Selection of audio features for music emotion recognition using production music

Chris Baume[1], György Fazekas[2], Mathieu Barthet[2], David Marston[1], and Mark Sandler[2]

[1]*BBC Research and Development, London, UK*

[2]*Queen Mary University of London, UK*

Correspondence should be addressed to Chris Baume (chris.baume@bbc.co.uk)

ABSTRACT

Music emotion recognition typically attempts to map audio features from music to a mood representation using machine learning techniques. In addition to having a good dataset, the key to a successful system is choosing the right inputs and outputs. Often, the inputs are based on a set of audio features extracted from a single software library, which may not be the most suitable combination. This paper describes how 47 different types of audio features were evaluated using a five-dimensional support vector regressor, trained and tested on production music, in order to find the combination which produces the best performance. The results show the minimum number of features that yield optimum performance, and which combinations are strongest for mood prediction.

1. INTRODUCTION

As the world's largest broadcaster, the BBC plays over 200,000 different music tracks every week. Currently, producers can only search for tracks using basic metadata such as title, artist and album. This makes it difficult for them to find tracks they do not already know, meaning that only a small proportion of the music library is used. This leads to well known problems such as popularity bias, and infrequent or no utilisation of the long-tail of these catalogues, resulting in the repetition of a small number of tracks across programmes.

The goal of the 'Making Musical Mood Metadata' (M4) project is to explore different avenues for facilitating access to the vast music libraries available to the BBC and enable their better utilisation in our programmes. M4 is a collaborative project between BBC R&D, Queen Mary University of London's Centre for Digital Music (QMUL C4DM) and music content provider I Like Music (ILM). ILM provides the BBC with an online music library service called 'Desktop Jukebox', which contains over a million commercial and about 400,000 production music tracks.

The core objective of the project is to enhance the service using semantic audio technologies (an overview of this field is presented in [1]), in order to make it easier for users to find the music that best matches their criteria. A large proportion of the project uses music emotion recognition (MER) to enhance the metadata of the music library to allow users to search for music in a more intuitive way based on mood (see Figure 1).

This paper details how machine learning techniques were used with a multi-dimensional mood model and production music library to find the optimum combination of audio features for mood recognition. Section 2 sets the work in the context of previous related work. Section 3 explains what production music is and why it might provide better training data in music emotion recognition. Section 4 introduces the multi-dimensional mood model that was used. Section 5 outlines the process that was used to extract a large number of audio features from the production music. Section 6 explains how these features were evaluated in order to find the best performing combination. Section 7 compares the performance of the best combination to using all features. Finally, the results are summarised and discussed in Section 8.

2. RELATED WORK

This research follows previous studies [2] which use machine learning methods to map from audio features to a mood representation. Traditionally the features used in mood prediction systems have been chosen because

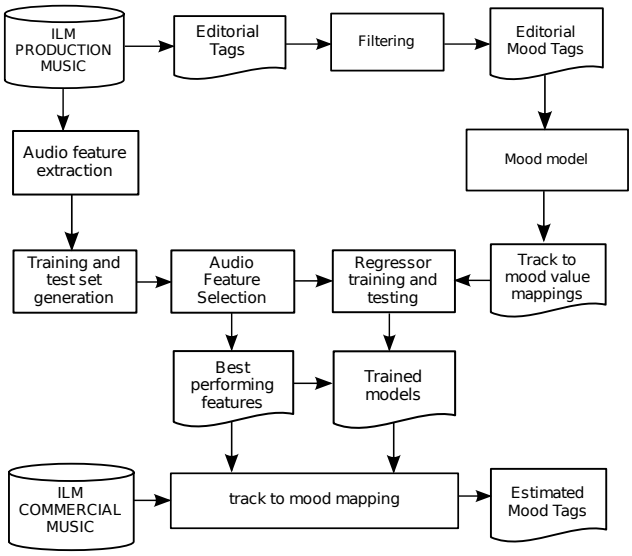

Fig. 1: Summary of tasks and experimental workflow within the project.

they belong to a single feature extraction software library, such as MIR Toolbox [3]. Therefore, features are often selected not because they provide the most optimal combination for mood prediction, but rather for convenience. By selecting the features that are most useful in the context of this task, the performance of the system can be improved, while the time required to extract the features and train/test the system can be reduced.

