AUTOMATIC CREATION OF MOOD PLAYLISTS IN THE THAYER PLANE: A METHODOLOGY AND A COMPARATIVE STUDY

Renato Panda
CISUC – DEI
University of Coimbra
panda@student.dei.uc.pt

Rui Pedro Paiva
CISUC – DEI
University of Coimbra
ruipedro@dei.uc.pt

ABSTRACT

We propose an approach for the automatic creation of mood playlists in the Thayer plane (TP). Music emotion recognition is tackled as a regression and classification problem, aiming to predict the arousal and valence (AV) values of each song in the TP, based on Yang's dataset. To this end, a high number of audio features are extracted using three frameworks: PsySound, MIR Toolbox and Marsyas. The extracted features and Yang's annotated AV values are used to train several Support Vector Regressors, each employing different feature sets. The best performance, in terms of R^2 statistics, was attained after feature selection, reaching 63% for arousal and 35.6% for valence. Based on the predicted location of each song in the TP, mood playlists can be created by specifying a point in the plane, from which the closest songs are retrieved. Using one seed song, the accuracy of the created playlists was 62.3% for 20-song playlists, 24.8% for 5-song playlists and 6.2% for the top song.

1. INTRODUCTION

Since the beginning of mankind music has always been present in our lives, serving a myriad of purposes both socially and individually. Given the major importance of music in all human societies throughout history and particularly in the digital society, music plays a relevant role in the world economy.

As a result of technological innovations in this digital era, a tremendous impulse has been given to the electronic music distribution industry. Factors like the widespread access to the Internet, bandwidth increasing in domestic accesses or the generalized use of compact audio, such as mp3, have contributed to that boom. The frenetic growth in music supply and demand uncovered the need for more powerful methods for automatically retrieving relevant songs in a given context from such huge databases. In fact, any large music database, or, generically speaking, any multimedia database, is only really useful if users can find what they are seeking in an efficient manner. Furthermore, it is also important that the organization of such a database can be performed as objectively and efficiently as possible.

Digital music repositories need, then, more advanced, flexible and user-friendly search mechanisms, adapted to the requirements of individual users. In fact, "music's preeminent functions are social and psychological", and so "the most useful retrieval indexes are those that facilitate searching in conformity with such social and psychological functions. Typically, such indexes will focus on stylistic, mood, and similarity information." [1]. This is supported by studies on music information behavior that have identified music mood[1] as an important criterion for music retrieval and organization [2].

Besides the music industry, the range of applications of mood detection in music is wide and varied, e.g., game development, cinema, advertising or the clinical area (in the motivation to compliance to sport activities prescribed by physicians, as well as stress management).

Compared to music emotion synthesis, few works have been devoted to emotion analysis. From these, most of them deal with MIDI or symbolic representations [3]. Only a few works tackle the problem of emotion detection in audio music signals, although it has received increasing attention in recent years. Being a recent research topic, many limitations can be found and several problems are still open. In fact, the present accuracy of those systems shows there is plenty of room for improvement. In a recent comparison, the best algorithm achieved an accuracy of 65% in a task comprising 5 categories [4].

Several aspects make music emotion recognition (MER) a challenging task. On one hand, the perception of the emotions evoked by a song is inherently subjective: different people often perceive different, sometimes opposite, emotions. Besides, even when listeners agree in the kind of emotion, there's still ambiguity regarding its description (e.g., the adjectives employed). Additionally, it is not yet well-understood how and why music elements create specific emotional responses in listeners [5].

For a long time, mood and emotions has been a major subject of psychologists and so several theoretical models have been proposed over the years. Such models can be divided into two approaches: categorical models or dimensional models. Categorical models consist of several states of emotion (categories), such as anger, fear, happiness and joy. Dimensional models use several axes to map emotions into a plane. The most frequent approaches

[1] Even though mood and emotion can be defined differently, the two terms are used interchangeably in the literature and in this paper. For further details, see [4].

uses two axes (e.g. arousal-valence or energy-stress), with some cases of a third dimension (dominance).

The advantage of dimensional models is the reduced ambiguity when compared with the categorical approach. However, some ambiguity remains, since each of the four quadrants represents more than one distinct emotion (happiness and excitation are both represented by high arousal and valence for example). Given this, dimensional models can be further divided into discrete (described above) and continuous. Continuous models, unlike discrete ones, view the emotion plane as a continuous space where each point denotes a different emotional state, thus removing the ambiguity between emotional states [5].

In order to reduce ambiguity, Thayer's mood model [6] is employed. Hence, the emotion plane is regarded as a continuous space, with two axes: arousal and valence. Each point, then, denotes a different emotional state and songs are mapped to different points in the plane.

In this paper we aim to automatically generate playlists by exploiting mood similarity between songs in the Thayer plane, based only on features extracted from the audio signal. To this end, we built on Yang's work [5], where a regression solution to music emotion recognition was proposed.

