). Then from each music track the sound signals over a period of 30 seconds after the initial 30 seconds were extracted in MP3. The distribution of different genres is: *Ambient* (109 files), *Classical* (164 files), *Fusion* (136 files), *Jazz* (251 files) and *Rock* (96 files). For both datasets, the sound files are converted to 22050Hz, 16-bit, mono audio files.

### 4.2. Experiment Setup

Figures 1 and 2 show the taxonomy structures for datasets A and B respectively. The hierarchies are manually generated by the second author.

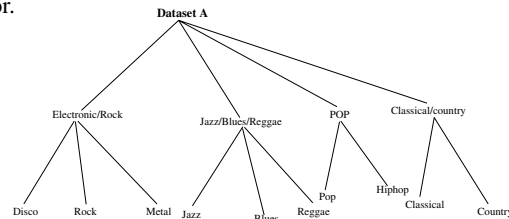
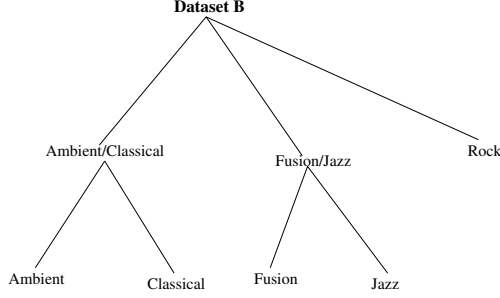


Fig. 1. Taxonomy Structure for Dataset A.

An obvious approach to utilization of the taxonomy is a top-down level-based approach that arranges the clusters in a two-level tree hierarchy and trains a classifier at each internal node. We first build a top-level classifier (L1 classifier) to discriminate among the top-level clusters of labels. At the second level (L2) we build classifiers within each cluster of classes. Each L2 classifier can concentrate on a smaller set of classes that confuse with each other.

To build classifiers we use Support Vector Machines [8] (SVM for short). Based on the theory of structural risk minimization,



**Fig. 2.** Taxonomy Structure for Dataset B.

SVMs are designed to learn a decision boundary between two classes by mapping the training examples onto a higher dimensional space and then determining the optimal separating hyperplanes between that space. SVMs have shown superb performance on music genre classification [6, 9]. Our SVM implementation is based on the LIBSVM [10], a library for support vector classification and regression. We use linear kernels in our experiments and all the experiments are performed on a P4 2GHz machine with 512M memory running Linux 2.4.9-31.

### 4.3. Experiment Results and Analysis

Table 1 gives the performance comparisons of flat classification with hierarchical classification. In the experiments, we randomly choose 70% of the data for training and assign the rest for testing. The results reported in Table 1 are the average values of five trials. From Table 1, we observe slight improvements on both datasets (about 0.7% on dataset A and 3% on dataset B).

Datasets	Flat	Hierarchical	Level-one
Dataset A	0.720137	0.726962	0.750853
Dataset B	0.675325	0.701299	0.835498

**Table 1.** Accuracy Table. The flat column gives the accuracy of flat classification, the Hierarchical and level-one columns present the hierarchical and level one accuracy respectively.

The slight improvement is partly due to the fact that we use the same set of features for all the classifiers, even though they are at different levels. Theoretically, Mitchell [11] shows that if the same feature space is used for all the classifiers and no smoothing is done, the accuracy of Bayesian hierarchical classification will be almost the same as that of a flat Bayesian classifier. However, since each classifier in hierarchical classification deals with a more easily separable problem, we can use an independently optimized feature set at each step. Hence to better utilize taxonomy for music genre classification, we should develop level-dependent and genre-specific feature extraction approaches for music signals and this is one of our future work.

## 5. TAXONOMY GENERATION

The taxonomy used in Section 4 is manually generated. Manual building of taxonomies is an expensive task since the process requires domain experts to evaluate the relevance of music genres. In addition, there may exist music collections in which there are no plausible human semantics and thus there are no natural taxonomies [1]. This motivates us to address the issue of automatically building genre hierarchies.