In previous studies on MER, different groups of features have been evaluated [4], and various feature selection algorithms have also been tested [5], but to date there has been little or no work on testing individual features. Additionally, the way in which different combinations of features affect the perfomance of machine learning algorithms hasn't been considered.

Machine learning requires the use of large labelled data sets. These data sets are not only expensive to create, they are typically unavailable for researchers due to licensing restrictions and commercial interests. In this project, the above issues are alleviated by having access to a large aggregated catalogue of production music with high-quality curated metadata. This metadata includes mood related tags entered by professionals at production music libraries using predefined guidelines and tagging policies.

Production music has previously been used at MIREX [6] as ground truth for the mood classification task, where 1,250 tracks were selected from the libraries of Associated Production Music (APM). A similar but alternative approach has been to use soundtracks [7,8], which (like production music) are primarily instrumental, but had to be hand-labelled.

The top overall mood classifier for MIREX in 2012 was based on support vector machines [9]. Support vector regressors have been shown to outperform multiple linear regression [10], Gaussian Mixture Models [11] and BoostR [10] for mood prediction. For these reasons, support vector regression was chosen as the machine learning method in our experiments.

3. PRODUCTION MUSIC

Production music is so named because it is often used in television, radio and film productions for background and 'mood' music. Typically, the music content is freely provided to content producers and the rights holders are paid for each time the music is used or broadcast.

With most commercial music, the rights for each track may belong to a number of parties. For instance, the musical works may be owned by one record company, whilst the sound recording may be owned by another. Potential licencees have to negotiate with all parties involved in order to use a music track.

Production music differs from commercial music in three major ways. Firstly, all of the rights are owned by one organisation, which greatly simplifies the licensing process. The majority of the music is instrumental, although it covers a wide range of genres including classical, jazz, electronic, rock and pop amongst others. Finally, it comes with a significant amount of manually-entered metadata.

As the income of the rights holders is based on how often their music is chosen, it is in their best interest for the metadata to be complete and accurate. However, each music supplier uses its own set of keywords and has different policies on how music is tagged. For instance, some use a tighly-controlled vocabulary whilst others allow free-tagging. This causes the data to be less organised than one might expect.

3.1. Music data

ILM have made 128,024 tracks of production music available to this project for the purposes of developing music classification algorithms. The music is sourced

from over 29 different suppliers and the tracks are stored as uncompressed 16-bit PCM wave files. 116,142 or 90.7% of them are sampled at 48kHz, with the remainder sampled at 44.1kHz. The mean track length is 01:59. For each track the metadata includes the library and label names; catalogue number; album name, composer and description; and track title, description, categories, mood tags, instruments, tempo, duration and International Standard Recording Code (ISRC [12]).

4. MOOD MODEL

In order to be able to describe the mood of the analysed music, a semantic mood model was developed to numerically represent mood. A data-driven dimensional model was chosen due to the large number of tracks and the richness of the metadata. It was created by analysing the frequency of co-occurrence of mood tags in the production music library. A detailed explanation of the mood model and how is was developed is available in [13]. A brief summary is provided below to help the understanding of the complementary contribution of the feature selection method detailed in this paper.

The production music metadata was analysed and mood tags that are used over 100 times throughout the collection were selected. Spelling mistakes, punctuation and duplicates were removed and a stemming algorithm was used to group tags with similar base parts (e.g. "joyful" and "joy"). This process resulted in a total of 453 stems. By considering the stem pairwise co-occurrences over all of the tracks (how often each pair of stems are used together on track descriptions), a stem dissimilarity matrix was generated. Multi-dimensional scaling [14] was then applied to the matrix, producing 3–, 5– and 11–dimensional representations of the mood tag relationships.

The optimum number of dimensions was chosen by conducting a listening test. Each of the tracks used in the experiment were labelled with mood values of 3, 5 and 11 dimensions. The test presented users with a seed track and four candidate tracks – three were randomly chosen and the fourth was chosen to minimise a distance criterion to the seed track's mood values. The users were asked to listen to the tracks and choose which candidate track sounded closest to the seed track in terms of the mood it portrayed. By using different numbers of dimensions for the candidate track selection, the results showed that a five-dimensional model performed, on average, better than three or eleven dimensions.