Thus, our first goal is to predict AV values for each song in the set. We employed the annotated values from the dataset created by Yang [5]. From each song, a high number of audio features are extracted, with recourse to three frameworks: PsySound, MIR Toolbox and Marsyas. The extracted features and Yang's AV annotated values are used to train Support Vector Regressors (SVR), one for arousal and another for valence. Given the high number of extracted features, the feature space dimensionality is reduced via feature selection, applying two distinct algorithms: forward feature selection (FFS) [7] and RReliefF (RRF) [8]. The highest results were achieved with a subset of features from all frameworks, selected by FFS, reaching 63% for arousal and 35.6% for valence, in terms of R^2 statistics. Results with RRF were slightly lower recurring to a smaller subset of features. Compared to the results reported in [5], the prediction accuracy increased from 58.3% to 63% for arousal, and from 28.1% to 35.6% for valence, i.e., an improvement of 4.7% and 7.5%, respectively. A classification approach, using quadrants in the Thayer plane (TP) to train and prediction instead of AV values was also tested. Still, results were very similar between different feature sets, reaching 55% accuracy in terms of quadrant matching.

Our second goal is to automatically create mood-based playlists. A playlist is "a list that specifies which songs to play in which order." [9]. The sequence of songs has three important aspects: the elements, i.e., the songs in the sequence; the order in which these elements appear; and the length of the sequence. Unlike playing random songs or listening to complete albums, many times users want to listen to music according to their mood or to some activity they are involved in (e.g., relaxing or running). In this work, we select the elements in the playlist based on their distance to a seed song according to their location in the Thayer plane (Euclidean distance is calculated). In this way, the songs in the playlist are organized in increasing distance order to the seed song. Additionally, the order of the songs can be specified with more flexibility by drawing a desired mood trajectory in the Thayer plane (see Section 4, Figure 3). As for the duration of the playlist, the number of songs to include is specified by the user.

The accuracy of this approach is measured by matching playlists generated with predicted AV values against playlists using the real AV values. With one seed song, the average accuracy of the created playlists is 62.3% for 20-song playlists, 24.8% for 5-song playlists and 6.2% for the top song only. We are not aware of any previous studies regarding the quantitative evaluation of mood-based playlists, so, to the best of our knowledge, this is an original contribution.

Finally, we have also built a working prototype to analyze music mood as well as to generate playlists based on a song or a mood trajectory (see Section 4, Figure 3).

This paper is organized as follows. In section 2, we describe relevant work that has been done in the area. In section 3, the feature extraction process and used frameworks are approached. Followed regression strategy and AV mood modeling is also addressed. In section 4, the quality of the ground truth is analyzed and experimental results are presented and discussed. Finally, conclusions from this study are drawn in section 5.

2. RELATED WORK

In 1989, Thayer proposed a two-dimensional mood model [6], offering a simple but effective way to represent mood. In this model, mood depends on two factors: Stress (happiness/anxiety) and Energy (calmness/energy) combined in a two-dimensional axis, forming four different quadrants: Contentment, representing calm and happy music; Depression, referring to calm and anxious music; Exuberance, referring to happy and energetic; and Anxiety, representing frantic and energetic music (see Figure 1). A key aspect of the model is that emotions are located away from the center, since closer to the center both arousal and valence have small values, thus not representing a clear, identifiable emotion. Thayer's mood model can fit in both sub-categories of dimensional models: it can be considered discrete, having four classes, but it can also be regarded as a continuous model, as approached by [5] and in this paper.

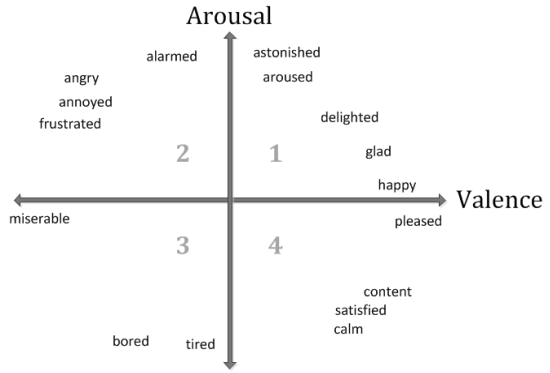

Figure 1. Thayer's model of mood (adapted from [10]).

Research on the relations between music and emotion has a long history, with initial empirical studies starting

in the 19th century [11]. This problem was studied more actively in the 20th century, when several researchers investigated the relationship between emotions and particular musical attributes such as mode, harmony, tempo, rhythm and dynamics [12]. However, only a few attempts have been made to build computational models. From these, most are devoted to emotion synthesis [13], elaborating on the relationships between emotion and music composition and music expressivity.

Only a few works addressing emotion detection in audio signals can be found. To the best of our knowledge, the first paper on mood detection in audio was published in 2003, by Feng et al. [14]. There, musical pieces are classified into 4 mood categories (happiness, sadness, anger and fear) using two musical attributes: tempo and articulation, extracted from 200 songs. These features are used to train a neural network classifier. The classifier is then validated on a test corpus of 23 pieces, with average precision and recall of 67 and 66%, respectively. This first attempt towards music mood detection had, naturally, several limitations. Namely, only two music attributes were captured and only four mood labels were employed. Regarding system validation, a reduced test corpus was utilized, making it hard to provide evidence of generality.