In this section, we present an approach for automatically generating music genre hierarchies. The core idea is to infer genre

relationships from the confusion matrix generated from some efficient classifiers. In practice, the confusion matrix can be constructed by applying the classifiers on a held-out validation set. We use linear discriminant projection [12]. Although it is possible to use other classification methods here, we choose linear discriminant projection because of its high efficiency and accuracy. In comparison, SVM is accurate but require long training time for multi-class problems. Naive Bayes is fast but it is not so accurate.

Linear discriminant projection [13] finds a discriminative feature transform that maximizes the ratio of intra-class scatter to the inter-class scatter and classification is then performed in the transformed space based on Euclidean distances. Mathematically, the transformation is determined by the eigenvectors associated with the largest eigenvalues of matrix  $T = \hat{\Sigma}_w^{-1} \hat{\Sigma}_b$  where  $\hat{\Sigma}_w$  is the intra-class covariance matrix and  $\hat{\Sigma}_b$  is the inter-class covariance matrix [12]. Table 2 shows an example of confusion matrix built on dataset B using our approach. We randomly select 30% of the data as a held-out validation set and the linear discriminant projection classifier is built using the remaining data.

Classes	1	2	3	4	5
<i>Ambient</i>	4	4	0	3	0
<i>Classical</i>	3	13	0	1	0
<i>Fusion</i>	0	0	4	8	1
<i>Jazz</i>	2	2	1	19	1
<i>Rock</i>	0	0	2	1	7

**Table 2.** Confusion Matrix for Dataset B.

The confusion matrix clearly shows the degrees of confusion among different classes. For example, there are a lot of overlaps between *Ambient* and *Classical*. Basically, the confusion matrix provides a domain-independent approach for inferring genre relationships. Using confusion matrix, each genre can then be represented as a multi-dimensional vector whose elements indicate the degree of confusions with other classes. To automatically generate taxonomies, we then apply hierarchical clustering [14]. By applying hierarchical clustering on Table 2, we actually get the taxonomy shown in Figure 2.

Table 3 shows an example of the confusion matrix built on dataset A<sup>1</sup>. Note that the generated genre relationships derived from Table 3 are not totally agree with that of Figure 1. For example, there is no relation between *Blues* and *Jazz* in Table 3. The difference comes from the fact that manually generated taxonomies are optimized for human use as they based on “human semantics” while the automatically generated taxonomies are optimized for computational classifiers. An important direction is to combine the automatic and manual approaches for generating both statistically significant and intuitively meaningful taxonomies.

## 6. RELATED WORK

A considerable amount of work has been reported on automatic music genre classification. Tzanetakis and Cook [7] propose a comprehensive set of features for direct modeling of music signals and explore the use of those features for musical genre classification using K-Nearest Neighbors and Gaussian Mixture models. Lambrou et al. [15] use statistical features in the temporal domain as well as three different wavelet transform domains to classify

<sup>1</sup>Due to space limit, we do not include the hierarchy generated from Table 3.

Classes	1	2	3	4	5	6	7	8	9	10
Blues	10	0	0	0	0	0	0	0	0	0
Classical	0	7	2	0	0	1	0	0	0	0
Country	0	2	5	0	0	2	0	1	0	0
Disco	0	0	2	6	0	0	0	0	2	0
Hip-hop	1	0	0	0	9	0	0	0	0	0
Jazz	0	0	1	1	1	6	0	0	0	1
Metal	0	0	0	0	1	0	9	0	0	0
Pop	0	0	1	0	0	3	0	6	0	0
Reggae	0	1	0	0	0	1	0	0	8	0
Rock	0	0	2	0	0	2	0	0	0	6

**Table 3.** Confusion Matrix for Dataset A.

music into rock, piano and jazz. Deshpande et al. [16] use Gaussian Mixtures, Support Vector Machines and Nearest Neighbors to classify the music into rock, piano, and jazz based on timbral features. Pye [17] investigates the use of Gaussian Mixture Modeling (GMM) and Tree-Based Vector Quantization in music genre classification. Soltau et al. [18] propose an approach of representing temporal structures of input signal. They show that this new set of abstract features can be learned via artificial neural networks and can be used for music genre identification. A comparative study on music genre classification is presented in [6].