Although the model's dimensions are abstract and do not necessarily represent any particular attributes, it was demonstrated that the five-dimensional model could be rotated so that three of the five dimensions are significantly correlated with the arousal, valence and dominance emotion dimensions [13]. However for recommendation applications, the unrotated model is used as the relative distance of the tracks is kept invariant through rotation.

5. FEATURE EXTRACTION PROCESS

There are many feature extraction software packages available, including MATLAB-based ones such as MIR Toolbox [3], MA Toolbox [15] and PsySound3 [16], or open source C++/Python libraries such as Marsyas [17], CLAM [18], LibXtract [19], Aubio [20] and YAAFE [21]. However, it remains difficult to know whether audio features computed by different audio feature extraction and analysis tools are mutually compatible or interchangeable. Moreover, if different tools were used in the same experiment, the outputs typically need conversion to some sort of common format, and for reproducibility, this wrapper code needs to evolve with the changes of the tools themselves [22].

To resolve these issues, we used the Vamp plugin architecture [23] developed at QMUL as a standardised way of housing feature extraction algorithms. A large number of algorithms are available as Vamp plugins, and as they can all be used from the command line using Sonic Annotator [24], it is easy to extract a wide variety of features. Five Vamp plugin collections were selected for use as part of the project – the QMUL plugin set, the BBC plugin set, NNLS Chroma, Mazurka, and libxtract. The plugins developed and used in this project are all available as open source software online[1].

A previous study by the first author [25] analysed 63 features computed from these Vamp plugins by using them as the input to a simple classifier aiming to categorise the songs into four basic moods. The results showed that some of the features (e.g. spectral kurtosis and skewness) had no correlation with the four basic moods which were considered in the experiments, so these were not included in the extraction processes. Of the remaining 47 algorithms, 40 were used with their default settings, while the remaining ones were set up with a variety of configurations, producing a total of 59 features. These are listed in Table 1.

[1] `http://www.vamp-plugins.org/download.html`

We collaborated with the University of Manchester to use the new N8 high performance computing (HPC) cluster to extract the features from 128,024 music tracks. The tracks were first down-mixed to mono in order to save time when transferring the files over FTP. Since all of the algorithms use a mono input, this does not affect the result. Once the files were on the cluster, a separate task was run for each music track in which Sonic Annotator [24] was used to extract each feature. As a result of parallelisation, all designated features were extracted from the collection in less than seven hours.

SQ1	MFCC (20 coefficients) [26]	HQ2	Tonal content function [27]
SQ2	MFCC (12 coefficients) [26]	HQ3	Key strength [28]
SQ3	MFCC (7 coefficients) [26]	HQ4	Chromagram
SM1	Spectral flatness [29]	HN1	NNLS harmonic change [30]
SL1	Spectral spread [29]	HN2	Consonance [31]
SL2	Spectral std dev [29]	HN3	NNLS chromagram [31]
SL3	Spectral slope [29]	HN4	NNLS bass chroma [31]
SL4	Spectral inharmonicity [29]	HL1	First tristimulus [29]
SL5	Spectral smoothness [32]	HL2	Second tristimulus [29]
SL6	Spectral sharpness [29]	HL3	Third tristimulus [29]
SL7	Spectral rolloff [29]	RQ1	Beat counts
SL8	Spectral irregularity (K [33])	RM1	Smooth power slope [34]
SL9	Spectral irregularity (J [35])	RM2	Scaled smooth pwr slope [34]
SL10	Spectral centroid [29]	RB1	Peak-valley ratio [36]
SL11	Bark coefficients	RB2	Rhythm strength [36]
SB1	Spectral valley (5 bands) [37]	RB3	Mean correlation peak [36]
SB2	Spectral valley (7 bands) [37]	RB4	Mean onset freq [36]
SB3	Spectral valley (9 bands) [37]	EL1	Total loudness [29]
SB4	Spectral peak (5 bands) [37]	EB1	RMS energy (10ms window)
SB5	Spectral peak (7 bands) [37]	EB2	RMS energy (20ms window)
SB6	Spectral peak (9 bands) [37]	EB3	Intensity [36]
SB7	Spectral mean (5 bands) [38]	TL1	Zero-crossing rate [29]
SB8	Spectral mean (7 bands) [38]	TL2	Variance [29]
SB9	Spectral mean (9 bands) [38]	TL3	Non-zero count
SB10	Spectral flux (L1 norm) [39]	TL4	Lowest value
SB11	Spectral flux (L2 norm) [39]	TL5	Highest value
SB12	Intensity ratio (5 bands) [36]	TL6	Average deviation [29]
SB13	Intensity ratio (7 bands) [36]	TB1	Temporal centroid [29]
SB14	Intensity ratio (9 bands) [36]	TB2	Log attack time [29]
HQ1	6D Tonal Content Space [27]		

