A BENCHMARK DATASET FOR AUDIO CLASSIFICATION AND CLUSTERING

Helge Homburg, Ingo Mierswa, Bülent Möller, Katharina Morik and Michael Wurst
University of Dortmund, AI Unit
44221 Dortmund, Germany

ABSTRACT

We present a freely available benchmark dataset for audio classification and clustering. This dataset consists of 10 seconds samples of 1886 songs obtained from the Garageband site. Beside the audio clips themselves, textual meta data is provided for the individual songs. The songs are classified into 9 genres. In addition to the genre information, our dataset also consists of 24 hierarchical cluster models created manually by a group of users. This enables a user centric evaluation of audio classification and clustering algorithms and gives researchers the opportunity to test the performance of their methods on heterogeneous data. We first give a motivation for assembling our benchmark dataset. Then we describe the dataset and its elements in more detail. Finally, we present some initial results using a set of audio features generated by a feature construction approach.

Keywords: Benchmark Dataset, Audio Classification, Audio Clustering, Meta Learning

1 CHALLENGES IN LEARNING ON AUDIO DATA

Information retrieval has started several efforts to automatic indexing [1] and retrieval (e.g., querying by humming [2]). Machine Learning has shown its benefits for text classification and ranked document retrieval with respect to user preferences [3]. It is straightforward to expect a similar benefit for the classification and personalized retrieval of music records. However, this area is still very challenging for several reasons. Unlike many other types of data used with Machine Learning, audio data consists of time series which are usually quite large. Given a sampling rate of 44100 Hz, a three minute music record has a length of about $8 \cdot 10^6$ values. Moreover, current approaches to time series indexing and similarity measures rely on a more or less fixed time scale [4]. The key problem for automatic audio processing based on Machine Learning is to obtain a fixed set of features from the wave forms [5, 6, 7, 8, 9].

Beside the problems connected with these characteristics of audio data, current applications lead to additional challenging problems. Firstly, different classification tasks ask for different feature sets. It is not very likely that a feature set delivering excellent performance on the separation of classical and popular music works well also for the separation of techno and hip hop music. Machine Learning methods should be able to adapt to different areas of the input space. This is usually referred to as local learning [10].

Secondly, for many kinds of audio data important additional information exist as title or artist. This information is often called meta data. Other useful information about songs could be the lyrics, ratings or comments provided by listeners. To integrate all this information for Machine Learning is a very challenging task usually referred to as Multi View Learning [11].

Finally, audio processing based on Machine Learning is often applied in user oriented applications, such as personal media organizers (e. g. iTunes). Such organizers typically help users to manage a collection of songs by automatically classifying songs, clusterings songs, searching for similar songs, etc. However, music is a highly personal issue, users often arrange their songs using very different viewpoints [12, 13, 14]. This leads to problems similar to the ones mentioned above: classification with respect to different viewpoints may ask for different representations. Think for example of a first user, who arranges songs according to mood and a second user arranging songs according to whether the singer is male or female (as shown in Figure 1). The second task may require a set of features that is completely different from the first one, even if the songs themselves are from the same genre. Furthermore the possible viewpoints are usually neither restricted nor anticipated when the system is designed. The Machine Learning methods must be flexible enough to handle any possible viewpoint and thus classification. Still another problem in end user applications is that datasets are of varying size. While some users only arrange very few items, others have large collections of songs. Methods have to provide both, a high accuracy for small datasets and efficiency for large datasets.