Much less work has been reported on hierarchical approaches for music genre classification. Pachet and Cazaly [19] analyze existing taxonomies of musical genres and discuss issues in building taxonomies. Burred and Lerch [3] describe a system for automatic audio signals classification using a hierarchical approach. Xu et al. [9] propose a multi-layer classifier based on support vector machines for music genre classification. Rauber, Pampalk, and Merkl [2] use self-organizing map to divide music collection into nine groups. Our contributions are three-fold. First, we analyze the advantages of using taxonomy in music genre classification. Second, we empirically evaluate the effect of incorporating hierarchies for genre classification. Finally, we present an approach based on linear discriminant projection to automatically generate music hierarchies.

## 7. CONCLUSIONS

In this paper, we investigate the use of hierarchical taxonomy in music genre classification. Specifically, we discuss the reasons for incorporating taxonomy, experimentally evaluate the effect of using taxonomy and propose an approach for generating hierarchical taxonomy.

Our future goals are: to develop level-dependent and genre-specific feature extraction approaches for music signals, to combine the automatic and manual approaches to generate both statistically significant and intuitively meaningful taxonomies, and to carefully create data collections for testing hierarchical approaches.

## 8. REFERENCES

- [1] J.-J. Aucouturier and F. Pachet, "Representing musical genre: A state of the art," *Journal of new musical research*, vol. 32, no. 1, pp. 83–93, 2003.
- [2] A. Rauber, E. Pampalk, and D. Merkl, "Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by sound similarities," in *Proceedings of the 3rd International Symposium on Music Information Retrieval*, 2002, pp. 71–79.
- [3] J. J. Burred and A. Lerch, "A hierarchical approach to automatic musical genre classification," in *Proceedings of the Sixth International Conference on Digital Audio Effects (DAFx-03)*, London, UK, 2003.
- [4] I. Daubechies, *Ten lectures on wavelets*, SIAM, Philadelphia, 1992.
- [5] A. David and S. Panchanathan, "Wavelet-histogram method for face recognition," *Journal of Electronic Imaging*, vol. 9, no. 2, pp. 217–225, 2000.
- [6] T. Li, M. Ogihara, and Q. Li, "A comparative study on content-based music genre classification," in *SIGIR'03*, 2003, pp. 282–289, ACM Press.
- [7] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, July 2002.
- [8] V. N. Vapnik, *Statistical learning theory*, John Wiley & Sons, New York, 1998.
- [9] C. Xu, C. N. Maddage, C. Xu, F. Cao, and Q. Tian, "Musical genre classification using support vector machines," in *Proceedings of ICASSP*, 2003.
- [10] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [11] T. Mitchell, "Conditions for the equivalence of hierarchical and flat Bayesian classifiers.," Tech. Rep., Carnegie-Mellon University, 1998.
- [12] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, Prentice Hall, 1988.
- [13] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, , no. 7, pp. 179–188, 1936.
- [14] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.
- [15] T. Lambrou, P. Kudumakis, R. Speller, M. Sandler, and A. Linney, "Classification of audio signals using statistical features on time and wavelet transform domains," in *Proc. Int. Conf. Acoustic, Speech, and Signal Processing (ICASSP-98)*, 1998, vol. 6, pp. 3621–3624.
- [16] H. Deshpande, R. Singh, and U. Nam, "Classification of music signals in the visual domain," in *Proceedings of the COST-G6 Conference on Digital Audio Effects*, 2001.
- [17] D. Pye, "Content-based methods for managing electronic music," in *Proceedings of the 2000 IEEE International Conference on Acoustic Speech and Signal Processing*, 2000.
- [18] H. Soltau, T. Schultz, and M. Westphal, "Recognition of music types," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing*, 1998.
- [19] F. Pachet and D. Cazaly, "A taxonomy of musical genres," in *Proceedings of Content-Based Multimedia Information Access Conference (RIAO)*, 2000.