UCLouvain

École polytechnique de Louvain

Importance of Instrumentals, Vocals and Lyrics in Music Emotion Classification, based on Random Forest and Logistic Regression

Author: **Thomas Berton**
Supervisors: **Jean-Charles Delvenne, Marco Saerens**
Readers: **Jean-Charles Delvenne, Marco Saerens, Bertrand Lebichot**
Academic year 2018–2019
Master [120] in Computer Science

Abstract

This thesis addressed the importance of the vocals, the instrumentals and the lyrics in Music Emotion Classification. More precisely, it investigated which of the three elements is better at predicting the emotion of a song. The combination of them has been analysed as well. Furthermore, we compared two machine learning, i.e. Random Forest and Logistic Regression, in order to determine which one is better in the context of this work. To our best knowledge, this work is unprecedented.

The classification is based on state-of-the-art features for both audio and lyrics. The instrumentals have been represented by a mix of high and low-level features, whereas the database of vocals only contains spectrum-based features. Regarding lyrics, various preprocessing approaches have been tested as well as tf-idf scores. The conceptualization of the emotions has been based on Hevner's adjectives checklist, taking into account four different emotions. They are the *calmness*, the *sadness*, the *happiness* and the *angriness*.

Regarding the database, a ground truth dataset containing around 300 songs has been created. This number can motivated by the fact that databases in this research area rarely exceed 1000 songs. For each song are associated the vocals, the instrumental and the lyrics. The separation between the voice and the instrumental has been achieved thanks to an audio software called RX7. The use of the latter is unprecedented in Music Emotion Recognition. Since the databases are a major problem in MER and a lot of time has been dedicated to building a ground truth dataset, we freely released the databases on which the experiments have been based. The reader will find them at the following url : https://github.com/thomasberton/musicemotionrecognition.

Experiments showed that the best accuracy rate for emotion classification is provided by instrumentals (75%) followed by a combination of the three elements (67%). Then, the vocals provided an accuracy of 63%, whereas the lyrics performed 49%. Regarding the models, the random forest approach has outperformed the logistic regression for the audio analysis, i.e. the vocals and instrumentals, and the multimodal analysis. However the logistic regression model has provided a better accuracy for the lyrical analysis. Additionally, this model has showed better performances regarding the prediction of happy songs when it only considered vocals. The models were foremost good at predicting songs that had an opposite arousal, i.e. calm songs versus angry songs. This can be explained by the fact we mostly used global features, which are known to induce this kind of effects.

Keywords— AMMD, MER, Multimodal Emotion Extraction, Random Forest, Logistic Regression

ACKNOWLEDGEMENT

First, I would to like to express my sincere gratitude to my supervisor and co-supervisor, namely Professor Jean-Charles Delvenne and Professor Marco Saerens, for giving me the opportunity to realize my master thesis on a topic that represents all my life : Music. Thank you for your indispensable guide during the exploration of this research field, your enthusiasm and your advises throughout the year.

I would like to express my sincere love to my parents and siblings. Thank you for your continuous support during these last five years. Thank you Mom for all the little (big !) things you did for me. I'm notably thinking about all the dishes you prepared for Sunday nights. Somehow, they always gave me comfort and motivation for the weeks. Dad, thank you for working that hard and giving me the opportunity, with mom, to attend the university. I never took this chance for granted and I always did my best to honor this privilege you gave me. Thank you, my brother Hadrien, for demonstrating me what perseverance means. Not a day goes without thinking about the day everything has changed for both of us, I was so lucky to have you by my side ! Thanks to my sister, Marie, for showing me what being strong is. The way you raise your three children is an example for any mom. I'm so proud of you !! Last but not least, I would like to thank all the rest of my family — grandparents, uncles, aunts and cousins — for always being so supportive during these last five years.

I would like to thank my "brother-in-arms", which almost shared everything with me during these last five years. Thanks to my best friend, Valentin Delaunoy. If I can't not summarize all the things we lived together, there is something I'd like to tell you though : it was a bless to share my studies with you ! Doing all the projects with you always lightened the amount of work to deliver. As I am often comparing ourselves to Frodo and Sam, there is something I'd like to admit : Thomas wouldn't have got far without Valentin.

Finally, I would like to express my affection to all my friends and people that made this 5-years experience as unique and special as it has been. I learned so many things about friendship, love, myself thanks to you all guys. Specially, thanks to Gilles Bodarwe and Derrick Nzarubara for both being reliable close friends and being so supportive regarding my musical project. Thanks to La Communauté, which I shared so many great memories with. Thanks to La Tournaisienne for being a life lesson and introducing me to so many people that are now really good friends. Thanks to Huawei for giving me the opportunity to discover China and meet awesome people. Finally, thanks to University Catholique de Louvain for sending me to the Politecnico di Milano, where I met so many interesting people. I can't cite you all but special thanks to my Spanish guys, the Belgian girls and boys, Dennis, Ivan, Louis and of course Etienne.

CONTENTS

Acronyms

2DES Two Dimensional Emotion Space

AMMD Automatic Music Mood Detection

API Application Programming Interface

BOW Bag-Of-Words

BPM Beat-Per-Minute

HP Hyperparameters

DCT Discrete Cosine Transform

ISMIR International Society for Music Information Retrieval Conference

LR Logistic Regression

MER Music Emotion Recognition

MFCC Mel-Frequency Cepstral Coefficients

MIR Music Information Retrieval

MIREX Music Information Research Evaluation eXchange

RF Random Forest

RMS Root Mean Square

SER Speech Emotion Recognition

TF-IDF Term Frequency - Inverse Document Frequency

WAV Waveform audio file format

CHAPTER 1

INTRODUCTION AND STATE OF THE ART OF MIR

This chapter introduces the objective of this degree project together with the related work that has already been achieved in the Music Information Retrieval field (abr. MIR). It ends with the underlying motivations of this thesis followed by the thesis outline.

1.1 Introduction and State-Of-The-Art Review

Since ancient times, the phenomenon of music gives a rhythm to humans' life. This art is part of their daily life, no matter they desire it or not : it is nowadays played on the radio, on TV and common areas such as stores and restaurants. The one that tried to spend one day without listening to any music has probably spent the day isolated in his/her bedroom.

Since the last two decades, the paradigm of music distribution has nevertheless known an important shift. Generally speaking, the XXIth century has been a decisive turning point for the music and its industry. Indeed, from vinyls, people switched to the tapes, cd's and usb keys. It goes without saying that nowadays, platforms of streaming such as Youtube or Spotify are the most common ways to listen to music [209, 247]. Posting music being consequently easier and faster, the growth of digital music and related metadata has exploded exponentially [241]. This required updating music information techniques and it received high interest both from academic and entertainment domains. This is the reason why and how the field of *Music Information Retrieval* saw the light [56]. This area of research is defined as the discipline between music and informatics that aims at retrieving information from music. It has many real-world applications and can be used to categorize, manipulate or create music. As examples of concrete applications, recommended systems (e.g. Spotify, Youtube) and audio identification (e.g. Shazam) can be mentioned.

In the context of this thesis, the focus is rather set on the classification of tracks based on the emotions they capture, and what influences, emotionally speaking, the most the listener. The branch of MIR devoted to identifying the emotions of a music is the *Music Emotion Recognition* field (abr. MER), whereas automatically determining the emotion of a track is referred as *Automatic Music Mood Detection* (abr. AMMD). Both are really intertwined. Even though they have been active fields of research in MIR for the past years [99, 122, 139, 142], these both fields are still at an early stage [248] due the multiple challenging problems it faces. First, music mood recognition systems are based on models of emotions, which remain an active topic of psychology and scientific research [247]. There exist multiple emotion taxonomies but unfortunately no consensus between researchers on which one should be used. Moreover, the collection of ground truth emotion labels, regardless the emotion representation that is used, tends to be a particularly effortful task due the subjective perception of emotion [208]. Finally, the intrinsic element of music that creates an emotional response for the listener is still far from being well-understood [247].

In order to develop and improve the field of Music Information Retrieval, researchers have developed a community-based framework since 2005 under the name of *Music Information Re-*

search Evaluation eXchange (abr. MIREX), aiming at evaluating MIR systems and algorithms [56]. MIREX has considered *Automatic Music Mood Detection* as a task in 2007. These evaluation campaigns are coupled with the ISMIR conferences, which are organized each year for researchers across the world and present advancements in the MIR field. The acronym stands for *International Society for Music Information Retrieval*, which is the world's leading forum for music-related data.

Psychologically speaking, the earliest work regarding music mood analysis is addressed to psychologist Kate Hevner. In her work [85], she defined groups of emotions and studies classical music to unveil correlations between emotions and characteristics of the music. Scientifically speaking, earliest MIR works were focused on audio features. If there exists a wide variety of them, they have been partially put together by Kim et al in [114]. In 2003, Li et al introduced features based related to the pitch and rhythm, which was unprecedented [127]. Testing classical audio features such as the Mel-Frequency Cepstral Coefficients has been proposed by [229, 162]. Few studies, however, exploited information form the vocals of the song, which has been proved to play an important role in emotion perception though [247]. Around 2010, studies [122, 100, 101] have shown that emotions were not fully encapsulated within the audio, promoting approaches with music metadata such as lyrics to improve accuracy. Until this day, none of these works analysed the emotion weight respectively carried by the instrumental, the vocals and the lyrics of a song. This shows the opportunity to investigate this unexplored task.

Due to its interdisciplinarity, the entirety of the Music Emotion Retrieval past is not covered here. For concision, we preferred to present previous works each time we introduced a new topic along the thesis.

1.2 Motivation and Contribution of this Thesis

Before expressing the research questions this thesis aims to answer, I'd like to express my personal motivation behind it. Music has always been an essential part of my life, whether as a pure music lover or as a multi-instrumentalist (i.e. guitarist, pianist and singer). For multiple personal reasons, pursuing a career in the musical industry has become a serious motivation (obsession) since few years. This is why this thesis was the perfect opportunity to gather my study and my passion : something in the middle of music emotion and Artificial Intelligence. However, as a true music lover, I didn't want to work on something that keeps the musicians at bay such as music generated by Artificial Intelligence. I wanted to provide something that musical industry and others music lover programmers could rely on. I feel that the art of music creation must stay in humans' hands, while informatics can provide really powerful and various tools to help musicians.

The goal of this thesis is to provide a unprecedented multimodal analysis of songs in order to determine whether lyrics, vocals or instrumentals achieve the highest accuracy rate for mood classification. The combination of them is also investigated. The four emotional labels that have been considered into this thesis are respectively the *calmness*, *sadness*, *happiness* and *angriness*. To our very best knowledge, analyzing separately the three components of songs and determining which support provides a better mood classification has not been done yet (or at least released publicly). The accuracy provided by the combination of the three different approaches is intended to be tested as well. Additionally, this research aims at comparing two well-known machine learning classifiers : Random Forest [30] and Logistic Regression [103], in order to determine which one is the best for *Music Mood Classification*. In other words, this project will aim to answer these questions :

- Relying on multiclass machine learning classifiers, which of the three components of the songs (lyrics/vocals/instrumentals) gives the best accuracy for emotion classification ? Does the combination of the three sources improves significantly the accuracy of the models ?
- Which model, namely Random Forest or Logistic Regression, fits the best this concrete application (in terms of result, e.g. accuracy) ?

1.3 Thesis Outline

This thesis will be structured as follows. The Chapter 2 covers relevant information about the emotion theory and presents the two most popular models of emotion representation that exist until this day. The Chapter 3 describes the underlying theory of the different approaches for emotion extraction and which features have been selected for each mode. Chapter 4 presents the different databases used in this thesis and the multiple steps that have been executed to gather pertinent information. In Chapter 5, the machine learning classifiers, namely Random Forest and Logistic Regression, are described in details with their hyperparameters. Accuracy metrics are covered in this chapter too. The results and analysis of the classification are presented in Chapter 6. The thesis ends with a conclusion and propositions for future work in Chapter 7.

CHAPTER 2

EMOTION THEORY

The emotion conception is an important cornerstone for Automatic Music Mood Classification. As the reader can expect, the way emotions are chosen to be represented will impact the results of the classification. Multiple representations exist, and there does not exist a better approach than another, each one having advantages and disadvantages.

This chapter settles the theoretical basis of emotion conceptualization. The Section 2.1 introduces the concept of emotion and how it can be represented. Then, Section 2.2 and Section 2.3 describe and illustrate respectively two different well-known approaches that exist in details.

2.1 Emotion Conceptualization and Perceptions

What is an emotion and how can it be conceptualized ? How people perceive emotions from music ? Here are two questions that should let some people bewildered. A huge amount of work in psychology and MER has been done for defining perceived emotions in music and a variety of mood representation exist [85, 188, 65].

Before any conceptualization, one may wonder how to capture emotions that listeners feel when they listen to music. In order to seize these emotions, psychologists often utilize people's verbal reports as emotion responses [111] : subjects are asked to write down, within a precise list or adjectives or not, the emotions they feel. Other approaches may be used, such as measuring stimulus responses from the subject's brain [196].

Regarding the conceptualization, two representations stood out. On one hand stands the categorical approach, which represents emotions by clusters of tags (e.g. *happy*, *dreamy*, etc). The fundamental basis of this representation is based on the Hevner's checklist [85]. As said before, it is one of the first works related to emotions in music. During her survey, listeners were asked to write adjectives that came to their minds as the most descriptive of the music played. Hevner discovered the existence of clusters of descriptive adjectives and laid them out in a circle in 8 clusters. She also noticed that labelling is consistent within a group that shares a similar cultural background. Later, the Hevner's adjectives were refined and regrouped into ten adjective groups by Farnsworth [65]. On the other hand, a continuous representation may be chosen. Russell [188] defined a 2-dimensional continuous space for embedding emotions. A point in this space is defined by its two coordinates, which are the valence (from negative to positive mood) and arousal (from calm to energetic mood) of an emotion. The space is thus divided in 4 parts, representing positive/high and negative/low values respectively. These both approaches are described in depth in the following sections.

As the reader may imagine, there exist other approaches to conceptualize the phenomenon of emotion. Another third approach can be notably cited. It consists of focusing on the dynamic process of music emotion and extract emotions for every short-time segment of a song, resulting in

a series of emotion predictions. This approach is referred as *Music Emotion Variation Detection* (abr. MEVD) [197]. As this approach is not used by a large community, it is not covered in details in this thesis.

The conceptualization of emotions and the associated emotion taxonomy are a central issue in the Music Emotion Retrieval field. Until this day, there is still no consensus on which emotion model should be used : whether emotions should be conceptualized as categories or considered as continuous. Moreover, questions arise for both described models. For the categorical approach, one may ask how many emotion categories should be used : based on Hevner's work or brand-new one. Regarding the dimensional approach, some researchers promote the idea of using a third dimension, arguing two dimensions don't capture the full concept of emotion. More details are covered in Section 2.2.

Methodology	Emotion conceptualization	Description
Categorical MER	Categorical (Hevner) [85]	Emotions are grouped by clusters
Dimensional MER	Dimensional (Russell) [188]	Emotions are expressed in terms of valence and arousal.
MEVD	Dimensional [197]	Emotion of the song is described by the different emotions expressed within the track.

Table 2.1: Different approaches that exist in Music Emotion Recognition [246]

Besides the choice of various approaches, one should not forget that the perception of emotion is subjective and thus subject to various interpretations. This means that songs can be labelled by two completely different emotions, depending on the listener's feelings and perceptions. This is even complicated by the fact the affective response to music is also sensitive to the environment, contexts of listening, or still mood [141]. For instance, subject may not feel the same emotion for the same song for two consecutive days. Additionally, some perceptual considerations must be explained and taken into account. Many studies showed that, regardless the chosen emotion conceptualization, people tend to confound the emotion(s) *expressed* by the music and the emotion(s) *induced* by the music [110]. For the remainder of this thesis, the focus is set to discern the emotions expressed, rather than induced by music. Last but not least, songs themselves can capture multiple emotions, e.g. an emotion at the introduction differing from the emotion at the end of the song. These challenging issues make AMMD systems hard to obtain extremely high results for classification. Moreover, the assumptions and choices regarding emotions consequently induce lack of cohesion between MER works [247].

2.2 Dimensional Conceptualization of Emotion

The dimensional approach focuses on identifying emotions based on their placement on a few number of emotion dimensions, which are intended to correspond to the internal human representation of emotion. Subjects are asked to use a large number of rating scales of affective terms to describe the emotion of music stimulus. *Factors* are then extracted from the correlations between the scales. Most of them correspond to the dimensions of emotions. They can be defined by different attributes, such as the *valence* (i.e. pleasantness : positive or negative affective states), the *arousal* (i.e. activation : energy and stimulation levels) or still the *potency* (i.e. dominance : a sense of control or freedom of act) [159, 167, 182, 226]. These features are often referred to as the three dimensions of an emotion. This representation can be motivated by the fact this continuous approach is closer to the human representation of emotion than the categorical method [85]. Humans usually don't cluster the emotions : it is generally an abstract combination of different factors.

One of the most used model for this scalar approach is the one proposed by Russell [188]. This is the *circumplex* model. The model consists of a two-dimensional and circular structure involving the dimensions of valence and arousal, as shown in Figure 2.1. For instance, high arousal values correspond to energetic moods such as "angry" and "excited", while negative values are synonyms of moods like "tired" or "sleepy". High valence values are linked to moods such as "happy" or

"content" whereas negative values gathers moods as "bored" and "anxious". Within this structure, emotions that are inversely correlated are placed across the circle from one another. The circumplex model is also referred as the *Two-Dimensional Emotion Space* (abr. 2DES). One of the strengths of the circumplex model is that it suggests a simple yet powerful way of organizing different emotions in terms of their affect appraisals (i.e. valence) and physiological reactions (i.e. arousal). Moreover, comparison of different emotions on two standard dimensions is directly allowable. The validity of this model space has been confirmed in multiple studies [111, 121].

However, the two-dimensional model is sometimes criticized. Indeed, some researchers argue that due to this continuous representation, psychological distinctions between emotions are blurred and tend to become ambiguous [123]. As example, one may notice in Figure 2.1 that the emotions "angry" and "afraid" are placed next to each other although they imply very distinct reactions for the listeners. To fix that, researchers have recommended using a third dimension, which is the *potency* [247]. It expresses the idea of dominance and submission [42]. If this extension helps to seize a more complete picture of emotion, it is however subject to speculation and disagreement [15]: during the annotation process, subjects are hence asked to annotate emotion in 3D, which is more difficult to achieve. One may imagine the difficulty of visualizing music in 3D rather than in 2D. In his work, Juslin et al [110] proved the two-dimensional model offers a balanced representation between a parsimonious definition of emotion and the complexity of the task.

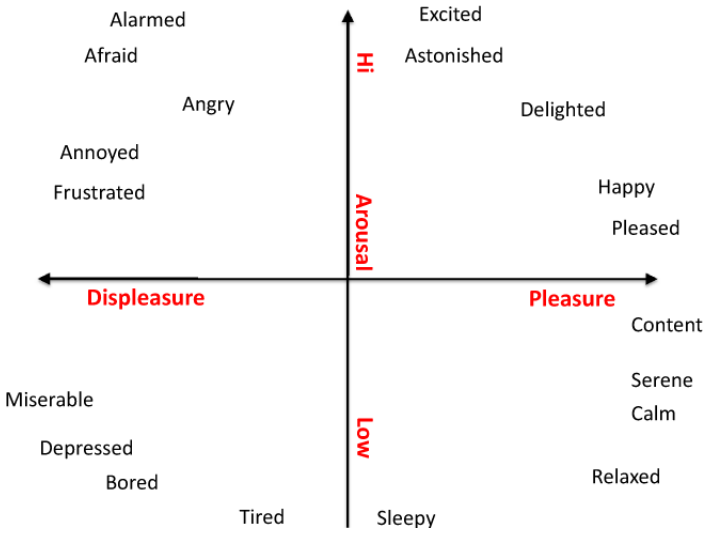


Figure 2.1: Circumplex Model of emotion by Russell [188]

2.3 Categorical Conceptualization of Emotion

The categorical conceptualization is based on the separation of emotions. They can be categorized as distinct from each other. One concrete model of this approach is based on the idea that there is a limited number of universal / primary emotion class such as *happiness*, *sadness*, *anger* or still *fear* from which all other secondary emotion classes can be derived [58]. If the basic emotions have the advantage to be found in all cultures, this conception is however not accepted by all the researchers : some have come up with different sets of basic emotions [111].