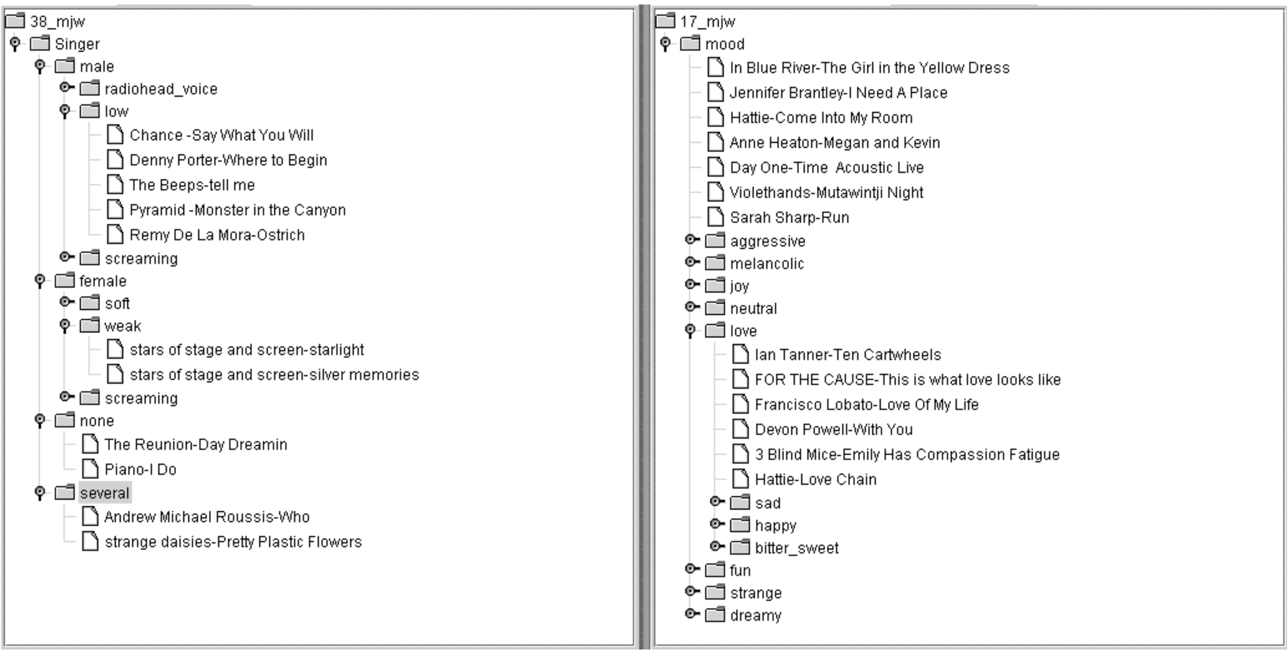

Figure 1: Two examples of user defined classification schemes.

The currently most popular freely available dataset is the RWC Music Database [15]. It provides audio samples together with extensive meta data and is well suited for the evaluation of many kinds of audio processing tasks. Unfortunately, the size of this data set is relatively small and hence does not meet the requirements of many Machine Learning methods. The dataset does also not contain different user viewpoints. We consider these heterogeneous viewpoints a major challenge for many real-world retrieval tasks.

The idea of our benchmark dataset is to provide a possibility to compare how different approaches and algorithms handle the described challenges. It reflects all of the above problems. It contains 1886 songs given as 10 s samples from 9 genres. Beside the audio data itself, meta data (band name, genre, etc.), user comments and partially even lyrics are available for each song. Also we provide 24 classification schemes created by our students using arbitrary personal viewpoints. This allows to test methods on very heterogeneous learning tasks, as could be expected in many real life user oriented scenarios. As the user classification schemes only cover parts of the songs, they also provide a way to test how well a given method can adapt to such local problems. Given audio and textual data, the dataset is especially well suited for Multi View Machine Learning. In the next section, the dataset is described in more detail.

2 THE DATASET

The dataset[1] consists of 1886 songs from the Garageband site. Garageband is a website that allows artists to upload their music and offer it for free download. Visitors of the site might download the audio clips, rate them or write comments. A group of students downloaded the songs together with some meta information. Then they created personal classification schemes on these songs described in section 2.4. The songs were taken from nine different genres: Pop, Rock, Folk/Country, Alternative, Jazz, Electronic, Blues, Rap/HipHop, Funk/Soul. The number of songs in each genre varies, Table 1 gives an overview.

Genre	Number
Blues	120
Electronic	113
Jazz	319
Pop	116
Rap/HipHop	300
Rock	504
Folk/Country	222
Alternative	145
Funk/Soul	47
total	1886

Table 1: Number of songs per genre.

2.1 Audio Samples

Each song is associated with a 10 second audio sample drawn from a random position of the corresponding song. Audio samples are encoded using mp3 with a sampling rate of 44100 Hz and a bitrate of 128 mbit/s.

2.2 Meta Data

The meta data for each song consists of several parts. These are the name and the length of the song, information about the genre, the band or artists name, and comments given by listeners. Lyrics are partially available.