Table 1: List of features extracted and tested. The letters accompanying each feature relate to the Vamp plugin which implements that feature (L = LibXtract, N = NNLS, Q = QMUL, B = BBC, M = Mazurka) and the category (S = Spectral, H = Harmonic, R = Rhythmic, E = Energy, T = Temporal)

6. FEATURE SELECTION

Feature selection is the process of selecting a subset of features for the purpose of removing redundant data. By utilising this pre-processing stage, the accuracy and speed of machine learning-based systems can be improved. Generally speaking, feature selection involves choosing subsets of features and evaluating them until a stopping criterion is reached.

6.1. Evaluation strategy

The different subset evaluation techniques broadly fall into three categories [40]: the filter model, the wrapper model and the hybrid model. The filter model relies on general characteristics of the data to evaluate feature subsets, whereas the wrapper model uses the performance of a predetermined algorithm (such as a support vector machine) as the evaluation criterion. The wrapper model gives superior performance as it finds features best suited to the chosen algorithm, but it is more computationally expensive and specific to that algorithm. The hybrid model attempts to combine the advantages of both.

Most studies which have employed feature selection in the context of music mood/genre classification have used the filter model [5, 9, 10], with the ReliefF algorithm [41] being particularly popular. Notably however, a few studies [7, 8] have successfully employed the wrapper model in feature selection for music mood prediction. Therefore, and due to its superior performance, this method was chosen as the feature selection evaluation strategy in our project.

6.2. Dataset

A reduced dataset was used for the evaluation as it would have been impractical to test so many features on all 128,024 tracks. The chosen tracks were randomly selected from the production music library, with each track coming from a different album (to avoid the 'album effect' [42]), being over 30 seconds in length (to avoid shortened versions of tracks) and containing explicitly labelled mood tags. This resulted in 1,760 tracks. Each feature was scaled to the range $[0, 1]$ before the tracks were randomly split into two-thirds training (1,173) and one-third testing (587). Where the feature is time-varying, the following six metrics were used to summarise the output: mean, standard deviation, minimum, maximum, median and mode.

Although some of these statistics assume a Gaussian distribution of audio features (an assumption which clearly does not hold in all cases), we found that the above combination of metrics provide a reasonable compromise. Using a bag-of-frames approach as an alternative would require storing large amounts of frame-wise feature data, which isn't practical given the size of the target music

collection in which our method will be applied within the project.

The features were evaluated by using combinations of them as the input, and the five-dimensional mood representation (described in Section 4) as the output.

6.3. Feature Selection algorithm

As there are over 5.7×10^{17} different possible combinations of the 59 features, it would have been impossible to perform an exhaustive search. To solve this a forward sequential search was used, where features are added successively. The feature selection algorithm is described as pseudocode below.

```
testFeats = combinations(featList, N)
for i = N:len(featList),
  bestFeats = sort(evaluate(testFeats))
  testFeats = []
  for j = 0:M-1,
    for k = 0:len(featList)-1,
      testFeats.add([bestFeats[j], featList[k]])
```

The process starts with generating a set containing every combination of N features. Each combination is evaluated and the M best combinations are chosen. A new set of combinations is generated by adding every one of the 59 features to each of the top M to make combinations with $(N+1)$ features. The best M combinations are chosen again and the process is repeated until all features are used. To maximise the computing time available in the project, the parameters were set as $N = 2$ and $M = 12$.

