RIADA: A Machine-Learning Based Infrastructure for Recognising the Emotions of *Spotify* Songs

P. Álvarez, J. García de Quirós, S. Baldassarri

Computer Science and Systems Engineering Department. María de Luna, 1, Ada Byron Building, Zaragoza, University of Zaragoza (Spain)

Received 28 January 2021 | Accepted 21 February 2022 | Early Access 21 April 2022

Abstract

The music emotions can help to improve the personalization of services and contents offered by music streaming providers. Many research works based on the use of machine learning techniques have addressed the problem of recognising the music emotions during the last years. Nevertheless, the results obtained are only applied on small-size music repositories and do not consider what the users feel when they listen to the songs. These issues prevent the existing proposals to be integrated into the personalization mechanisms of the online music providers. In this paper, we present the RIADA infrastructure which is composed by a set of systems able to annotate emotionally the catalog of songs offered by Spotify based on the users' perception. RIADA works with the Spotify playlist miner and data services to build emotion recognition models that can solve the open challenges previously mentioned. Machine learning algorithms, music information retrieval techniques, architectures for parallelization of applications and cloud computing have been combined to develop a complex result of engineering able to integrate the music emotions into the *Spotify*-based applications.

Keywords

Affective Annotation, Cloud Computing, Emotion Recognition, Machine Learning, Music, Spotify.

DOI: 10.9781/ijimai.2022.04.002

I. Introduction

Currently, the music streaming services are facing the chal-lenge of offering personalised media contents to their users [1]. The huge size of their music catalogs has promoted the develop-ment of innovative tools that help users to find among so many choices the songs that best suit their tastes. Most of these tools analyse the users' profiles and listening habits applying artificial intelligence techniques (such as collaborative filtering or content-based filtering), and then make personalised music recommenda-tions to the users [2]. These automatic tools are compatible with other types of content access services, for example, with services that publish the playlists created by other users or with social networks in which the users can share their listening experience. In all these solutions there are some factors that play a relevant role in the process of selecting the music, such as the musical genre and the popularity of the songs, the listening context and the activity that the user is doing, or certain cultural criteria, for instance. Nevertheless, other interesting factors have not had too much prominence among the tools offered by the streaming ser-vices, for example, the music emotions.

The relationship between music and emotions has been widely studied during the last years and the interest of including the users' emotions as a factor for the content personalization has promoted the research area commonly refereed to as Music Emotion Recognition (MER) [3]. The goal of this area is to annotate automatically the songs from an emotional point of view. These annotations usually represent the perceived or the felt emotions by the users when listening the songs, that is, the perception of emotions or the induction of emotions [4]. These two emotional dimensions are clearly different: the former is related to the emotions expressed by the music through the songs' structure and sound properties, whereas the second depends on the listener's experience and is influenced by her/his mood and context, among other factors. During the last years machine learning and deep learning techniques are being widely used to determine automatically both types of emotions in order to improve the music retrieval and recommendation systems [5].

The MER systems that work with perceived emotions are mainly based on the songs' audio. These audio files are processed by specialised tools in order to extract the acoustic characteristics of the songs, called audio features. Then, some of these features are manually selected and used to build a recognition model that acts as a classifier. The recognition function determines the emotions that the listener perceives when listening to an input song from its audio features. The resulting emotions are finally translated to affective tags that enhance the songs' attributes. Although most of these recognition approaches obtain acceptable accuracy results, some works focus on including new features that can improve the classifiers, for example, features related to the songs' lyrics [6], [7]. On the other hand, the deep learning based approaches automate the extraction of features by providing more expressive representations of the music low-level and high-level characteristics [5]. Learning algorithms (mainly, different classes of neural networks [8]–[10]) are applied on music spectrograms for determining gradually the features of interest, and then for finding the relationship between these features and the output emotional categories. These solutions require less domain knowledge than machine learning approaches, but have a higher computational cost.

* Corresponding author.

E-mail addresses: alvaper@unizar.es (P. Álvarez), jgarciaqg@unizar.es (J. García de Quirós), sandra@unizar.es (S. Baldassarri).

Please cite this article in press as:
P. Álvarez, J. García de Quirós, S. Baldassarri. RIADA: A Machine-Learning Based Infrastructure for Recognising the Emotions of *Spotify* Songs, International Journal of Interactive Multimedia and Artificial Intelligence, (2022), http://dx.doi.org/10.9781/ijimai.2022.04.002

Regardless of the learning method applied, the previous solutions present some drawbacks. Firstly, the lack of public large-size datasets that contain high-quality annotations about the songs' emotions. The reference datasets in the field of MER research are usually small (most of them have between 250 and 2, 000 songs) and have not solved the challenge of the subjective perception (the annotations are usually based on the users' feedback, which is influenced by different emotional and contextual factors that cause the quality of these annotations less than desirable) [11]. Secondly, their emotion recognition methods are usually applied on their own datasets or some of the reference datasets (mainly, the MediaEval Database for Emotional Analysis in Music [12] or the MIREX mood dataset [13]), but not on the music catalogues of the streaming services. The application to these catalogues would require to develop systems that integrate the recognition solutions with the technological infrastructure of the streaming providers. Thirdly, there is no consensus on which type of learning method is the best option, and it is even difficult to compare the existing approaches between them. Each proposal applies different feature extraction algorithms, selects different features to build the models, creates the models from different datasets and/or validates the results with different metrics and methodologies [3]. And, finally, most of the approaches determine the emotions perceived by the listeners, instead of considering the emotions that they feel. The problem of determining the emotional response of each user is complex. Nevertheless, wearable technology is demonstrating to be a good opportunity to make progress on the recognition of the listeners' feelings [14].

In this paper, we propose an infrastructure of services, called RIADA, for annotating emotionally the catalog of songs available in *Spotify*. The infrastructure interacts with the *Spotify* service platform and can be used to include the emotional dimension in the music recommendation services offered by the streaming provider. As part of the solution, we have built an automatic music emotion recognition system that classifies and annotates the songs according to the emotions perceived by the listeners. These emotions have been deduced from the playlists that the registered users publish in *Spotify*. The recognition system is based on machine learning techniques and the audio feature services available in the provider's service platform. A parallel version of the system has been programmed to be deployed and executed on cloud environments in order to be applied on large-size music datasets. The main contributions of the proposal with respect to the existing solutions are:

- it consists in a complex result of engineering able to solve a real-life problem related to the emotion recognition,
- the *Spotify* playlists have been used for deducing the emotions that the users perceive when listening to certain types of songs and for creating the dataset of annotated songs involved in the building of the recognition models,
- the emotion recognition is based on a set of multi-label classification models that work from the information published by the *Spotify* data services,
- finally, the system prototype has been successfully tested in a real cloud-based operating environment and, therefore, it has achieved a TRL-6 maturity level in the scale *Technology Readiness Level* [15].

The rest of the paper is structured as follows. Section II presents a review of the music emotion recognition systems based on machine learning techniques. It also reviews the music systems that have been programmed by integrating the *Spotify* services paying attention to those that consider the users' emotions. Section III describes the software architecture of the RIADA infrastructure. The process of building and validating the emotion recognition models is presented in detail in Sections IV and V. Section VI details the parallel and cloud-based implementation of the recognition system and shows its application to large-size music repositories. And, finally, Section VII discusses the main conclusions obtained and the future work.

II. Related Work

In this section, a review of the Music Emotion Recognition (MER) systems based on machine learning techniques and the*Spotify*-based systems that combine music and emotions are presented.

A. MER Systems Based on Audio Features and Machine Learning

There are many research works that propose automatic systems for the recognition of music emotions based on the combination of audio features and machine learning methods [3], [11]. These proposals differ from each other in terms of the method used for extracting the music features, the form of mapping those music features to emotions and, finally, the machine learning algorithms applied in the building of the recognition systems. In the following paragraphs these three issues are detailed from the perspective of the existing solutions in the field of MER research.

The first step of a typical MER system is the extraction of music features. In this review we are specially interested in those features extracted directly from the songs' audio files, called *audio features*. Several studies have analysed the relationship between certain audio features and the emotions that they produce in the listeners [16], [17]. Unfortunately, there is no consensus about which audio features are most appropriate to recognise the music emotions. Therefore, the process of feature selection is a difficult task that is usually based on researchers' experience and knowledge. This problem gets worse since there is a wide variety of processing audio tools that can be used for the feature extraction, such as *MIR toolbox* [18], *Marsyas* [19], *PsySound* [20], *OpenSmile* [21] or *JAudio* [22]. These tools apply different processing methods and, therefore, they compute different features. For this reason many works combine these toolkits for obtaining a large variety and number of features. Intuitively, we may think that it is a good option for increasing the accuracy of emotion recognition models, but some experiments have demonstrated that too many features lead to performance degradation [23].

On the other hand, it is necessary to determine and represent the emotions ascribe to the songs (the perceived or induced emotions, as was discussed in the introduction). This relationship between emotions and songs is affected by a strong subjectivity, because it depends on the listeners' character, musical preferences, genre or cultural factors, for instance. Therefore, the process of annotating manually the music emotions requires to involve many and diverse participants and, as consequence, it is time-consuming and prone to faults and impressions. With respect the representation of emotions, two different models are usually used in the field of the MER research: categorical and dimensional models [24]. The former conceptualise the emotions as a set of distinct categories (such as the *Hevner* model [25] or the *MIREX mood clusters* [13]); whereas the seconds map the emotions onto a two-dimensional space characterised by those emotions' feeling and intensity (such as the *Russell*'s affective model [26], the *Tellegen-Watson-Clark* model (TWC) [27] or the *Thayer* model [28]). Most of the MER systems use the *Russell*'s model, probably the most popular dimensional model in the development of emotion-based systems. Some proposal even work with simplifications or variations of this affective model. According to the presented in the above paragraphs, the creation of datasets that can be used for building MER models is a complex and difficult process. Most of these datasets are small in size and usually contain the songs' audio features and annotations that describe the emotions perceived by the users when listening to those songs. There is some reference datasets in the field of MER research, such as the *MIREX mood* dataset, which is the largest

one, containing about 2, 000 songs [13], the *DEAM* dataset (Database for Emotional Analysis of Music) composed by 1, 800 songs [12], or the *Allmusic* dataset composed by 900 songs. A more detailed description of the released and freely available datasets can be found in [11]. The advantage of using these datasets is that their songs are already emotionally annotated. In particular, the annotations of the *MIREX* dataset are based on their mood clusters (a categorical approach), and the annotations of *DEAM* and *Allmusic* on the *Russell*'s model (a dimensional approach). In any case, the challenge of having large-size datasets that contain the appropriate audio features and the emotional annotations with low levels of subjectivity is still open.