The categorical approach is often represented and based on one of the earliest studies provided by Kate Hevner, in 1936. In her work she published a categorization of emotions with 66 adjectives, arranged into 8 groups [85]. The adjectives within each cluster are similar, and the meaning of neighboring clusters slightly varies until reaching a contrast in the opposite position. Illustration of the Hevner's adjectives checklist can be found in Figure 2.2. In 1954, Farnsworth proposed an update of this checklist and regrouped emotions into ten adjective groups [65]. Later, in 2003, a representation with nine adjective groups was proposed by Schubert [198]. In 2010, MIREX evaluations for Automatic Music Mood Classification have categorized songs into five mood clusters,

shown in Table 2.2 [88]. Many categorical studies highlight an advantage of this approach : such form of emotional groups can be intuitive and consistent, regardless the listener's musical training [198]. Hence, this reduces the risk of mistakes during the annotation process.

Such as the dimensional conception, the categorical approach is not deprived of drawbacks. One of the major criticisms addressed to the categorical approach is that the number of emotion classes is too small compared to the richness of music emotion perceived by humans [246]. However, using more detailed and precise clusters does not solve the problem neither, because it leads to more ambiguity during the annotation process. Indeed, distinction between similar emotions, yet important for this task, depend on the subjects' language and thus varies between annotators [110]. Furthermore, using a larger number of emotion classes would overwhelm the subjects and is impractical for psychological studies [111]. Once again, the reader can notice the challenging problems in MER research field due to the lack of cohesion.

This is the categorical approach that has been chosen for emotion conceptualization in this work. From a machine-learning perspective, choosing this kind of approach results in a multi-label classification problem [127]. Indeed, music sounds can be classified into multiple classes simultaneously. For instance, a chord progression can sound *calm* as well as *sad* . For the seek of simplicity, many works assume a multiclass classification [247]. In this thesis, we also assumed a multiclass classification between four distinctive emotions, namely *calm*, *sad*, *happy* or *angry*. Motivations and more details about this assumption are covered in Chapter 4.

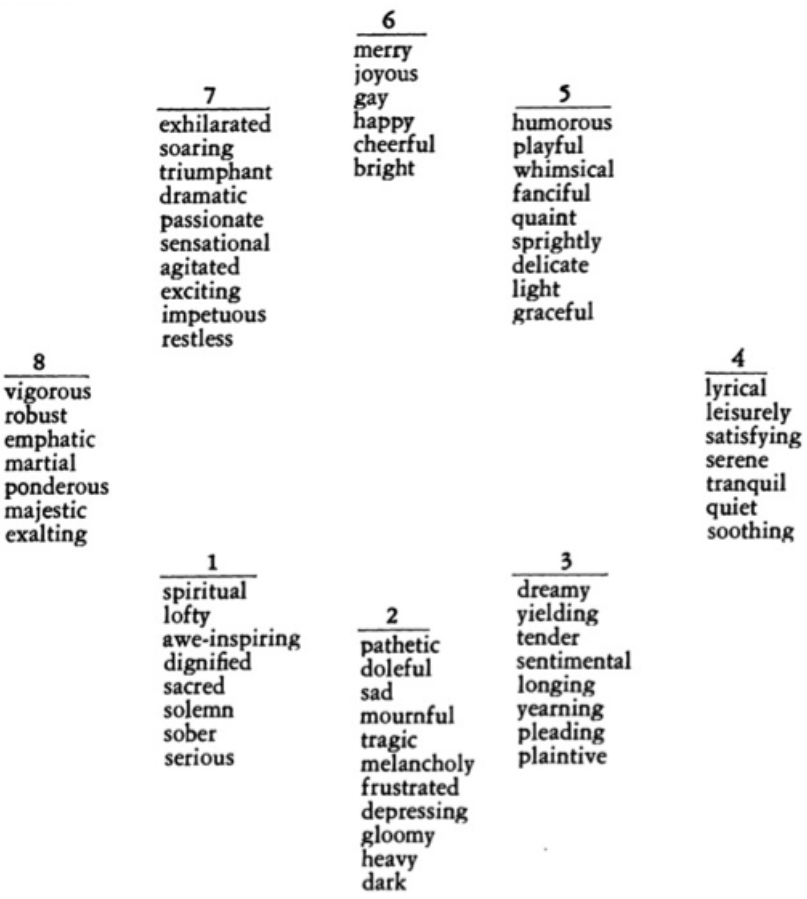


Figure 2.2: Hevner's adjectives checklist [86]

Clusters	Mood Adjectives
Cluster 1	passionate, rousing, confident, boisterous, rowdy
Cluster 2	rollicking, cheerful, fun, sweet, amiable/good natured
Cluster 3	literate, poignant, wistful, bittersweet, autumnal, brooding
Cluster 4	humorous, silly, campy, quirky, whimsical, witty, wry
Cluster 5	aggressive, fiery, tense/anxious, intense, volatile, visceral

Table 2.2: Mood adjectives clusters defined by MIREX in 2008 [177]

CHAPTER 3

THEORY OF MULTIMODAL EMOTION EXTRACTION

This chapter provides the theoretical principles of the three different approaches used for retrieving emotions. Section 3.1 explains in details the fundamentals of speech emotion recognition, the existing methods to extract emotions from voices and the features that have been used in this work. Section 3.2, focused on the instrumental analysis, defines how and which features have been extracted from this support. This chapter ends with Section 3.3, covering the foundations of the lyrical analysis.

3.1 Voice Analysis

This section is focused on the emotion related to the voice of the singer(s) and features that have been used to represent this emotion. For any music listener, the voice carries an important part of the *expressed* emotion. Extracting the latter requires to focus on a domain called *Speech Emotion Recognition* (abr. SER), a subfield of Natural Language Processing [17, 138, 164, 130]. This is nowadays a hot topic that has gained a lot of interest for numerous researchers.

This chapter is organized as follows. First a brief introduction of the SER research field is presented, as well as a state-of-the-art review . Section 3.1.2 highlights the relevant features that are commonly used in this research area and Section 3.1.3 explains the features that have been exploited in this thesis.

3.1.1 SER and Related Problems

The purpose of the relatively recent[1] *Speech Emotion Recognition* research field (abr. SER) is to extract the emotional state of a speaker from his or her speech. Increasing attention has been lately directed to the study of this specific topic and many works have been dedicated to this task [1, 116, 40].

If Blanton has written that "*effect of emotions upon the voice is recognized by all people (...) the language of the tones is the oldest and most universal of all our means of communication*"[21], it is nevertheless not an easy task for a computer to discover the underlying emotions. Indeed, the task of speech emotion recognition is very challenging for many reasons. The first and principal problem is related to the features selection : there does not exist one holy-grail approach that advocates better features than others for representing the emotion expressed by a song. The lack of cohesion and the freedom of selecting different features makes the process complex and subject to different results [20]. Secondly, there may be more than one perceived emotion from the same utterance[2], which makes the analysis even more complicated. Thirdly, the acoustic variability of the speech is a problem too : different speakers and speaking styles make speech signals highly

[1] SER is said to be born after the seminal work of Daellert et al. in 1996.
[2] The smallest unit of speech.

variable and directly impact the extracted features [72, 7]. Other limitations are added to the development of this task. For instance, a challenging issue is that how a certain emotion is expressed generally depends on the speaker's culture and environment [102]. For these reasons, most of the research works have been focused on a monolingual emotion classification, assuming there is no cultural difference among speakers [60].

Besides these multiple challenges, there exist other important aspects that must be taken into account during the creation of Speech Emotion Recognition system. One can notably cite the design criteria of the databases used for training the classifier [60, 37]. For instance, an important criterion is the degree of naturalness of the speeches. Core questions regarding relevant features are focused on how well should the databases simulate a real-world environment. Should it use real-world emotions or acted ones ? [242] Who utters the emotions ? [61] How to simulate the utterances ? [27] These are the questions that come with the creation of a SER database. Hopefully, in this work, these questions are directly solved by the fact our database is composed by real-world records of human's voices. More details about the considered databases are expressed in Chapter 4.

Once again, the set of considered emotions is a decisive aspect of the SER systems. Specific common sets have been determined , such as the one from O'Connor and Arnold [156]. However, due their large number of emotions — 300 emotional states — the classification remains difficult. For this reason many researchers agree with the *palette theory*, which refers to a concept mentioned earlier : any emotion is composed of primary emotions, the same way any colour is composed of some basic colours [45]. These primary emotions can be the anger, joy, sadness or disgust.

3.1.2 Relevant Features for SER

The most important choice in the design of a speech emotion recognition system, besides the origin of speeches, is the selection of relevant features that *efficiently* represent the underlying emotion. As declared in May 2018 by Schuller, the design of ideal features is still an ongoing active subfield of research in the SER domain [20].

Ideally, these features should be robust against environmental noises, varying languages or cultural influences. This is, in practice, not an easy task. Multiple assumptions must be considered during a features extraction [60]. The first decision concerns the region of analysis used for features extraction. So far, the community of researchers is divided into 2 approaches : either dividing speech signals into small intervals called frames, from which local features vectors are extracted, or extracting global statics from the whole speech utterance [89, 176, 199]. The second issue regards the features types to be analysed, e.g. pitch, energy, etc. Which are the best ones remains an opened question [35]. Following paragraphs provide more details about these challenging questions.

Before any feature extraction per se, one may decide if audio files are analysed from a local or global point-of-view. Since speech signals are not stationary at global view, it is common to split the speech signal into small segments called frames. Within each frame, the signal can be assumed to be approximately stationary [176]. Global features are statistically computed on features extracted from an utterance. If researchers are divided between the local and global approaches, some works [89, 199, 233] have proved that classifications with global features achieve a higher accuracy rate than with local ones. Furthermore, the relatively small number of global features increases the fastness of the algorithms.

The global approach is nevertheless not deprived of drawbacks. Researchers proved it is only efficient at distinguishing energetic emotions (e.g. anger, fear, joy) from low-energy ones (e.g. sadness). Consecutively, they fail in classifying emotions which share similar arousal, e.g. *anger* versus *joy* (see Figure 2.1 for reminder). Additionally, the small number of features may prevent the system to use classifiers such as Hidden Markov Model [84] or Support Vector Machine [19], which require numerous training vector to reliably estimate the model parameters. With these classifiers, a large number of local feature vectors provides hence a higher accuracy rate [60]. With a global approach, the temporal information of the speech is lost too.

No matter the choice of a global or local approach, one should now focus on the extraction of

speech features that efficiently characterize the emotion of the speech. As said previously, they should be independent of the speaker and the lexical content, which is in practice more complicated. Many features have been explored during the past years but scientists have not identified the best speech features for this task yet. So far, there exist four groups of speech features : the continuous features, qualitative features, spectral features and TEO-based features. The Figure 3.1 illustrates the different existing categories and shows examples for each type of features. Following paragraphs describe the advantages and drawbacks of each category, even though it is a common practice in SER to combine features that belong to different categories.

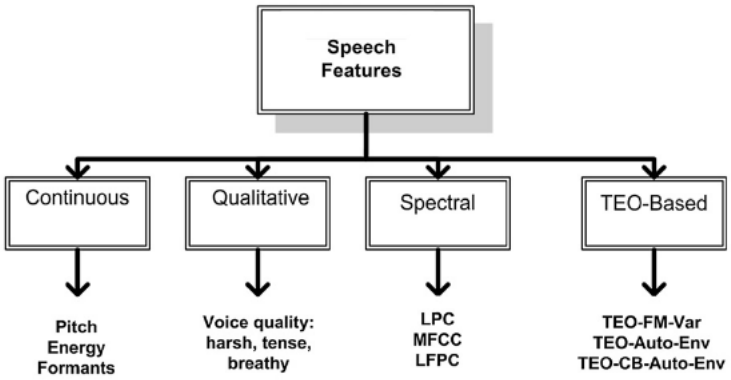


Figure 3.1: The four existing categories of speech features [60]

First of all, the category that has gained a lot of attention is the group of continuous features, i.e. varying in time. Gathered under the name of prosody[3], characteristics of the voice such as the pitch have been proved to convey emotional content [23, 34, 112]. To fully understand what pitch means, the reader should first know what the fundamental frequency of the voice means. It is defined as the frequency of vocal folds vibration and measured in Hertz [90, 95]. Since it is not stationary in time, it is computed multiple times along a speech duration and then averaged. More details about the calculation are given in the following section. This frequency is often denoted as F0. The term *pitch* is often used to talk about the same concept although F0 and pitch are not fundamentally identical. Pitch originally refers to the human *perception* of this fundamental frequency, i.e. the way the auditory system perceives it. Since it does not perceive pitch in linear manner[4], it is measured with the *Mel* scale [236]. The misuse of the latter leads unfortunately to some ambiguities and misunderstanding. For the rest of the thesis, the term *pitch* will refer to the fundamental frequency, as it is commonly used in the SER literature [60].

The energy of the voice is a common-used continuous feature too. It refers to the variations in amplitude of the signal [51, 230]. Other well-known features can be cited, such as the *formants* [80]. They represent the resonances of the voice in the vocal tract (area from the nose down the throat). More scientifically speaking, a formant is defined as "a concentration of acoustic energy around a particular frequency in the speech wave" [243], which is particularly observable in a spectogram [93]. Research studies have investigated the relationship between the mentioned speech features and the basic emotions, and they proved prosodic features provide a reliable indication of the emotion [7, 161, 22]. However, as the reader may imagine, contradictory reports exist though. For instance, Murray and Arnott [149] indicate that a high speaking rate is associated with emotion of anger, while Oster and Risberg [161] advocate an opposite conclusion. Furthermore studies have shown some emotions such as *anger* or *fear* share similar characteristics for the fundamental frequency [176].

The second category of speech features is related to the quality of the utterance. Indeed, it has been proved that the voice quality of an utterance is strongly related to the emotional content of the speech [45, 78]. More precisely, the quality of a voice can be defined by adjectives such as *breathy*, *harsh* or still *tense*. It seems to be linked with *full-blown* emotions, i.e. emotions that

[3]The term *prosody* refers to the rhythmic and intonational aspect of language
[4]More details are given in the following section, especially where the MFCC are introduced

directly impact people [45]. This is opposed to *underlying* emotions [45]. However, this kind of features has not been widely explored yet. This can be mainly explained by two reasons. First the voice quality is defined by adjectives such as *breathy* or *harsh*, which have various interpretations among researchers [78]. The second problem is related to the difficulty of automatically deciding these voice quality features directly from the speech signal [176]. Due to these two reasons and by the fact this kind of features is still at a early stage, they have not been considered for the classification of this thesis.

The third features category gathers the spectral-based speech features. This means the time-based signal has been converted into its spectrum in the frequency domain [25]. In their work, Nwe et al [155] demonstrated that the emotion content of an utterance has an impact on the distribution of the spectral energy across the speech range of frequency. For instance, a speech that contains a happy feeling shows high energy at high frequency range while a sad speech reveals small energy at the same range [7]. If there exist various features that represent the spectral information of a speech, a common approach is to use the Mel-Frequency Cepstrum Coefficients (abr. MFCC) [98, 1, 60, 233]. For concision and due to the theoretical aspects they cover, details about them are given in the following section and in Appendix B. Bou-Ghazale et al showed in their work that MFCC outperform the performance of other existing coefficients [24].

Finally, a fourth and last category of features exists. Referred to as Teager-Energy-Operator based features, they are particularly useful to detect stress in voice [36]. This approach is based on the fact that the air flow in the vocal system is influenced by the stress, due to the muscles tension. However, in the context of this work, the assumption that the singing voices are deprived of stressed conditions has been made. For this reason and due to their relative complexity, the TEO-based features have not been used.

3.1.3 Considered Features

One should remember that until this day, there does not exist *holy-grail* features that would extract the emotion of the vocals better than others. However, some features are more commonly used and give some clues to tackle the problem. Following paragraphs aim at describing the multiple features that have been considered during the classification of the emotions extracted from the vocals.

In his work *Features Importance Analysis for Emotional Speech Classification* [221], Tao classified different acoustic features according to their importance. In this ranking, the top three most relevant features were related to F0. This gives us the intuition that the classifier should take this feature into account. This can be further motivated by the fact F0 and related features are ones of the most used features regarding SER [7]. As a quick reminder, the fundamental frequency is the vibration frequency of the vocal folds when pronouncing voiced sounds. In other words, it measures how high or low the frequency of someone's voice sounds. As example, a adult male's voice has a lower fundamental frequency than a baby's voice. This is this frequency that people use to add intonation in their speeches. During a song, the singer controls the fundamental frequency of his/her voice according to the melody of the song [63].

The algorithm used for the estimation of F0 values is based on the *MELODIA* algorithm [192], provided by Essentia [59]. The idea of the algorithm is to estimate the fundamental frequency that corresponds to the perceived pitch of the main melody [79, 151]. It was first designed to extract the predominant melody from a polyphonic music, i.e. when different instruments play or a single instrument plays multiple notes at the same time [192]. However, since the vocal part has been isolated from the instrumental, a monophonic-based algorithm has been used [165]. Working with such isolated signals is easier for the algorithm since it can be assumed there is no superposition of different musical sources. This avoids the difficulty of attributing frequency bands and energy levels to corresponding instruments notes [193].

However, the melody extraction process faces multiple challenges. The first one regards the treatment a song and its instruments receive during the creation step. For music producers and sound engineers, it is a common practice during the mixing and mastering processes to add reverberation or compression [8]. If this kind of processes is essential for a song, it nevertheless blurs the

original signal and reduces the chances to extract *clean* features [192]. Furthermore, controversial debates exist over what the main melody of a song is, and definitions vary according the context [228, 189]. In this context, the melody will be referred as "the single (monophonic) pitch sequence that a listener might reproduce if asked to whistle or hum a piece of polyphonic music, and that a listener would recognize as being the 'essence' of that music when heard in comparison" [168].

The original algorithm can be described in four main steps. The first one consists of enhancing the middle frequencies, where the melody is expected to be found, and providing a spectral representation of the signal. Details about this filter is given at Section 3.2.3. Based on this representation, the spectral peaks are located. They are used to constructing a representation of the pitch salience[5] over the time, defined as a *salience* function [192]. These peaks are considered as potential F0 candidates. They are then grouped into pitch contours, i.e. sequences of salience peaks. Within each frame, the weakest peaks are filtered out so that only predominant candidates are preserved. Then, the algorithm needs to determine which contours belong to the melody. For this, some characteristics are computed for each contour such as the pitch mean or the pitch deviation. This kind of information helps the algorithm to determine the contours related to the melody. Moreover, the algorithm must be particularly careful to not select a F0 of the harmonic of the melody, instead of the melody itself. One way proposed is to iteratively calculate the melody pitch mean, i.e. the pitch trajectory over the time [192]. This approach is particularly useful to distinguish octaves duplicates. Finally, the melody selection is based on the remaining peaks that have not been filtered out during the previous steps. The different steps of this *MELODIA* algorithm are illustrated in Figure 3.2.