6.4. Machine learning

The system was implemented by using five support vector regressors (SVR), each based on a polynomial kernel. The implementation used the scikit-learn [43] Python module. Although the RBF kernel has often been used for MER [9, 11], a recent study has shown the polynomial kernel to be faster and more accurate [4]. Two-fold cross-validation was used with each regressor to perform a grid search for the parameters C and gamma. For C, the search range was $[2^{-5}, 2^{5}]$ and for gamma it was $[2^{-13}, 2^{-3}]$. Each regressor was trained using the optimum parameters from cross-validation and evaluated against the test set, using absolute error as the performance metric.

6.5. Results

Figure 2 shows the best absolute error achieved for each regressor (one for each dimension of the mood model) when combining up to 19 different features. The first local minima marked with triangles. Table 2 shows which features were used to achieve those minima. The full results, which show the features that were used for every point of the graph in Figure 2, are available online[2]. The overall minimum mean error achieved by using the best feature combinations for each regressor was 0.1699.

From the shape of the plots in Figure 2, we can see that using more features produces diminishing returns before reaching a baseline.

Table 2 shows that mood prediction benefits from a wide variety of features (32 in this case) from every category. A recent previous experiment [4] found that "inclusion of dynamics features with the other classes actually impaired the performance of the classifier while the combination of spectral, rhythmic and harmonic features yielded optimal performance". Our results generally support these findings, with two notable exceptions – a dynamics feature (RMS energy) was required to optimise the performance of one of the regressors, and temporal features are also important for optimum mood prediction.

Some of the regressors were more reliant on certain categories than others. For example, SVR4 doesn't use any harmonic features. This suggests that it would be advantageous to optimise the features for individual dimensions.

Finally, there is a large difference in error rates between the regressors. Although the error of SVR4 is notably worse than the others, it was later found that its error improved significantly with a larger training set, bringing it in line with the error of the other regressors. It is hypothesised that because SVR4 uses fewer features, it needs information from more tracks.

It is interesting to note that the three best performing regressors (2, 3 and 5) are those which are most highly correlated with arousal, valence and dominance (see Table 1 in [13]). Not only that, but the best performing regressor (3) has the highest correlation[3]. This strong relationship may suggest that the features optimised for SVR3 (and to a lesser extent SVR2 and SVR5) are particularly relevant when using an arousal/valence/dominance representation.

7. SELECTION EVALUATION

In order to see how well the selected features perform, they were compared against the use of all features with

[2] Full results available at `http://bbcarp.org.uk/m4/aes53`

[3] Valence (r=0.47, p<0.001) and Arousal (r=0.33, p<0.001)

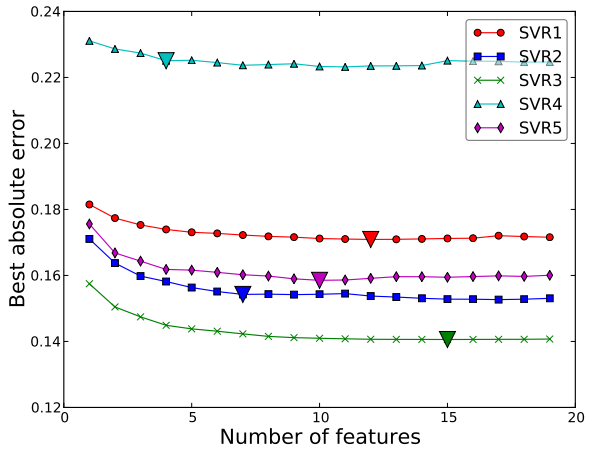

Fig. 2: The absolute error of the best performing combinations for each of the five regressors. The first local minima are marked with triangles.

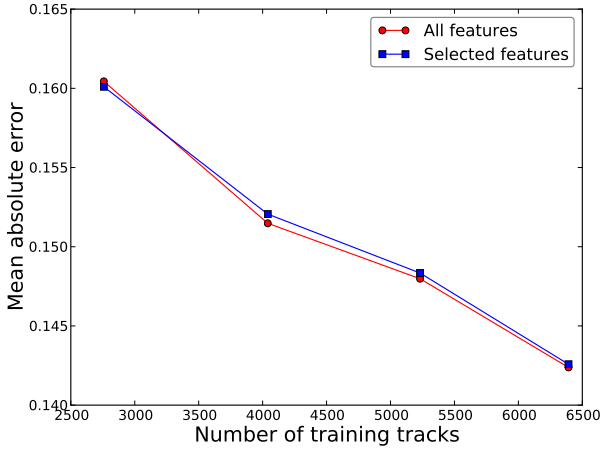

Fig. 3: Comparison of performance between using all features and using the features chosen by the selection process.