Finally, the different methods based on machine learning that are applied in the creation of computational models able to annotate automatically the songs' emotions are revised. We are interested in those methods that use the combination of audio features and emotion annotations. Most of these MER solutions are based on classification algorithms. Their goal is to obtain one or more emotion labels from the input song's features (single-label and multi-label classification, respectively). Recently, the multi-label classification has gained popularity because it takes in account the inaccuracy of human annotations and classifies each song into a number of different emotion categories. Different machine learning algorithms have been used for creating these classifiers, such as *Support-Vector Machines* (SVM) [29]–[32], *Random Forest* (RF) [33]–[35], *K-Nearest Neighbor* (KNN), *Decision Trees* (DT) [36], *Naïve Bayes* (NB) [33], [37], [38], *Linear Discriminant* (LD) [39] or *Gradient Boosting Machines* (GBM) [40], etc. Among these, SVM is the most used and a good option option for recognising the music emotions from the songs' audio features [33], [34], [39], [40]. This supervised method usually achieves good accuracy results with low computational power. Nevertheless, during the last years SVM has been usually combined with other classification methods in order to improve the classification results [41], [42]. In [3] a detailed review of the emotion classifiers proposed between 2003 and 2017 is presented and discussed (Table 4, pages 384-386). Regardless of the classification method used for the MER, in many cases is necessary to reduce the dimension of the feature space before building the recognition models. The choice of the appropriate features is many times more important than the machine learning method selected. *Principal Component Analysis* (PCA) [30], [33], [40], [43] and the ReliefF algorithm [32], [39] are two techniques commonly used for the feature reduction in the field of the MER research. These techniques help to create a more meaningful representation of the feature space by selecting the features of interest from the recognition point of view, and to improve the final results obtained by the emotional classifiers.

As conclusions, firstly, it is difficult to compare the results of the reviewed proposals because they work with different feature extraction tools, heterogeneous emotion annotated datasets and different classification strategies and methods. The same conclusion was reached by [44], as part of its interesting state of the art about the MER systems based on audio features. And, secondly, future MER solutions should address some drawbacks of interest, such as to avoid the necessity of having the audio of the songs for extracting their features, to have available large-scale reference datasets, or to improve the accuracy of learning-based recognition by applying a multi-method approach.

B. Music Intelligent Systems Based on Spotify

In recent years, a wide variety of intelligent systems based on the *Spotify* services have been proposed. We are especially interested in those that extract knowledge from the songs' audio features and that help users to discover songs and to create their playlists. Within this review, our focus is set on how these proposals integrate the emotional dimension into their solutions.

Spotify offers a data service for accessing the audio feature of the songs available in its music catalogue. These features have been used to predict the future success of a song [45]–[47] or to determine the influence of music on the walking practice in urban space [48], for instance. These solutions analyse the audio features that are determinant for explaining the popularity of a song or the different way of walking, respectively, and then use these features to create machine-learning models (mainly, regression models) that solve the problem. On the other hand, the *Spotify* audio features have been also used for making music recommendations [49]–[52]. These recommendation systems combine the user preferences with the features of songs that she/he usually listens to. The preferences are determined by utilizing the users' past interactions [52] or by processing the messages published by those users in social networks, such as *Twitter* [51] or *Facebook* [50]. Then, different content and collaborative filtering techniques are applied to determine the similarity between songs based on their audio features and the similarity between users based on their preferences in order to make the recommendations. The same approach is even used by *Spotify* as part of its recommendation algorithms [53]. As conclusion, despite the recent interest in using the songs' audio features to develop *Spotify*-based intelligence systems, these solutions ignore the music emotions.

Other works related to the exploitation of playlists created on *Spotify* consider the emotions. These works apply different procedures for determining the emotions of playlists, as will be presented in the following paragraphs.

In some cases, these emotions are deduced by applying natural language processing over the titles of the songs contained in the playlist [54] or over the songs' lyrics [55]. In [54] the songs' titles are concatenated to build a sentence, and then linguistic analysis techniques are used to infer the emotions that will be possibly produced in the listeners. The author concludes that the results are not as expected and only the affection of love may be detected. On the other hand, in [55], a music emotion recognition method based on the sentimental analysis of the words contained in a song's lyric is proposed. The method consists in the building of a recognition model that combines machine learning and natural language processing techniques. This model is trained using the dataset *MoodLyrics4Q* and manually applied over a reduced dataset of songs in order to validate the approach.

In other cases, the emotions of a *Spotify* playlist are recognised by processing the audio features of the songs included in it [56]. A Support Vector Machine model classifies each song of the playlist as happy, sad or angry, and then a voting strategy is used to determine the emotion of that playlist. The classifier recognises the emotions from some of the audio features offered by *Spotify* for describing their songs. Despite the similarities with our work, this proposal is a work in progress that presents some relevant weaknesses: the dataset used for building the model was manually created and consists of a small number of songs (579 songs) reducing the reliability of the classifier, only 3 different emotions are recognised, the features used in the recognition were intuitively selected and are a restricted set, and finally the results are not formally validated (a playlist is only labelled as example).

Instead of analysing the existing playlists, other works provide tools for searching *Spotify* songs applying emotional criteria and supporting the creation of new playlists. In [57], the users classify emotionally the songs based on their personal experience listening to music. Each song is manually annotated using a colour scale that represents the different vibes produced in the listener. It makes difficult the application of this solution to large-size repositories of songs. Then, an user can introduce an input colour and find songs that could produce the desired effect. As an alternative, in [58], a prototype for searching

Spotify songs according to the user's mood is presented. The emotions of the songs are not explicitly recognised, but the authors assume that certain *Spotify* audio features can be mapped directly to moods (the validation of this assumption is not discussed). The mood-based search of songs is programmed applying similarity techniques over the features of interest and integrated into a prototype of application.

III. Description of the Proposal

In this section a high-level description of the RIADA infrastructure is presented. It is composed of a set of systems that collaborate for annotating emotionally the *Spotify* songs using net-accessible data resources. The semantics of these annotations and the affective model used for representing them are two relevant issues that are discussed in advance. After that, the architecture of the proposed system is presented.

A. Music and Emotions

The goal is to build a large-size database of emotionally annotated songs. These annotations represent the emotions that a user perceives when she/he listens to a song. In this subsection, the music data source and the affective model selected for implementing the songs' annotation are briefly explained. *Spotify* is the most popular online music streaming provider with more than 35 million of songs and 100 million of subscribers. Besides, it has recently published a platform of Web services and online tools for accessing the songs' metadata, searching the registered users' playlists, browsing the listeners' habits or making simple music recommendations [59]. These data services are available for encouraging the development of novel *Spotify*-based applications. As today, the emotions that the user perceives or feels when listening to the songs have not been included in the data offered by the music provider. Nevertheless, other data available on its platform could be combined in order to integrate the emotional dimension in its products, and to solve the open challenge of annotating a large-size catalog of songs.

On the other hand, the *Russell's* affective model has been selected for representing the emotions [26]. In this model, the affective states are represented over a two-dimensional space defined by *valence* (X-axis) and *arousal* (Y-axis) dimensions. The valence represents the intrinsic pleasure/displeasure (positive/negative) of an event, object or situation, and the arousal the feeling's intensity. The combination of these two dimensions (valence/arousal) determines four different quadrants: the *happy* (positive/positive), the *angry* (negative/positive), the *sad* (negative/negative) and the *relaxed* (positive/negative) quadrant. Then, each emotion is mapped to a point in the two-dimensional space and, therefore, is also located into one of the mentioned quadrants. Alternately, the emotions can be also represented as a probability vector of four values, one per each of the Russell's quadrants. These values are the probability that the emotion represented belongs to the corresponding quadrant. For example, the "I want to hold your hand" song by "The Beatles" has the following emotional annotation [0.174, 0.765, 0.155, 0.006] which represents that is a *happy* song with a 0.765 probability (the sad, angry and relaxed probabilities are 0.174, 0.155 and 0.006, respectively).

Therefore, the proposal consists of annotating the songs considering the four quadrants of the *Russell's* affective model. The probability of that the emotions ascribe to a song belong to each of those quadrants is mainly estimated from the song's audio features. Those features can be obtained from the *Spotify* data services and, therefore, are available without the need for having the song's audio file. This last issue is very important from our proposal point of view because it will allow us to apply the solution on a large-scale, although it involves delegating the feature extraction process to Spotify.

B. Architecture of the Proposed System

The *RIADA* infrastructure presented in this section is composed of the set of software systems that are responsible for creating and updating the database of emotionally annotated songs. The infrastructure has been integrated into a *multi-tier architecture* [60], [61] in order to make easier the logical and physical decomposition in different tiers of functionality involved in the global solution.

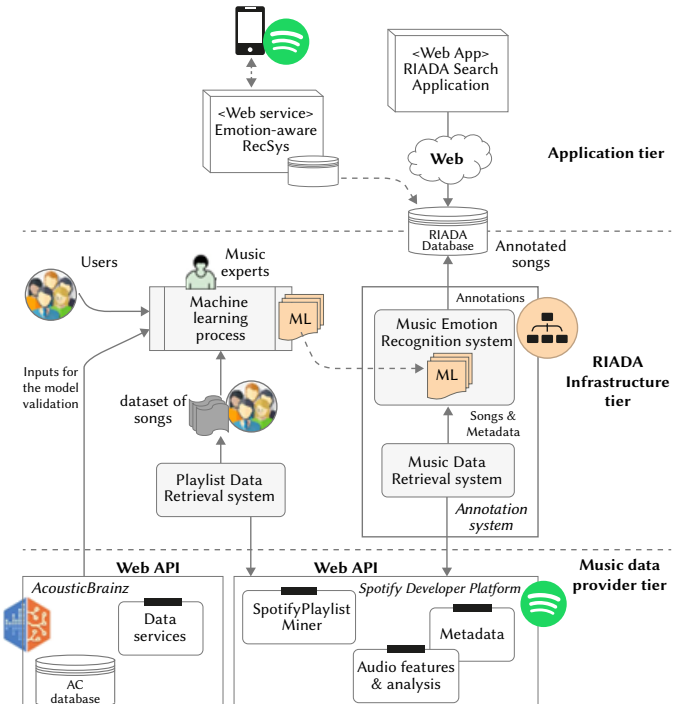


Fig. 1. High-level architecture of the solution.