Most of the described limitations were still present in following research works (e.g., [13], [15], [16]). Overall, in each approach a different (and limited) set of features, mood taxonomies, number of classes and test sets are proposed. Also, some studies constrain the analysis to a particular musical style, e.g., [13], [16].

One of the most comprehensive approaches so far is proposed by Lu et al. [13]. The system is based on Thayer's model of mood, employing its 4 music moods and using features of intensity, timbre and rhythm. Mood is then detected with recourse to a hierarchical framework based on Gaussian Mixture Models and feature decorrelation via the Karhunen-Loeve Transform. The algorithm was evaluated on a test set of 800 songs, reaching 86.3% average accuracy. This value should be regarded with caution, since the system was only evaluated on a corpus of classical music using only 4 classes. Its main limitations are the absence of important mood-related features, such as mode and articulation, and its short number of mood categories. Some interesting points are the usage of a hierarchical framework and different weights for each feature, according to their.

Contrasting to most approaches, based on categorical mood models, Yang et al. [5] maps each music clip to a point in Thayer's arousal-valence plane. The authors evaluated their system with recourse to R^2 statistics, having achieved 58.3% accuracy for arousal and 28.1% for valence. We base our approach in this work.

In a recent evaluation that took place in MIREX'2010 [4], the accuracy of several algorithms in a 5-class mood classification task was compared. The best algorithm achieved 65% accuracy. For a more comprehensive survey on MER see [10]. To sum up, we can see, from the lack of accuracy and generality of the current approaches, that there is plenty of room for improvement.

Regarding automatic playlist generation (APG), most current approaches are based on the specification of one or more seed songs, creating playlists based on the distance between the seed and remaining songs, according to some distance function, e.g., [17-19]. Playlist ordering is usually defined according to the distance to the seed.

Other approaches rely on the usage of user-specified constraints based on metadata, e.g., [9], [20], [21]. Those constraints usually include criteria such as balance (e.g., don't allow two consecutive songs of the same artist) or progress (e.g., increase tempo or change genre at some point) among others [9]. Besides metadata-based constraints, audio similarity constraints can also be employed (e.g. timbre continuity through a playlist) [22].

In this paper, we follow the first approach, i.e., a seed song is specified and the playlist is created according to distances of the songs in the dataset to this seed song. Additionally, the order of the songs can also be specified by drawing a desired mood trajectory in the Thayer plane.

3. FEATURE EXTRACTION AND AV MOOD MODELING

3.1 Feature Extraction

Several authors have studied the most relevant musical attributes for mood analysis. Namely, it was found that major modes are frequently related to emotional states such as happiness or solemnity, whereas minor modes are associated with sadness or anger [23]. Simple, consonant, harmonies are usually happy, pleasant or relaxed. On the contrary, complex, dissonant, harmonies relate to emotions such as excitement, tension or sadness, as they create instability in a musical piece [23]. In a recent overview, Friberg [12] lists and describes the following features: timing, dynamics, articulation, timbre, pitch, interval, melody, harmony, tonality and rhythm. Other common features not included in that list are, for example, mode, loudness or musical form [23]. Several of these features have already been studied in the MIDI domain, e.g., [24]. The following list contains many of the relevant features for music mood analysis:

- Timing: Tempo, tempo variation, duration contrast
- Dynamics: overall level, crescendo/decrescendo, accents
- Articulation: overall (staccato/legato), variability
- Timbre: Spectral richness, onset velocity, harmonic richness
- Pitch (high/low)
- Interval (small/large)
- Melody: range (small/large), direction (up/down)
- Harmony (consonant/complex-dissonant)
- Tonality (chromatic-atonal/key-oriented)
- Rhythm (regular-smooth/firm/flowing-fluent/irregular-rough)
- Mode (major/minor)
- Loudness (high/low)
- Musical form (complexity, repetition, new ideas, disruption)

However, many of the previous features are often difficult to extract from audio signals. Also, several of them require further study from a psychological perspective. Therefore, it is common to apply low-level audio descrip-

tors (LLDs), studied in other contexts (e.g., genre classification, speech recognition), directly to mood detection. Such descriptors aim to represent attributes of audio like pitch, harmony, loudness, timbre, rhythm, tempo and so forth. LLDs are generally computed from the short-time spectra of the audio waveform, e.g., spectral shape features such as centroid, spread, skewness, kurtosis, slope, decrease, rolloff, flux, contrast or MFCCs [25]. Other methods have been studied to detect tempo and tonality.

To extract the referred features, an audio framework is normally used. The main differences between frameworks are the number and type of features available, stability, ease of use, performance and the system resources they require. In this work, features from PsySound, MIR Toolbox and Marsyas were used, measuring the relevance of each one in MER. Although PsySound is cited in some literature [5] as having several relevant features to emotion, there is no known comparison between this and other frameworks.