	Feature	SVR1	SVR2	SVR3	SVR4	SVR5
SQ1	MFCC (20 coeffs)	•				
SQ2	MFCC (12 coeffs)			•		
SQ3	MFCC (7 coeffs)		•			
SM1	Spectral flatness	•		•		•
SL1	Spectral spread	•				
SL4	Spectral inharmonicity			•	•	
SL5	Spectral smoothness				•	
SL6	Spectral sharpness	•				•
SL8	Spectral irregularity (K)			•		
SL10	Spectral centroid					•
SL11	Bark coefficients			•		
SB1	Spectral valley (5 bands)			•		
SB2	Spectral valley (7 bands)	•				
SB3	Spectral valley (9 bands)		•			
SB14	Intensity ratio (9 bands)			•		
HQ1	Tonal content space	•				
HQ2	Tonal content function		•			
HQ3	Key strength	•		•		
HN1	NNLS harmonic change					•
HN2	Consonance		•			
RM1	Smooth power slope	•		•		•
RM2	Scaled smooth pwr slp	•	•	•	•	
RB1	Peak-valley ratio		•			
RB2	Rhythm strength			•		•
RB3	Mean correlation peak	•				
RB4	Mean onset frequency		•	•		•
RQ1	Beat counts	•				•
EB1	RMS energy			•		
TL1	Zero-crossing rate	•		•		•
TL3	Non-zero count				•	
TL4	Lowest value					•
TL5	Highest value			•		

Table 2: Best feature combinations for each regressor.

a larger dataset. Unfortunately, the resources required to train/test the system using all 128,024 tracks were not available, so a variety of reduced sets were used.

7.1. Method

Four datasets were created by selecting tracks from the production music library. As in Section 6.2, each track was over 30 seconds and contained explicit mood tags. For each album in the library, $[3,5,7,9]$ tracks were chosen at random to create the four datasets. The features were scaled to the range $[0,1]$ and randomly split into two-thirds training, one-third testing. The same machine learning method detailed in Section 6.4 was employed, including use of two-fold cross-validation.

7.2. Results

The evaluation results are shown in Figure 3. It can be seen that the features chosen in Section 6 match the performance of using all of the features together. It also shows that the system does not suffer from overfitting, and that doubling the number of training items gives approximately a 10% error reduction. As the error does not tail off, it is not known how many tracks are needed for optimum performance.

8. CONCLUSIONS

This paper set out to find the combination of audio features that performs best when used as part of a music emotion recognition system. 1,760 production music tracks were trained and tested on a machine learning system that used support vector regressors with a polynomial kernel. The output of the system was a five-dimensional mood representation outlined in Section 4 and detailed in [13]. Various combinations of 59 features were used as the input in order to find the combination that produced the best results.

It was found that there is an optimum combination of features to use, and that using all available features is not necessary. The best performing combination used 32 features (shown in Table 2) from a wide variety of categories, including spectral, harmonic, rhythmic and temporal features. This shows that to achieve the best results, many different types of features needed to be taken into account, including rhythmic and temporal features. The optimum combinations of features were very different for each of the five dimensions of the mood model, showing that (in this case at least) features should be selected on a per-dimension basis.

Figure 3 shows that using the selected features matches the performance of using all features. Although 32 features were needed to produce the optimum results, the plots in Figure 2 converge quite quickly, so good results should be achievable with only a dozen or so well-chosen features.

9. FURTHER WORK

The work described here only shows results for training with 6,392 tracks out of the 128,024 production music tracks for which feature data is available. By using a larger training set, it is expected that the error will be further reduced, but it is not known yet by how much. Experiments will be conducted to see how many music tracks are needed to minimise the error.

As the production music metadata contains keywords covering a variety of topics, it would be possible to use the dataset to train classifiers for genre and instrumentation, as well as mood. Additionally, as the time-varying feature data has been retained, it may be possible to try and map information such as mood over time, within a track.

10. ACKNOWLEDGEMENTS

This research was carried out as part of the BBC Audio Research Partnership[4]. It was part-funded by a grant from the Technology Strategy Board (Grant No. TS/J002283/1).