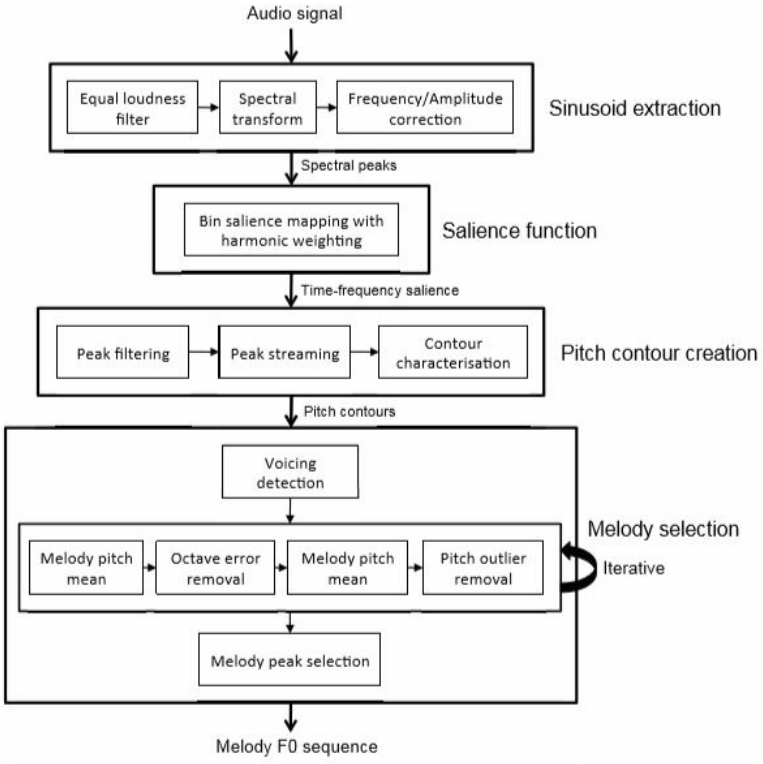


Figure 3.2: Creation steps of Melody F0 sequence [192]

Secondly, the focus has been set on Mel-Frequency Cepstrum Coefficients [48, 38, 81, 108]. These are often referred as the best in the speech recognition domain since they take human perception sensitivity with respect to frequencies into consideration [92]. The MFCC are commonly computed in 6 steps, as shown in Figure 3.3. Due to the multiple concepts it refers to, all the details about these sixth steps are given in Appendix B.

[5]The term *salience* refers to the capacity of standing out, of being particularly noticeable

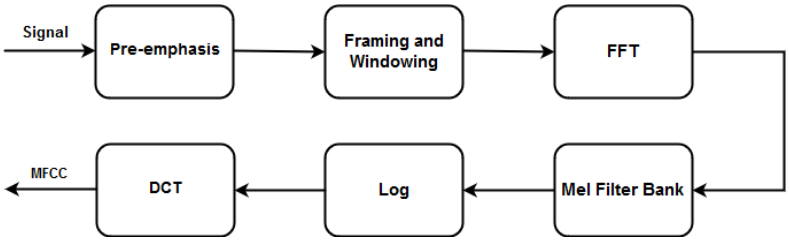


Figure 3.3: MFCC coefficients extraction process [48]

The Mel-Frequency Cepstrum Coefficients highlight an important point : human perception of frequency is non-linear [217]. For this reason, another scale is used, which is the *mel* scale [50]. During their experiments, Steven et al [217] designed the scale as linearly below 1000 Hz and logarithmic above. Indeed, the human ears act as filters and are non-uniformly spaced on the frequency axis : more filters are present at the low frequencies, whereas fewer filters are found at high frequencies. To mimic this human perception, researchers have come up with the concept of *mel filter bank* [129], shown in Figure 3.4. It groups overlapping triangular bandpass filters (denoted by different colours) which are linearly equally-spaced below 1 kHz and logarithmic equally-spaced above.

Besides this theoretical aspect, the reader should know that among the extracted coefficients ($\pm$ 40), only the first 13 coefficients are kept since they are the ones that contain the most important information [114].

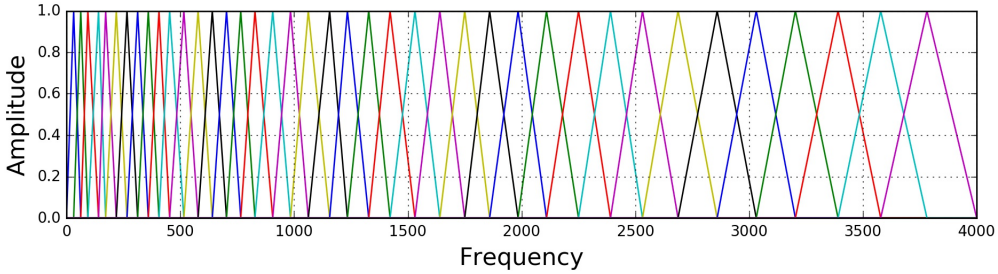


Figure 3.4: Mel filter bank from Young et al, 1997 [129]

Additionally, other features such as the spectral centroid [160], the spectral roll-off [160] or the zero-cross rate [178] have been tested during the classification based on the vocals. An explanation about these features is given in the following section, since they have been used for the instrumental analysis too. The importance of the features are discussed in the *Experiments and Results* Chapter.

3.2 Instrumental Analysis

As any music listener would express, the experience of music listening is multidimensional. Different emotion perceptions of music are usually associated with different patterns of acoustic cues. Emotion perception is rarely dependent on a single music factor but a combination of them [85]. As example, faster tempos are associated with high-energy songs, and slower tempos with lower energy, sadder songs. Loudness, or intensity of a song can be connected with anger, while softer songs would suggest tenderness, sadness, or fear [9]. All these features are described below.

To extract the relevant characteristics related to the emotion of an instrumental, multiple features have been tested (the reader will find the different categories of audio features in Appendix A, in Figure A.1) and analysed empirically. Their relevance during the classification are discussed in the *Experiments and Results* Chapter. The following section focus on explaining low-level features, which are directly obtained from the audio signal by applying techniques like Fourier Transform [25] or cepstral analysis (MFCC). The second section presents the middle-level features, which are basically extracted on the top of low-level ones [47].

3.2.1 Low-level Features

Low-level descriptors are computed from the audio signal in a direct or derived way, e.g. from the frequency representation. They have little meaning to users, but they contain relatively important information of the analysed song. They have been widely used in music classification, due to the simple procedures to obtain them[6] and their good performance [47].

The first low-level feature that has been taken into the consideration is a temporal feature[7] : the *zero-crossing rate* [178]. It computes the number of sign changes within a frame of time domain waveform. In others words, the zero-crossing rate is calculated by counting the number of times that the signal crosses zero (in terms of amplitude) within a given window. An illustration is given in Figure 3.5. It is an indicator for noisiness of the signal. Let $x_i[n]$ be the n^{th} sample of the i^{th} frame of a signal of size N, then the zero-cross rate Z_i can be computed as in Equation (3.1) [117].

$$Z_i = \sum_{n=1}^{N} |sign(x_i[n]) - sign(x_i[n-1])| \tag{3.1}$$

where

$$sign(x_i[n]) = \begin{cases} 1 & \text{if } x_i[n] \geq 0 \\ -1 & \text{if } x_i[n] < 0 \end{cases} \tag{3.2}$$

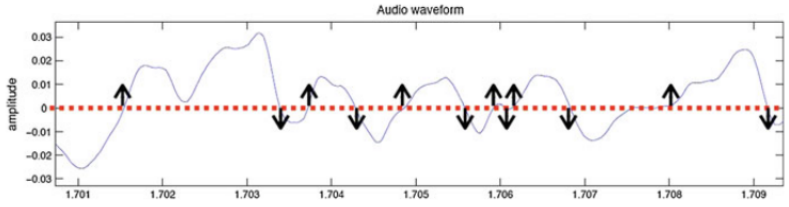


Figure 3.5: The zero-crossing rate is incremented each time the signal crosses zero [47]

Two others low-level features have been used, yet based on a spectral approach [152]. This means these features are calculated from the short time Fourier transform and are calculated for every short-time frame of the audio signal [31, 179].

The first one is the *spectral centroid* [160]. It indicates where the *centre of mass* of the spectrum is located. It is calculated as the weighted mean of the frequencies present in the signal, with their

[6] Many programming languages and softwares propose built-in functions for that purpose [207, 171, 59]
[7] which means based on a temporal distribution analysis

magnitude as the weights. In other words, the centroid is a measure to characterize the spectral shape. For instance, higher centroid values indicate higher frequencies. This feature is interesting, because it is defined as a measure of brightness of a sound [47, 160]. Let $x_i[n]$, $n = 0, 1, ..., N-1$ be the nth sample of the i^{th} frame of a signal, with $X_i[k]$, $k = 0, 1, ..., N-1$ the corresponding Discrete Fourier Transform coefficients. The spectral centroid C for each frame i can be thus defined as equation (3.3) [117] :

$$C_i = \frac{\sum_{k=0}^{N-1} k \cdot X_i[k]}{\sum_{k=0}^{N-1} X_i[k]} \tag{3.3}$$

The second spectral feature that has been considered is the spectral *roll-off* [160]. It is defined as the frequency below which some percentage of the magnitude distribution of the spectrum is located. Common values for the percentage are 85 % or 95 % [47, 147]. Let $X_i[k]$, $k = 0, 1, ..., N-1$ be the corresponding Discrete Fourier Transform coefficients of the original discrete signal frame $x_i[n]$, $k = 0, 1, ..., N-1$ and M the number of frames, the spectral roll-off can be defined as :

$$r_{0.85} = \sum_{m=0}^{M-1} r_m = 0.85 \sum_{k=0}^{N-1} X_m[k] \tag{3.4}$$

3.2.2 Middle-level Features

Middle-level features are usually extracted on top of low-level ones. This kind of features captures properties of music as humans can perceive, e.g. the rhythm [47].

First, one may give some interest to the intensity of the song [64]. The latter, which measures the loudness of a song, is considered to be the average volume across its entirety. The unit of intensity is the decibel (dB) [219]. This feature can be a strong indicator to indicate the happiness, sadness or calmness of a song [154]. During the experiments, two methods have been tested to extract the loudness.

The first one refers to *Root Mean Square* pressure [115]. It is one of the most used feature because it is directly related to the intensity. The RMS pressure is defined a squared root of the average of the square of the pressure of the sound signal [205]. A higher RMS would indicate a higher intensity, and a lower RMS would suggest a lower intensity, calmer song. Let $p(n)$, $n = 0, ...N-1$ be pressure points measured along the sound signal. Then the RMS pressure can be expressed as :

$$p_{RMS} = \sqrt{\frac{1}{N}\left(p[1]^2 + p[2]^2 + ... + p[n]^2\right)} \tag{3.5}$$

However, during the classification, it showed quite disappointing results compared to the second method, based on *Loudness* algorithm from the *Essentia* library [133]. This algorithm computes the loudness of an audio signal based on the Steven's power law, which defines the perceived strength of a stimulus as proportional to its physical intensity raised to a power [218]:

$$P = cI^b \tag{3.6}$$

In this equation, P is equal to the perceived magnitude, i.e. how we perceive the related sound. I is equal to the intensity of the actual stimuli : our perception depends thus on the intensity of physical stimulus, which is logical. The letter c equals to a scaling constant. The most important term of this equation is the exponent b, which is the power to which the intensity is raised. During his experiments, Stevens defined various exponents to use according the application field. For instance, one may use a coefficient of 1.3 to measure the perception of tasting salt, or a coefficient of 5 for brightness perception [218]. The required exponent for loudness has been set at 0.67 by Stevens. This induces that a large increase in intensity produces a small increase in sensation. This is motivated by the fact humans are less sensitive to small differences between high amplitudes than with low amplitudes [66, 218].

The second middle-level feature that draw some attention is the *timbre*. Timbre is the tonal quality of a sound[8], caused by its frequency components. This is precisely this combination of frequencies that gives to an instrument its own unique sound. For one single musical note, the difference in timbre between two instruments can be easily determined by analyzing their frequency responses over the note's duration. This difference is however not suitable for an entire song or at least a sample of it. A common used feature to characterize the timbre is the Mel-Frequency Cepstral Coefficients, which have been explained in the previous section. As a reminder, they model the spectral energy distribution as it is humanly perceived.

Additionally, the pitch is an important characteristic of a song [90, 95]. If it was defined in terms of the vocal folds in the previous section, the reader should know that this frequency is also computed for instruments. In this context, the pitch is redefined by the speed of vibrations of the instrument and the size of these vibrations. High frequency sounds are produced by fast vibrations. This kind of sounds are generally a good cue of a positive emotion, e.g. happiness. On the contrary, sounds produced by slower vibrations, and thus low-pitched, indicate songs that express negative feelings such as the sadness [154]. The pitch of the instrumental has been computed by a function *PredominantPitchMelodia*, coming from the *Essentia* library [173]. This function is based on the *MELODIA* algorithm [192], which has already been explained in the previous section. The difference with the function that extract the F0 sequence from the vocals comes from the fact that a polyphonic signal is assumed here. Indeed, the instrumentals are a combination of multiple instruments playing at the same time. Technically speaking, the idea behind the algorithm is the same as before, but it is now focused to extract the predominant melody sequence, as it may exist multiple melodies produced by the instruments. During the experiments, the vector of estimated melody pitch has been averaged as a single value in order to be compared with other single-value features.

Easily noticeable to the ears, the rhythm is another important feature for creating emotions among the music listeners too [154]. The tempo, or the speed of the music, plays an important role in the rhythm of a song. It can be identified through extracting a beat spectrum from the audio [71], then determining the frequency (in terms of occurrences) of the beats. A beat is defined as an accented (i.e. stronger or louder) pattern in the spectrum and that is generally repeated periodically over the song. Most of the time, a beat is produced by a drum kick[9]. One way to illustrate this is to imagine the drum pattern of a rock song [9]. Most of the time, assuming a 4/4 time signature[10], the drum kick hits at the first, second and the fourth beat. On the third beat, the drum snare[11] hits. This kind of pattern is easy to detect since the amplitude of the music is louder on beats where kick and snare hit than the ones where drum does not play. One can then determine the length of this 4/4 pattern and determine the number of *Beats Per Minute* (abr. BPM) of the song. As it can be expected, *sad* songs are usually linked with lower tempos, while music related to *anger* have a faster bpm [154]. This information has been extracted by the *Essentia* library [183].

Last but not least, another middle-level feature has been analysed : the *danceability* of the songs, which measures how well can people dance on a song [213]. Computed via a *Essentia* function [46], the algorithm is derived from the *Detrended Fluctuation Analysis* [163]. Due to its relative complexity, details about its method are not covered in this thesis. The output of this algorithm is the danceability of the audio signal, and is expressed as a real number between 0 and 3. The higher the values are, the more danceable is the song.

[8] Musicians sometimes talk about it as the colour of the sound

[9] Part of the drum kit that produces the lowest sounds, and located at the feet of the drummer

[10] The two numbers in the time signature define how many beats are in each measure of the song : 4 beats in this case.

[11] Part of the drum kit that produces a rattling sound

Mood	Intensity	Timbre	Pitch	Rhythm
Happy	medium	medium	very high	very high
Sad	medium	very low	low	low
Calm	very low	very low	medium	very low
Energetic	very high	medium	medium	high

Table 3.1: Examples of different moods and the expected values for their attributes [154]

3.2.3 Vocals and Instrumentals Preprocessing

Regarding the vocal and instrumental analysis, a simple pre-processing has been applied. Indeed each song has been loaded by a *Essentia* function called *EqloudLoader* [62]. This method is working in three steps. After loading the raw audio as mono signal, it is normalized with a *replayGain* filter [186]. It measures the perceived loudness and normalizes it so that each song share the same loudness. This kind of filter is often used by media players so that the listener does not have to adjust manually the volume level between each track. Finally, an *equal-loudness* filter is applied on the audio file [187].

This filter lowers the frequencies to which human ears are less sensitive, and increases the ones for which humans are more sensitive. As a reminder, the human auditory system is sensitive to frequencies from 20 Hz to around 20 kHz. In 1933, Fletcher and Munsion [94] made an experiment : subjects were listening to pure tones at different frequencies and over 10 dB increments. For all the frequencies and intensities, the subjects listened to a reference tone at 1 kHz too. The reason of it was to adjust this reference tone until the listener perceived the same loudness as the test tone. From this experiment were born the *Equal Loudness Contours* illustrated in Figure 3.6 [187]. To fully understand this concept, here is a concrete example coming from these contours. Let's consider the line marked as 60 at 1 kHz. The y value of this line, at 0.5 kHz, is close to 55 db. This means that a 500 Hz tone at 55 dB sounds as loud as a 1000 Hz at 60 dB for a human listener.

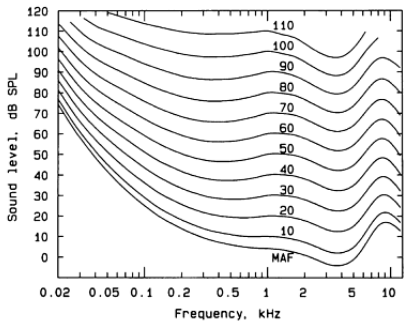


Figure 3.6: Equal Loudness Contours [187]

The higher the curves are, the less sensitive are the ears. As the reader may notice, the humans are hence more sensitive between 2 and 5 kHz. Taken this into consideration, an ideal filter would take the inverse of these lines in order to reduce as much as possible the undesired frequencies. Averaging the curves, an inverse equal loudness curve can be illustrated as the blue line in Figure 3.7. Approximating filters exist, such as the pink line drawn on the same figure. If this filter gives an excellent approximation at high frequencies, it does not at low frequencies. This is why this filter is mixed with another filter (2^{nd} order Butterworth high pass filter [187], green line in the figure) to attenuate the first filter at low frequencies. Once filtered, the middle frequencies are hence boosted and this is where melody can be notably expected to be, making easier the pitch melody extraction.

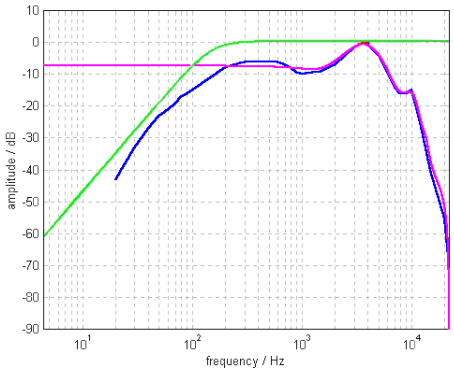


Figure 3.7: The average inverse equal-loudness curve [186]

3.3 Lyrical Analysis

This section examines how to extract features from the lyrics and which are the most relevant in the context of emotion classification. In 2004, the seminal work of Aucouturier and Pachet [2] revealed a "glass ceiling" in spectral-based MER, due to the fact that many high-level (e.g. semantic) music features are not discernible using spectral-only techniques. This showed the opportunity to consider the lyrics of song as relevant source of emotions. This was further motivated by the fact that, during the same year, a survey from Juslin and Laukka [110] mentioned that 30% of people consider the lyrics as a relevant factor of how music expresses emotions. There was born the *Lyrical Music Information Retrieval* field [91, 121].

In 2018, expressing the emotion of a song based on the lyrics was still considered as a challenging task [177]. This can be explained by the fact lyrics have generally a limited vocabulary and some words (e.g. *love*) are often used, but in different contexts e.g. both in a sad and a happy song. Another challenge comes from the music style of the song : lyrics have more or less importance according to the genre of music the song it is related [184]. For instance, lyrics in dance music have less impact than the audio, whereas in poetic music it is the opposite.

The following section provides more details and information about how the lyrical analysis has been led. It explains the preprocessing step that has been applied on the lyrics as the process of extracting relevant features from the lyrics.

3.3.1 Lyrics Preprocessing and Features Selection

The data quality is a first and foremost requirement before running an analysis. This is why the preprocessing of the data, step during which irrelevant or incorrect values are notably removed, is crucial [74]. In the context of Lyrical Music Information Retrieval, this step is even more important since various questions exist about this step. As example, one may wonder if the database should only contain English lyrics or accept other languages [32]. Until this day, there does not exist a common decision on which MER researchers have all agreed. The following paragraphs explain the different processes that have been applied on the lyrics before classification.