In his work [5], Yang used PsySound2 to extract a total of 44 features. At the time, PsySound was available only for Mac PowerPC computers. Since then, the program was rewritten in MATLAB, resulting in PsySound3. Still, the current version contains inconsistencies and lacks features present in the previous version, making it impossible to replicate Yang feature set and thus compare the results between PsySound2 and 3. For this reason, we employ the exact same PsySound2 features extracted and kindly provided by Yang. From PsySound, a set of 15 features are said to be particularly relevant to emotion analysis [26]. Therefore, another feature set was defined by Yang [5], containing these 15 features. This set is denoted as Psy15 hereafter, while the full PsySound, Marsyas and Music Information Retrieval (MIR) Toolbox feature sets will be denoted as Psy44, MAR and MIR respectively.

The MIR Toolbox is an integrated set of functions written in MATLAB, that are specific to the extraction of musical features such as pitch, timbre, tonality and others [27]. A high number of both low and high-level audio features are available.

Marsyas (Music Analysis, Retrieval and Synthesis for Audio Signals) is a framework developed for audio processing with specific emphasis on MIR applications. Marsyas has been used for a variety of projects in both academia and industry, and it is known to be computationally efficient, due in part to the fact of being written in highly optimized C++ code. On the less bright side, it lacks some features considered relevant to MER.

A brief summary of the extracted features and their respective framework is given in Table 1. Regarding Marsyas and MIR Toolbox, the analysis window size used for frame-level features is 23 ms, later transformed to song-level features by the MeanVar model [28], which represents the feature by mean and variance. All extracted features were normalized to the [0, 1] interval. A total of 12 features extracted with Marsyas returned the same (zero) value for all songs, thus not being used in the experiment.

Framework (features)	***Description***
PsySound2 (44)	Loudness, sharpness, volume, spectral centroid, timbral width, pitch multiplicity, dissonance, tonality and chord, based on psycho acoustic models.
MIR Toolbox (177)	Among others: root mean square (RMS) energy, rhythmic fluctuation, tempo, attack time and slope, zero crossing rate, rolloff, flux, high frequency energy, Mel frequency cepstral coefficients (MFCCs), roughness, spectral peaks variability (irregularity), inharmonicity, pitch, mode, harmonic change and key.
Marsyas (237)	Spectral centroid, rolloff, flux, zero cross rate, linear spectral pair, linear prediction cepstral coefficients (LPCCs), spectral flatness measure (SFM), spectral crest factor (SCF), stereo panning spectrum features, MFCCs, chroma, beat histograms and tempo.

Table 1. Frameworks used and respective features.

3.2 AV Mood Modeling

A wide range of supervised learning methods are available and have been used in MER problems before. From those, we opted for regression algorithms as a solution, similarly to what was done by Yang. The idea behind regression is to predict a real value, based on a previous set of training examples, which proved to be a fast and reliable solution [29].

Since we employ Thayer's model as a continuous representation of mood, a regression algorithm is used to train two distinct models – one for arousal and another for valence. To this end, the algorithm is fed with each song feature vector, as well as the AV values, previously annotated in Yang's study. The created models can then be used to predict AV values for a given feature vector.

Support Vector Regression (SVR) was the chosen algorithm, since it achieved the best results in Yang's study [5], when compared with Multiple Linear Regression (MLR) and AdaBoost.RT. We used the libSVM library [30], a fast and reliable implementation of SVR and classification (SVC). A grid parameter search was also carried out to discover the best SVR parameters.

To reduce the dimensionality of the feature space while increasing prediction accuracy, achieving a subset of features that are better suited to our problem, we tested two feature selection algorithms: Forward Feature Selection (FFS) [7] and RReliefF [8]. FFS is a simple algorithm, starting with an empty "ranked" set of features. All the remaining features are tested one at a time, moving the best performing one to the "ranked" set. The procedure continues iteratively, with one feature being added to the "ranked" set in each iteration, until no more features are left. One of its main limitations in FFS is the fact that it does not take into consideration the relation

that might exist between groups of features, resulting in big subsets of features. RReliefF is another algorithm to measure features' importance. Unlike FFS, RRF does not assume feature independence. In addition, it also provides a weight to each feature in the problem under analysis. Since the algorithm uses k-nearest neighbors (KNN), a proper value of K is of major importance. Using a small value may give unreliable results. On the other hand, if K is high it may fail to highlight important features. Taking this into consideration, several values of K for each feature set were tested to obtain better results. Given the differences of each feature selection algorithm, it may be interesting to compare each ranking and respective performance.

The dimensionality of the feature space can also be reduced with recourse to Principal Component Analysis (PCA) [31]. This is a widely used technique whose basic idea is to project the computed feature matrix into an orthogonal basis that best expresses the original data set. Moreover, the resulting projected data is decorrelated. As for the selection of the principal components, we kept the ones that retained 90% of the variance. Regarding implementation, we made use of the PCA MATLAB code provided in the Netlab toolbox [32].