As shown in Fig. 1, the solution has been divided in three tiers: the *music data provider tier*, the *RIADA infrastructure tier* and, finally, the *application tier*. The first is composed of the online services and the tools offered by the music data providers, in particular, the *Spotify* and *AcousticBrainz* [62] solutions have been integrated into this data tier. The second tier contains the systems involved in the annotations processes and the resulting database of annotated songs. These systems works with the music data providers for building the music emotion recognition models and applying these models over the *Spotify* catalog of songs. The RIADA database is the interface of this second tier from the applications point of view. These RIADA-based applications constitute the last tier of the architecture.

Following the different functional elements of the system are described in more detail. The music data tier is mainly composed of the net-accessible services integrated into the *Spotify Developer Platform*. These offer a set of Web APIs that allow to access the music database of the provider and to retrieve information about *Spotify* songs and the most popular playlists published by registered users. Additionally, the *AcousticBrainz* services have been also included in this data tier, and provide functionality for extracting the songs' acoustic characteristics and for accessing to high-level data computed from those characteristics. Some of these high-level data are related to the mood.

On the other hand, the core component of the RIADA infrastructure is the music emotion annotation system (represented in the right side of the RIADA tier). It consists of a *Music Emotion Recognition* (MER) system which integrates a set of machine-learning models for annotating emotionally *Spotify* songs. These models work with the songs' audio features and predict the emotions that the users perceive

when they listen to each of these songs. Then, these predictions are translated to emotional labels (probability vectors based on the Russell's quadrants) which are stored into the RIADA database. In the recognition process is involved the *Music Data Retrieval* (MDR) system which is responsible for interacting with the *Spotify* data services in order to get the information needed to make the emotional predictions. Note that the annotation system has been programmed implementing parallelism techniques to be applied over large-sized catalogs of songs, as will be presented in Section VI.

A fundamental component of the MER system are the machine learning models used for the emotion recognition. Before building these models, it is necessary to have a dataset of songs emotionally annotated. In the proposal, this dataset is created by the *Playlist Data Retrieval* (PDR) system. Its functionality is based on the *Spotify Playlist miner API* which aggregates the top songs from the most popular playlists created by the *Spotify*'s users. The PDR system processes the names and descriptions of these top songs and from that textual information deduces the emotions that the users can perceive when listening to them. This process is explained in detail in Section IV.

Then, a *Machine learning process* is responsible for building the recognition models from the dataset of annotated songs. This process implements a multi-model hybrid method in which a different model is created for recognising the emotions contained in each of the Russell's quadrants. A detailed description of the process will be presented in Section V. Finally, the models are integrated into the MER system in order to support the massive annotation of *Spotify* songs.

Finally, the emotionally annotated songs are stored into the *RIADA database*. The attributes and annotations of these songs are stable and do not require to be downloaded or computed again. Nevertheless, as *Spotify* is continuously adding new songs to its online catalog, it is necessary to update periodically the contents of the RIADA database. These updates are made by executing the music emotion annotation system previously presented. The system can be configured to work in update mode, and in this case it will process and annotate those songs that are not already included in the database.

IV. Creation of a dataset Based on the *Spotify* Playlists

A dataset of emotionally annotated songs has been created to be used in the building and training of the emotion recognition models. *Spotify* provides certain information about its playlists, but not about the emotions that the users perceive when they listen to those playlists. In this work, a method for deducing those emotions from the playlists that are available through the *Spotify Playlist miner* is presented.

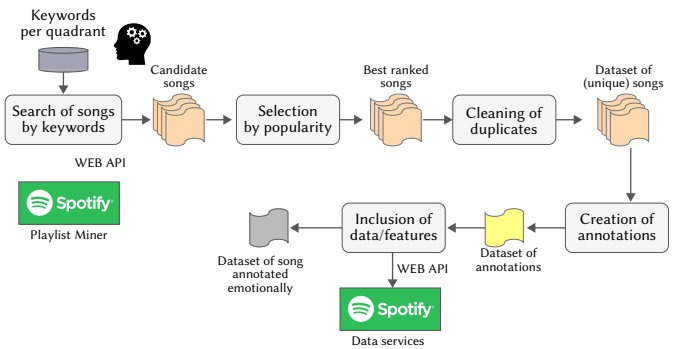


Fig. 2. Description of the data preprocessing process.

Fig. 2 shows the process followed for creating the dataset of annotated songs and the tools involved in it. Before starting the process, a set of keywords have been defined for each Russell's quadrant. These keywords correspond with emotions mapped to each particular quadrant, for example, the keywords *happy*, *joy*, *motivating* or *excited* are some of those included in the *Happy* quadrant.

The process begins executing the task *Search of songs by keywords*. This task invokes to the *Spotify Playlist miner API* which returns aggregations of songs contained in the most popular playlists published by the *Spotify*'s users. These aggregations are created from search criteria based on keywords which are matched with names and descriptions of published playlists. We have assumed that a song contained into a playlist called "Motivating music for running" is likely that conveys positive energy and emotions. Therefore, that song could be annotated as *happy*. Considering this, a set of requests are executed for each quadrant. The search criterion of a request contains a subset of the keywords defined for the quadrant of the interest and some unwanted keywords. These latter are selected from among those included in the other three quadrants. Different non-repeating combinations of keywords have been calculated for each quadrant in order to determine its set of search criteria. For example, *"Happy AND Joy AND Motivating AND NOT Sad AND NOT Relaxed"* or *"Joy AND Motivating AND NOT Angry"* are some of the criteria configured for getting songs that are probably contained in the *happy* quadrant. In the future these criteria could be improved by analysing the combinations of keywords that return the most appropriated playlists. Sentiment analysis techniques based on text could be applied for deducing the playlists' emotions from the titles or the lyrics of their songs, such as in [56] or [55], respectively. Each playlist could be emotionally characterised from the emotions obtained, and then the results contrasted with the quadrant of interest in order to evaluate the quality of the search criteria. In any case, the result of this first task is a set of candidate songs for each Russell's quadrant.

Secondly, since the songs returned by the miner have an attribute that represents their popularity, the task *Selection by popularity* processes these songs to select the most popular. The selection is achieved by applying the inverse frequency, a numerical statistic widely used in the field of information retrieval that is intended to reflect how important a song is in the returned playlists. Those songs that have an inverse frequency greater than 2.5 are discarded (this threshold have been experimentally determined). Then, the best ranked songs are filtered to remove those that appear in more than one quadrant, and therefore that could generate confusion in the creation and training of future classification models (task *Cleaning of duplicates*). Finally, the songs of each set are annotated with their respective label (in the case of the example, they will be annotated with the label *Happy*) in order to create an unique dataset of annotated songs (task *Creation of annotations*).

The last task of the process consists in completing the data of annotated songs. The general purpose attributes (such as the artist, the album, etc) and the audio features of these songs are obtained from the *Spotify* Web data services. More specifically, the list of audio features returned by *Spotify* is: loudness, energy, tempo, acousticness, valence, liveness, speechiness, instrumentalness, danceability, key, duration, and mode. The definition, unit of measurement and representation format of these features are available in [63], as part of the specification of the *Spotify AudioFeaturesObject*. All these data are recorded jointly with the emotional annotation into the dataset as result of the task *Inclusion of data and features*.

Finally, the results of the data processing stage are briefly summarised. At the beginning of the stage, we obtained 83,078 *Spotify* songs from the *Playlist Miner*. More specifically, the number of songs for each of the four requests was: 19,092 songs that probably convey emotions located into the *Happy* quadrant, 17,661 into the *Angry* quadrant, 23,931 into the *Sad* quadrant, and 22,394 into the *Relaxed* quadrant. After applying the inverse frequency, there were selected

3,055 songs for the *Happy*, 1,817 for the *Angry*, 2,943 for the *Sad*, and 1,671 for the *Relaxed* quadrants. This selection process reduces significantly the number of available songs, but increases confidence in results concerning the users' perceived emotion. Finally, the songs located into more than one quadrant were eliminated, obtaining a final dataset composed of 1,644 songs for the *Happy*, 1,307 for the *Angry*, 1,737 for the *Sad*, and 504 for the *Relaxed* quadrant. Therefore, the *prepared data* database used for the training of the models contained a total of 5,192 songs.

V. Building of *Spotify*-based Learning Models

Once presented the dataset of annotated songs, the three stages directly involved in the building of the machine learning models are following described: the analysis and extraction of the features of interest, the application and the training of algorithms, and the validation of the models. Besides, an experiment with real users has been carried out to corroborate the validity of these models before integrating them into the *Music emotion recognition system*.

A. Analysis and Extraction of Features

As will be discussed later, we have decided to build four classification models to recognise the music emotions, one per each of Russell's quadrants. Each model will predict whether or not the emotions that the users perceive when they listen to a song belong to the corresponding quadrant. The decision of considering four different hypothesis aims at creating more accurate models. Nevertheless, it is necessary to identify first the audio features that must be involved in the building of these models. We have decided to analyse these features from the perspective of each Russell's quadrant, that is, we are supposing that a feature may be significant to identify a class of emotions, but irrelevant to others. In the literature this analysis is usually carried out by applying three different approaches [3]: by selecting the same features used in other similar research works, consulting the opinion of music experts, or evaluating and interpreting certain statistical tests frequently used in the machine learning field.

In this paper, we have applied a combination of the three approaches. Firstly, statistical tests have been calculated to evaluate the degree of features' relevance in each quadrant. Then, the test results have been contrasted with the conclusions published by other similar works in the field of MER research and refined by a group of music experts in order to determine the features to be finally selected. Table I shows the result for each quadrant after calculating the tests (first step). Additionally, the audio features selected after considering the research works and the experts' conclusions have been highlighted in green color (second and third steps).