[1] `www-ai.cs.uni-dortmund.de/audio.html`

prediction \ true	**Blues**	**Electronic**	**Jazz**	**Pop**	**HipHop**	**Rock**	**Folk/Country**	**Alternative**	**Funk/Soul**
Blues	18	4	26	6	16	23	1	6	0
Electronic	2	17	12	6	11	9	0	10	0
Jazz	29	42	171	37	39	64	0	34	0
Pop	4	3	14	15	5	15	0	10	0
HipHop	10	21	25	15	187	21	0	10	0
Rock	55	19	58	31	38	358	1	60	0
Folk/Country	0	0	0	0	0	0	213	0	37
Alternative	2	7	13	6	4	14	1	15	0
Funk/Soul	0	0	0	0	0	0	6	0	10

Table 2: The confusion matrix for k-NN on the genre data.

2.3 Audio Features

Based on the approach described in [7] a total number of 49 features were extracted from each song. These audio features are also part of the benchmark dataset. The set of features cover temporal features, spectral features, and some unusual features extracted in the the phase space of the audio data.

2.4 User Classification Schemes

A group of users created 24 classification schemes without any further specification. These schemes are of varying size and cover different subsets of the songs. The aspects used for classification differ considerably. For example, users arranged the songs according to genre, quality or preference, mood, time of day, instruments, singer, etc. The classification schemes are tree like structures in which every node has a label. Songs are allowed to be anywhere in the tree.

We think that these user defined classification schemes offer a challenging problem to audio classification and clustering, as they are very heterogeneous, mostly small and cover different subsets of the songs, thus require the ability for local adaptations from the algorithm.

3 INITIAL RESULTS

We performed some initial experiments on our dataset. All experiments were performed with the Machine Learning environment YALE [16]. YALE is available as open-source software under the GNU Public License (GPL)[2]. The next sections describe the performance which can be achieved on genre classification and on the user defined classification schemes.

3.1 Classifying Global Genres

A first learning task on our dataset is classification according to genre. The genre information is given as part of the meta data. Classification is done on the 49 features described in section 2.3. The learning schemes used were C4.5 decision trees, k-nearest neighbors with an adaptive distance metric, Naive Bayes, and a random classifier as baseline. Results were measured with a 10-fold cross validation. Table 3 shows the performance for all learning schemes.

[2] `http://yale.cs.uni-dortmund.de/`

	Accuracy
Random	26.72
C4.5	45.44
Naive Bayes	43.69
k-NN	53.23

Table 3: The accuracy for the genre classification.

	Accuracy
Random	44.07
C4.5	49.52
Naive Bayes	49.92
k-NN	49.63

Table 4: The averaged accuracy for the user tasks.

The confusion table for the complete dataset for genre classification with Nearest Neighbor is shown in Table 2. The ability of the algorithm to classify audio clips depends on the genre. For some genres, as alternative, it is very hard for the algorithm to find the correct classification. However, we can assume that even human judgment would not come to a definite agreement in this case. Small genres, as Funk/Soul, lead to poor classification performance as well. This can be explained by the small number of examples in these classes. We plan further experiments using multi aspect learning combining textual information and audio information.

3.2 Classifying User Schemes

The user defined classification schemes are well suited for diverse evaluation tasks as audio classification, audio clustering or similarity search. In this section we present results on audio classification. A hierarchical classification scheme can be interpreted as a set of nested classification tasks (using every inner node as splitting point). Using only inner nodes having child nodes with at least ten items, we obtained 27 flat classification problems. We used several learners on these problems and calculated the average performance. The results are presented in Table 4. All learners yield a poor result. This motivates the hypothesis that a the feature set is more important than a particular learning scheme. Especially, we expect that an optimal feature set is highly dependent on the given learning task. The empirical evidence for this hypothesis is given in [17]. In a second experiment we applied the feature