Before the classification process, each sentence is split into words and each word is considered as a *token* [137]. If humans would consider tokens such as *The* and *the* as identical, programming languages would not. Converting to lowercase is thus a first mandatory preprocessing step. In the context of this experiment, lyrics have been lowercased before being entered into the dataset.

Secondly, the focus has been set on the *stop* words [225]. They are the words that occur the most in texts and hence don't provide supplementary value to the lyrics. For instance, common *stop* words are *about*, *I* or still *such*. Removing them helps to only analyse content-bearing words but reduces the number of features to consider too. Additionally, punctuation marks have been removed.

Word clouds of different emotion categories can be found in Figure 3.8. They help to understand which are the words that are the most used for a related mood. As the reader can notice, "la la" or "yeah" are common words in lyrics of *happy* songs. However — and that's a reason why LMIR is challenging — some words appear frequently in lyrics labelled to different emotions. This notably the case of the tokens like *know*, *one* or *love*. Furthermore, the fact words are sometimes chosen for their sound or for rhyming makes the classification even more complicated.

Stemming is very important and well-known part of the preprocessing [137, 6]. It consists of converting all the words to their respective stem, i.e. the *root* word [191]. For instance, *loving* and *loved* are different tokens, but they represent the same meaning : the action of *love*. In this thesis, two different stemmers have been tested : the Porter-stemmer [169] and the Snow-stemmer, which is an improved version of Porter [170]. The algorithm behind these stemmers consists of five steps of word reductions, within each one linguistic rules are selected [220].

If stemming identifies and removes the affix (e.g. look-ing) of a word, it must not be confused with the concept of *lemmatization* [220]. Unlike stemming, the lemmatization approach assures

(a) Word cloud of *calm* lyrics

(b) Word cloud of *sad* lyrics

(c) Word cloud of *happy* lyrics

(d) Word cloud of *angry* lyrics

Figure 3.8: Different word clouds before the stemming process

that the reduced word is a word that exists. Whereas the stemming is sometimes defined as a crude heuristic that removes the end of a work, lemmatization is said to be a cleaner approach. Table 3.2 illustrates the difference between stemming and lemmatization.

Removing stop words or still the use of stemming have shown mixed effects in MER classification [101]. In this thesis, we removed the stop words from the lyrics, assuming they are noisy. Regarding the stemming, the classification has been tested with and without it. The lemmatization has not been considered, mostly for time constraints.

Word	**Lemmatization**	**Stemming**
was	be	wa
studies	study	studi
studying	study	study

Table 3.2: Differences between the lemmatization and stemming

To reduce the number of considered words and improve the classification, one may want to discard tokens that are not related to emotions. For instance, the set of features (e.g. words) should better contain tokens such as *bruise* or *fear* than words like *chair* or *door*. This can be achieved by analysing the semantics of the words [26]. Without going into the details of this topic, there exists a simple method to distinguish the words that bear emotional meaning from these that do not. It consists of checking if the words are in a list of affective words, i.e. a sentiment lexicon [131]. The lexicon that we used was provided by [181].

During the experiments, a *tf-idf* approach has been considered too, but it didn't show any improvement. If the reader wants to know more about this approach, a full description of this method is given in Appendix C.

Applying or not these multiple processes leads to consider a different number of words for the classification. These different number of words are shown in Table 3.3. As it can be observed, applying a stemming (both Porter and Snowball) reduces the number of words, which is logical. Only considering words included in a sentiment lexicon helps to reduce the number of words too : it discards around 2500 words. The influences of these multiple processes are given and explained in the *Experiments and Results* Chapter.

Applied processes	**Number of words**
no stemming	4029
Porter stemming	3281
Snowball stemming	3275
no stemming + sentiment lexicon	711
Porter stemming + sentiment lexicon	609
Snowball stemming + sentiment lexicon	605

Table 3.3: Number of considered words for the lyrics-based classification

CHAPTER 4

THE DATABASE

In Music Information Retrieval research, the databases on which experiments are made are one of the central interrogations. The major time of this thesis has been dedicated to the creation of a reliable database. At this date, it does not exist a common database that gained general acceptance in MER field. Most existing works compile their own database, for multiple reasons. For instance, works may focus on only one type of songs, e.g. jazz, or on songs sharing one same language, i.e. monolinguistic [32]. Another reason that prevents from using a common shared database is a problem that has already been mentioned earlier : how emotions should be represented and how many emotions should be taken into account ? The fact this work is unprecedented justifies the creation of a brand-new database too.

Depending on the emotion representation, researchers proposed different databases. Assuming a categorical approach, they based their work on a emotion taxonomy of three, four, six or more emotion classes [227, 70, 238, 245]. The reader should remember that due to inherent subjectivity of music perception, there is not an universally accepted number of mood categories. Once this choice has been set, one must attribute an emotion for each song : should it be made by experts or music listeners ? [110] In 2007, an analysis of Audio Mood Classification has revealed 30% of discrepancy regarding the expressed emotion among listeners [88]. This can give a simple glimpse about the difficulty of creating a relevant MIR database.

This chapter details, in its first section, which databases have been considered for this research and how the databases of this thesis have been created. The Section 4.2 is dedicated to explaining the multiple challenges and difficulties that have been encountered during the creation of these databases.

4.1 Data Collection

Since exploring separately the “emotionality” of the voice, the instrumental and lyrics is unprecedented, there does not exist a database that gathers such data. However, there exist accessible databases and platforms that provide interesting data such as the metadata or the lyrics of the song [144, 209, 14].

To fully understand the process of collecting the necessary data, the rest of this section is organized as follows. The first Section 4.1.1 explains where the data have been found and extracted. Information about the plateforms that have been mentioned above is given in this section too. How the audio files and the lyrics have been collected is described in 4.1.2. This section ends up by explaining how the vocals have been isolated from the instrumentals, and how the songs have been labelled with their related emotions.

4.1.1 Metadata, Audio Files and Lyrics Collection

As said previously, it does not exist a dataset that gained general acceptance. However, some ones stood out from the others and are more often used. For instance, it is notably the case of the *Million Song Database*, largely used for MIR tasks [14]. For its well-known reputation and ease of use, this is one of the datasets we used for this thesis.

The *Million Song Dataset* (abr. MSD) is provided by The Echo Nest[1]. It is defined as a "freely-available collection of audio features and metadata for a million contemporary popular music tracks." [14]. In other words, the dataset gathers metadata and some features analysis for one million songs. One may precise that it does not include any audio nor emotions. Due to the large set of this set (originally 280 GB) only a subset of it, i.e. 10,000 songs (1% of the global dataset), has been considered. This subset is also provided by The Echo Nest, and songs are said to be collected randomly from the original MSD. The Table 4.1 shows some of the 45 fields associated with each track in the database.

Features	**Value**
artist_mbid	db92a151-1ac2-438b-bc43-b82e149ddd50
artist_name	Rick Astley
title	Never Gonna Give You Up
danceability	0.0 (which means not analysed)
loudness	-7.75
year	1987

Table 4.1: Example of some features for a track in the MSD

By looking at the features which have been already computed in the MSD, one may ask the interest of having audio files if some of their characteristics have already been computed. The reader should remember that these features have been computed on the whole song and not on the instrumental nor isolated vocals. Downloading the audio files to recompute the different features respectively for the vocals and the instrumental is thus necessary.

After collecting information about 10,000 songs, one must obtain the corresponding audio files. To achieve this, a process of two steps has been followed. First the *Spotify* API [209] provides, given the title and the author of a song, a 30 seconds preview for a song given the name of the artist and the song. Only analyzing 30 seconds of a song for the vocals and instrumental parts is a very common practice in MER, and has been proved to be a good practice [127]. Experimental results show that gives better performance for detecting the sentiment of the song rather than the whole song [214]. However, the preview is given in the form of an url, which is not directly downloadable and thus not analysable. Therefore a second step is required : converting each url into a *wav* file. This step has been achieved by an online converter [157].

As said previously, the Million Song Dataset does not include emotion tags in the set itself. To tackle this problem, a common practice consists of linking the songs with another data source, *Last.fm*, which collects the corresponding tags [144, 88]. It is the official website that groups song tags and song similarities of the Million Song Dataset. These tags can be extracted from the online website via the *Last.fm* API or by the corresponding offline database. In this thesis, the second option has been opted. The database contains two columns : one for the unique song identifier, and one for the corresponding tag(s). The fact the MSD also provides this unique id allows to connect the two databases. Since the tags are subject to some discussion, Section 4.1.3 is dedicated to giving more details about them and their related emotions.

Regarding the lyrics, multiple data sources exist such as *musiXmatch* dataset, which is the official lyrics collection of the Million Song dataset [150]. The lyrics are gathered in a bag-of-words format [137] : the lyrics are defined as a binary vector, where each element of this vector is set a 1 if it is present in a most used 5,000 words dictionary, 0 otherwise. Beyond the fact it is probably easier to use for multiple purposes, distributing lyrics under this format avoids copyright issues

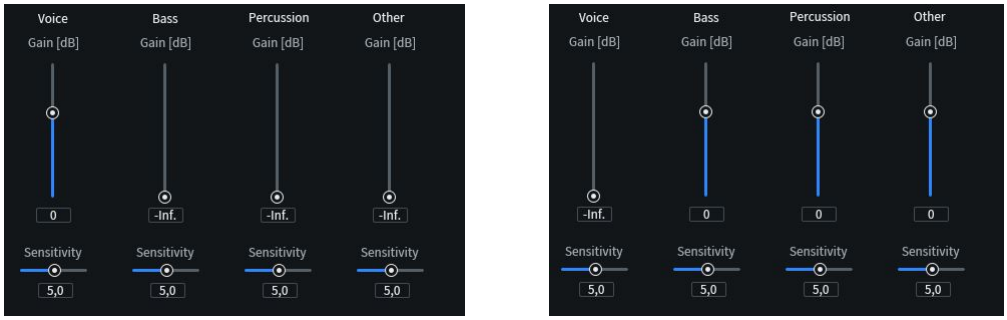
[1]Music platform for developers, which was originally a research spin-off from the MIT Media Lab to understand the audio and textual content of recorded music

too [150]. This database has been used in the beginning of the thesis, but since it removed some freedom (e.g. it only considered 5000 imposed words) it was finally dropped. Another database has been thus used : Lyricwiki.org [135]. It is an online database, which is well-known for its broad coverage and classical format [140]. Song title, artist must be given to correctly identify the lyrics.

4.1.2 Isolating Instrumental and Vocals

Once the audio files and lyrics have been collected, one must separate the vocals from the instrumental. This is a problem known as *source separation* problem [39]. Unfortunately easy-to-use algorithms don't exist for a such task and the quality of the results varies [12].

Hopefully there exists a well-known software, used both by NLP scientists and music producers, called RX7[2] [105, 12]. It provides a powerful interface for audio experiments and it is often referred as an excellent software for noise removal [195]. However, using this software in the context of Music Information Retrieval is unprecedented. This software has been chosen because it provides a *Music Rebalance* option, which is particularly useful for this source separation problem. Once the original audio file is loaded in the software, this option proposes multiple modifications of the audio, as shown in Figure 4.1 : tweaking the loudness of the voice, bass and so on. In our case, two setups were important : either isolating the voice, either remove it. This infers the following adjustments shown in (a) and (b). The *Gain* option is used to adjust the level (dB) of the voice/bass/percussion or the others. The *sensitivity* rate defines the percentage of the input signal considered as e.g. Voice by the separation algorithm. Set at low values, this induces the separation algorithm to narrowly define the vocal content of the signal. If the resulting signal contains less audio bleed from the other instruments, this reduces the vocal clarity due to its sharpened approach. Higher values help to reduce this lack of clarity but consider more bleeding from the other elements. For this work, *sensitivity* has been set to its default value : 5,0.



(a) Parameters to isolate the acapella

(b) Parameters to remove the vocals

Figure 4.1: The two different operations during the source separation process

4.1.3 Identifying Mood Categories

As said previously, the perception of emotion is subjective. This means that one song may be considered as *calm* for a person, while another person would judge this song as *happy*. This, once again, proves the challenging difficulties the MIR field must deal with. This arises the question of "how to correctly label a song ?". This is even more complicated by the fact that perceived emotions are influenced by the mood of the listener and the place he listens to music [102]. Platforms such as *All Music Guide* have invested a lot of time, money and human resources to annotate their music databases with high-quality emotional tags [4, 114]. For this reason they are unfortunately unlikely to share their data with the MIR research community.

Hopefully, there exist free approaches to collect emotion annotations from human listeners. One may cite *Last.fm*, which has already been mentioned above. This website allows users to associate social tags to songs through their audio player interface. In 2007, no more than 960,000 different tags were identified and used for annotate songs [145]. This is the dataset that has been used to

[2]manufactured by iZotope

collect the tags and determine the emotion of the songs.

Still, as the reader can expect, this dataset of tags is not deprived of weaknesses. For instance, it does not always include direct useful or desired emotion-related tags, such as *sadness* or *happiness*. It provides a range of different tags, which may be not relevant in the context of sentiment classification. Examples of irrelevant tags are related to a non-affective aspect ("beat", "trance") or are a matter of personal taste (e.g. "bad", "good"). Others tags may relate the style of the songs or the period of release (e.g. "80s"), which are not useful in this context neither. Besides the fact that social tags contain irrelevant information, there exist ambiguous tags that can cause confusion. This is notably the case of the "love" tag. Is the song about love or did the annotator mean he loved the song needs to be answered.

Table 4.2 shows the most 10 frequent tags that can be found in the *Last.fm* dataset. It confirms the idea that social tags can be noisy and not directly emotion-related. The fact the "love" term is ranked 5^{th} in the top 10 most frequent tags can give some clues about its ambiguous meaning too.

This shows the need of some clean-up. For this reason junk tags and tags with no affective meanings have been filtered out. A linguistic resource, *WordNet-Affect*, has been particularly helpful for this task [212]. It is an extension of WordNet, which is defined as "a large lexical database of English nouns, verbs, adjectives and adverbs grouped into sets of synonyms i.e. synsets" [143, 244]. The term *synset* is used to represent a distinct concept. WordNet-Affect, its extension, provides a powerful support by linking each non-noun or noun tag to the emotional synset it refers to. Hence, it helps to filter tags which bear an emotional meaning from the non-interesting tags. For instance *WordNet-Affect* would output, given the tag "sadly", the synset "sad" whereas it would output nothing for a "80s" tag.

Tag	**Total count**
rock	101,071
pop	69,159
alternative	55,777
indie	48;175
electronic	48,175
female vocalist	42,565
favorites	39,921
Love	34,901
dance	33,618
00s	31,432

Table 4.2: Top 10 most frequent tags in the *Last.fm* dataset

The junk tags being filtered out, the useful tags have been converted into emotional tags. However, by looking closer at some of them, it existed some emotional tags that didn't represent distinguishable meanings. In fact, many of them were synonyms and needed to be grouped together. Hence the 150 remaining emotional tags belonging to and being derived from the same upper synset in WordNet-Affect were grouped together. Due to its tree-based structure, Wordnet-Affect can easily find the "upper" emotional concept of an emotion. The reader can find an illustration of this tree in Appendix A, in Figure A.3. As result the tags were merged into 19 groups, as shown in Table 4.3.

For the classification experiments, each category should have enough samples to build a reliable model. This is why categories with fewer than 20 songs were dropped, resulting in a database labelled with 15 emotional tags categories. The categories that have been dropped are highlighted in blue in the Table 4.3.

n°	Emotional categories
1	calm, comfort, quiet, serene, mellow, chill out, chill
2	sad, sadness, unhappy, melancholic, melancholy, misery,...
3	happy, happiness, happy songs, happy music,...
4	romantic, romantic music, affection, tender, love,...
5	upbeat, gleeful, high spirits, zest, enthusiastic,...
6	depressed, blue,dark, depressive, dreary,...
7	anger, angry, choleric, fury, outraged, rage, hostility, jealousy,..
8	grief, heartbreak, mournful, sorrow, sorry,...
9	dreamy, captivation, admiration, awe, worship,...
10	cheerful, cheer up, festive, jolly, jovial, merry,...
11	brooding, contemplative, meditative,...
12	aggression, aggressive, belligerence, bang,...
13	confident, encouraging, encouragement, optimism, confidence,...
14	angst, anxiety, anxious, jumpy, nervous,...
15	earnest, heartfelt, earnestness,...
16	desire, hope, hopeful
17	pessimism, cynical, pessimistic, weltschmerz,...
18	excitement, exciting, exhilarating, thrill, ardor,...
19	shamefacedness, confusion, shame,...

Table 4.3: The 19 emotional categories resulting from the Last.fm analysis

4.2 Workflow and Related Difficulties

As the reader may have understood until this point, a lot of challenges and ambiguity exist around the construction of a ground truth data. This section explains the multiple difficulties that have been encountered during this process and the multiple decisions that have been consecutively taken. As shown in Figure 4.3, the workflow of adding one reliable song in the database is particularly subject to various considerations.

Starting from the metadata provided for the 10,000 songs of the subset of the *Million Song Dataset*, the first question one should answer is if the song is linked to some tags from the *Last.fm* dataset. If not, the song is not taken into account. If yes, however, one should check if the tags can reveal emotions. As said in Section 4.1.3, they can contain no useful information or being not convertible into an emotion. In this case, the song is neither considered. Thirdly, the song has to pass the *Spotify* test. Indeed, the streaming platform does not provide a thirty seconds preview for every song, which can void the previous steps. If the *url* preview has been provided, then it is converted into a *wav* file thanks to the online converter. This step is time and memory-consuming.

After reception of the downloaded audio file, one may want to know if the 30 seconds of song contain a vocal part or if it is a pure instrumental song. This can be achieved by listening to the track, or, more scalable, noticing the 13 computed MFCC of the voice are all set to 0. One way to avoid this step should have been to check if the song had lyrics after *Last.fm* tags step. But it is only possible for the songs that have no lyrics. Even though a song contains lyrics, Spotify can provide an 30 preview with no vocals and it is not possible to detect it before analyzing the audio file itself.

Once this has been checked, the audio file is then processed by the RX7 software in order to separate the voice and the instrumental. Unfortunately, multiple conditions reduce the capacity of perfectly extracting the vocals from the song. For instance, the frequencies of the instruments, e.g. guitar or drums, may be intertwined with the frequencies of the voice, which complicates the extraction and makes it less satisfying. Furthermore, as it has already been mentioned previously, some production processes such as the mastering make this separation even more complicated. As result, it implies an audio file containing bleeding from other instruments, and vocals sound muffled, i.e. are not correctly distinguishable. This implies miscalculations of the related features. During the construction process of database, around 2000 songs have reached this step.

However, a major weak point had not been discovered yet. Indeed, during the first classification experiments, the accuracy rate and other metrics were very low. For this reason, the database has been rigorously inspected, revealing a major drawback of the *Last.fm* database : the songs are not correctly labelled with emotion they capture. Some songs, capturing a *happy* emotion, were labelled as *angry* for instance. The fact all the listeners on *Last.fm* can associate songs with the tags they want may give some cues about the fact that the annotation can be wrong. This conclusion confirmed the idea that either one must pay for a correctly emotion-labelled set of songs or one must recruit emotions experts to annotate songs carefully. These two options being unrealistic in the context of this thesis, re-evaluating these two thousand songs to ensure all of them were correctly labelled required unfortunately too much human labor, subjectivity and time.

For these reasons, we decided to reduce the dataset about 300 songs, being thus humanly analysable. If the number seems to be low at first glance, this can be nevertheless supported by many works [101]. Indeed, MER experiments are rarely evaluated against databases of more than 1,000 music pieces. The labelling step being chaotic, we decided to reduce the set of considered emotions to four distinguishable emotions. They are the *angriness*, the *happiness*, the *calmness* and the *sadness*. The choice of these four emotions has been based on two assumptions. First, songs can be almost objectively annotated with these emotions. Indeed, a *metal* song can be, regardless some subjectivity, classified as *angry*. The second motivation is that one can expect that these four emotions have relatively distinguishable features values. Values for a *happy* song can be expected to be opposed to values of a *sad* song. The same expectation can be assumed for *calm* songs and *angry* songs. Additionally, the percentage of songs related to each emotion has been maintained even (see Figure A.4 in Appendix A).