TABLE I. Analysis of Songs' Audio Features

Happy	Angry	Sad	Relaxed
valence	acousticness	energy	instrumentalness
acousticness	energy	acousticness	energy
danceability	speechiness	valence	loudness
energy	loudness	loudness	acousticness
instrumentalness	danceability	liveness	valence
loudness	liveness	duration	danceability
duration	tempo	tempo	speechiness
speechiness	instrumentalness	instrumentalness	duration
tempo	mode	key	tempo
key	duration	mode	mode
mode	valence	danceability	liveness
liveness	key	speechiness	key

In more detail, three statistical tests have been calculated, specifically, the *Chi Squared*, *ANOVA F-value* and *Mutual information* tests. These tests order the features from most to least relevant. Then, a voting strategy has been applied to combine the results of the three tests, as shown Table I for each quadrant.

Then, the features considered in other *Music Emotion Recognition* systems have been reviewed [3], [44]. Most of these systems work with features extracted from the audio of the songs. In general, they are mainly interested in extracting *timbral* and *rhythmic* features and in determining the *intensity* of the songs. Each solution uses a different audio processing tool, which makes it difficult to compare their results (it is even unknown how the features are calculated by *Spotify*). Nevertheless, these conclusions can be interpreted from the *Spotify* point of view. According to our interpretation, acousticness, instrumentalness or speechness are audio features related to the songs' timbre, tempo or danceability to the rhythm, and finally energy and valence to the intensity. Therefore, those *Spotify* audio features must be included in the final selection. For example, the tests determined that the valence and danceability features could have a low relevance for the *Angry* and *Sad* quadrants, respectively. However, after analysing the existing MER proposals we have decided to include them among the selected features.

Thirdly, an activity was organised with the participation of three music experts. The goal was to gather their opinions about the importance that the *Spotify* audio features can have in the emotion recognition. The activity had two stages. In the first each expert individually studied the information published by *Spotify* about these features (definition, units of measurement, feature extraction procedures, etc.) and listened to a collection of songs for understanding the intrinsic nature of those features. Then, the second stage consisted of a discussion group in which the experts contrasted their individual opinions and collaboratively made a list of the most relevant features. They concluded that the most significant features are: energy, valence, danceability and tempo. These conclusions are consistent with those of the existing proposals [3] and reinforce the decision to include the valence and danceability features for the case of the *Angry* and *Sad* quadrants. Besides, they believed that the features key and duration are the least relevant ones. The rest of features could have a moderate influence depending on the emotion to be recognised.

Therefore, the final proposal consists of using different audio features for building of each classification model (this type of approach was already considered by [64]). As described above, the audio features that have been finally selected for each classification model are represented in green color in Table I.

B. Model Selection and Training

In this stage, the goal is to build a machine learning model for each of Russell's quadrant. The *target function* of these models is defined as: the input are the song's audio features, while the output is a pair of values (a logical value and a real value) that predicts whether the emotions perceived by the listeners are located into the corresponding quadrant. Therefore, the emotional annotation of a *Spotify* song will consist of two vectors of four values. For example, the "I want to hold your hand" song by "The Beatles" will have the following emotional annotation ([*true*, *false*, *false*, *false*], [0.765, 0.155, 0.174, 0.006]) which represents that is a *happy* song with a 0.765 probability. The angry, sad and relaxed probabilities (0.155, 0.174 and 0.006, respectively) are lower than the classification threshold and, therefore, the song is also classified as not sad, not angry and not relaxed.

For the building of the models, three types of machine learning algorithms have been considered: *Support Vector Machine* (SVM), *K-Nearest Neighbours* (KNN) and *Random Forest* (RF). These have been widely used with good results in the recognition of emotions [65], [66]. Nevertheless, we have also considered the possibility that the use

of an unique algorithm is not the best option for building the different classification models. Therefore, the best machine learning algorithm for each quadrant (its model) is also studied.

Before comparing the algorithms, there must be defined the positive and negative datasets that will be used in the training and testing of the resulting models. The starting point is the dataset of annotated songs that was created during the preprocessing stage (described in Section IV). For each quadrant, this dataset has been been divided into two parts. On the one hand, the songs that were annotated with the emotional value of that quadrant and, on the other hand, the rest of songs. For example, for the *Happy* quadrant, the first dataset is composed by the songs annotated as *happy* (positive class), and the second by those annotated as *angry*, *sad* and *relaxed* (negative class). This partitioning strategy has been replicated for the four quadrants.

Then, the three selected machine learning algorithms have been applied in the building and training of the models. The choice of input audio features is determined by the results of the previous analysis. Besides, the range of input hyperparameters has been varied in order to find the best configuration. The library *Scikit randomized search* has been used for this evaluation since it provides an efficient procedure for the analysis of the possible permutations [67].

Table II shows the results for the different combinations of algorithms and quadrants. The best combinations have been highlighted in green color. Each of these combinations has been configured with the optimal input of audio features and hyperparameters. The models have been trained by performing a *Repeated 5-fold cross validation*. The use of this validation approach is especially important when the models are built from small-sized or unbalanced datasets, as in this case. A ratio 70/30 was applied to split the original dataset into two sets, a training set and a testing set. This ratio was experimentally chosen and it seems to be a good option for this specific classification problem. The data splitting was manually made to maintain the original percentage of songs of each quadrant in the training and testing datasets. Besides, the cross validation was configured to use the *Stratified* library of *Scikit learn* to preserve the percentage of samples for the positive and negative classes. As conclusions, *Random Forest* models offer good accuracy and F1-score results for the four quadrants. These results contradict the initial assumptions of applying different algorithms for building the model of each quadrant in order to improve the models' accuracy. The mean accuracy is 88.75%, a good result compared to the other similar studies presented in Section II.

TABLE II. Comparative of Different Models/Quadrants

Algorithm	Tests	Happy	Angry	Sad	Relaxed
SVM	accuracy	0.767	0.872	0.8036	0.929
	f1	0.752	0.821	0.783	0.733
	precision	0.7475	0.8435	0.7792	0.8624
	recall	0.7715	0.8059	0.7991	0.6801
K-NN	accuracy	0.843	0.876	0.842	0.935
	f1	0.822	0.824	0.816	0.784
	precision	0.8256	0.8516	0.8185	0.8505
	recall	0.8198	0.8055	0.8142	0.7428
Random forest	accuracy	0.844	0.899	0.862	0.945
	f1	0.820	0.860	0.839	0.801
	precision	0.8307	0.8828	0.8488	0.9299
	recall	0.8083	0.8446	0.8353	0.7392

The confusion matrices of the *Random Forest* models reaffirm the good performance of the classification models, as can be seen in Fig. 3. Nevertheless, it is also important to analyse the *false positives* in order to understand where the models fails.

Table III shows a comparison of the predictions (rows) versus the true emotions (columns) for each quadrant. The diagonal of the matrix corresponds to the true positives (highlighted in grey color), while the rest of values in each row corresponds to false positives. Firstly, the results of the models *Happy* and *Angry* have been analysed. As explained, these two affective quadrants have the same arousal (the feeling's intensity), but different valence (the intrinsic pleasure/displeasure) in the Russell affective model. The model *Happy* predicts 91 false positive of which 40 were incorrectly annotated as *angry* (48% of the total false positives), and the model *Angry* predicts 50 false positive of which 39 are songs that were annotated as *happy* (78% of the total). Therefore, these wrong predictions may be due to the valence of those songs is near zero (the zero value represents the axis that separates the two quadrants), and in those cases the models are not able to classify correctly. On the other hand, the results of analysing the models *Sad* and *Relaxed* are similar (both quadrants have the same arousal, but different valence again). In this case, the model *Sad* predicts 96 false positives of which 50 were annotated as *relaxed* (49% of the total), and the model *Relaxed* predicts 7 false positives having been all these songs annotated as *sad*. As conclusion, we suppose that the songs that are mapped to a point close to the affective quadrants' axis may be wrong classified in some cases. Nevertheless, the results of models are good being the percentage of false positives very low.

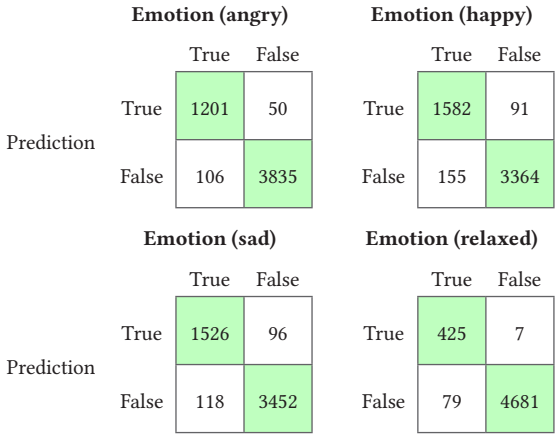


Fig. 3. Confusion matrices of the *Random Forest* models.

TABLE III. Matrix of Positive Predictions Versus True Emotions

	Happy	Angry	Sad	Relaxed
Happy	1582	40	29	22
Angry	39	1201	8	3
Sad	24	22	1536	50
Relaxed	0	0	7	425

C. Validation of the Models

The next stage is the validation of the models. From a methodological point of view, we have selected music database published by the project *AcousticBrainz* [62] for analysing the accuracy of the models built in the previous stage. This repository contains over 11 million of songs, but the version that can be downloaded is only composed by half a million (songs released before 2015). Each song has an attribute that represents the emotion conveyed by it. More specifically, this attribute is a vector of four numerical values, where each of them determines the probability of conveying an emotion belonging to a Russell quadrant. These values have been generated from users' opinions published in the music Website *Last.fm*. For that reason, these values can be especially interesting for validating the decision of creating the emotional annotations from the *Spotify* playlists (in both cases, the users' opinions and the metadata of the playlists represent the

emotional perception that the users have of the songs) and of building the recognition models using these annotations.

The downloaded dataset has been preprocessed for selecting those songs that have a high probability value in one emotion and a low probability value in the other three (in other words, a quadrant stands out from the others). After the preprocessing, the dataset size has been reduced to 60,000 songs (around 15,000 songs per quadrant in order to have a balanced sample). Then, the audio features of these songs have been obtained by invoking the *Spotify* Web data services. In this way, the features and an emotion for each song contained into the dataset are obtained. Afterwards, the goal is to validate the models using this set of *AcousticBrainz* songs.

Table IV shows the validation results. In general, the results get worse with respect to those presented in Table II: the average accuracy drops from 0,887 to 0,724, and the average f1 from 0,83 to 0,696. Nevertheless, these results were expected because two different types of annotations are "compared": the emotions deduced from the *Spotify* playlists (used for building the classification models) and the emotions extracted from users' opinions (for validating them). In any case, the most important issue is that the accuracy results are still quite good, with a mean accuracy over 72%. Besides, these results are interesting since the *Random Forest* models are particularly sensitive to changes in input data. Therefore, it is concluded that the *Random Forest* models can be a good option to recognise the emotions that the users perceive when they listen to *Spotify* songs.

