Automatic Genre Classification as a Study of the Viability of High-Level Features for Music Classification

Cory McKay

Faculty of Music, McGill University
cory.mckay@mail.mcgill.ca

Abstract

This paper examines the potential of high-level features extracted from symbolic musical representations in regards to musical classification. Twenty features are implemented and tested by using them to classify 225 MIDI files by genre. This system differs from previous automatic genre classification systems, which have focused on low-level features extracted from audio data. Files are classified into three parent genres and nine sub-genres, with average success rates of 84.8% for the former and 57.8% for the latter. Classification is performed by a novel configuration of feed-forward neural networks that independently classify files by parent genre and sub-genre and combine the results using weighted averages.

1 Introduction

There has been significant research into using features based on information derived directly from signal properties (referred to here as "low-level features") to classify audio recordings into categories based on qualities such as genre and style. Although this research is certainly very valuable, the current lack of reliable polyphonic transcription systems makes it difficult to extract and use features based on musical abstractions (referred to here as "high-level features") from audio recordings, as this requires precise knowledge of information such as the pitch and timing of individual notes.

This is unfortunate, as high-level features have the potential to provide information that could be highly characteristic of particular categories. Although low-level features have been shown to be useful and should certainly continue to be researched and used, parallel research involving high-level features should also be pursued.

There is a large body of existing recordings in symbolic formats from which high-level features can be extracted, including scores and digital formats such as MIDI, Humdrum, MusicXML and GUIDO. Optical music recognition techniques can also be used to process scores into digital files from which high-level features can be extracted.

If one has access to both audio recordings and symbolic recordings, it is then possible to take advantage of both high and low-level features. High-level features also make it possible to classify scores, be they paper or digital, when audio recordings are not available. Furthermore, future improvements in transcription systems may make it possible to extract high-level features from audio recordings. Research now into establishing useful high-level features could then be incorporated immediately into audio classification systems, which could take advantage of both types of features.

The potential of high-level features is explored in this paper by using them in a system that classifies MIDI recordings by genre. MIDI was chosen here because a diverse range of training and test recordings are available in this format. Although it is true that genre classification of MIDI files in particular is not a particularly pressing problem from a practical perspective, the features discussed here could just as easily be extracted from other formats such as Humdrum. In any case, the classification performed here is intended primarily as an examination of the viability of high-level features, not as an end in itself.

This being said, automatic genre classification is a particularly interesting and potentially useful task. Browsing and searching by genre can be very effective tools for users of the rapidly growing networked music archives. It is currently necessary to perform manual classifications in many cases, which is both time-consuming and inconsistent. Genre classification is only one possible application of the techniques used here, however, which could be applied to any type of classification that makes use of supervised learning.

2 Related Work

There have been a number of exciting studies on automatic genre classification of audio files. For example, Tzanetakis et al. (Tzanetakis, Essl & Cook 2001; Tzanetakis & Cook 2002) used a variety of low-level (and a few high-level) features to achieve success rates of 61% when classifying between ten genres.

Additional research has been performed by Grimaldi, Kokaram and Cunningham (2003), who achieved a success rate of 73.3% when classifying between five categories. Kosina (2002) achieved a success rate of 88% with three genres. Xu et al. (2003) achieved a success rate of

93% with four categories. Deshpande, Nam and Singh (2001) constructed a system that correctly classified among three categories 75% of the time. McKinney and Breebaart (2003) achieved a success rate of 74% with seven categories. Jiang et al. (2002) correctly classified 90.8% of recordings into five genres.

There has been somewhat less research into the classification of symbolic data. Shan and Kuo (2003) achieved success rates between 64% and 84% for two-way classifications using features based solely on chords and melodies. Recordings were classified into categories of Enya, Beatles, Chinese folk and Japanese folk. Chai and Vercoe (2001) were successful in correctly distinguishing between Austrian, Genrman and Irish fold music 63% of the time using only melodic features.

There has also been some important research (Whitman and Smaragdis 2002) on combining features derived from audio recordings with "community metadata" that was derived from text data mined from the web. Although beyond the scope of this paper, this line of research holds a great deal of potential, although the metadata can be difficult to find, parse and interpret.