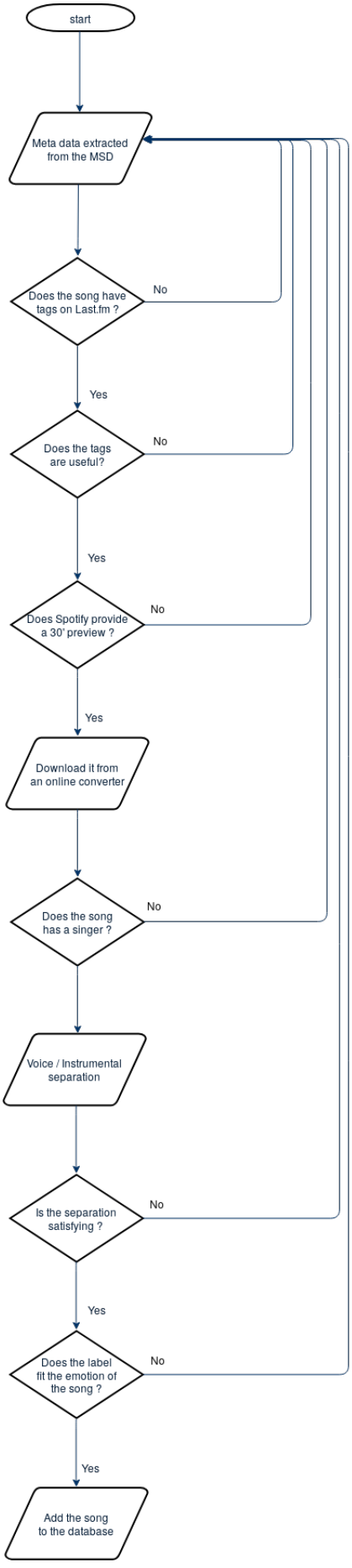


Figure 4.2: Workflow for adding a song in the database

CHAPTER 5

MACHINE LEARNING CLASSIFIERS

Now that all the concepts related to the Music Emotion Recognition field have been introduced, let's focus on the classification *per se* and the machine learning principles that are behind it. This fifth chapter is devoted to explaining the two different models that have been considered for the classification, namely the *Random Forest* [30, 84] and the *Logistic Regression* [19, 103].

After introducing some notions of machine learning in Section 5.1, both models are respectively described in the following Section 5.2 and Section 5.3. For each one of them, the mathematical concepts and fundamentals are detailed. Which hyperparameters have been considered and how their optimal values have been found are covered too.

5.1 Multiclass Supervised Classification

The type of classification that has been considered during this thesis is a *multiclass supervised classification*, topic widely covered in the machine learning literature [3, 216, 19, 84]. Before explaining what *multiclass* and *supervised* mean, a definition of classification must be given.

The classification aims at predicting one categorical output related to an object, i.e. *example*, defined by multiple features. Mathematically speaking, it is equivalent to define a function that assigns a class label y among a set of possible labels Y, to a vector of features $x \in X$, the space of all possibles features. This function f can be modelled as :

$$f : X \to Y : x \to y \tag{5.1}$$

In the context of MER, a single music sound may be characterized by more than one label, e.g. both "dreamy" and "cheerful". In this thesis we assumed a *multiclass* classification, which means there exist multiple possible outputs for each example, but only *one* is assigned. For various reasons that have been mentioned in Section 4.2, four possible labels have been considered. They are *happy, sad, angry* and *calm*. The function f that has been mentioned above can be redefined as follows :

$$f : X \to \{angry, calm, happy, sad\} : x \to y \tag{5.2}$$

The learning algorithm behind the classification needs data in order to define the function f. Broadly speaking the data can be defined a set of n *examples*, represented by a vector of attributes x and their related label y. This representation is shown in Equation (5.3). In our case, each x_i can be composed by audio features whereas the y_i is the corresponding expressed emotion. The

fact the output is already defined for each example before the classification refers to the principle of *supervised* learning.

$$\{(x_1, y_1), ..., (x_n, y_n)\} \tag{5.3}$$

A common way to train and evaluate the classification algorithm is to split the whole dataset into two subsets, i.e. a *train* set and a *test* set [84]. The train set is the one on which the algorithm designs the prediction function whereas the test set is used to evaluate this function. There exist multiple ratios to split on. For instance, common ratios are 75%-25%, 80%-20% or 90%-10% respectively for the training and test set [82]. In the experiments, we opted for a ratio of 75%- 25%.

Another approach to train and evaluate the learning algorithm consists of using the k-fold cross validation technique [84, 202]. In a k-fold cross validation, the data is split into k equal-sized subsets. One is used as a test set, whereas the *k-1* other folds define the training set. This step is repeated k times in order to use each fold as a test set. The final accuracy is the average of the k results. This approach is often used when the amount of data is limited, since it can consider all the data for the training step [83]. An illustration of a 10-fold cross validation is given in Appendix A in Figure A.5.

For each classifier, are defined parameters and hyperparameters. The parameters are defined as configuration variables that are internal to the model. Their values define the performance of the model and are learned from the dataset. Hence they are often not set manually by the practitioner. Examples of model parameters include weights for a neural network or still coefficients in a logistic regression [103]. They should not be confused with the model *hyperparameters*, which are external variables of the model. This means their values are not estimated from data. These model properties are fixed before the model is trained or tested. Their values are generally specified by the practitioner or found by heuristics.

There exist two strategies to find the best values for hyperparameters [67]. The first one is called the *grid search* [11]. Given a set of values for each hyperparameters, it evaluates the model with each combination of the values of the hyperparameters. As the number of hyperparameters and values can be huge, the grid search can be computationally expensive, i.e. exponential. To tackle this problem, one may use the second approach, namely the *random search*. It is a technique based random combinations of values of the hyperparameters, which are tested to find the best solution for building the model. This approach has been proved, empirically and theoretically, to be more efficient for hyperparameters optimization than *grid search* [11]. In this thesis, both approaches have tested to find optimal values of hyperparameters.

5.2 Random Forest Model

The first model we introduce is the *Random Forest* model [30]. It is an easy-to-use machine learning algorithm that has been proven to be extremely successful in multiple fields [232, 73, 210]. It is one of the most used learning algorithms, thanks to its simplicity and its related properties.

The first Section 5.2.1 describes these properties and what are the fundamentals of the model. The second Section 5.2.2 presents the parameters of random forests, as well as their hyperparameters.

5.2.1 Fundamentals of RF

The Random Forest model relies on two principles : the *decision trees* [211, 202, 175] and the *ensemble methods* [54, 84].

First, the decision tree is a class of machine learning models known for its easy interpretation and clarity of information representation. It is defined as a rule-based classifier based on a tree-like structure, where each internal node tests an attribute and each branch corresponds to a value of this attribute [57]. The leaf nodes represent the corresponding labels of the classification. Figure 5.1 illustrates an example of such model. It illustrates a tree that decides whether one should play tennis based on weather's features [146]. For instance, if the *outlook* is *rain* and the *wind* is *strong*,

one should not play since the leaf node of this conjunction of attribute values is *no*. From a global point of view, the whole tree can be seen as a disjunction of conjunctions.

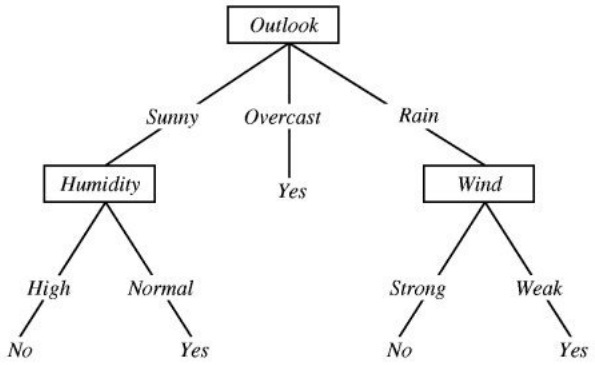


Figure 5.1: Example of a decision tree [146]

ID3 is the initial algorithm invented by Ross Quinlan and widely used to generate decision trees from a dataset [174]. Algorithm 1 gives the pseudo-code of ID3. By reading the first line of code, one may ask how to select the *best* attribute ?

Data: A training set $S = \{(x_1, y_1, ..., (x_n, y_n)\}$
Result: A decision tree based on the set S
Init: Single node tree with all training examples assigned to it
$x_j \leftarrow$ the "best" decision attribute for current node ;
assign x_j as decision attribute for node ;
for ***each*** *value of* x_j **do**
  create new descendant of node ;
end
Assign training examples to the leaf nodes ;
if *training examples are perfectly classified* **then**
  return tree ;
else
  run Main loop over new leaf nodes;
end

Algorithm 1: ID3 pseudo-algorithm [57]

The answer is found by using *The Theory of Shannon* [204]. To fully understand the concept, the idea of *entropy* must be introduced first [204]. The entropy measures the uncertainty of a set of examples S. Inversely, it highlights the purity of this set. The scale of this measure is defined between 0 and 1. A high entropy means a high level of disorder, i.e. a low level of purity. Let S be the sample of training examples at one node, c the number of different classes and p_i is the probability of an example of class i. The entropy $E(S)$ is mathematically defined as :

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 \left(p_i\right) \tag{5.4}$$

For each iteration of the algorithm, the algorithm tests every unused attribute x_j and calculates the information gain $Gain(S,\, x_j)$ for the set of examples S. This measures the expected reduction in entropy due to sorting on this attribute. Since one wants to reduce the entropy, ID3 selects the attribute which has the higher gain. Mathematically speaking, $Gain(S,\, x_j)$ of the attribute x_j is defined as :

$$Gain(S, x_j) = E(S) \;\; - \sum_{v \in Values(x_j)} \frac{|S|}{|S_v^j|} E(S_v^j) \tag{5.5}$$

where

$$S_v^j = \{x \in S | x_j = v\} \tag{5.6}$$

However, ID3 algorithm suffers from some limitations [202, 206]. For instance, the model may be overfitted, which happens when it corresponds too closely or exactly to a particular set of data [84, 202]. Furthermore, it can't handle continuous values efficiently [57]. It can't neither handle training with missing attribute values nor attributes with differing costs.

Therefore the *C4.5* algorithm was introduced, as an improvement of its precursor and fixing the previous problems [30]. The overfitting problem is solved by the fact C4.5 prunes trees after creation, which was not done with ID3 [84]. Pruning is the action of reducing the size of decision trees by removing sections of the tree that does not provide a lot of information for classifying the examples.

A third algorithm for decision tree may be used, namely the *CART* algorithm [28, 84]. It relies on the *Gini* impurity principle, which measures the homogeneity of the nodes [84]. A node is said to be pure if all examples assigned to this node have the same class label. The decision tree finds the features to separate the data at its best by minimizing the *Gini* impurity. Let the sample S containing C different classes, and $p_S(y)$ denotes the proportion of examples labelled y in S. Splitting S with an attribute x_j results in a drop of impurity expressed as :

$$Drop(S, x_j) = Gini(S) \ - \sum_{v \in Values(x_j)} \frac{|S|}{|S_v^j|} E\big(S_v^j\big) \tag{5.7}$$

where

$$Gini(S) = 1 - \sum_{y=1}^{C} p_S\big(y\big)^2 \tag{5.8}$$

The second concept that defines the random forests is related to *ensemble methods* [54, 84]. An ensemble method is defined as a combination of multiple classifiers whose predictions are combined in order to improve the overall performance and provide a better accuracy than a single classifier would achieve. Within ensemble methods there exist two main categories [29]. The first one, called *bagging* or *bootstrap aggregating*, splits the training set in subsets and uses each set for each classifier. The second approach, namely *boosting*, attributes the subset for each classifier according the previous classifier : instances that were not correctly classified in the previous classifier are prioritized in the following classifier to improve their predictions [190]. Bagging helps in reducing the risk of overfitting, while Boosting can generate a combined model with lower errors. Random Forests are based on the first approach, whose pseudo-code is given by Algorithm 2.

Data: A label set $S = \{(x_1, y_1, ..., (x_n, y_n)\}$
A total number B of bagging rounds
A learning algorithm *Alg* returning a class. $X \rightarrow \{1,..,N\}$
Result: A combined classifier
for $b \leftarrow$ *1 to B* **do**
  S_b = Resample(S) ;
  f_b(x) = Alg(S_b) ;
end
return $\frac{1}{B} \sum_{b=1}^{B} f_b(x)$)

Algorithm 2: Bagging algorithm [29]

The random forests combine the two above concepts. It uses an arbitrary set of decision trees and defines the predicted class of an example based on a majority vote from all the results of the different trees. By averaging them, it significantly reduces the risk of overfitting and outperforms the learning algorithm as it was applied once on the set. However, one disadvantage is that using a large amount of trees can be very computationally heavy and ineffective. Using a large number

of RF reduces the capacity of visualizing and understanding how the model works too.

Regarding the process of each tree, the attribute which is chosen to be split on is the one that maximizes the impurity (*Gini index* or *Information Gain*) among a random selection of v attributes out of the d possible attributes. Values of v may be between 1 (then one attribute is chosen and no computation of impurity is needed) and d. Typical values are $v = \log_2(d)$ or $v = \sqrt{d}$ [76]. The randomness selection is important to avoid the correlation between the decision trees : if one or a few features are very strong predictors for the response variable (target output), these features would be selected in many trees, causing them to become correlated.

5.2.2 RF Hyperparameters

Here are discussed the hyperparameters that can used to custom the random forests [30]. Since their values are in most of the cases defined empirically, the reader should know there are no reference values : it mostly depends on the data on which the classification is based. We based our approach on multiple machine learning blogs, such as [166]. Technically speaking, we used the implementation of random forests provided by Scikit-learn [200].

The first hyperparameter that should be introduced is the number of trees that defines the forest. It is referred as `n_estimators` and set at few hundred to several thousand trees, depending on the size and nature of the training set. In general, a higher number of trees increases the performance and makes the predictions more stable, but it also slows down the computation. In this thesis, the following values have been tested : 100, 200, 300, 1000.

Regarding the splitting `criterion`, the two approaches that have been previously presented have been tested. As a reminder, there are the *Information Gain* and *Gini Impurity*.

Another hyperparameter that has already been mentioned is the number of considered features when decision trees need to split. It is defined as `max_features`. As we mentioned above, it can be defined by taking the square root of the total number of features or the binary logarithm of it. Both approaches have been tested.

The hyperparameter `max_depth` is used to define the maximum depth of the trees. It can be set to $None$, specifying the nodes are expanded until all leaves are pure, i.e. only containing instances from the same class. We tested the following values : 80, 90, 100, 110 and 1000.

The `min_samples_split` defines the minimum number of samples needed to split an internal node. If the default value is set at 2, other values such as 3, 4, 5 have been tested too. Similarly, the `min_samples_leaf` hyperparameter refers to the minimum number of samples required to be at a leaf node. We tested a range of values between 1 and 5.

The random forests have more hyperparameters than the six ones mentioned above. For instance, one may cite `min_weight_fraction_leaf` or `max_leaf_nodes`. They have remained unchanged during the classification experiments. There exist other hyperparameters, but they are not purely related to the random forests. One may cite `n_jobs`, which can make the model faster. It tells the computer how many processors it is allowed to use. A value of 1 means the use of one processor, a value of -1 meaning there is no limit. Another — and very important — hyperparameter is `random_state`, which is the seed used by the random number generator. Passing a specific seed makes the model's output replicable. In other words, the model will always produce the same results when it has a definite value of `random_state`.

5.3 Logistic Regression Model

Let's now focus on the second model : the *Logistic Regression* model [84, 103]. It has become a standard method of analysis in many fields [103]. The goal of this model is to explain the relationship between one dependent binary variable and multiple independent variables.

The first Section 5.3.1 describes the fundamentals of the model as well as its underlying concepts. The second Section 5.3.2 presents the hyperparameters of a logistic regression and which values have been considered to optimize them.

5.3.1 Fundamentals of LR

As stated before, a logistic regression is a mathematical modeling approach that can be used to describe the relationship between several variables X's and a dependent variable Y. Before explaining the logistic regression with multiple classes, let's first introduce the binary logistic regression, i.e. when it only considers two classes : 1 or 0. Let x be an observation, the probability of x to belong to the particular class $y = 1$ can be formulated as :

$$p(x) = P(Y = 1|X = x) \tag{5.9}$$

Expressing this a posteriori probability $p(x)$ as linear combination of explanatory variables (i.e. features) is however not possible :

$$p(x) = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n \tag{5.10}$$

Indeed, if the right-hand side of Equation (5.10) is theoretically defined between [-∞, +∞], the probability $p(x) \in [0, 1]$. The Figure 5.2 illustrates this problem and the distinction between both models.

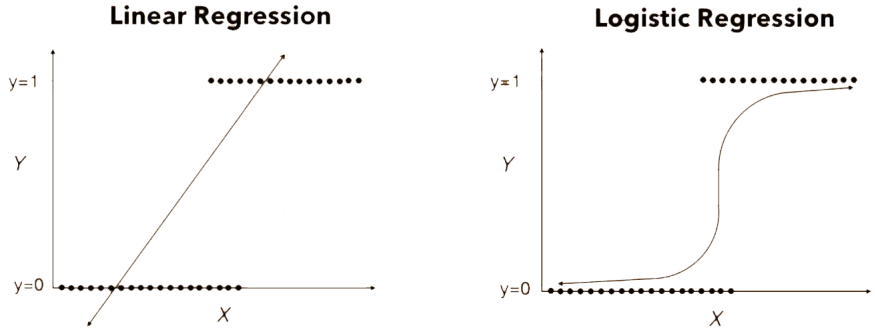


Figure 5.2: Difference between linear regression and logistic regression [5]

As we can see, we need a *S-shaped* curve to correctly predict the output. One way to solve this is to model $p(x)$ by using a *logistic* function [103]. Most of the time, the logistic function is defined as a sigmoid function, which is a special case of it [103, 84, 239]. Let x be a number, then the logistic function $f(x)$ can be described as :

$$f(x) = \frac{e^x}{1 + e^x} \tag{5.11}$$

To express $p(x)$ as function of $\beta_0+\beta_1 x_1+...+\beta_n x_n$, one must apply a logit transformation, expressed in Equation (5.12) [103, 84]. This corresponds to the logarithm of the odds of the observation x. The odds are defined as the probability of this event occurs, divided by the probability that this event does not occur. Taking the logarithm of the odds gave the term *logit*.

$$logit(x) = \ln\left[\frac{x}{1 - x}\right] \tag{5.12}$$

We can thus rewrite Equation (5.10) as :

$$logit(p(x)) = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n \tag{5.13}$$

Once isolated, $p(x)$ can be expressed as in Equation (5.14) , which corresponds to a logistic function $f : \mathbb{R} \rightarrow [0,1]$:

$$p(x) = \frac{e^{\beta_0+\beta_1 x_1+...+\beta_n x_n}}{1 + e^{\beta_0+\beta_1 x_1+...+\beta_n x_n}} \tag{5.14}$$

That being said, one needs to estimate the unknown parameters β_0, ..., β_n. In linear regression, the most used method is the *least squares* approaches, which minimizes the sum of squared deviations of observed values from the predicted values [84]. In the context of logistic regressions the approximation of these coefficients, namely $\hat{\beta_0}$, $\hat{\beta_1}$, ..., $\hat{\beta_n}$, is calculated by the method of *Maximum Likehood* [103, 84]. This method is based on the *likehood* concept. Assuming a binary classification, the Maximum Likelihood method estimates the parameters β=$\{\beta_0,...,\beta_n\}$ such that the probability $p(y = 1|x)$ for the observations x belonging to y=1 is as close to 1 as possible and vice-versa for the observations x belonging to y=0.