TABLE IV. Results of the Model Validation

Model	Test	Happy	Angry	Sad	Relaxed
Random forest	accuracy	0.694	0.705	0.771	0.729
	f1	0.623	0.700	0.745	0.719

D. Assessment With Real Users

As a complement to the *AcousticBrainz*-based validation, an experiment with real users has been programmed to corroborate the validity of the resulting annotations. In the design of the experiment the "Pick-A-Mood" (PAM) model [68] has played a relevant role. PAM a cartoon-based pictorial instrument for representing the possible user's emotional states based on the Russell's affective model. In particular, PAM expresses eight different mood states, two for each of the four quadrants: excited and cheerful (*happy* quadrant), irritated and tense (*angry* quadrant), sad and bored (*sad* quadrant), and relaxed and calm (*relaxed* quadrant). Also, the model includes a neutral state. The added value of PAM is that its visual representation requires little time and effort of the respondents, which makes it suitable for the design of experiments in which the users must introduce their emotions.

At the beginning, a playlist composed by 12 *Spotify* songs was created, three songs of each of Russell's quadrants. These songs were selected from the dataset annotated emotionally using the *Random Forest* models, and randomly ordered in the new playlist. The experiment consisted in playing each of the songs and in asking the user what emotions she/he perceived when listening to that song. The user must listen to the entire song before responding the question since we are interested in annotating at the song level (the *Spotify* audio features used for creating the classification models are calculated processing the entire audio of songs). A *Google form* survey has been created to gather the users' responses. The survey presents a visual representation of the PAM model after playing a song and allows the user to select a maximum of two emotional states. The duration of the experiment is about 40 minutes (three and a half minutes per song, approximately).

In the experiment 25 users participated. Table V summarises the results obtained. The structure of the table is the following. It has 12 data rows, one for each song (S_1-S_{12}). Each row contains information about the emotions perceived by the users when listening to the song *Si* (these have been determined applying the recognition models built and are represented in the columns Em_{main} and $Em_{secondary}$), and about the users' responses after listening to that song (rest of columns). The column *Emmain* determines the emotion the listener is most likely to perceive and the corresponding probability value. For example, the song *S*1 ("Sorry, I'am a lady" by the duo "Baccara") was annotated as ([*true*, *false*, *false*, *false*], [0.66, 0.084, 0.014, 0.28]) which represents that is a *happy* song with a 0.66 probability (column Em_{main}). Likewise, the column $Em_{secondary}$ determines the emotional quadrant with the second highest probability value. Considering the previous example, the song *S*1 is *relaxed* with a 0.28 probability.

On the other hand, the rest of columns contains the users' responses, specifically, a column for each of the PAM states (from *Excited* to *Calm*). These columns have an integer value that represents the number of users that perceived the corresponding emotion. In green color it has been highlighted the most selected emotion, and in yellow color the second most selected. These eight columns are grouped according to the Russell quadrants, for example, the columns *Excited* and *Cheerful* correspond with the quadrant *Happy*, as is represented at the headline of the table. An extra column has been added to represent the response "Don't Know" (the column *DK*).

Following, the results obtained are briefly discussed:

- The users mostly perceived a happy emotion (*Excited* or/and *Cheerful*) when they listened to a song annotated as *happy* (songs *S*1-*S*3). The same good results are achieved when they listen to a song annotated as *angry* (songs S_4-S_6). The most of users respond that they perceive a *Tense* or/and *Irritated* emotion, the two states corresponding with the quadrant *Angry*.
- The results of the songs *sad* (songs S_7-S_9) are not as conclusive as in the two previous cases. The users mostly ascribed relaxed and/or sad emotions when listened to these songs. Although the majority of opinions correspond with these two quadrants, the responses lean towards the quadrant *Relaxed*. This fact can be due to both quadrants have the same arousal in the Russell model, but they differ in the intensity of the emotion. It could have influence in the users' responses. Besides, the high probability values of secondary emotions could have also influence in the users' opinions. For example, the songs S_7 and S_8 have high values of relaxed probability, and it could also affect to the responses. As conclusion, the results are not as satisfactory as in the previous cases, but they are not bad either.
- Finally, the high probability values of secondary emotions seems to influence the results of the songs *relaxed* (songs S_{10}-S_{12}). For example, the users mostly perceived a happy emotion when they listened to the song S_{10}. Its value of happy probability is 0.45 and, therefore, it is high value. Besides, it is important to remark that the rest of user responses concentrate on the quadrant *Relaxed* (9 users felt relaxed). The same applies to the song *S*11, but in this case the quadrants *Sad* and *Relaxed* are the most selected (the value of sad probability is also high in this case). Finally, the song S_{12} is clearly relaxed, from the users point of view. Therefore, in our opinion, the results are good and show an interesting correlation between the emotional annotations and the users' opinions. We think that we should have also included into the playlist some relaxed song in which the secondary emotion had a low probability value.

As conclusion, although the number participants and the number of songs played regarding the size of the *Spotify* catalog are low, the results obtained are very promising. And, therefore, the method of emotional labelling based on the *Spotify* playlist and the *Random Forest* models built from those annotations can be a good approach for determining the emotions that the users perceive when listen to these songs.

TABLE V. Results of Experiment With Real Users

					Happy		Angry		Sad		Relaxed		
	Em_{main}		$Em_{secondary}$		Excited	Cheerful	Tense	Irritated	Sad	Bored	Relaxed	Calm	DK
S_1	happy	0.66	relaxed	0.28	4	19	0	3	0	1	1	0	1
S_2	happy	0.73	angry	0.25	13	11	0	2	3	7	0	2	1
S_3	happy	0.68	angry	0.14	9	13	0	5	0	0	0	0	2
S_4	angry	0.61	happy	0.42	8	0	15	11	1	1	0	0	0
S_5	angry	0.62	happy	0.51	6	1	14	12	0	0	0	0	1
S_6	angry	0.59	happy	0.40	5	2	11	13	0	0	0	0	0
S_7	sad	0.66	relaxed	0.32	4	0	1	2	1	4	13	8	0
S_8	sad	0.94	relaxed	0.58	1	1	0	3	5	10	6	4	4
S_9	sad	0.56	happy	0.49	1	1	0	3	3	7	8	8	2
S_{10}	relaxed	0.61	happy	0.45	12	7	0	0	1	1	6	3	4
S_{11}	relaxed	0.66	sad	0.50	1	1	1	0	2	9	6	12	0
S_{12}	relaxed	0.59	happy	0.48	5	5	0	0	1	4	7	11	1

VI. An Automatic System for Annotating Emotionally Songs

In this section the design of the two systems involved in the annotation of songs is presented in detail: the *Music Data Retrieval* (MDR) system and the *Music Emotion Recognition* (MER) system. The goal is that these systems work automatically and are able to process efficiently a large number of songs by using the classification models previously created.

A. Description of the Annotation Process

Fig. 4 shows the stages and the data involved in the process proposed for annotating emotionally the *Spotify* songs. The green stages represent interactions with the *Spotify* data services; whereas the red stage represents the recognition actions executed by the MER system. The input is a database of artists which was previously created applying mining techniques over the data services offered by the music provider. The output is the RIADA database.

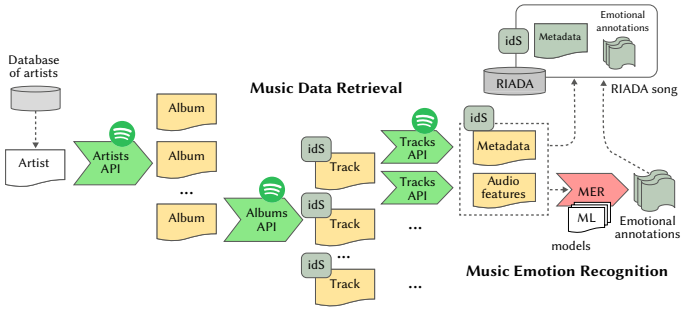


Fig. 4. Stages and data involved in the annotation process.

The MDR system is responsible for executing the first part of the process. It consists of a sequence of invocations to the *Spotify* data endpoints. Firstly, the *Artists endpoint* is invoked for getting the list of albums published by each artist. Then, each album is individually processed. A request to the *Albums endpoint* is executed for getting information about all the songs (or tracks in *Spotify* terminology) contained into that album. Each track contains a unique *Spotify ID* that will be reused to identify the song in the RIADA database. This decision facilitates the integration of the *Spotify* tools in the RIADA-based future applications. Optionally, the metadata of each song can be also obtained invoking the *Tracks endpoint*. An independent request is executed for each song of the album. Some metadata of the songs can be finally stored into the RIADA database (in grey color), if their are available, for example, the song's author, album, title, musical genre, or year of publication.

Subsequently, the MER system is in charge of annotating emotionally these songs, as shown in the right side of Fig. 4. Before, it must obtain the audio features of the songs invoking again the *Tracks endpoint* (a request for each song). Then, the MER processes each song's features and applies the four *Random Forest* models to compute the emotions that the users will perceive when listen to that song (specifically, the probability vector based on Russell's quadrants that represents those emotions). Finally, the MER creates a *RIADA song* structure which contains the song's *Spotify ID*, the emotional annotations and the metadata obtained during the retrieval phase.

Obviously, the data retrieval is a time consuming task due to it involves a large number of invocations to the endpoints and requires to process a large number of response files (in JSON format) for extracting the information of interest. These invocations are independent of each other, making possible to apply parallelism techniques to improve the efficiency of the systems involved. On the other hand, the emotion recognition also consists of a large-size bag of independent tasks (the execution time of each task is relatively small), and therefore it also requires high computing capacity for achieving an efficient processing.

B. Architectural Design of the System

The two systems involved in the annotation process have been designed according to the *master-worker architecture* [69]. It is a high-level design pattern that facilitates the parallel execution of applications composed by a set of independent tasks. The pattern consists of two class of processes: a master and a pool of workers. The former is responsible of assigning tasks to workers and guaranteeing that all of them are correctly completed; whereas the workers simply execute the assigned tasks. This architectural model is highly scalable by increasing (or decreasing) the size of pool of workers according to the execution requirements.