Although there has been a great deal of work on analyzing and describing particular types of music, there has been relatively little research on deriving features from music in general. Alan Lomax and his colleagues in the Cantometrics project (Lomax 1968) have performed the most extensive work, by comparing several thousand songs from hundreds of different cultural groups using thirty-seven features. These features provide a good starting point for developing a library of high-level features. Although there have been a few other efforts to list categories of features, they have tended to be overly broad. Works such as Phillip Tagg's "checklist of parameters" (1982) are still useful as a general guide, however.

3 Choice of Features

One approach to devising a catalogue of features would be to make use of the large body of work on analytical musical techniques. Unfortunately, most of this work has limited applicability outside the particular types of music it was designed in response to. Furthermore, the successful automation of many sophisticated analytical systems remains an unsolved problem.

It is suggested here that a better approach is to keep features simple, at least initially. Ideally, one would like to get simple numbers for each feature that is extracted from recordings. This makes storing and processing features both simpler and faster. Features that represent an overall aspect of a recording are particularly appropriate in this respect. Features based on averages and standard deviations allow one to see the overall behavior of a particular aspect of a recording, as well as how much it varies. In some cases, vectors of features can also be useful, such as a list of numbers, each describing the fraction of notes played by different instruments, or sequences of melodic intervals.

Such features should take advantage of the high-level information that is available in symbolic representations, namely the knowledge of the pitch, timing, voice, instrumentation and potentially dynamics of each note. Seven broad classes of features are suggested here:

- **Instrumentation:** What types of instruments are present and which are given particular importance relative to others? The importance of non-pitched instruments and their interaction with pitched instruments could be of particular interest.
- **Texture:** How many independent voices are there and how do they interact (e.g. polyphonic, homophonic, etc.)? What is the relative importance of different voices?
- **Rhythm:** The time intervals between the attacks of different notes and the durations of each note can be considered. What kind of meters and rhythmic patterns are present? Is rubato used? How does rhythm vary from voice to voice?
- **Dynamics:** How loud are notes and how much variation in dynamics is there?
- **Pitch Statistics:** What are the occurrence rates of different notes? How tonal is the piece? What is the range? How much variety in pitch is there?
- **Melody:** What kinds of melodic intervals are present? Is there a lot of melodic variation? What kinds of melodic contours are used? What types of phrases are used and are they repeated often?
- **Chords:** What kinds of notes occur simultaneously? Are there any obvious harmonic progressions? Is there a drone? Are particular vertical intervals particularly prominent?

Ideally, one would like to create a large catalogue of such features, and then apply feature selection techniques to select the features that are most appropriate for a particular taxonomy. This would be especially useful if one is using a hierarchal taxonomy. One could first make a coarse classification with a certain set of features, and then use different sets of features to make progressively finer classifications as one progresses down the hierarchy, depending on the results of earlier classifications.

Given that this was an initial investigation of the viability of high-level features, only twenty features were implemented for this experiment. They are described in Table 1. A number of these features are based on the periodicity and pitch histograms used by Tzanetakis and Cook (2002) and Brown (1993), which provide a rich resource for features. The periodicity histograms consisted of beats-per-minute bins that were constructed using autocorrelation to derive the frequencies of lags between MIDI note-ons. All of the features consisted of one value per feature per recording except for the Orchestra-

tion feature, which was made up of a vector of 128 on/off values, one for each General MIDI patch. This feature had one network dedicated to it.

These features in particular were selected because they were easy to implement and give a general description of recordings without being optimized to the particular genre taxonomy that was used. Although there is no doubt that twenty better features could be devised, these particular features were chosen simply to show that even non-optimal features could still perform well.

