Exploring Music Review Score Prediction (2012)

Patrick Marchwiak
pd@marchwiak.com

ABSTRACT

Much of the popular and independent music released today is critically reviewed and judged on its quality and artistic merit. This is typically considered a subjective art performed by a human. This paper seeks to explore how well a computer can predict the general quality of a piece of music using music information retrieval and data mining techniques.

1. DATA GATHERING AND INTEGRATION

Manipulation of data and metadata needed to be repeated multiple times and with various parameters thus a modular pipeline architecture was used where the output of the previous process or script was used as the input to the next. In steps where data was acquired from an external source, it was saved to disk and processed from disk by the next step rather than in memory to minimise the amount of communication needed with the external sources.

Metacritic.com was used as the starting point for all data. Metacritic is a review aggregation website which normalizes scored reviews (those assigned a letter grade or numeric score in some range) to their own 0-100 scale and assigns scores to reviews without scores based on their overall sentiment. This methodoloy is somewhat controversial as it reduces all reviews to a single score losing all of the rich descriptive textual information. However this type of data works very well in the context of machine learning since reasoning about a “metascore” is simpler as well as gets closer to the notion of an objective measure of quality for an album.

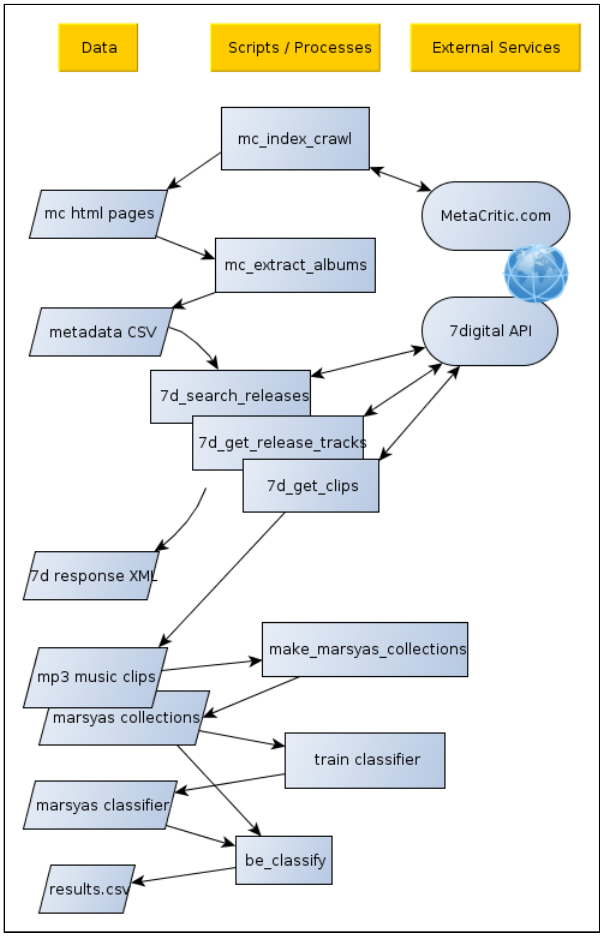

Figure 1: Metadata and audio processing pipeline

The Metacritic website displays reviews grouped by genre and sorted by date. These index pages were downloaded as HTML. The next script parsed out artist, album, and metascores and wrote them to a flat file (CSV).

A number of approaches were considered for acquisition of music files. Firstly, the use of personal music libraries was evaluated. These were found to be unsatisfactory as the overlap between them and the data on metacritic was not substantial. Additionally, people naturally tended to own more music that was rated highly which would result in an unbalanced sample set.

The second approach considered and the one ultimately used was the 7digital API service. 7digital is primarily a digital media distribution company and their API provides access to a wide range of metadata from their music catalog. With artist and album names in hand, these were used as input into another script which searched for releases with those names using the "release/search" resource. For each matching release, a track list was obtained using the "release/tracks" resource. The obtained track ids were finally used to obtain 30-second preview clips in MP3 format.

2. FEATURE EXTRACTION AND CLASSIFIER

Given a set of digital audio samples in the form of 30-second MP3 files, the next task was to extract features that could be used for classification or regression. Much research has been done on music similarity in the music information retrieval community. Mel-Frequency Cepstral Coefficients are one commonly used feature, initially popularized with speech recognition [3] but also shown to be effective for music similarity applications. MFCC frames capture the timbral attributes of a music signal [4] which is useful when judging the quality of a piece of music.