Assuming N observations denoted x, the likelihood $L(\beta)$ is mathematically defined as follows :

$$L(\beta) = \prod_{i=1,y=1}^{N} p(x_i) \prod_{i=1,y=0}^{N} 1 - p(x_i) \tag{5.15}$$

Instead of calculate this product, a common practice consists of calculating the logarithm of this equation [103, 202]. The resulting equation is defined in Equation (5.16) and is referred as the *log-likelihood*. The intermediary equation between (5.15) and (5.16) can be found in Appendix A, in Equation (A.1).

$$\ell(\beta) = \sum_{i=1}^{N} y_i \log\big(p(x_i)\big) + (1 - y_i) \log\big(1 - p(x_i)\big) \tag{5.16}$$

To solve it, its derivatives are set to zero. Numerical methods are commonly used to solve it, such as the *Newton-Raphson* algorithm [84].

$$\frac{\partial \ell(\beta)}{\partial \beta} = 0 \tag{5.17}$$

Once the coefficients have been replaced by their estimators $\hat{\beta_i}$, the output y of an observation $x = (x_1, ..., x_n)$ is given by the following rule :

$$y = \begin{cases} 1 & \text{if } \hat{\beta_0} + \hat{\beta_1} x_1 + ... + \hat{\beta_n} x_n > 0 \\ 0 & \text{if } \hat{\beta_0} + \hat{\beta_1} x_1 + ... + \hat{\beta_n} x_n \leq 0 \end{cases} \tag{5.18}$$

In terms of probability, it is equivalent to :

$$y = \begin{cases} 1 & \text{if } p(x) > 0,5 \\ 0 & \text{if } p(x) \leq 0,5 \end{cases} \tag{5.19}$$

Similarly to random forests, the logistic regressions can be subject to overfitting [84]. To tackle it, a common practice is to use a regularization [103]. Before introducing this concept, let us first define the cost function J(β) of this model. Maximizing the *log-likelihood* is equivalent of minimizing its averaged inverse [53]. The Equation (5.16) can be thus redefined as :

$$J(\beta) = -\frac{1}{N} \sum_{i=1}^{N} \Big(y_i \log\big(p(x_i)\big) + (1 - y_i) \log\big(1 - p(x_i)\big) \Big) \tag{5.20}$$

The first regularized model is the *ridge regression* [84, 185]. It adds the squared magnitude of the coefficients as a penalty in the cost function, as highlighted in blue in (5.21). The λ term is

defined as the regularization factor. Set at zero, there is no regularization. If it is large, then it adds much weight and it leads to underfitting [185]. The value of this hyperparameter is discussed in the following section.

$$J(\beta) = -\frac{1}{N}\sum_{i=1}^{N}\Big(y_i \log\big(p(x_i)\big) + (1-y_i)\log\big(1-p(x_i)\big)\Big) + \lambda\sum_{i=1}^{N}\beta_i^2 \tag{5.21}$$

The second approach is named the *lasso regression* [84, 19, 185]. It adds the absolute value of magnitude of the coefficients as a penalty. Its equation is given in Equation (5.22), where the penalty term is highlighted in blue. This regularization can lead to zero coefficients. In other words, it helps to reduce the overfitting problem, but it does a feature selection by setting some coefficients at zero [185]. The value of this hyperparameter is also discussed in the following section.

$$J(\beta) = -\frac{1}{N}\sum_{i=1}^{N}\Big(y_i \log\big(p(x_i)\big) + (1-y_i)\log\big(1-p(x_i)\big)\Big) + \lambda\sum_{i=1}^{N}|\beta_i| \tag{5.22}$$

So far, we only considered a binary logistic regression. But, in the context of this thesis, there exist multiple classes since we consider four emotions. There are different methods for generalizing a two-class classifier. The one that have been used in for Logistic Regression is the *one versus all* approach [84].

Let's assume C different classes. According to this method, C binary classifiers are trained, each one separating one class and all the other ones. Given a training set $S = (x_1,y_1),..., (x_n, y_n)$, C different training sets $S_1,...,S_C$ are built, where $S_i = \big(x_1, (\text{-}1)^{y_1 \neq i}\big),...,\big(x_n, (\text{-}1)^{y_n \neq i}\big)$. This means the set S_i contains instances labelled as "1" if their label in S was i, and "-1" otherwise. Then, we train C predictors h_i, each one based on their respective set S_i. These predictors h_i define the confidence of their prediction. One should thus consider the prediction of the most confident classifier. Equation (5.23) defines this decision. Concretely, the confidence could be the distance between the observation x an the hyperplane w defined by the classifier [84].

$$h(x) \in \underset{i \in C}{\arg\max}\, h_i(x) \tag{5.23}$$

5.3.2 LR Hyperparameters

Multiple hyperparameters exist for the logistic regression classifiers. The hyperparameters that will be discussed here are the ones of the Python implementation provided by Scikit-learn [200, 132]

The first hyperparameter that has been tweaked is the type of penalty. The regularization is applied by default, but one can choose between "l1" or "l2". The first term refers to the Lasso regularization, and the second one to the Ridge regularization. Both have been tested.

The `tol` hyperparameter may be adjusted as well. It represents the tolerance for stopping criterion and is set at $1e^{-4}$ by default. Said differently, it tells to the optimization algorithm when to stop, i.e. when it is closed enough from the minimum or maximum [41, 240]. Other arbitrary values have been tested.

A third hyperparameter that we adjusted is `C`, which represents the inverse of regularization strength. Let λ be the regularization term used in Equation (5.21) and Equation (5.22), we can define this hyperparameter as (5.24). Its default value is 1.0. Lowering this value strengthen the λ regulator.

$$C = \frac{1}{\lambda} \tag{5.24}$$

Additionally, one can decide if a constant, i.e. an `intercept`, should be added to the decision function. This is the term β_0 in Equation (5.10), which represent the expected value of y when all the x_i equal zero. Note that it is a binary hyperparameters, so we decide if the decision function should add or not this term.

As the reader can expect, there exist other hyperparameters that can be adjusted but they have remained unchanged during the experiments. For instance, we can mention the fact that different weights can be specified for each class via the `class_weight` parameter. If not, all classes share a weight of one. As mentioned in Section 5.2.2, hyperparameters such as `n_jobs` or `random_state` can be adjusted too.

5.4 Metric Functions in Machine Learning

The evaluation of the performances of classifiers is an important topic in the machine learning field [153]. The following paragraphs describe metrics that are commonly used to evaluate the performances of the classifier [137, 231].

The first and most known metric function is the *accuracy*, which represents the correctness of the model predictions [107]. It is achieved by computing the ratio of examples that are correctly classified with the correct label. The underlying concept consists of checking if the predicted label of the considered example is the same as the actual label of this example. Let's define this concept mathematically. Let N be the number of observations, y^p the true label of the observation p and $\hat{y}^p$ the predicted label. Mathematically speaking, this can be defined as :

$$accuracy = \frac{1}{N}\sum_{p=1}^{N} ||\hat{y}^p = y^p|| \tag{5.25}$$

However, this metric does not grasp one important consideration [107, 33]. Indeed, it does not distinguish the error of wrongly predicting the positive class or wrongly predicting the negative class. Taking the example of a unbalanced dataset containing 5 % of positive labels, any classifier that would always assign the negative label would achieve an accuracy of 95%.

For this reason, one should better look at the *confusion* matrix, i.e. the *error* matrix, which shows differently the performance of the classifier [107]. An illustration of the confusion matrix is given in Table 5.1. Each row of the matrix represents the instances of the actual class while each column represents the instances of the predicted label. In other words it provides the types of errors that are made.

		Predicted class	
		P	*N*
Actual Class	*P*	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

Table 5.1: Confusion matrix and its components for binary classification [180]

As the reader may notice, assuming a binary classification, the classifiers may output four different types of results : True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN). The first two outputs represent respectively correct positive and negative predictions. FP, however, mean incorrect positive predictions : the instances are predicted as positive, when in reality they are not. The FN term represents the same idea but for the negative predictions.

Based on this matrix, multiple measures can be derived from. One may start with the *error* rate, representing the number of all incorrect prediction divided by the total number of instances [107]. The range of this measure is contained between 0 and 1, 0 being the ideal rate the classifier should achieve.

$$error_{rate} = \frac{FP + FN}{TP + TN + FN + FP} \tag{5.26}$$

The accuracy rate, mentioned earlier, can be redefined with the terms of the confusion matrix. Defined between 0 and 1, the best rate one may desire is 1.

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP} = 1 - error_{rate} \tag{5.27}$$

Other measures exist and are more focused on the error costs. Depending on the context of the classification, one may want to avoid false negatives more than false positives or vice versa. A typical example can be illustrated as follows : it is preferably better to wrongly predict cancer (to a non-threatened patient) than non predicting cancer to a threatened patient [107].

First, let's consider the *precision score*, which allows to identify the percentage of positive labelled instances which are actually positive [107]. It is, in terms of confusion matrix, the number of positive examples that are correctly classified, divided by the number of all positive results predicted by the classifier.

$$precision = \frac{TP}{TP + FP} \tag{5.28}$$

The *recall* score highlights the proportion of positive labelled instances that have been correctly identified as positive [107]. This is referred to as the number of correct positive results divided by the number of samples that should have been classified as positive.

$$recall = \frac{TP}{TP + FN} \tag{5.29}$$

The evaluation of a model must take into account these two measures. For this reason, they can be gathered into a single score called the F_β-score [107]. It is defined as the weighted harmonic[1] mean of precision and recall [137], where β is a parameter controlling the impact weight of precision and recall. If in certain applications more importance is given to one of them, $\beta = 1$ is commonly used to equivalently balance the weight of the two measures.

$$F_\beta = \frac{(\beta^2 + 1) \cdot precision \cdot recall}{\beta^2 \cdot precision + recall} \tag{5.30}$$

$$F_1 = \frac{2 \cdot precision \cdot recall}{1 \cdot precision + recall} \tag{5.31}$$

So far, only evaluations of binary classification have been considered. In this work, one needs to consider a confusion matrix for multiple classes as shown in Table 5.2. Assuming n different classes, FN, FP, TN and TP_{all} rates are computed for each class i according to Equations (5.32), (5.33), (5.34) and (5.35) [136]. Note that the TP for each class i is the intersection of the i^{th} column and the i^{th} row.

		Predicted Number			
		Class 1	Class 2	...	Class n
Actual Number	Class 1	x_{11}	x_{12}	...	x_{1n}
	Class 2	x_{21}	x_{22}	...	x_{2n}
	.	.	.	.	.
	.	.	.	.	.
	.	.	.	.	.
	Class n	x_{n1}	x_{n2}	...	x_{nn}

Table 5.2: Confusion matrix for multiple classes [136]

The number of FN's for each class i is the sum of values in the corresponding row, excepting the TP.

[1] A harmonic mean is defined as the reciprocal of an arithmetic mean

$$FN_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ij} \tag{5.32}$$

The number of FP's for class i is computed as the sum of values in the related column, excepting the TP.

$$FP_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ji} \tag{5.33}$$

For clas i, the number of TN's is the sum of all rows and columns, except the ones belonging to the considered class.

$$TN_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i}}^{n} x_{jk} \tag{5.34}$$

The total true positive rate can be calculated as the sum of the diagonal of the confusion matrix.

$$TP_{all} = \sum_{j=1}^{n} x_{jj} \tag{5.35}$$

To compute the precision and the recall for each class, one may use Equations (5.36) and (5.37). Overall accuracy can be obtained via Equation (5.38) [136].

$$precision_i = \frac{TP_i}{TP_i + FP_i} \tag{5.36}$$

$$recall_i = \frac{TP_i}{TP_i + FN_i} \tag{5.37}$$

$$accuracy_{all} = \frac{TP_{all}}{\#instances} \tag{5.38}$$

CHAPTER 6

EXPERIMENTS AND RESULTS

Now that the theoretical part has been covered, let's focus on the practical aspect of this thesis. The goal of this section is to explain the different results obtained for the three approaches as well as the combined one. Which model, from Random Forest or Logistic Regression, predicts the best the related emotion is analysed too. The results are expressed in terms of indicators that have been mentioned earlier. Since the *F-measure* is a combined metric, results are discussed in terms of *accuracy*, *precision* and *recall* since they are more intuitive. For concision, the emotions *calmness*, *sadness*, *happiness* and *angriness* are sometimes referred as category 1, 2, 3, 4.

Before going any further, the reader must understand something. In the Music Emotion Recognition field, comparison with other works should be carefully analysed for some reasons that have already been mentioned before. First and foremost most of existing works compile their own databases, which can lead to different conclusions due to the inherent subjectivity of the songs. Furthermore, existing works don't always consider the same machine learning algorithm for the classification : support vector machine, neural networks,... [18, 70]. The number of considered emotions, and the choice of different features to represent the emotions lead to contradictory reports [246]. Finally, due to the unprecedented works of this task, the comparison is even more complicated.

During the experiments, both random forest and logistic regression models have been based on a train and a test sets with a respective proportion of 75%-25%. To obtain reliable and representative results, the experiment has been repeated ten times and so the evaluation metrics have been averaged on this basis.

The first section explains the results of the instrumentals analysis. Section 6.2 describes the results of the vocals analysis, while the third Section 6.3 presents the outcomes of the lyrical analysis. The Section 6.4 describes the results of the combination of three features. Even though a global conclusion is given in the next chapter, partial deductions are drawn at the end of each analysis.

6.1 Instrumental Analysis Results

Initially, eight features have been tested. There were the *loudness range*, the *loudness average*, the *RMS*, the *MFCC*, the *pitch*, the *bpm*, the *danceability* and the *zero-cross* rate. As we could expect, none of these features contributed equally to the classification. This was notably the case of the *MFCC*. Reduced as single value (hence a mean of the 13 first MFCC), one should not be surprised that it did not contain relevant information anymore. Indeed it didn't contribute a lot to the classifier : even worse, it decreased the accuracy rate of the classification. For this reason this feature was not used for the final instrumental analysis. The reader should know that a classification only based on the 13 first MFCC has been tested as well, showing low results i.e. an accuracy of 33%.

Based on their classification importance, five features have been finally kept for the final classification. There were three low-level features : the *loudness*, the *pitch*, the *zero-cross* rate as well as two high-level features : the *danceability* and the *bpm*. The importance of these five features is shown in Table 6.1.

Feature	**Importance** (%)
Zero-cross	37
Bpm	21
Pitch	15
Danceability	13
Loudness	11

Table 6.1: Importance of the different features for RF classifier

The random forest model predicted emotion labels with an accuracy of 70 %. Based on a 3-fold cross validation, a greedy search helped out to find the best values for the hyperparameters. These values are presented in Table A.1 in Appendix A. This improved the correctness of the classifier by 5%, providing a final accuracy rate of 75%. The confusion matrix, precision, recall and F1 values obtained with the best parameters can be found in Table 6.2 and Table 6.3.

Regarding the confusion matrix, one may notice some errors happen more often than others. For instance, errors such as songs belonging to the *calm* label but predicted as the *sad* label are common. This can be explained by the fact a calm song can share a lot of common characteristics with a sad song. The fact the bpm is generally in the same range of values makes the classification harder, since this feature is considered as the second most important in Table 6.1. Most of the time, a calm song is generally a sad song, which complicates the distinction. Surprisingly, some *sad* songs are predicted as *happy* songs. The fact all the *sad* and *happy* songs are not respectively labelled as *sad*/*happy* songs directly impacts their *recall* rate. Unsurprisingly, this score is lower for these two classes (see Table 6.3).

On the contrary, by looking at the columns of the confusion matrix, we can see that songs predicted as *calm* or *angry* are effectively related to the *calmness* and the *angriness*. This is reflected by the *precision* rate. By looking in Table 6.3, one can read that this score is higher for the two aforementioned emotions but lower for the other ones.

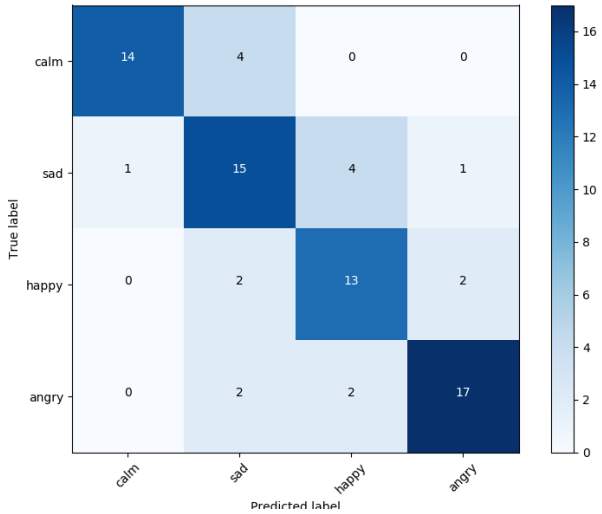


Table 6.2: Confusion Matrix of RF

c	**Precision**	**Recall**	**f1-score**
1	0.93	0.78	0.85
2	0.65	0.71	0.68
3	0.68	0.76	0.72
4	0.85	0.81	0.83

Table 6.3: Precision, Recall and F1 of RF

The logistic regression model achieved a lower accuracy rate : 52 % with hyperparameters set at their default values. The latter have been optimized by a 3-fold cross validation grid search, improving the accuracy to 65 %. These optimal values can be found in Figure A.2, in Appendix

A. The confusion matrix and related metrics are provided in Table 6.4 and Table 6.5.

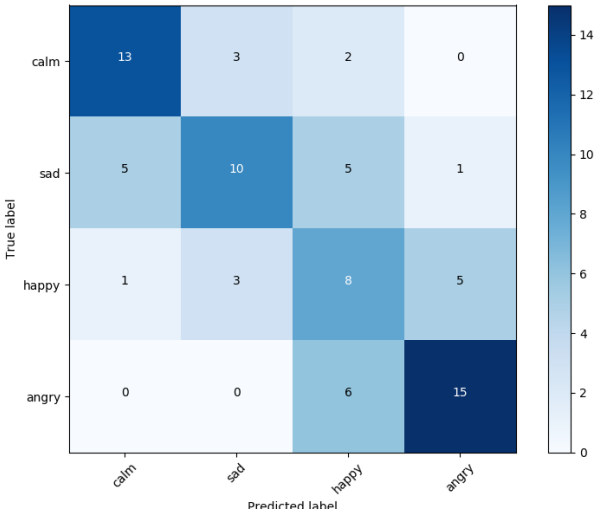


Table 6.4: Confusion Matrix of LR

c	Precision	Recall	f1-score
1	0.68	0.72	0.70
2	0.67	0.48	0.54
3	0.38	0.47	0.42
4	0.71	0.71	0.71

Table 6.5: Precision, Recall and F1 of LR

Such as the random forest model, the *angry* and *calm* categories have higher *precision* and *recall* than the two other emotions. However, *angry* songs tend to be more predicted as *happy* songs and inversely, which was not the case with random forests. *Calm* songs are now predicted as *happy* songs too. For this reason, the recall of these categories is lowered.

As we can see, *sad* songs are more often predicted as *calm* songs than with random forests. This largely impacts the recall of this category, which is reduced to 48 %. Similarly, the prediction error has increased for the *happy* songs. They are more often predicted as *angry* or *sad* songs. This is also reflected in the recall score, which is now lower.

By looking at both models, the random forest approach gives better performances. It provides a better accuracy, but foremost it provides better precision and recall rates for the four classes. This is reflected in the f1-score.

As a partial conclusion, one may say that, assuming a random forest model, the instrumental analysis gives good results at distinguishing songs that have an opposite arousal. As a reminder, the *arousal* term refers to the energy of a song, e.g. if it expresses more an *angry* or a *sleepy* feeling. As shown in Table 6.2, the model is foremost good at correctly predicting *calm* and *angry* songs.