The master-worker architecture requires an asynchronous communication mechanism that makes possible the uncoupled

interactions between the processes involved. *Message brokers* have been usually used for this purpose, demonstrating their adaptability and effectiveness in this model of architectural solutions.

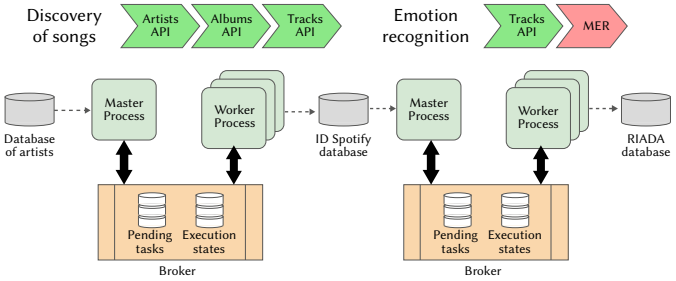


Fig. 5. Components and connectors of the architecture.

Fig. 5 shows the concrete design of the solution. It consists of two master-worker systems, one for implementing the MDR system and another for the MER system, connected between them by a shared database that contains the songs available in the music provider. Each master-worker system coordinates their processes through a broker based on message queues. Two queues have been internally declared: the *pending tasks queue*, in which the master publishes the tasks to be executed by some of the workers, and the *execution states queue*, in which the workers report to the master about the final state of executing each of their tasks (this state also includes the performance metrics concerning the execution of the task). On the other hand, both master-worker systems create their tasks from the data available into their input databases, and store the results computed by the workers into an output database. The granularity of the tasks depends on the restriction imposed by *Spotify* on the use of its services.

The process of getting the metadata of the songs published by 50 artists is an independent task in the MDR system. The master accesses to the database of artists, creates tasks that contains the identifiers of the artists to be processed (in blocks of 50), and then publishes these tasks into the broker. The workers execute the pending tasks when they are available, store the songs discovered into the output database, and finally notify the execution state of the task. These states are then used by the master for applying fault recovery strategies based on retrying the failed tasks and for generating reports of execution. On the other hand, in the MER system a task consists in annotating emotionally 50 songs, being the behavior of the system similar to that described above. In this case, the workers are responsible for getting the songs' audio features and for determining the emotional annotations applying the *Random Forest* models.

C. Cloud-based Deployment and Performance Analysis

A generic master-worker architecture has been programmed using the Python programming language. Besides, it integrates a *RabbitMQ server* as message broker in order to the processes can be executed and deployed in distributed computing environments, such as in a cloud infrastructure, for instance.

Fig. 6 shows the system configured for annotating emotionally the songs available in *Spotify*. The processes are executed on virtual machines of the *OVH cloud* (https://www.ovh.com/). Each master is running in a dedicated virtual machine in which it has been also deployed its input database. These databases have been designed and managed using *MongoDB* technology. The workers are running on a pool of machines so that these instances' computing resources are always busy. The message server has been installed as a service in the *CloudAMQP* (https://www.cloudamqp.com/), and therefore it is also deployed over cloud-based resources. Finally, the RIADA database in which the final results are stored has been installed in *mLab*, a cloud database service that hosts *MongoDB* repositories (https://mlab.com/). Therefore, the technological solution has been deployed and executed in a real environment. According to the *Technology Readiness Levels* scale (TRL, [15]) this solution has achieved a TRL-6 level, being a system prototype that may evolve into a final product.

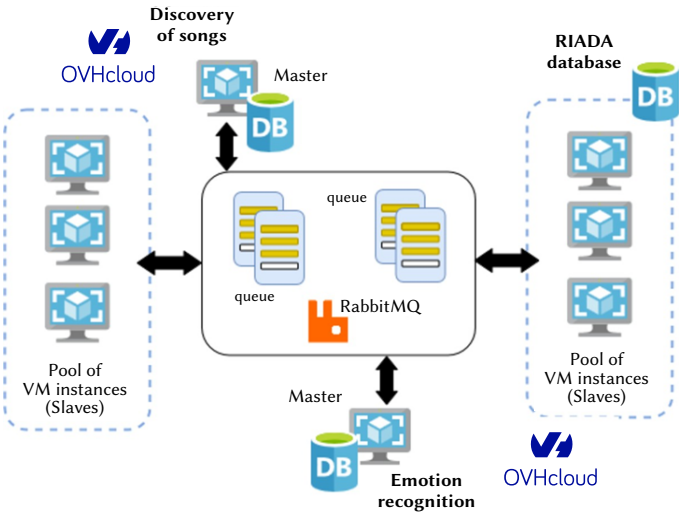


Fig. 6. Deployment over the OVH cloud resources.

A set of experiments have been also programmed for demonstrating the flexibility of the system to adapt to different resource provisioning scenarios and for analysing the scalability of the solution. Each experiment has consisted in annotating 750, 000 *Spotify* songs. Different computing instances have been hired for the execution of the master process, and different sizes of pools configured for the workers. These instances have been selected from those available in the *OVH cloud* on the basis of the authors' experience. In the future the selection criterion could be based on optimization techniques able to reduce the execution costs of the provisioning and to maximise the system performance. The use of these techniques would imply a detailed evaluation of the behavior of the deployed system, which it is out of the scope of this paper.

Table VI shows the results of the experiments. The table is structured as follows. The first column defines the type of OVH virtual machine (VM) hired to execute the master. The second column determine the number of the VM instances that compose the pool in which the workers are being executed. In all the cases, the pool is composed by *b2-7* computing instances, a general purpose virtual machine provided by OVH (2 cores at 2 GHz with 7 GB of RAM and a SSD storage of 50 GB). The MDR system executes a worker in each instance of its pool (these workers are continuously invoking to the *Spotify* data services -more than 60, 000 requests per experiment-, and the streaming provider generates response delay when two or more processes invoke it from the same machine), and the MER system executes two workers per instance (the number of interactions with *Spotify* is less, around 25, 000 requests). The third column is the total execution time needed for annotating all the songs. It is the sum of the times required to complete the execution of the MDR system and the MER system. The execution times of both systems are broken down in the fifth and sixth columns (these represent the CPU time considering all the cores involved and the real time needed to complete the execution, respectively). Finally, the last column is the mean execution time to complete a task in each of the parallel systems.

The first row of Table VI presents the results of executing sequentially the annotation system (the MDR system and the MER system are only composed by a worker). The total execution time is

TABLE VI. PERFORMANCE RESULTS OF THE DIFFERENT CLOUD-BASED EXPERIMENTS

Master VM	Number of VM instances	Total time (hh:mm:ss)	System	CPU time (hh:mm:ss)	User time (hh:mm:ss)	Mean time per task (in seconds)
b2-7	1	11:31:35	MDR	1:19:15	1:19:15	47.55
			MER	10:12:20	10:12:20	2.51
b2-7	2	3:01:51	MDR	1:14:23	0:38:12	44.63
			MER	9:34:30	2:23:39	2.35
b2-7	3	2:00:24	MDR	1:21:35	0:28:09	48.95
			MER	9:13:08	1:32:15	2.25
b2-7	4	1:37:56	MDR	1:14:09	0:15:44	44.49
			MER	10:54:54	1:22:12	2.68
b2-7	5	1:18:58	MDR	1:08:19	0:12:34	44.49
			MER	11:01:09	1:06:24	2.70
r2-15	5	1:26:02	MDR	1:36:52	0:20:46	58.12
			MER	10:49:43	1:05:16	2.65
c2-7	5	1:17:25	MDR	1:35:01	0:19:27	57.01
			MER	9:36:59	0:57:58	2.42

more than 11 hours. Then, different experiments increasing the number of virtual machines are executed, from 2 instances to 5 instances (rows 2-5, respectively). The speedup obtained (considering this metric as the ratio between the sequential execution time and the parallel execution time of each experiment) is near to the number of workers that are being executing: 3.7X in the case of 2 instances and 4 workers, 5.7X in the case of 3 instances and 6 workers, 7.6X in the case of 4 instances and 8 workers, and finally 8.8X in the case of 5 instances and 10 workers. This behavior is a good result from the parallelization point of view. On the other hand, we have also evaluated the possibility of executing the master in other type of virtual machine, for example, in an instance with optimised CPU/RAM ratios and accelerated IOPS (specifically, a *r2-15* instance, with 2 cores with 5 GB of RAM, a SSD storage of 50 GB and a public network connection of 250 Mbps guaranteed), or in an instance for processing parallel workloads (a *c2-7* instance, with 2 cores at 3 GHz with 7 GB of RAM, a SSD storage of 50 GB and a public network connection of 250 Mbps guaranteed)). The results are shown in the two last rows of Table VI. The execution times are similar to those obtained in the experiment in which the master is executing in a *b2-7* instance (a pool of 5 instances), but a small improvement is observed in the MER execution time when a *c2-7* instance is hired.

D. A Prototype of RIADA-based Application

After executing the cloud-based system, the RIADA database contains the emotional annotations of 10 million of *Spotify* songs. As discussed in Section III, this database can be reused for developing different emotion-based applications. A Web application for searching songs applying emotional criteria has been developed as an example of RIADA-based application. The application also allows to filter the results according to the songs' musical genre or popularity, and to play a fragment of the songs found (30 seconds) through the *Spotify* music streaming service. Fig. 7 shows the interface of this application which is available in https://riada.djrunning.es/. It is hosted on *OVH hosting service* and its back-end is running on an OVH virtual private server. This back-end works directly with the RIADA database deployed in *mLab*.

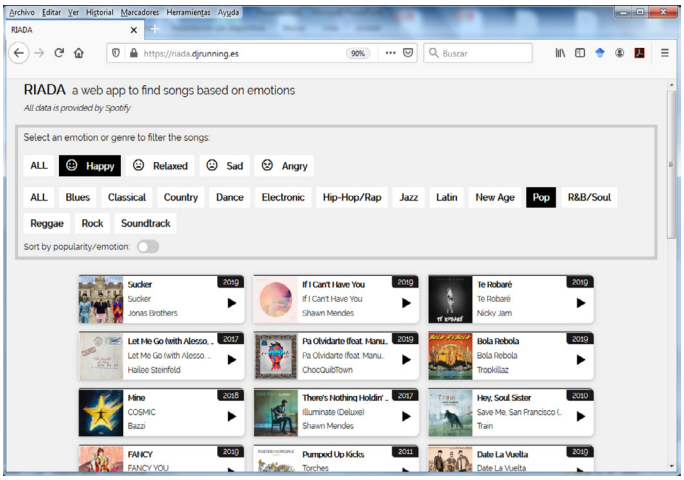


Fig. 7. Web interface of the search application based on RIADA.