In order to measure performance of the regression models we used the R^2 statistics, "which is the standard way for measuring the goodness of fit for regression models" [5]. Moreover, we want a direct comparison between our results and Yang's. R^2 is defined as follows, (1):

$$\boldsymbol{R^2 = 1 - \frac{SSE}{SST}}, \quad (1)$$

where SSE represents the sum square error (SSE) and SST the total sum of squares (SST). SSE measures the total deviation of the predicted values from the original annotations (2).

$$\boldsymbol{SSE = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2}, \quad (2)$$

where $\boldsymbol{y_i}$ is the annotation and $\boldsymbol{\hat{y}_i}$ the predicted value. The SST is used to measure the deviation of each annotation to the mean value of the annotations (3).

$$\boldsymbol{SST = \sum_{i=1}^{N} (y_i - \bar{y})^2}, \quad (3)$$

where $\boldsymbol{y_i}$ is the specific annotation and $\boldsymbol{\bar{y}}$ the average of all annotation values. An R^2 of 1 means the model fits the data perfectly, while negative values indicate that the model is even worse than simply taking the sample mean.

4. EVALUATION

4.1 Ground Truth Analysis

As previously mentioned, we employ the dataset and AV annotations kindly provided by Yang and used in his work [5]. The AV annotations are fundamental to the results, since they are used in the regressor training process and to measure the playlist results. According to Yang, the dataset is made of 25 seconds clips, of various genres, that better expressed the emotion present on each song, for a total of 195 songs, balanced between quadrants. The ground truth was created using 253 volunteers with different backgrounds, in a subjective test, with each song being labeled by at least 10 different subjects. The volunteers were asked to annotate the evoking emotion in AV values, between [-1, 1]. Details on the subjective test can be found in [5].

There are several problems with the ground truth that may have a negative influence on the results. One of them is the proximity of the AV values with the origin of the graph. Thayer's model places the emotions far from the center, where the reference values are relevant, with a high positive or negative valence and arousal. However, most of the annotations are near the center, as shown in Figure 2, where 70% are at a distance smaller than 0.5. In it, the position of each point represents the average AV value given by annotators, while the marker type represents the expected quadrant for each song by Yang. One possible reason for this is the fact that the AV annotations result from averaging several annotations by different subjects, which can vary greatly, once again showing the subjectivity existent in emotions perception.

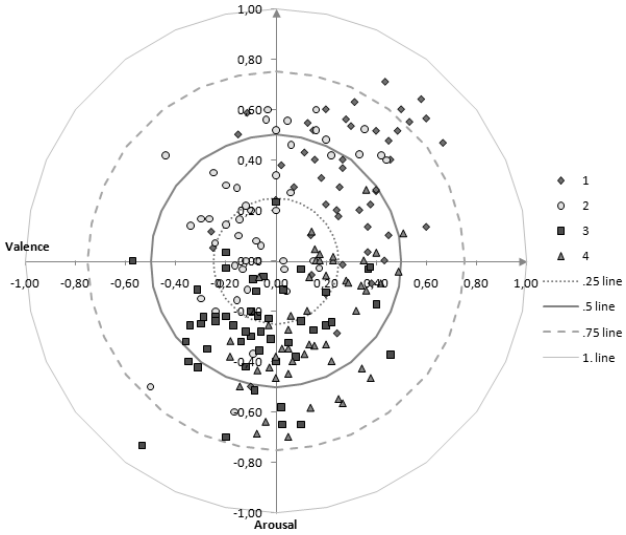

Figure 2. Yang annotations placed on Thayer's model

Another issue is the unbalanced song distribution. The original balance was disrupted since the AV values given by the subjects often placed the songs in a different quadrant than initially predicted by Yang. This affected specially the second quadrant, having only 12% of the songs.

Finally, a few inconsistencies were found between the provided data. Originally, the article [5] mentions 195 songs. However, due to some mismatches between arousal and valence annotations in the data provided by the author, we could only employ 189 songs. In any case, the difference is not significant.

4.2 Experimental Results

4.2.1 Mood Regression

In the regression tests, 20 repetitions of 10-fold cross validation were run, ensuring that all songs are used in different groups for training and testing.

Various tests were run in order to perceive the importance of each framework and its features on mood detection. From these tests, the best results were obtained with FFS, using a combination of all feature sets, reaching 63% for arousal and 35.6% for valence, using a total of 53 and 80 features respectively. The number of used features was high, in part due to the FFS working mode. Although RRF results were lower, they were in many cases obtained resorting to less features, helping us to identify the most important features for both problems (AV). For instance, using only the first ten features selected with RRF resulted in 31.5% for arousal and 15.2% for valence. On the other hand, FFS achieved only 0.8% and 2.0% for arousal and valence respectively..

The remaining tests highlighted MIR Toolbox features as achieving better results, especially on valence with R^2 attaining 25.7%. PsySound followed, with a valence accuracy of 21% and Marsyas scored the lowest, only 4.6%, proving to be quite ineffective for valence prediction. In terms of arousal, all the frameworks had a close score, ranging from 56% (Marsyas) to 60.3% (MIR Toolbox). A summary of the results is presented in Table 2 (values refer to average ± variance). For some unknown reason, we were unable to replicate Yang results [5], using either the Psy15 features or the list of features resulting from the feature selection algorithm[2]. We also conducted the same tests with PCA, normally used to reduce correlation between variables, without any noticeable improvement in results but actually leading to lower R^2 values.