6.2 Voice Analysis Results

The vocals analysis was initially based on seven features, as shown in Table 6.6. For the same reason as the instrumental analysis, some of them were dropped. The features highlighted in blue are the ones that have been kept for the final analysis. The vocal analysis showed lower accurate results than the instrumental analysis. Indeed the accuracy rate did not exceed 63%, which is 12% less than the instrumental-based approach.

In this vocals-based analysis the random forests achieved an initial accuracy rate of 55 %, improved to 63% thanks to optimized hyperparameters (Figure A.3 in Appendix A). The importance of the features with random forests is shown in Table 6.7. As the reader can see, the pitch and the spectrum centroid are the two first most important features. This means that the degree of highness or lowness of a voice, and its brightness play important roles in the expressed emotion. Then comes the zero-cross rate, indicator of noisiness, and finally the roll-off.

The confusion matrix of the random forest classification is presented in Table 6.8. The derived metrics are shown in Table 6.9. Compared to the random forest approach for instrumental analysis,

Feature
MFCC
F0 range
F0 max
Pitch
Spectral centroid
Roll-off
Zero-cross

Table 6.6: Features for the vocals analysis

Feature	Importance (%)
Pitch	33
Spectrum centroid	27
Zero-cross	20
Roll-off	18

Table 6.7: Importance of features for RF classifier

one may notice that all the metrics are lower. This is specially the case of the *calm* songs, which have now a precision rate of 63 % instead of 93 %, and a recall score lowered by 15 %.

Even though this approach gives lower metrics, a similar pattern occurs : the *calm* and *angry* songs have higher precision rates than the *sad* and *happy* songs. This is particularly true for the *angry* songs. This can probably be explained by the genre of music this emotion is related to. Most of the time, this latter is represented by *metal* songs, which have relatively distinguishable vocals parts from any other genres.

Another interesting pattern we can observe is that the classifier tends to predict *calm* songs as *sad* songs and inversely. A reason that can explain this comes from the fact that vocals in *sad* and *calm* songs can sound similar and thus share similar characteristics.

One can notice that the recall for *happy* songs is noticeably high compared to the instrumental-based analysis. This means it better identified the *happy* songs than with the instrumentals.

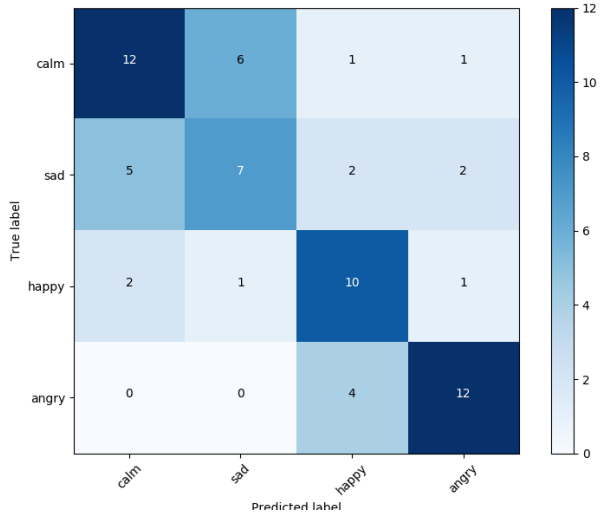


Table 6.8: Confusion Matrix of RF

c	Precision	Recall	f1-score
1	0.63	0.60	0.62
2	0.50	0.44	0.47
3	0.59	0.77	0.65
4	0.80	0.8	0.8

Table 6.9: Precision, Recall and F1 of RF

The accuracy rate with a logistic regression has remained steady around 54 %. Optimizing the hyperparameters raised this rate to 55 % (see Figure A.5 in Appendix A for values). The confusion matrix with the logistic regression model and the accuracy metrics can be observed in Table 6.10 and Table 6.11.

With this kind of classifier, one may notice that the errors related to *sad* and *calm* songs tend to increase. Indeed, more *calm* songs are predicted with the *sad* label, such as more *sad* songs are classified as *calm* songs. These misclassifications are reflected in the *recall* rate of the two categories.

On the other side, *happy* songs are identified with a precision of 91%, which is better than

during the instrumental analysis. In other words, it predicts better the *happy* songs as *happy*.

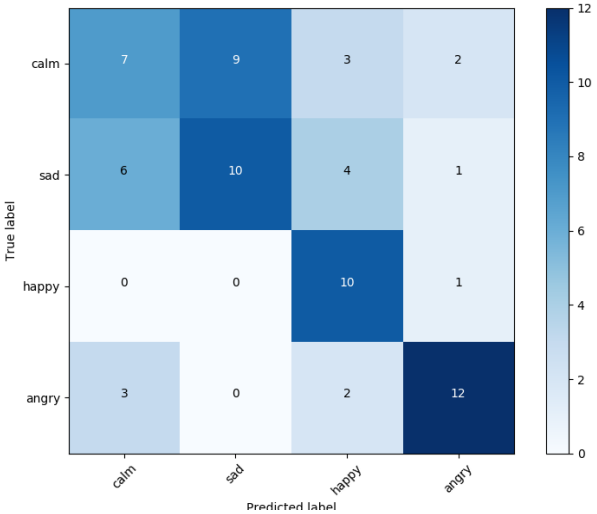


Table 6.10: Confusion Matrix for LR

c	Precision	Recall	f1-score
1	0.44	0.33	0.38
2	0.53	0.48	0.50
3	0.53	0.91	0.67
4	0.75	0.71	0.73

Table 6.11: Precision, Recall and F1 with LR

To conclude this analysis, we can say that this approach provides a lower accuracy rate than the instrumental approach. At its best, it only predicts observations with an accuracy of 63 % compared to 75 % with the previous approach. Neither random forests nor logistic regressions gave better f1-scores than the random forests in the instrumental analysis.

This approach revealed nevertheless an interesting pattern. With a logistic regression, it outperforms the instrumental analysis when it comes to predict the songs related to *happiness*. It showed a recall of 91 %, compared to 76% with the instrumentals.

The low accuracy rate can be partially explained by the loss of quality information during the separation process. Indeed, since it does not exist a software that perfectly extracts the vocals part from the original track, the algorithm may not have taken the full spectrum of the voice into consideration. This results into a voice as it was passed through a high-pass filter. Chances to tackle this kind of separation problems would be to only work with perfectly separated voice-instrumental. Until this day, the only way to get this approach right is by getting what we referred as the *stems* of the songs. They represent the official instrumental and voice files which have been directly recorded in the studio while production. Unfortunately, the music labels almost never share this kind of files in order to keep their rights and royalties under control.

Furthermore, to grasp the difficulty of this approach, one should test to classify manually the different songs according to their respective vocals. If some vocals would be easily classifiable such as vocals from metal songs (and thus related to *angriness*), some others are not : vocals from *sad* and *calm* tend to sound the same and reduce thus the chance to obtain great results.

6.3 Lyrical Analysis Results

Regarding the classification based on the lyrics, both random forest and logistic regression models have been tested with multiple preprocesses. As it can be observed in Table 6.12, the two approaches have been considered without stemming, but also with Porter and Snowball stemmers. Additionally, they were tested with the concept of white list introduced earlier. It is denoted as *wl* in the table.

As we can observe, the concept of *sementic lexicon* was particularly useful. With the random forests, it increased the accuracy of 4 % whereas for the accuracy of logistic regressions was raised by 10%. Regarding the stemming, two conclusions can be drawn. Porter approach gave a better results than Snowball when no semantic lexicon was considered, but Snowball outperformed Porter when it was the case. One can notice that the less words the classifier consider, the higher the accuracy is.

Classifier	Approach	ACC (%)
Random Forest	without stem. (4029 w.)	31.2
	Porter (3281 w.)	34.0
	Snowball (3275 w.)	32.0
	without stem. + wl (711 w.)	36.1
	Porter + wl (609 w.)	36.3
	Snowball + wl (605 w.)	**38.2**
Logistic Regression	without stem. (4029 w.)	24.0
	Porter (3281 w.)	31.5
	Snowball (3275 w.)	31.0
	without stem. + wl (711 w.)	41.0
	Porter + wl (609 w.)	47.0
	Snowball + wl (605 w.)	**49.0**

Table 6.12: The different accuracy rates obtained from the lyrics-based classification

Let's first analyse in depth the random forest model that provided the best accuracy rate. Its confusion matrix is given in Table 6.13. The related metrics are shown in Table 6.14, highlighting the poor performance of this model.

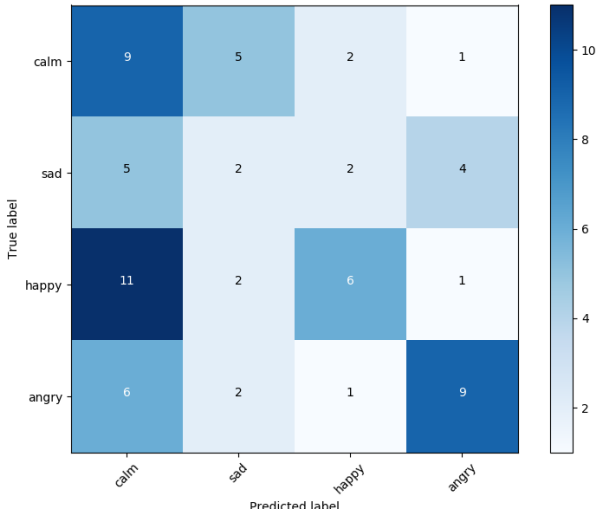


Table 6.13: Confusion Matrix for RF

c	Precision	Recall	f1-score
1	0.29	0.53	0.38
2	0.18	0.15	0.17
3	0.55	0.30	0.39
4	0.60	0.50	0.55

Table 6.14: Precision, Recall and F1 with RF

The lowest precision rate belongs to the *sad* songs. This means the classifier predicts a song to be *sad* but it is not. Indeed, *calm* songs are often wrongly predicted as *sad*. This observation

explains this low precision rate, i.e. 18%. Inversely, the *sad* songs tend to be predicted as *calm* songs, but also as *angry* ones too. This pattern is measured by a low recall, i.e. 15 %.

One unprecedented characteristic is that a lot of songs related to the *sadness*, the *happiness* and *angriness* are predicted as *calm* songs. This is specially the case for the *happy* songs, for which eleven of them are wrongly labelled as *calm*.

Let's now consider the logistic regression model that gave the highest accuracy, i.e. 49 %. The confusion matrix, the precision, the recall and F1 scores are given in Table 6.15 and 6.16. Compared to the random forests, the f1-scores have all increased except for the *angry* class.

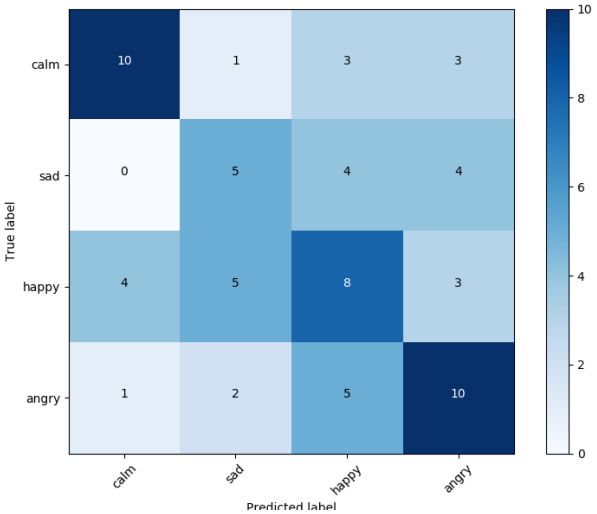


Table 6.15: Confusion Matrix for LR

c	Precision	Recall	f1-score
1	0.67	0.59	0.62
2	0.38	0.38	0.38
3	0.40	0.40	0.40
4	0.50	0.56	0.53

Table 6.16: Precision, Recall and F1 with LR

The logistic regression model does not predict *sad* songs as *calm*, and barely not *calm* songs as *sad* ones. This differs from the random forest approach. However, these both classes tend to be both predicted as *happy* or *angry*, which was not the case with the other approach.

With the random forests, the *happy* songs were foremost predicted as *calm* songs. With a logistic regression, these songs are also predicted as *calm*, but less frequently. They are more labelled with the *sadness* or the *angriness*. One may notice that *angry* songs are, with this approach, more predicted as *happy*, where it used to be more predicted as *calm* songs with random forests.

A partial answer to these low rates comes from the fact a lot of words are common in any songs. As it was shown in Figure 3.8, words such as *love* or *know* are used in various contexts. This induces confusion for the classification. Another reason that explains the low accuracy can be found in the way song writers choose their lyrics. Most of the times, words are chosen for their sounds or their rhymes, rather than for their meaning.

6.4 Multimodal Analysis Results

After analyzing separately the three elements of a song, let's focus on combining all these features to determine if it provides a better accuracy and related metrics than the other supports.

The features that have been considered are the five features from the instrumental-based analysis, the four of the vocals analysis and the 605 words from the lyrical analysis. Regarding the models, the random forests performed an accuracy of 67 %, whereas the logistic regression provided 47%. For this reason, the following paragraphs are only focused on the random forest model and not on the logistic regression. However, if the reader wants to observe the confusion matrix and the other metrics, they are given in Table A.6 and Table A.7 in Appendix A.

The confusion matrix of the random forest approach is illustrated in Table 6.17. The precision, recall and F1-scores are given in Table 6.18. As it can be observed, the F1-scores are not higher than the ones observed earlier, i.e. higher than the instrumental analysis based on random forests.

Calm songs are correctly identified. As shown in the confusion matrix, it is predicted as *sad* only once. This is reflected by a high recall rate. The *angry* class is also well predicted, yet with a bias for the *happy* category. It is also reflected by a high recall compared the other classes.

Regarding the precision, the *calm*, the *sad* and the *happy* classes share a lower rate compared the *angry* one. This can be explained by the fact that, for example, *sad* songs are predicted as *calm* and then it increases the number of false positives. This reduces the precision rate. Respectively, *happy* songs tend be predicted as *calm* or as *sad* too.

If, at first sight, we could expect a better performance based on the combination of these features, we can imagine how it adds complexity for the classifier. Indeed, due to the variety of emotions expressed separately by the instrumental, the vocals and the lyrics, the distinction between classes tend to become fuzzy with that much sparsity.

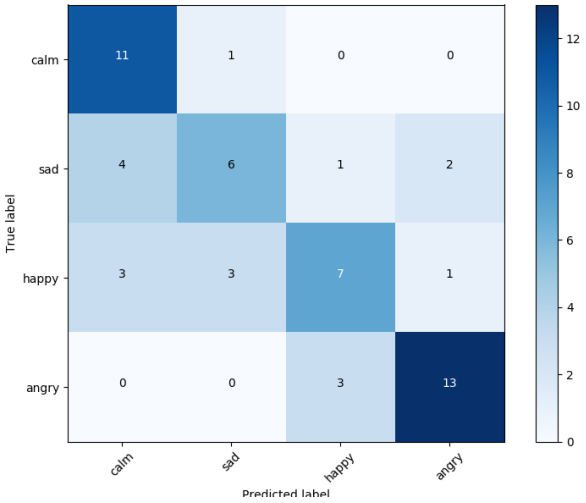


Table 6.17: Confusion Matrix for RF

c	Precision	Recall	f1-score
1	0.61	0.92	0.73
2	0.60	0.46	0.52
3	0.64	0.50	0.56
4	0.81	0.81	0.81

Table 6.18: Precision, Recall and F1 with RF

CHAPTER 7

CONCLUSION AND FURTHER WORK

In this thesis, we presented an unprecedented application of the emergent field of Music Emotion Retrieval which gathers music, emotions and machine learning techniques. After reviewing the state of the art of this relatively recent field and presenting the theory of MER, we investigated separately the instrumental, the vocals and the lyrics of songs to determine which support carries the most emotional content. The combination of three elements has been analysed as well. In order to achieve this task two machine learning models, namely *Random Forest* and *Logistic Regression*, have been considered. To the best of our knowledge, this work is unprecedented.

Table 7.1 summarizes the best accuracy rates that have been obtained for each of the four analysis, both by random forest and logistic regression. The first conclusion that can be drawn is that the accuracy rate of the instrumental analysis outperforms the rates of the other approaches, from 10% to almost 20%. This means the instrumental is better at predicting the emotion of the song than the vocal part, the lyrics or a combination of the three. In second position comes the combination of three elements with an accuracy of 67 %, then the vocal part (63%), and finally the lyrics (49 %).

	Instrumental		**Vocals**		**Lyrics**		**Combined**	
	RF	LR	RF	LR	RF	LR	RF	LR
accuracy (%)	75	65	63	55	38.2	49	67	47

Table 7.1: Summary of accuracy rate for the instrumental, vocal and lyrics analysis

We can thus give an answer to the first question of this thesis : "*which support, from lyrics, vocals or the instrumental, gives the best accuracy for emotion classification ? Does a combination of them give better results ?*". Experiments showed, in the context of this thesis, that the instrumentals provide the most accurate predictions, followed by a combination of the three supports and then the vocals and the lyrics.

The low accuracy rate for the lyrics is consistent with the conclusion of some previous works : lyrics alone are not as effective as audio for mood classification [128, 139]. Furthermore, regarding the combination of three features, the variety of emotions separately expressed by the instrumental, the vocals and the lyrics makes the classification fuzzier rather than clearer. For this reason, we understand why previous works based their prediction on the complete audio i.e. vocals and instrumentals grouped, rather than isolated [49, 247]. Regarding the lyrics, experiments showed that a better prediction can be obtained with a semantic lexicon combined with Snowball stemmer.

The second question can be answered too. As a reminder, it asked to determine which model, from Random Forest or Logistic Regression, fits the best the concrete application of this thesis. By looking at the Table 7.1 and the previous chapter, the random forest approach has outperformed

the logistic regression for the audio analysis (vocals, instrumentals) and the multimodal analysis. However the logistic regression model has provided a better accuracy for the lyrical analysis. Additionally, this model has showed better performance regarding the prediction of the *happy* songs when it only considered vocals. By analyzing the confusions matrices and the related metrics, we noticed that the models were foremost good at predicting songs that have an opposite arousal, i.e. *calm* songs versus *angry* songs. This can be explained by the fact we mostly used global features, which are known to induce this kind of effects.

The reader may have understood it, multiple challenges exist in the research field. This is why we would like to suggest some paths that could be explored into the future. One major drawback of MER is the lack of cooperation across research institutes. Each one uses a different database and conclusions from previous works are unfortunately not consistent. As result, these conclusions cannot be extended to other datasets [60].

To address this problem, MER researchers should establish emotional benchmark databases so that results would come from the same database and be comparable. A common accessible database, analysed by experts, would be time-saving and would reduce errors such as songs labelled with a non-corresponding emotion. As far as we can contribute to the Music Emotion Recognition field, we freely released all the databases that have been used in this thesis. The reader, and any enthusiastic person, will find them at the following url : https://github.com/thomasberton/musicemotionrecognition

Before suggesting multiple paths that could be explored in the future, one should remember all the assumptions and choices that have made before the classification process. First and foremost, we opted for a categorical approach of the emotions. This means emotions are not represented by coordinates but are separated in groups. Four categories have been taken into account, namely the *calmness*, the *sadness*, the *happiness* and the *angriness*. These four emotions represent two groups of two opposed categories in the Hevner's adjectives checklist. Furthermore, one should look back at the features that have been considered for each approach. For the instrumentals, a mix of high and low-level features has been finally kept. There were the *zero-cross* rate, the *bpm*, the *pitch*, the *danceability* and the *loudness*. The vocals have been analysed under the light of four features, which were the *pitch*, the *spectral centroid*, the *roll-off* and the *zero-cross* rate. Finally, the lyrics have been preprocessed before classification : punctuation has been removed as well as stop words, assuming they don't contribute the emotion classification. All the words have been lower-cased too. Classification has been tested with and without stemming, with and without semantic consideration.