This work made use of the facilities of N8 HPC provided and funded by the N8 consortium and EPSRC (Grant No. EP/K000225/1). The Centre is co-ordinated by the Universities of Leeds and Manchester.

11. REFERENCES

[1] György Fazekas, Yves Raimond, Kurt Jacobson, and Mark Sandler. An overview of Semantic Web activities in the OMRAS2 project. *Journal of New Music Research*, 39(4):295–311, 2010.

[2] Mathieu Barthet, György Fazekas, and Mark Sandler. Music Emotion Recognition: From Content to Context-Based Models. In *From Sounds to Music and Emotions (Lecture Notes in Computer Science Volume 7900)*, pages 228–252. Springer, 2013.

[3] Olivier Lartillot, Petri Toiviainen, and Tuomas Eerola. A Matlab Toolbox for Music Information Retrieval. In *Proc. Conference of the Gesellschaft für Klassifikation*, 2008.

[4] For more information, see `http://www.bbc.co.uk/rd/projects/audio-research-partnership`

[4] Yading Song, Simon Dixon, and Marcus Pearce. Evaluation of Musical Features for Emotion Classification. In *Proc. International Society for Music Information Retrieval Conference*, pages 523–528, 2012.

[5] Shyamala Doraisamy, Shahram Golzari, Noris Mohd. Norowi, Md. Nasir B. Sulaiman, and Nur Izura Udzir. A Study on Feature Selection and Classification Techniques for Automatic Genre Classification of Traditional Malay Music. In *Proc. International Society for Music Information Retrieval Conference*, pages 331–336, 2008.

[6] Xiao Hu, JS Downie, Cyril Laurier, and Mert Bay. The 2007 MIREX audio mood classification task: Lessons learned. In *Proc. International Society for Music Information Retrieval Conference*, pages 462–467, 2008.

[7] Tuomas Eerola and Olivier Lartillot. Prediction of multidimensional emotional ratings in music from audio using multivariate regression models. In *Proc. International Society for Music Information Retrieval Conference*, pages 621–626, 2009.

[8] Pasi Saari, Tuomas Eerola, and Olivier Lartillot. Generalizability and Simplicity as Criteria in Feature Selection: Application to Mood Classification in Music. *IEEE Transactions on Audio, Speech and Language Processing*, 19(6):1802–1812, August 2011.

[9] Renato Panda and Rui Pedro Paiva. Music Emotion Classification: Dataset Acquisition and Comparative Analysis. In *Proc. International Conference on Digital Audio Effects (DAFx)*, pages 1–7, 2012.

[10] Yi-Hsuan Yang, Yu-Ching Lin, Ya-Fan Su, and Homer H. Chen. A Regression Approach to Music Emotion Recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 16(2), 2008.

[11] Byeong-jun Han, Seungmin Rho, Roger B. Dannenberg, and Eenjun Hwang. SMERS: Music Emotion Recognition using Support Vector Regression. In *Proc. International Society for Music Information Retrieval Conference*, pages 651–656, 2009.

[12] ISO. ISO 3901:2001 - International Standard Recording Code (ISRC), 2001.

[13] Mathieu Barthet, David Marston, Chris Baume, György Fazekas, and Mark Sandler. Design and Evaluation of Semantic Mood Models for Music Recommendation. In *Proc. International Society for Music Information Retrieval Conference*, 2013.

[14] J B Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29:1–27, 1964.

[15] Elias Pampalk. A MATLAB Toolbox to Compute Music Similarity from Audio. In *Proc. International Society for Music Information Retrieval Conference*, 2004.

[16] Densil Cabrera, Sam Ferguson, and Emery Schubert. PsySound3: Software for Acoustical and Psychoacoustical Analysis of Sound Recordings. In *Proc. International Conference on Auditory Display*, pages 356–363, 2007.

[17] George Tzanetakis and Perry Cook. MARSYAS : A framework for audio analysis. *Organised Sound*, 4(3), 2000.

[18] Xavier Amatriain, Jordi Massaguer, David Garcia, and Ismael Mosquera. The CLAM Annotator : A Cross-platform Audio Descriptors Editing Tool. In *Proc. International Society for Music Information Retrieval Conference*, 2005.