	All features		***FFS***		***RReliefF***	
	A	**V**	**A**	**V**	**A**	**V**
Psy15	58.7% ± 15.6	12.7% ± 18.4	60.3% ± 14.7	21.0% ± 15.4	60.1% ± 16.0	21.1% ± 16.4
Psy44	57.3% ± 15.9	7.9% ± 14.0	57.3% ± 15.6	19.1% ± 13.4	60.5% ± 15.2	16.3% ± 15.0
MIR	58.2% ± 14.2	8.5% ± 19.5	58.7% ± 13.3	25.7% ± 18.9	62.1% ± 9.9	23.3% ± 15.7
MAR	52.9% ± 16.2	3.7% ± 14.9	56.0% ± 14.6	4.6% ± 20.2	60.0% ± 12.4	10.4% ± 10.7
ALL + PCA	56.5% ± 13.6	23.4% ± 18.2	61.8% ± 11.0	27.2% ± 22.5	61.4% ± 16.2	17.0% ± 20.6
ALL	57.4% ± 15.6	19.4% ± 12.3	62.9% ± 8.8	35.6% ± 14.7	62.6% ± 13.7	24.5% ± 14.3

Table 2. Results of the regression and classification tests.

A list of the top ten features for both arousal and valence is presented on Table 3. The list was obtained by running the RReliefF algorithm on the combined feature set of all frameworks (referred as "ALL" in Table 3).

Arousal			***Valence***		
Feature	Set	Weight	Feature	Set	Weight
SFM19 (std)	MAR	0.0186	spectral diss (S)	Psy15	0.0255
RMS energy (kurtosis)	MIR	0.0153	tonality	Psy15	0.0239
key strength minor (max)	MIR	0.0139	key strength major (max)	MIR	0.0210
MFCC2 (kurtosis)	MIR	0.0136	key clarity	MIR	0.0158
pulse clarity	MIR	0.0135	fluctuation (kurtosis)	MIR	0.0147
spectral kurtosis (skw)	MIR	0.0129	MFCC6 (skw)	MIR	0.0132
LAmin	Psy44	0.0128	fluctuation (skw)	MIR	0.0129
spectral skewness (kurtosis)	MIR	0.0126	pulse clarity	MIR	0.0118
Nmin	Psy44	0.0112	tonal centroid 1 (std)	MIR	0.0118
chroma (kurtosis)	MIR	0.0110	key strength major (std)	MIR	0.0117

Table 3. Top ten features selected by RRF (using the combined feature set from the three frameworks).

4.2.2 Playlist Generation

As mentioned before, for playlist quality evaluation we tested a regressor-based distance strategy. In this method, distances are calculated using the predicted AV values returned by the regression models. The predicted distances were compared to the reference distances resulting from the real AV annotations.

To this end, the dataset was randomly divided in two groups, balanced in terms of quadrants. The first, representing 75% of the dataset was used to train the regressor. Next, the resulting model was used to predict AV values for the remaining 25% songs[3]. From this test dataset, a song is selected and serves as the seed for automatic playlist generation. Using the seed's attributes, similarity against other songs is calculated. This originates two playlists ordered by distance to the seed, one based on the predicted and another on the annotated AV values. The annotations playlist is then used to calculate the accuracy of the predicted list, by matching the top 1, 5 and 20 songs. Here, we only count how many songs in each top are the same (e.g., for top5, a match of 60% means that the same three songs are present in both lists). The entire process is repeated 500 times, averaging the results.

Results obtained for playlist generation were very similar between the three audio frameworks. Several tests were run using all the combinations of features referred before. The similarity ranking was calculated using predicted values from the regressor. The best results were

[2] It is worth mentioned that, in order to try to replicate Yang's results, we employed the SVR parameters mentioned in his web page: http://mpac.ee.ntu.edu.tw/~yihsuan/MER/taslp08/.

[3] This 75-25 division was necessary so that the validation set was not too short, as we want to evaluate playlists containing up to 20 songs. On the other hand, the 90-10 division was employed before for the sake of comparison with Yang's results

accomplished using FFS for the combined feature set of all frameworks, with a matching percentage of 6.2% for top1, 24.8% for top5 and 62.3% for top20. Detailed results are presented in Table 4. The lower results in smaller playlists are mostly caused by the lack of precision when predicting valence. Still, best results are obtained with longer playlists, as normally used in a real scenario.