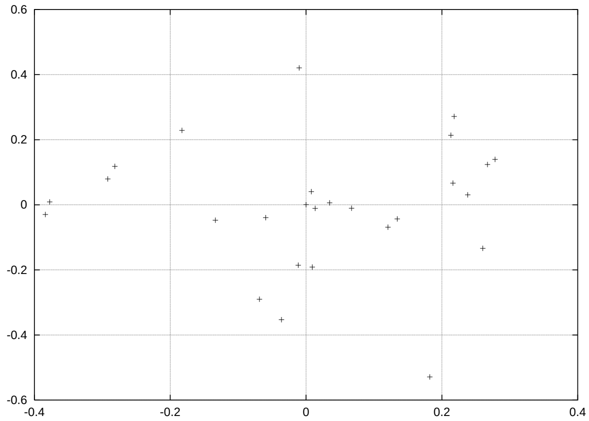

Figure 2: Feature weights of the user classification after a dimensionality reduction.

construction scheme described there in this work. This leads to 27 feature weight vectors describing the utility of each feature for each of the 27 classification tasks. To visualize the resulting matrix, we performed a dimensionality reduction based on singular value decomposition. The result is shown in Figure 2. Each point represents a classification task. Tasks that are close to each other employ similar feature weights. Although we can see that some tasks resemble each other to some extent, in general different tasks require different features. This observation supports our thesis and gives rise to a meta learning approach [17].

We strongly believe that heterogeneity poses an important challenge to future audio applications. We hope that our dataset is a humble contribution to the scientific work in this domain.

ACKNOWLEDGEMENTS

We would like to thank the members of the student project Nemoz for creating and providing their personal classification schemes.

REFERENCES

[1] F. Kurth and M. Clausen. Full-text indexing of very-large audio data bases. In *110th Convention of the Audio Engineering Society*, 2001.

[2] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith. Query by Humming: Musical Information Retrieval in an Audio Database. In *Proc. of ACM Multimedia*, pages 231–236, 1995.

[3] T. Joachims. *Learning to Classify Text using Support Vector Machines*. Kluwer International Series in Engineering and Computer Science. Kluwer, 2002.

[4] E. Keogh and P. Smyth. An enhanced representation of time series which allows fast classification, clustering and relevance feedback. In *Procs. of the 3rd Conference on Knowledge Discovery in Databases*, pages 24–30, 1997.

[5] G. Guo and S. Z. Li. Content-Based Audio Classification and Retrieval by Support Vector Machines. *IEEE Transaction on Neural Networks*, 14(1):209–215, 2003.

[6] Z. Liu, Y. Wang, and T. Chen. Audio Feature Extraction and Analysis for Scene Segmentation and Classification. *Journal of VLSI Signal Processing System*, 1998.

[7] I. Mierswa and K. Morik. Automatic feature extraction for classifying audio data. *Machine Learning Journal*, 58:127–149, 2005.

[8] G. Tzanetakis. *Manipulation, Analysis and Retrieval Systems for Audio Signals*. PhD thesis, Computer Science Department, Princeton University, 2002.

[9] T. Zhang and C. Kuo. Content-based Classification and Retrieval of Audio. In *SPIE's 43rd Annual Meeting - Conference on Advanced Signal Processing Algorithms, Architectures, and Implementations VIII*, 1998.

[10] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2001.

[11] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Annual Conference on Computational Learning Theory (COLT-98)*, 1998.

[12] S. Baumann, T. Pohle, and V. Shankar. Towards a socio-cultural compatibility of mir systems. In *Proc. of the 5th International Conference on Music Information Retrieval (ISMIR 2004)*, 2004.

[13] S. Jones and S. J. Cunningham. Organizing digital music for use: an examination of personal music collections. In *Proc. of the 5th International Conference on Music Information Retrieval (ISMIR 2004)*, 2004.

[14] E. Pampalk, S. Dixon, and G. Widmer. Exploring music collections by browsing different views. In *Proc. of the 4th International Conference on Music Information Retrieval (ISMIR 2003)*, 2003.

[15] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. Rwc music database: Popular, classical, and jazz music databases. In *Proc. of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, pages 287–288, 2002.

[16] S. Fischer, R. Klinkenberg, I. Mierswa, and O. Ritthoff. Yale: Yet Another Learning Environment – Tutorial. Technical Report CI-136/02, Collaborative Research Center 531, University of Dortmund, Dortmund, Germany, 2002.

[17] I. Mierswa and M. Wurst. Efficient case based feature construction for heterogeneous learning tasks. Technical Report CI-194/05, Collaborative Research Center 531, University of Dortmund, 2005.