A number of libraries were considered for the task of feature extraction. YAAFE [5] was initially explored. Its improvement upon other similar frameworks such as Marsyas, is that for a set of features to be extracted, a feature extraction plan is created that removes redundancy by decomposing and reordering transformations. This results in faster feature extraction times. While YAAFE was easy to use and provided many extractable features, its main drawback for the purpose of this paper was that it provided no support for building models using the extracted features; that was up to the user.

The next library considered and the one that was ultimately chosen was Marsyas [7](as previously mentioned). It is a general purpose framework for supporting audio analysis and synthesis applications with emphasis on music information retrieval. The included "bextract" command line tool [1] was found to be very useful. It extracts means and variances of the timbral features (Zero Crossings, Spectral Centroid, Rolloff, Flux, and MFCC). The results can be stored in a .arff file (used in the Weka data mining library). This tool also has the ability to build a classifier in the form of a plugin which can be later used to classify new audio files. For the following experiments the Support Vector Machine classifier was used.

3. EXPERIMENTS

To simplify processing, rather than using the full range of scores in the Metacritic data set (15-95) and attempting to predict a score in this large range, the scores were binned using Metacritic's criteria. The range 100-81 is considered "Universal Acclaim", 61-80 "Generally Favorable", 40-60 "Mixed or Average", 20-39 "Generally Unfavorable", and 0-19 "Overwhelming Dislike" [2]. These categories were mapped to "vg" ("very good"), "g" ("good"), "f" ("fair"), "p"("poor"), and "vp" ("very poor"), respectively. Another decision that was made was to train and predict within genres as music varies significantly between them, making prediction much more difficult. Three genres were chosen. Country and rap were chosen for their vast difference in sound but relative similarity across works within their respective genres. The indie genre (short for independent) was also chosen, for its relative broad range of sound.

A collection of songs was created for each grouping of genre and category. Each collection's category was used as the label for the purpose of classification. A script was used to randomly select a user provided number of songs from each collection to build classifiers for each genre. The number of songs used to train each classifier was varied in order to avoid overtraining it as well as to reduce model computation time as it was found to be time intensive. An additional parameter that was varied was the length of the audio clip over which to extract features. Lengths of 5 and 10 seconds were used.

After each classifier was built, a random selection of 50 songs (not used during training) was used to test the accuracy of the classifier. Each song was classified and the results stored back into CSV. Results were varied, but in general were no better than random guessing. Accuracies for classifiers built with 10 samples , 5 seconds each were .33 (3 classes in test data), .47 (2 classes in test data), and .56 (2 classes in test data) for rap, indie, and country respectively. Increasing parameters such as length of audio clips and number of songs used to train classifiers produced similar results.

4. FUTURE WORK

There are many furthur directions to explore in further research.

This paper used the same set of features and the same type of classification algorithm for all experiments. Other classification algorithms that have shown promise in the literature are Guassian Mixture Models and K-NN type classifiers [6].

There are many additional features that can be extracted. In the audio realm, other spectral features should be considered alongside the timbral features. Additionally, different features may prove to be more useful for different genres. For example, a feature that captures rhythmic characteristics may be more appropriate when classifying music from the dance genre.

Lyrics are an important component of many musical works of art and were completely ignored in these experiments. Natural language processing techniques could be used to determine how lyrics factor into an album's metascore. Other textual metadata embedded into digital music formats such as artist, album, track names, years, as well as genre could be analyzed too.

CSV was used for storing metadata and while it was simple to start with it became evident that a relational database would work better.

5. SUMMARY

This paper discussed an exploration of using music information retrieval techniques to predict reviews scores. A pipeline for acquiring reviews, metadata, audio clips, and performing classification was successfully built. The results of the classifiers showed that there is still much work to be done and that music review score prediction is a difficult problem.

6. REFERENCES

[1] Marsyas user manual. http://web.archive.org/web/20100213173902/http://marsyas.info/docs/manual/marsyas-user/bextract.html, 2010.

[2] How we create the metascore magic. http://www.metacritic.com/about-metascores, 2012.

[3] M. Hunt, M. Lennig, and P. Mermelstein. Experiments in syllable-based recognition of continuous speech. 5:880 – 883, apr 1980.

[4] K. Jacobson. A multifaceted approach to music similarity. *Proceedings of International Conference on Music Information Retrieval*, 2006.

[5] B. Mathieu, S. Essid, and T. Fillon. Yaafe, an easy to use and efficient audio feature extraction software. *11th ISMIR*, (Ismir):441–446, 2010.

[6] G. Tzanetakis. Audio information retrieval (AIR) tools. *International Conference on Music Information Retrieval*, pages 1–2, 2000.

[7] G. Tzanetakis and P. Cook. MARSYAS: a framework for audio analysis. *Organised Sound*, 4(3):169–175, Dec. 2000.