		Psy15	Psy44	MIR	MAR	ALL
Top1	All	4.2 ± 20.7	4.1 ± 18.6	3.6 ± 22.0	4.0 ± 20.7	4.2 ± 20.9
	FFS	5.6 ± 21.0	3.8 ± 18.6	5.2 ± 23.6	4.4 ± 19.8	6.2 ± 20.7
	RRF	5.1 ± 22.0	4.6 ± 19.0	5.6 ± 22.0	4.6 ± 22.6	5.2 ± 20.6
Top5	All	21.1 ± 18.1	20.9 ± 17.1	22.8 ± 19.0	18.1 ± 17.6	21.0 ± 17.8
	FFS	21.5 ± 18.3	21.2 ± 17.9	22.0 ± 19.3	19.8 ± 18.5	24.8 ± 18.3
	RRF	21.9 ± 18.1	22.1 ± 17.9	23.3 ± 18.4	18.7 ± 17.8	23.3 ± 18.4
Top20	All	61.9 ± 11.6	60.5 ± 12.3	62.7 ± 14.1	58.5 ± 13.6	60.7 ± 14.1
	FFS	62.0 ± 11.9	61.9 ± 12.4	62.5 ± 13.9	60.0 ± 13.6	62.3 ± 13.6
	RRF	61.0 ± 12.2	60.8 ± 12.8	61.7 ± 13.7	57.4 ± 13.0	61.6 ± 13.8

Table 4. Regression-based APG results (in %)

Finally, we have also built a working prototype to analyze music mood as well as to generate playlists based on a song or a mood trajectory. This is illustrated in Figure 3, where a desired mood trajectory was specified by drawing in the Thayer plane (black dots), giving rise to the playlist represented by the larger colored circles.

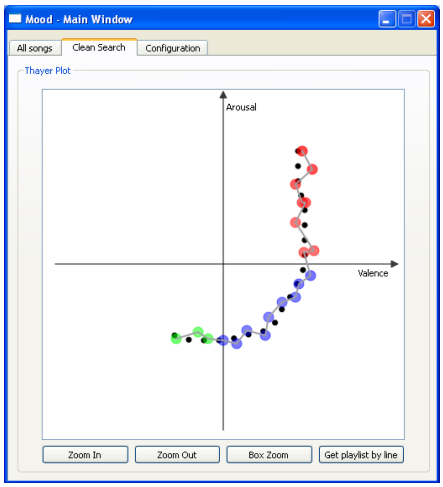

Figure 3. Automatic playlist generation prototype.

5. CONCLUSIONS

In this paper, we proposed an approach for the automatic creation of mood playlists in the Thayer plane, based on previous work by Yang [5] on continuous mood modeling.

Regarding AV prediction accuracy, we were able to outperform Yang's previous results using forward feature selection on a set of features extracted from three frameworks (PsySound, MIR Toolbox and Marsyas), reaching 63% average accuracy for arousal and 35.6% for valence, in terms of R^2 statistics. RReliefF was also important to highlight the most interesting features to the problem.

Regarding the playlist generation and similarity analysis, matching for top1 was low, averaging 5% between all frameworks, with top20 presenting some reasonable results, of around 60%. From all the tests, a slightly higher accuracy was attained using the FFS selection of features from the combination of all frameworks, with 6.2%, 24.8% and 62.3% for top1, top5 and top20 respectively. Still, the results are very similar between feature selection algorithms to classify one as better suited. The same is verified in relation to frameworks, with MIR Toolbox having a slight advantage.

In both cases, to decrease the influence that the outliers may have in the results we pretend to repeat the tests using median values instead of the current arithmetic mean. Despite the achieved improvements, we can see, from the lack of accuracy and generality of both our and other current approaches, that there is plenty of room for improvement. Also, several key open problems can be identified, namely in terms of extraction, selection and evaluation of meaningful features in the context of mood detection in audio music, extraction of knowledge from computational models (as all known approaches are black-box) and the tracking of mood variations throughout a song. In order to tackle the current limitations, we believe the most important problem to address is the development of novel acoustic features able to capture the relevant musical attributes identified in the literature, namely features better correlated to valence.

As stated in previous studies [10], the lyrical part of a song can have a great influence in the transmitted mood. The emotional response to the lyrics, obtained through natural language processing and commonsense reasoning, contributes to both the context and mood classification of the song [25].

As for playlist creation, it would be interesting to add some constraints regarding song ordering, for example, in terms of balance and progression.

Acknowledgments

This work was supported by the MOODetector project (PTDC/EIA-EIA/102185/2008), financed by the Fundação para Ciência e Tecnologia - Portugal.

6. REFERENCES

[1] T. Fritz et al., "Universal Recognition of Three Basic Emotions in Music," *Current Biology*, vol. 19, no. 7, pp. 573-6, Apr. 2009.

[2] K. Hevner, "Experimental Studies of the Elements of Expression in Music," *American Journal of Psychology*, vol. 48, no. 2, pp. 246-268, 1936.

[3] M. Zentner, D. Grandjean, and K. R. Scherer, "Emotions Evoked by the Sound of Music: Characterization, Classification and Measurement," *Emotion*, vol. 8, no. 4, pp. 494-521, Aug. 2008.

[4] J. S. Downie, "2010: MIREX2010 Results," 2010. [Online]. Available: http://www.music-ir.org/mirex/wiki/2010:MIREX2010_Results. [Accessed: 19-May-2011].

[5] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A Regression Approach to Music Emotion Recognition," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 448-457, Feb. 2008.