VII. CONCLUSIONS AND FUTURE WORK

The paper presents the systems involved into the RIADA infrastructure. These systems collaborate among them to annotate emotionally the *Spotify* catalog of songs. The processes of building the required machine learning models and of using those models to recognise the music emotions are based on the playlist and data services provided by *Spotify*. The integration of these services allows to apply the solution to a large-size catalog of songs, and it is an alternative to the usual approach based on the processing of songs' audio files. Besides, a parallel implementation of the RIADA systems has been proposed in order to improve the efficiency of the annotation processes. It is based on the master-worker architecture and has been deployed in different cloud-based environments.

On the other hand, the playlists published by the *Spotify* registered users play a relevant role in the solution. These playlists have been used to extract knowledge about what the users emotionally perceive when listen to a song, and then this knowledge has been applied in the building of the emotion recognition models. The proposal is innovative and it considers explicitly the user point of view. The resulting recognition models have been validated by using the *AcousticBrainz* dataset and by involving real users, obtaining good results in both cases. Moreover, the validation based on *AcousticBrainz* is interesting because it demonstrates that the models are only applied on the *Spotify* songs, but they can be applied successfully on other music repositories.

Although it has not been included in the paper, other alternatives to our recognition approach have been studied, for example, the possibility of building only one model able to solve a multi-class classification problem. In that case, the *target function* of this multi-class model was defined as: the input are the song's audio features,

while the output is a vector of four logical values ([*is_happy*, *is_angry*, *is_sad*, *is_relaxed*]) that determine in which Russell's quadrants could be located the emotions perceived by the listeners. We have built various models applying different machine learning algorithms and using the same dataset of songs. The *Random Forest* models are again the best option, obtaining an accuracy and f1 of 0.78 and 0.75, respectively. Therefore, the results are slightly worse than our proposal. In our opinion, the good results of our approach are due to: the splitting of the classification problem into four subproblems simplifying the classification constraints to be considered, and the adaptation of the building model stages (the selection of features and algorithms, and the training of models) to the characteristics and particularities of each quadrant.

Finally, some of the challenges that could be addressed in the future are briefly outlined:

- despite the good results obtained, to validate experimentally that the hypothesis formulated for annotating songs from playlists are really suitable in order to obtain an accurate dataset
- to publish the dataset (or a part of the dataset) so that it can be reused by other MER researchers (the *Spotify* terms of service and developer policies are being studied in order to find a viable option for its publication)
- to include the songs' lyrics in the emotion recognition in order to propose a multi-modal approach
- to explore the possibility of building *Spotify*-based accurate models able to recognise the emotions of each song's segments
- to build alternative recognition models based on fuzzy logic and to compare them with the models presented
- to analyse the execution behavior of the cloud-based system in order to optimise its configuration and to reduce the costs of its resource provisioning
- to create an emotion-aware music recommendation system based on the RIADA functionality and the content personalization and recommendation services provided by *Spotify*
- to reuse the RIADA technology for the generation of affective playlist. It is an open and interesting challenge in the field of the affective computing
- and, finally, to use wearable devices to detect the emotions induced to the listeners through the music. These devices could be used to include a new emotional dimension into the dataset or to study the correlation between the perceived emotions (the songs' annotations) and the induced emotions

Acknowledgment

This work has been supported by the TIN2017-84796-C2-2-R and RTI2018-096986-B-C31 projects, granted by the Spanish Ministerio de Economía y Competitividad, and the DisCo-T21-20R and Affective-Lab-T60-20R projects, granted by the Aragonese Government.

References

[1] G. Knox, H. Datta, "Streaming services and the homogenization of music consumption," 2020. [Online]. Available: https://research.tilburguniversity.edu/en/publications/streaming-services-and-the-homogenization-of-music-consumptionf, [Online; accessed 19-July-2020].

[2] M. Schedl, H. Zamani, C.-W. Chen, Y. Deldjoo, M. Elahi, "Current challenges and visions in music recommender systems research," *International Journal of Multimedia Information Retrieval*, vol. 7, pp. 95–116, 03 2018, doi: 10.1007/s13735-018-0154-2.

[3] X. Yang, Y. Dong, J. Li, "Review of data features-based music emotion recognition methods," *Multimedia Systems*, vol. 24, pp. 365–389, July 2018, doi: 10.1007/s00530-017-0559-4.

[4] A. Pannese, M.-A. Rappaz, D. Grandjean, "Metaphor and music emotion: Ancient views and future directions," *Consciousness and Cognition*, vol. 44, pp. 61–71, 2016, doi: https://doi.org/10.1016/j.concog.2016.06.015.

[5] J. Nam, K. Choi, J. Lee, S. Chou, Y. Yang, "Deep learning for audio-based music classification and tagging: Teaching computers to distinguish rock from bach," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 41–51, 2019, doi: 10.1109/MSP.2018.2874383.

[6] C. Gökalp, "Music emotion recognition: a multimodal machine learning approach," Master's thesis, School of Management, Sabanci University, 2019.

[7] G. Liu, Z. Tan, "Research on multi-modal music emotion classification based on audio and lyirc," in *2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, vol. 1, 2020, pp. 2331–2335.

[8] Y. Dong, X. Yang, X. Zhao, J. Li, "Bidirectional convolutional recurrent sparse network (bcrsn): An efficient model for music emotion recognition," *IEEE Transactions on Multimedia*, vol. 21, no. 12, pp. 3150–3163, 2019, doi: 10.1109/TMM.2019.2918739.

[9] M. Russo, L. Kraljević, M. Stella, M. Sikora, "Cochleogram-based approach for detecting perceived emotions in music," *Information Processing & Management*, vol. 57, Sept. 2020, doi: 10.1016/j.ipm.2020.102270.

[10] R. Sarkar, S. Choudhury, S. Dutta, A. Roy, S. K. Saha, "Recognition of emotion in music based on deep convolutional neural network," *Multimedia Tools and Applications*, vol. 79, pp. 765––783, 2020, doi: 10.1007/s11042-019-08192-x.

[11] S. Zhao, S. Wang, M. Soleymani, D. Joshi, Q. Ji, "Affective computing for large-scale heterogeneous multimedia data: A survey," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 15, no. 3s, pp. 1–32, 2019, doi: 10.1145/3363560.

[12] M. Soleymani, A. Aljanaki, Y. Yang, "DEAM: Mediaeval database for emotional analysis in music." http://cvml.unige.ch/databases/DEAM/, 2016. [Online; accessed 19-July-2020].

[13] X. Hu, J. Downie, C. Laurier, M. Bay, A. Ehmann, "The 2007 mirex audio mood classification task: Lessons learned," 01 2008, pp. 462–467.

[14] B.-J. Han, S. Rho, S. Jun, E. Hwang, "Music emotion classification and context-based music recommendation," *Multimedia Tools and Applications*, vol. 47, no. 3, pp. 433–460, 2010.

[15] E. Commission, "Horizon 2020 work programme 2014 – 2015. european commission decision c(2015)8621," 2015. [Online]. Available: https://ec.europa.eu/research/participants/data/ref/h2020/wp/2014_2015/annexes/h2020-wp1415-annex-ga_en.pdf, [Online; accessed 25-July-2021].

[16] A. Gabrielsson, E. Lindstrom, *The influence of musical structure on emotional expression*, pp. 223–248. Oxford University Press, 2001.

[17] R. E. Thayer, R. J. McNally, "The biopsychology of mood and arousal," *Cognitive and Behavioral Neurology*, vol. 5, no. 1, p. 65, 1992.

[18] O. Lartillot, P. Toiviainen, "A matbal toolbox for musical feature extraction from audio," in *Proceedings of the 10th International Conference on Digital Audio Effects, DAFx-07*, Bordeaux, France, 2007, pp. 1–8.

[19] G. Tzanetakis, "Marsyas-0.2: A case study in implementing music information retrieval systems," *Intelligent Music Information Systems: Tools and Methodologies*, pp. 1–48, 2007, doi: 10.4018/978-1-59904-663-1.ch002.

[20] D. Cabrera, "Psysound: A computer program for psychoacoustical analysis," in *Proceedings of the Australian Acoustical Society Conference*, 1999, pp. 47–54.

[21] OpenSMILE, "OpenSMILE audio feature extraction." https://www.audeering.com/opensmile/, 2020. [Online; accessed 19-July-2020].

[22] D. McEnnis, C. McKay, I. Fujinaga, P. Depalle, "jaudio: An feature extraction library.," in *Proceedings of the 6th International Conference on Music Information Retrieval, ISMIR 2005*, London, UK, 01 2005, pp. 600–603.

[23] J. L. Zhang, X. L. Huang, L. F. Yang, Y. Xu, S. T. Sun, "Feature selection and feature learning in arousal dimension of music emotion by using shrinkage methods," *Multimedia systems*, vol. 23, no. 2, pp. 251–264, 2017, doi: 10.1007/s00530-015-0489-y.

[24] P. Zachar, R. Ellis, *Categorical versus dimensional models of affect: A seminar on the theories of Panksepp and Russell*. John Benjamins Publishing Company, 2012.

[25] E. Schubert, "Update of the hevner adjective checklist," *Perceptual and motor skills*, vol. 96, no. 3, pp. 1117–1122, 2003, doi: 10.2466/pms.2003.96.3c.1117.

[26] J. Russell, "A circumplex model of affect," *Journal of personality and social psychology*, vol. 39, no. 6, pp. 1161–1178, 1980.

[27] A. Tellegen, D. Watson, L. A. Clark, "On the dimensional and hierarchical structure of affect," *Psychological Science*, vol. 10, no. 4, pp. 297–303, 1999, doi: 10.1111/1467-9280.00157.

[28] R. E. Thayer, "Toward a psychological theory of multidimensional activation (arousal)," *Motivation and Emotion*, vol. 2, no. 1, pp. 1–34, 1978, doi: 10.1007/BF00992729.

[29] Y.-H. Chin, C.-H. Lin, E. Siahaan, I.-C. Wang, J.-C. Wang, "Music emotion classification using double-layer support vector machines," in *Proceedings of the 1st International Conference on Orange Technologies (ICOT 2013)*, 2013, pp. 193–196.