Feature	Explanation
Orchestration	Which of the 128 MIDI instruments are played
Number of instruments	Total number of instruments played
Percussion prevalence	Fraction of note-ons belonging to unpitched instruments
Dominant pitch prevalence	Fraction of note-ons corresponding to the most common pitch
Dominant pitch class prevalence	Fraction of note-ons corresponding to the most common pitch class
Dominant interval	Number of semi-tones between the two most common pitch classes
Adjacent fifths	Number of consecutive pitch classes separated by perfect 5ths that represent at least 9% of the notes
Pitch class variety (common)	Number of pitch classes that represent at least 9% of the notes
Pitch class variety (rare)	Number of pitch classes played at least once
Register variety	Number of pitches played at least once
Range	Difference between highest and lowest pitches
Pitchbend fraction	Number of pitch bends divided by total number of note-ons
Dominant periodicity	Magnitude of the highest periodicity bin
Second dominant periodicity	Magnitude of the second highest periodicity bin
Combined dominant periodicities	Combined magnitude of the two highest periodicity bins
Dominant periodicity strength ratio	Ratio of the frequencies of the two highest periodicity bins
Dominant periodicity ratio	Ratio of the periodicities of the two highest periodicity bins
Number of strong periodicities	Number of periodicity bins with normalized magnitude > 0.1
Number of moderate periodicities	Number of periodicity bins with normalized magnitude > 0.01
Number relatively high periodicities	Number of periodicity bins with frequencies at least 25% as high as the highest magnitude

Table 1: Features extracted from MIDI files and fed into neural networks.

4 Details of the Experiment

The training and testing data consisted of 225 MIDI files hand classified hierarchically into three parent genres (Classical, Jazz and Pop) and nine sub-genres (Baroque, Romantic, Modern Classical, Swing, Funky Jazz, Cool Jazz, Rap, Country and Punk). The particular files that were chosen were selected so as to represent each category as broadly as possible (e.g. the Baroque category included operas, violin concertos, harpsichord sonatas, etc., not just organ fugues, for example). This significantly increased the difficulty of the task, as each sub-genre only had 20 training recordings (five recordings were reserved for testing in each run) to learn a broad range of music. This was done in order to truly test the viability of the system and its features.

The recordings were classified using an array of eight feed-forward neural networks that consisted of four networks for identifying parent genres and four networks for identifying sub-genres. Each network had a single hidden layer. This division into two groups made it possible to classify parent genres independently from sub-genres.

The input units of each network took in different groups of features (orchestration, pitch statistics, rhythm statistics or stylistic), thus making it possible to study the relative success of the different features in classifying the test data. This made it possible to compare how well different feature groups performed.

A coordination system considered the certainty score output by the networks for each sub-genre in combination with the certainty for each parent genre, and produced a final classification using weighted averages.

This particular classification system was used because it allowed the independent comparison of different groups of features as well as a comparison of how well parent genres were classified relative to sub-genres.

5 Results

A five-fold cross-validation was used to test the performance of the system. The results are shown below:

	Set 1	Set 2	Set 3	Set 4	Set 5	Average
Classical	93	80	100	93	100	**93.2**
Jazz	73	80	60	53	40	**61.2**
Pop	100	100	100	100	100	**100.0**
Average	88.7	86.7	86.7	82.0	80.0	**84.8**

Table 2: Classification success rates (in percentages) for parent genres for all five cross-validation testing runs.

	Set 1	Set 2	Set 3	Set 4	Set 5	Average
Baroque	80	40	80	80	80	**72.0**
Romantic	0	40	0	20	40	**20.0**
Modern	100	40	100	40	80	**72.0**
Swing	40	80	20	40	20	**40.0**
Funky Jz.	60	40	60	40	0	**40.0**
Cool Jz.	40	20	20	20	0	**20.0**
Rap	80	60	80	60	20	**60.0**
Country	80	100	100	100	100	**96.0**
Punk	100	100	100	100	100	**100.0**
Average	64.4	57.8	62.2	55.6	48.9	**57.8**

Table 3: Classification success rates (in percentages) for sub-genres for all five cross-validation testing runs.

Overall success rates of 84.8% were achieved for parent genres and 57.8% for sub-genres across all five train-

ing runs. These results were fairly consistent across cross-validation testing runs, as can be seen by the standard deviations of 3.6% and 6.1% respectively. There was also a consistent difference in which categories were successfully classified, with Punk and Country performing very well and Cool Jazz and Romantic performing very poorly.

Interestingly enough, even when the system was tested using training data, success rates of only 93.4% and 77.1% were achieved for parent and sub-genres respectively. This did not change significantly, even when the number of training epochs was increased and the number of hidden nodes was varied.