[6] R. E. Thayer, *The Biopsychology of Mood and Arousal*. Oxford University Press, USA, 1989, p. 256.

[7] S. L. Chiu, "Selecting input variables for fuzzy models," *Journal of Intelligent and Fuzzy Systems*, vol. 4, no. 4, pp. 243-256, 1996.

[8] M. Robnik-Šikonja and I. Kononenko, "Theoretical and Empirical Analysis of ReliefF and RReliefF," *Machine Learning*, vol. 53, no. 1-2, pp. 23-69, 2003.

[9] M. P. H. Vossen, "Local Search for Automatic Playlist Generation," M.S. thesis, Technische Universiteit Eindhoven, 2005.

[10] Y. E. Kim et al., "Music Emotion Recognition: A State of the Art Review," in *Proc. 11th Int. Society for Music Information Retrieval Conf.*, 2010, pp. 255-266.

[11] A. Gabrielsson and E. Lindström, "The Influence of Musical Structure on Emotional Expression," in *Music and Emotion*, vol. 8, Oxford University Press, 2001, pp. 223-248.

[12] A. Friberg, "Digital Audio Emotions - An Overview of Computer Analysis and Synthesis of Emotional Expression in Music," in *Proc. 11th Int. Conf. on Digital Audio Effects*, 2008, pp. 1-6.

[13] L. Lu, D. Liu, and H.-J. Zhang, "Automatic Mood Detection and Tracking of Music Audio Signals," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14, no. 1, pp. 5-18, Jan. 2006.

[14] Y. Feng, Y. Zhuang, and Y. Pan, "Popular Music Retrieval by Detecting Mood," *Proc. 26th Annu. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, vol. 2, no. 2, p. 375, 2003.

[15] D. Yang and W. Lee, "Disambiguating Music Emotion Using Software Agents," in *Proc. 5th Int. Conf. on Music Information Retrieval*, 2004, p. 52–58.

[16] D. Liu and L. Lu, "Automatic Mood Detection from Acoustic Music Data," *Int. J. on the Biology of Stress*, vol. 8, no. 6, pp. 359-377, 2003.

[17] B. Logan, "Music Recommendation from Song Sets," in *Proc. 5th Int. Conf. on Music Information Retrieval*, 2004, pp. 425-428.

[18] E. Pampalk, T. Pohle, and G. Widmer, "Dynamic Playlist Generation Based on Skipping Behavior," in *Proc. 6th Int. Conf. on Music Information Retrieval*, 2005, pp. 634-637.

[19] A. Flexer, D. Schnitzer, M. Gasser, and G. Widmer, "Playlist Generation Using Start and End Songs," in *Proc. 9th Int. Conf. of Music Information Retrieval*, 2008, pp. 173-178.

[20] J. J. Aucouturier and F. Pachet, "Scaling Up Music Playlist Generation," in *Proc. 2002 IEEE Int. Conf. Multimedia and Expo*, 2002, vol. 1, p. 105–108.

[21] S. Pauws, W. Verhaegh, and M. Vossen, "Fast Generation of Optimal Music Playlists Using Local Search," in *Proc. 6th Int. Conf. on Music Information Retrieval*, 2006, pp. 138-143.

[22] J.-J. Aucouturier and F. Pachet, "Finding Songs that Sound the Same," in *Proc. IEEE Benelux Workshop on Model-Based Processing and Coding of Audio*, 2002, pp. 91-98.

[23] C. Laurier, M. Sordo, J. Serrà, and P. Herrera, "Music Mood Representations from Social Tags," in *Proc. 10th Int. Society for Music Information Conf.*, 2009, pp. 381-386.

[24] Z. Cataltepe, Y. Tsuchihashi, and H. Katayose, "Music Genre Classification Using MIDI and Audio Features," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, pp. 275-279, 2007.

[25] O. C. Meyers, "A mood-based music classification and exploration system," M.S. thesis, Massachusetts Institute of Technology, 2007.

[26] E. Schubert, "Measurement and Time Series Analysis of Emotion in Music," *Emotion*, vol. 1, 1999.

[27] O. Lartillot and P. Toiviainen, "A Matlab Toolbox for Musical Feature Extraction from Audio," in *Proc. 10th Int. Conf. on Digital Audio Effects*, 2007, p. 237–244.

[28] A. Meng, P. Ahrendt, J. Larsen, and L. K. Hansen, "Temporal Feature Integration for Music Genre Classification," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 275-9, 2007.

[29] A. K. Sen and M. S. Srivastava, *Regression Analysis: Theory, Methods and Applications*. Springer, 1990, p. 362.

[30] C.-C. Chang and C.-J. Lin, "LIBSVM: A Library for Support Vector Machines," *Computer*. pp. 1-30, 2001.

[31] C. M. Bishop, "Neural Networks for Pattern Recognition," *Journal of the American Statistical Association*, vol. 92, no. 440, p. 1642, 1995.

[32] I. Nabney and C. Bishop, "Netlab Neural Network Software," *Pattern Recognition*. 1997.