[19] Jamie Bullock. LibXtract: A Lightweight Library for Audio Feature Extraction. In *Proc. International Computer Music Conference*, pages 3–6, 2007.

[20] Paul M Brossier. The Aubio Library at MIREX 2006. In *Proc. International Society for Music Information Retrieval Conference*, 2006.

[21] Benoit Mathieu, Slim Essid, Thomas Fillon, Jacques Prado, and Gael Richard. YAAFE: An Easy to Use and Efficient Audio Feature Extraction Software. In *Proc. International Society for Music Information Retrieval Conference*, 2010.

[22] György Fazekas, Sebastian Ewert, Alo Allik, Simon Dixon, and Mark Sandler. Shared open vocabularies and semantic media. In *Proc. International Society for Music Information Retrieval Conference*, 2012.

[23] Chris Cannam, Christian Landone, Mark Sandler, and Juan Pablo Bello. The Sonic Visualiser : A Visualisation Platform for Semantic Descriptors from Musical Signals. In *Proc. International Society for Music Information Retrieval Conference*, 2006.

[24] Chris Cannam, Michael O Jewell, Christophe Rhodes, and Mark Sandler. Linked Data And You : Bringing music research software into the Semantic Web The Semantic Web. *Journal of New Music Research*, 2010.

[25] Chris Baume. Evaluation of acoustic features for music emotion recognition. In *Proc. Audio Engineering Society Convention*, volume 134, 2013.

[26] S Davis and P Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics Speech and Signal Processing*, 28(4):357–366, 1980.

[27] Christopher Harte, Mark Sandler, and Martin Gasser. Detecting harmonic change in musical audio. In *ACM workshop on Audio and music computing multimedia (AMCMM)*, page 21, New York, New York, USA, 2006. ACM Press.

[28] Katy C Noland. *Computational Tonality Estimation: Signal Processing and Hidden Markov Models*. PhD thesis, 2009.

[29] G. Peeters. A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Technical report, 2004.

[30] Matthias Mauch and Simon Dixon. Approximate Note Transcription for the Improved Identification of Difficult Chords. In *Proc. International Society for Music Information Retrieval Conference*, number 1, 2010.

[31] Matthias Mauch. *Automatic Chord Transcription from Audio Using Computational Models of Musical Context*. PhD thesis, 2010.

[32] Stephen McAdams. Perspectives on the Contribution of Timbre to Musical Structure. *Computer Music Journal*, 23(3):85–102, 1999.

[33] J. Krimphoff, Stephen McAdams, and S. Winsberg. Caractérisation du timbre des sons complexes. *Journal de Physique IV*, 4:2–5, 1994.

[34] Craig Stuart Sapp. Mazurka project plugins (mazurka.org.uk/software/sv/plugin).

[35] Kristoffer Jensen. *Timbre Models of Musical Sounds*. PhD thesis, 1999.

[36] Lie Lu, Dan Liu, and Hong-Jiang Zhang. Automatic Mood Detection and Tracking of Music Audio Signals. In *IEEE Transactions on Audio, Speech and Language Processing*, volume 14, pages 5–18, 2006.

[37] Dan-Ning Jiang, Lie Lu, and Hong-Jiang Zhang. Music type classification by spectral contrast feature. In *IEEE International Conference on Multimedia and Expo*, pages 113–116, 2002.

[38] Vincent Akkermans, Joan Serrà, and Perfecto Herrera. Shape-based spectral contrast descriptor. In *Proc. Sound and Music Computing Conference*, number July, pages 23–25, 2009.

[39] Simon Dixon. Onset Detection Revisited. In *Proc. International Conference on Digital Audio Effects (DAFx)*, pages 133–137, 2006.

[40] Huan Liu and Lei Yu. Toward Integrating Feature Selection Algorithms for Classification and Clustering. In *IEEE Transactions on Knowledge and Data Engineering*, volume 17, pages 491–502, 2005.

[41] Marko Robnik-Sikonja and Igor Kononenko. Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning Journal*, (53):23–69, 2003.

[42] Youngmoo E Kim, Donald S Williamson, and Sridhar Pilli. Towards Quantifying the 'Album Effect' in Artist Identification. In *Proc. International Society for Music Information Retrieval Conference*, 2006.

[43] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.