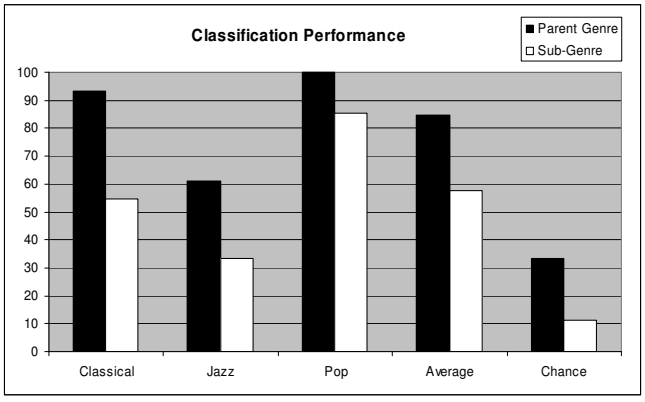

Figure 1: Average classification success rates on test sets. The sub-genre bars give the average success rates of the sub-genres belonging to the corresponding parent genres.

6 Conclusions

As can be seen from Figure 1, the test set was classified at a success rate significantly higher than chance in all cases. Furthermore, the system achieved success rates comparable to existing audio classification systems using similar numbers of categories (see Section 2). This is particularly encouraging, given the limited feature set, small training sample and broad categories used here. This is good evidence that high-level features hold a good deal of potential for application to musical classification, and future research is certainly warranted.

The fact that the neural networks did not entirely converge during training and that particular categories consistently performed badly could be due to a number of factors. It could be that the particular features used did not provide sufficient information to effectively distinguish these categories from others. Alternatively, the categories may have been too broad given the number of training samples, so that there was not a clear enough pattern. The classification system itself may have been at fault as well, as neural networks can have a tendency to fall into local minima where certain categories are effectively ignored.

All of this justifies future research with a larger training/testing set that is analyzed with an improved catalogue of high-level features that could be used to more accurately describe different categories. Alternative classifiers could be used as well, as could more narrow and better defined categories. As discussed in Section 3, a hierarchical classification system that made use of feature selection techniques to choose the best features for each level of classification could be particularly effective.

7 Acknowledgments

Thanks to Ichiro Fujinaga for his invaluable advice and to the *Fonds Québécois de la recherche sur la société et la culture* for their generous support, which has helped to make this research possible.

References

Brown, J. C. 1993 Determination of meter of musical scores by autocorrelation. Journal of the Acoustical Society of America 94 (4): 1953-1957.

Chai, W., and B. Vercoe. 2001. Folk music classification using hidden Markov models. *Proceedings of the International Conference on Artificial Intelligence*.

Deshpande, H., U. Nam, and R. Singh. 2001. Classification of music signals in the visual domain. *Proceedings of the Digital Audio Effects Workshop*.

Grimaldi, M., A. Kokaram, and P. Cunningham. 2003. Classifying music by genre using a discrete wavelet transform and a round-robin ensemble. *Work Report*. Trinity College, University of Dublin, Ireland.

Jiang, D. N., L. Lu, H. J. Zhang, J. H. Tao, and L. H. Cai. 2002. Music type classification by spectral contrast feature. *Proceedings of Intelligent Computation in Manufacturing Engineering*.

Kosina, K. 2002. Music genre recognition. *Diploma thesis*. Technical College of Hagenberg, Austria.

Lomax, A. 1968. *Folk song style and culture*. Washington, D.C.: American Association for the Advancement of Science.

McKinney, M. F., and J. Breebaart. 2003. Features for audio and music classification. *Proceedings of the International Symposium on Music Information Retrieval*. 151–158.

Shan, M. K., and F. F. Kuo. 2003. Music style mining and classification by melody. *IEICE Transactions on Information and Systems* E86-D (3): 655–659.

Tagg, P. 1982. Analysing popular music: Theory, method and practice. *Popular Music* 2: 37–67.

Tzanetakis, G., and P. Cook. 2002. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing* 10 (5): 293–302.

Tzanetakis, G., G. Essl, and P. Cook. 2001. Automatic musical genre classification of audio signals. *Proceedings of the International Symposium on Music Information Retrieval*. 205–210.

Whitman, B., and P. Smaragdis. 2002. Combining musical and cultural features for intelligent style detection. *Proceedings of the International Symposium on Music Information* Retrieval. 47–52.

Xu, C., N. C. Maddage, X. Shao, F. Cao, and Q. Tian. 2003. Musical genre classification using support vector machines. *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*. V 429–432.