[30] J. Deng, *Emotion-based music retrieval and recommendation*. PhD dissertation, Hong Kong Baptist University, 2014.

[31] N. Nalini, S. Palanivel, "Music emotion recognition: The combined evidence of mfcc and residual phase," *Egyptian Informatics Journal*, vol. 17, no. 1, pp. 1–10, 2016, doi: https://doi.org/10.1016/j.eij.2015.05.004.

[32] R. Panda, R. Malheiro, R. P. Paiva, "Novel audio features for music emotion recognition," *IEEE Transactions on Affective Computing*, vol. Early access, 2018, doi: 10.1109/TAFFC.2018.2820691.

[33] P. F. Vale, "The role of artist and genre on music emotion recognition," Master's thesis, Information Management School, 2017.

[34] Y. Ospitia-Medina, J. R. Beltrán, S. Baldassarri, "Emotional classification of music using neural networks with the mediaeval dataset," *Personal and Ubiquitous Computing*, vol. April (online), pp. 1–13, 04 2020, doi: 10.1007/s00779-020-01393-4.

[35] M. Rumiantcev, O. Khriyenko, "Emotion based music recommendation system," in *Proceedings of the 26th Conference of Open Innovations Association FRUCT*, Yaroslavl, Russia, 2020, pp. 639–645.

[36] M.-C. Chiu, L.-W. Ko, "Develop a personalized intelligent music selection system based on heart rate variability and machine learning," *Multimedia Tools and Applications*, vol. 76, pp. 15607––15639, 09 2016, doi: 10.1007/s11042-016-3860-x.

[37] K.-A. Bodarwé, J. Noack, P. Jean-Jacques, "Emotion-based music recommendation using supervised learning," in *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia*, New York, NY, USA, 2015, pp. 341–344, Association for Computing Machinery.

[38] F. Paolizzo, N. Pichierri, D. Casali, D. Giardino, M. Matta, G. Costantini, "Multilabel automated recognition of emotions induced through music," *CoRR*, vol. abs/1905.12629, 2019.

[39] J. H. Juthi, A. Gomes, T. Bhuiyan, I. Mahmud, "Music emotion recognition with the extraction of audio features using machine learning approaches," in *Lecture Notes in Electrical Engineering. Proceedings of ICETIT 2019, Emerging Trends in Information Technology*, vol. 605, 2020, pp. 318–329, Springer International Publishing.

[40] K. W. Cheuk, Y.-J. Luo, B. B, T, G. Roig, D. Herremans, "Regression-based music emotion prediction using triplet neural networks," in *Proceedings of the International Joint Conference on Neural Network, IJCNN*, Glasgow, 07 2020, IEEE.

[41] A. Ma, I. Sethi, N. Patel, "Multimedia content tagging using multilabel decision tree," in *Proceedings of the 11th IEEE International Symposium on Multimedia*, 2009, pp. 606–611.

[42] S. Das, S. Debbarma, B. Bhattacharyya, "Building a computational model for mood classification of music by integrating an asymptotic approach with the machine learning techniques," *Journal of Ambient Intelligence and Humanized Computing*, vol. May (online), pp. 1–13, 05 2020, doi: 10.1007/s12652-020-02145-1.

[43] R. Panda, R. P. Paiva, "Music emotion classification: Dataset acquisition and comparative analysis," in *15th International Conference on Digital Audio Effects, DAFx-12*, 10 2012, pp. 1–7.

[44] R. Panda, *Emotion-based Analysis and Classification of Audio Music*. PhD dissertation, Universidade de Coimbra, 2019.

[45] E. Georgieva, M. Suta, N. Burton, "Hitpredict: Predicting hit songs using spotify data," 2018. [Online; accessed 19-July-2020].

[46] M. Sciandra, I. Spera, "A model based approach to spotify data analysis: A beta GLMM," *SSRN Electronic Journal*, vol. 3, pp. 1–18, 01 2020, doi: 10.2139/ssrn.3557124.

[47] J. H. Oh, S. Ouwejan, S. T. Kim, I. Ng, "Music intelligence: Granular data and prediction of top ten hit songs," *SSRN Electronic Journal*, pp. 1–12, 05 2020, doi: 10.2139/ssrn.3585176.

[48] R. Oi, "Spotify on the streets: walking and listening to music in urban spaces," Master's thesis, Lundt University, 2019. http://lup.lub.lu.se/student-papers/ record/8976269.

[49] M. Dittenbach, R. Neumayer, A. Rauber, "Playsom: An alternative approach to track selection and playlist generation in large music collections," in *Proceedings of the Workshop of the EU Network of Excellence DELOS on Audio-Visual Content and Information Visualization in Digital Libraries (AVIVDiLib 2005)*, 2005, pp. 226–235.

[50] A. Germain, J. Chakareski, "Spotify me: Facebook-assisted automatic playlist generation," in *IEEE 15th International Workshop on Multimedia Signal Processing (MMSP 2013)*, Sep. 2013, pp. 25–28.

[51] M. Pichl, E. Zangerle, G. Specht, "Combining spotify and twitter data for generating a recent and public dataset for music recommendation," in *Proceedings of the 26th GI-Workshop Grundlagen von Datenbanken (GvDB 2014)*, Ritten, Italy, 2015, pp. 35–40.

[52] F. Fessahaye, L. Pérez, T. Zhan, R. Zhang, C. Fossier, R. Markarian, C. Chiu, J. Zhan, L. Gewali, P. Oh, "Trecsys: A novel music recommendation system using deep learning," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 1–6.

[53] M. Madathil, "Music recommendation system spotify - collaborative filtering," 2017. Reports in Computer Music. Aachen University, Germany.

[54] N. F. R. Fauzia, "The use of song titles in spotify playlists to express the affection," in *International Seminar on Sociolinguistics and Dialectology: "Changes and Development of Language in Social Life" 2017*, 2017, pp. 185–189.

[55] S. Giammusso, M. Guerriero, P. Lisena, E. Palumbo, R. Troncy, "Predicting the emotion of playlist using track lyrics," in *19th International Society for Music Information Retrieval Conference*, Paris, France, 2018.

[56] G. Subramaniam, J. Verma, N. Chandrasekhar, K. Narendra, K. George, "Generating playlists on the basis of emotion," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2018, pp. 366–373.

[57] H. Abderrazik, G. Angela, H. Brouwer, H. Janse, S. Lutz, G. Smitskamp, S. Manolios, C. C. S. Liem, "Spotivibes: Tagging playlist vibes with colors," in *Proceedings of the 6th Joint Workshop on Interfaces and Human Decision Making for Recommender Systems co-located with 13th ACM Conference on Recommender Systems, RecSys 2019*, vol. CEUR, 2450, 2019, pp. 55–59.

[58] P. Helmholz, M. Meyer, S. Robra-Bissantz, "Feel the moosic: Emotion-based music selection and recommendation," in *32nd Bled eConference: Humanizing Technology for a Sustainable Society*, Bled, Slovenia, 06 2019, pp. 203–221.

[59] *Spotify* for developers, "*Spotify* web api." https://developer.spotify.com/documentation/web-api/, 2020. [Online; accessed 19-July-2020].

[60] M. D. Team, *Microsoft Application Architecture Guide, 2nd Edition (Patterns & Practices)*. Wiley, 2009.

[61] L. Liu, M. T. Özsu Eds., *n-Tier Architecture*, pp. 1924–1924. Springer US, 2009.

[62] T. M. project, "AcousticBrainz." http://acousticbrainz.org/, 2015. [Online; accessed 19-July-2020].

[63] *Spotify* for developers, "Description of the *Audio Feature Object*," 2020. [Online]. Available: https://developer.spotify.com/documentation/web-api/reference/#objectaudiofeaturesobject, [Online; accessed 26-July-2021].

[64] R. Panda, R. Malheiro, R. P. Paiva, "Novel audio features for music emotion recognition," *IEEE Transactions on Affective Computing*, vol. Early access, 2018, doi: 10.1109/TAFFC.2018.2820691.

[65] C. Laurier, M. Sordo, J. Serrá, P. Herrera, "Music mood representations from social tags," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009)*, Kobe, Japan, 2009, pp. 381–386.

[66] K. Trohidis, G. Tsoumakas, G. Kalliris, I. Vlahavas, "Multi-label classification of music by emotion," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2011, p. 4, Sep 2011, doi: 10.1186/1687-4722-2011-426793.

[67] J. Bergstra, Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281–305, Feb. 2012.

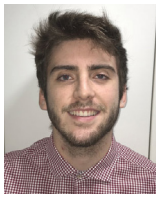
[68] P. Desmet, M. Vastenburg, V. Bel, D., N. Romero, "Pick-a-mood; development and application of a pictorial mood-reporting instrument,"

in *Proceedings of the 8th International Conference on Design and Emotion: Out of Control - Proceedings*, 09 2012, pp. 1–12.

[69] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal, *Pattern-Oriented Software Architecture, Volume 1, A System of Patterns*. Wiley, 1996.

Pedro Álvarez

Pedro received the Ph.D. degree in computer science engineering from the University of Zaragoza, Zaragoza, Spain, in 2004. He works as Lecture Professor at this University, since 2000. His current research interests focus on two main aspects. First, on integration problems of network based systems and the use of novel techniques and methodologies for solving them. And, secondly, on the application of formal analysis techniques and artificial intelligence techniques to extract knowledge from logs, databases and/or IoT systems.



Jorge García de Quirós

Jorge received the B.E. degree in computer science engineering from the University of Zaragoza, Zaragoza, Spain, in 2018. He works as Researcher at this University, since 2018. His current research interests focus on affective computing and music emotion recognition. He has also interest in other topics as cybersecurity and sport data analysis.

Sandra Baldassarri

Sandra received the Ph.D. in Computer Science Engineering from the University of Zaragoza, Spain, in 2004. She is Associate Professor in Computer Science Department at the University of Zaragoza (Spain) and founder member of the AffectiveLab Research Group and member of the Engineering Research Institute of Aragon (I3A), both at the University of Zaragoza. Her research interests include affective computing, multimodal interfaces, tangible and natural interaction, virtual humans and their application in educational fields. In these areas she published a numerous papers in conferences and journals and participates as part of the scientific and organizer committees of several Human Computer Interaction national and international conferences.