These multiple assumptions pave the way for exploring new paths for further research. First, future works should investigate a larger set of emotions or a dimensional approach, where emotions are defined in terms of *arousal* and *valence*.

Besides, it is suggested to explore features that have not been considered in the thesis, e.g. TEO-features for vocals, in order to determine if they can increase or reduce the performances. One can explore to separate the vocals from the instrumental in a different way than the *RX7* software too. Moreover, other preprocessing steps (e.g. lemmatization) could be tested for the lyrics.

Regarding the machine learning classifier, the thesis only considered *random forest* and *logistic regression* models. Even though they achieved some acceptable results, especially for the instrumentals, one should explore other machine learning techniques in hope to improve automatic music emotion classification in the future.

APPENDIX A

ADDITIONAL FIGURES

Category	Examples
Top Level labels	**Genre** - *pop, blues, classical,etc* **Mood** - *happy, sad, angrily, etc* **Instrumental** - *piano, guitar, violin, drum, etc* **More** - *artist, style, similar song, etc*
Middle Level Labels	**Rhythm** **Pitch** **Harmony**
Low Level Labels	**Timbre** - *zero crossing rate, spectral centrol, spectral rolloff, etc* **Temporal** - *SM, AM,ARM, etc*

Figure A.1: Different categories of audio features [47]

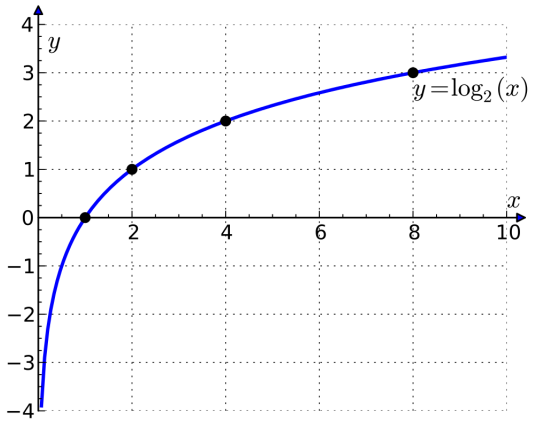


Figure A.2: Graph of $\log_2$(x) as a function of a positive real number x [16]

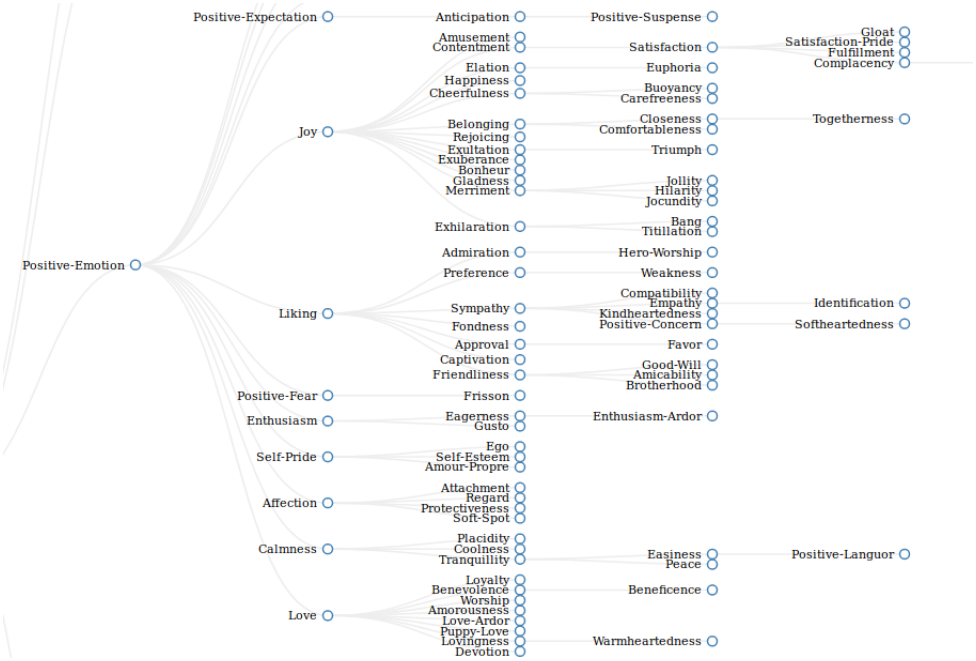


Figure A.3: Part of the tree-based structure of WordNet-Affect [212]

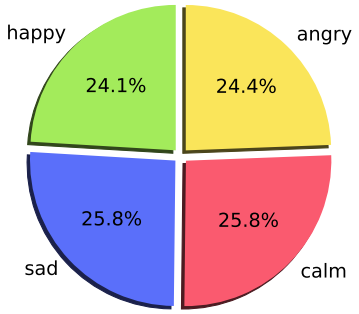


Figure A.4: Presence in % of each emotion in the database

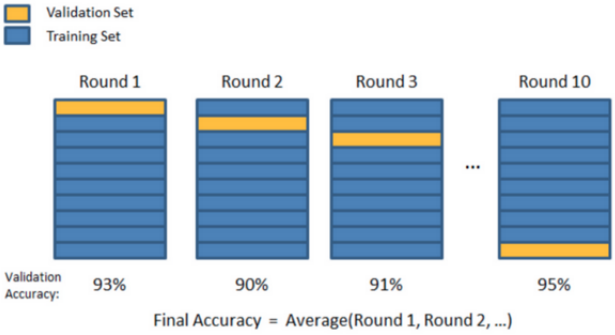


Figure A.5: Illustration of a 10-fold cross validation process [113]

)

Hyperparameters	Values
criterion	entropy
bootstrap	True
max_depth	90
max_feature	'auto'
min_samples_leaf	3
min_samples_split	8
n_estimators	300

Table A.1: Best values for the RF hyperparameters in instrumental analysis

Hyperparameters	Values
penalty	'l1'
C	100.0
fit_intercept	true
max_iter	100
tol	0.0001

Table A.2: Best values for the LR hyperparameters in instrumental analysis

Hyperparameters	values
criterion	entropy
bootstrap	False
max_depth	80
max_feature	'sqrt'
min_samples_leaf	1
min_samples_split	10
n_estimators	1000

Table A.3: Best values for RF hyperparameters in vocal analysis

(hyper) parameters	values
C	1
class_weight	None
dual	false
fit_intercept	true
intercept_scaling	1
max_iter	100
multi_class	'warn'
n_jobs	none
penalty	'l1'
random_state	none
solver	'warn'
tol	0.0001
verbose	0
warm_start	false

Table A.4: Best values for LR hyperparameters in instrumental analysis

Class.	Approach	ACC (%)	PRE (%)	REC (%)	F1 (%)
LR	without stem.	24.0	18.0 21.0 38.0 33.0	18.0 54.0 15.0 17.0	18.0 30.0 21.0 22.0
	Porter	31.5	26.0 26.0 38.0 33.0	29.0 54.0 15.0 17.0	28.0 35.0 21.0 39.0
	Snowball	31.0	26.0 26.0 40.0 42.0	29.0 54.0 20.0 28.0	28.0 35.0 27.0 33.0
	/ + wl	**44.0**	**45.0 24.0 50.0 58.0**	**59.0 31.0 25.0 61.0**	**51.0 27.0 33.0 59.0**
	Porter + wl	44.0	45.0 29 40 57	53.0 38.0 20.0 67.0	49.0 33.0 27.0 62.0
	Snowball + wl	43.0	42.0 29 36 57	47.0 38.0 20.0 67.0	44.0 33.0 26.0 62.0

Table A.5: Accuracy metrics for the lyrical analysis with LR

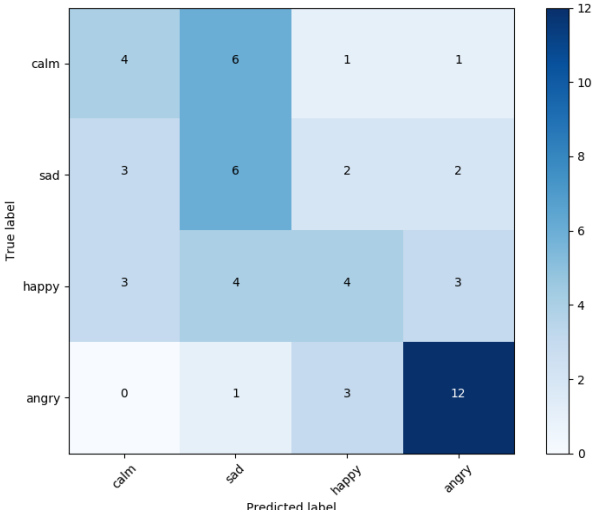


Table A.6: Confusion Matrix for LR

c	Precision	Recall	f1-score
1	0.40	0.33	0.36
2	0.35	0.46	0.40
3	0.40	0.29	0.33
4	0.67	0.75	0.71

Table A.7: Precision, Recall and F1 with LR

APPENDIX B

MFCC

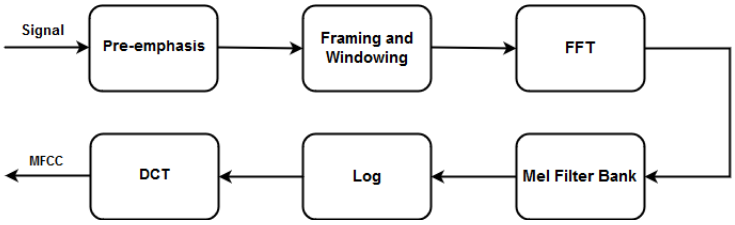


Figure B.1: MFCC coefficients extraction process

Pre-emphasis emphasis

In the very first step, the pre-emphasis one, the signal is passed through a filter which emphasizes high frequencies, i.e. increases the energy of signal at higher frequency [176]. This is required since high frequency components have small amplitude compared to low frequency components. Let X be the discrete-time signal input, the output Y of the pre-emphasis filter can be formulated as (B.1). The parameter n is integer number between 1 and N-1, N being the length of the signal. The α term is a positive parameter between 0 and 1 and controls the degree of pre-emphasis filtering. Classical value for this parameter is 0,97 [119].

$$Y[n] = X[n] - \alpha \times X[n-1] \tag{B.1}$$

Framing and Windowing

The second step regards the framing and windowing processes. Framing is the process of segmenting the speech signal into small duration blocks (e.g. between 20∼30 ms) referred as frames in the time domain [75, 148, 203]. The frames contain N samples and are overlapped within each others by M samples ($M < N$). This overlaps generally correspond to 50∼70% of the length of the frames. This framing step is required since speech is a time varying signal : we assume the overall signal is non-stationary, but stationary and periodic for a short period of time [176].

After the framing step, each individual frame is multiplied by a window. The framing process breaking the continuity between the different blocks, one must reduce these differences by smoothing the transition between frames [55]. This is the role of windowing. A window function can be defined as a mathematical function which outputs zero outside a defined region, one otherwise [172]. There exist multiple window function but the one used in the context of MFCC is the *Hamming* function [96]. It is a bell-shaped window which sets the beginning and the end of each frame at almost zero. The formula of Hamming window is described in (B.2), where N is the length of the window and n a sample in this window. Applying this window on each frame is expressed in Equation (B.3), where X is the discrete input signal, Y the discrete output signal and $w(n)$ the

Hamming window.

$$w[n] = \begin{cases} 0.54 - 0.46\cos(\frac{2\pi n}{N-1}) & 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{B.2}$$

$$Y[n] = X[n] \times w[n] \tag{B.3}$$

Fast Fourier Transform

The third step computes the Discrete Fourier Transform of a signal [31, 179]. DFT is the result of converting each frame of the signal from time domain into frequency domain. One existing algorithm to generate this DFT is the *Fast Fourier Transformation* [44]. Let N be the number of bins in the input signal x, the discrete Fourier transform $X[k]$ is expressed as follows :

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-jkn\pi N} \qquad 0 < k \leq N-1 \tag{B.4}$$

Mel Filter Bank

The fourth step highlights an important point : human perception of frequency is non-linear [217]. For this reason, another scale is used, which is the *mel* scale [50]. During their experiments, Steven et al [217] designed the scale as linearly below 1000 Hz and logarithmic above. The conversion of a frequency on the mel scale is expressed in Equation (B.5), f being the frequency to be converted. The frequency f_0 is chosen such that below it, the mel scale changes linearly and above it, it changes logarithmically [158]. Common value for this *corner* frequency is between 700 and 1000. The constant C is chosen such that 1000 Hz correspond to 1000 mels [158]. Assuming 700 as f_0 this gives the constant $C = 2595$.

$$Mel(f) = C \times \log_{10}(1 + \frac{f}{f_0}) \tag{B.5}$$

The human ears act as filters and are non-uniformly spaced on the frequency axis : more filters are present at the low frequencies, whereas fewer filters are found at high frequencies. To mimic this human perception, researchers have come up with the concept of *filterbank* [129], shown in Figure B.2. It groups overlapping triangular bandpass filters (denoted by different colours) which are linearly equally-spaced below 1 kHz and logarithmic equally-spaced above.

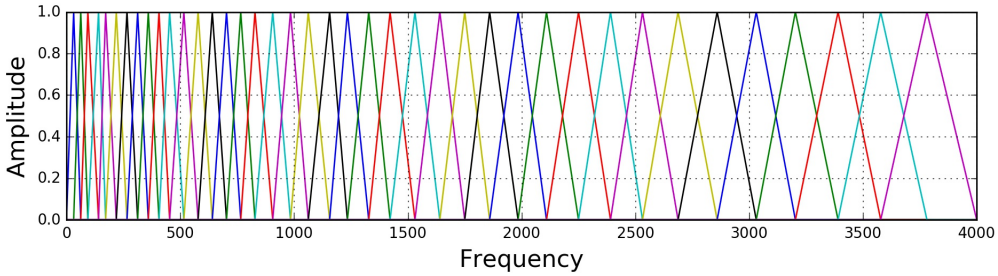


Figure B.2: Mel scale filter bank from Young et al, 1997 [129]

They are applied to the spectrum, giving thus the mel spectrum, and each filter output is the sum of its filtered spectral components. Assuming M filters, this output Y_m of the m^{th} filter can be expressed as (B.6). The term $H_m[k]$ is the frequency magnitude response of the m^{th} filter and $X[k]$ is the FFT spectrum of the windowed speech [69]. The output of the m^{th} filter can be seen as the magnitude of the speech in its frequency region, weighted by the filter response [69].

$$Y[m] = \sum_{k=0}^{K} X[k] \times H_m[k] \qquad 0 < m \leq M-1 \tag{B.6}$$

The Log Of Powers

The fifth step, often grouped with the previous one, is a reformulation of the previous equation. It computes the logarithm of the squared magnitude $X[k]$. This is motivated by the fact human response to signal level is logarithmic [66, 218]. This means humans are less sensitive to small differences between high amplitudes than with low amplitudes. Given that, one can rewrite the previous equation as (B.7), where $E[m]$ denotes now the m^{th} *log-energy* mel spectrum coefficient.

$$E[m] = \ln\left[\sum_{k=0}^{K} |X[k]|^2 \times H_m[k]\right] \qquad 0 < m \leq M-1 \qquad \text{(B.7)}$$

The Discrete Cosine Transform

The sixth and last step in the extraction process of MFCC is converting the log mel spectrum into a time domain. This can be achieved by a Discrete Cosine Transform [179]. Understanding the motivation of this step requires to introduce the source-filter model [217]. According this model, a signal is composed by the source, the vocals folds, and shaped by a filter, the vocal tract. The source is not important for ASR because most of the important information is carried in the filter [222]. One way to solve this problem is to compute the spectrum of the log of the spectrum, which is defined as the *cepstrum*[1].

In the context of the MFCC, we compute the cepstrum of the mel spectrum, which produces the *mel-frequency cepstrum*. Hence, applying a DCT on the mel-frequency coefficients produces the Mel-Frequency Cepstrum Coefficients. Most widely DCT used for speech processing is DCT-II [106, 179]. Due to its energy compacting, it implies more concentrated coefficients at lower indices. Higher coefficients tend to be noisy and carry less information. This property allows approximating the speech signal with fewer coefficients [92] and this is why a lot of researchers only consider the first 13 MFCC coefficients [109]. Each MFCC c_i is computed according to the Equation (B.8).

$$c_i = \sum_{m=1}^{M} E[m] \cos\left[\frac{i\pi}{M}\left(m - \frac{1}{2}\right)\right] \qquad 0 < m \leq M-1 \qquad \text{(B.8)}$$

[1]The term *cepstrum* is derived by reversing the first four letters of "spectrum"

APPENDIX C

TF-IDF SCORE

As for the audio analysis, feature extraction is one of the key stages of Lyrics-based Music Information Retrieval. There exist multiple approaches to extract relevant features. For instance, some previous works based the emotion of the lyrics on the most important words of the song [120, 246, 88]. Differently, other researchers focused on interjection words (e.g. "ooh", "ah"") and special punctuation marks (e.g. "!", "?") [177, 100].

In the thesis, the focus has been set on the words themselves and their related importance. Assuming this, one may ask how to estimate the importance of each word ? There exists a metric to compute it, which is called the *term frequency - inverse document frequency* [13, 194]. It is a well-known method in information retrieval and text mining [220, 97].

With a tf-idf approach, each word is given a weight that represents a statistical measure to evaluate how important is this word in a document, in a collection of document[1]. The intuition behind this score can be described as follows : the words that appear frequently are less important than the terms that are rare. Formally speaking, the tf-idf weight is composed of two terms.

The first one represents the *term-frequency* (abr. tf). It measures the importance of a term within a particular document [191]. Mathematically speaking, it is the number of times a word appears in a document, divided by the total number of words in that document. This normalization is necessary to tackle the bias that appears with long documents : a term would naturally appear much more times in long documents than shorter ones. Let i be a term in the document j, $\mathrm{f}_{i,j}$ the frequency of this word in the document and n_w the number of words in the document. The term frequency tf_{ij} is hence defined as follows :

$$tf_{ij} = \frac{f_{ij}}{\sum_{i=1}^{n_w} f_{ij}} \tag{C.1}$$

The second term, i.e. the *inverse document frequency* (abr. *idf*), measures how important a term is regarding a particular document, compared to all other documents. Broadly speaking, it measures the general importance of the word. This quantity is particularly useful since common words such as *is* or *a* may appear very frequently in many documents, but don't provide useful information. For this reason, excessively frequent words are weighted down, while rare ones are scaled up. The logarithm scale is used for that purpose. Let $P(w_i)$ be the a priori probability of a word w_i to appears in a document, the idf score of this word is measured as (3.16). Note that the result of the binary logarithm is reversed since it outputs negative values between]-∞,1] (see the graph of the function in Figure A.2 in Appendix A).

$$idf_i = -\log_2 \left[P(w_i)\right] \tag{C.2}$$

Finally, the *term frequency - inverse document frequency* is calculated as in Equation (C.3). A high tf-idf score for a word w_i is thus achieved by a high *term frequency* and low *inverse document*

[1] The documents are, in this case, the lyrics of the songs.

frequency. Hence, the weights tend to filter out common terms. Indeed the more a word w_i is expected to be in the document the more high is its probability $P(w_i)$. Let $P(w_i)$=1, this results as $\log_2(1) = 0$ and thus *idf* and tf-*idf* equal to zero.

$$tf\text{-}idf_{ij} = idf_i \cdot tf_{ij} \tag{C.3}$$

The tf-*idf* score has been computed for each word from the lyrics via a Python function called `Tfidfvectorizer` [200, 224]. Once this step has been reached, one must define the `min_df` and `max_df` parameters. They respectively represent the document frequency below / above the terms are ignored. Unfortunately, there does not exist common values to set these parameters. They have to be defined empirically [237].

BIBLIOGRAPHY

[1] C.N. Anagnostopoulos, T. Iliou and I. Giannoukos, "Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011," *Artificial Intelligence Review 43*, 2, 2015, pp. 155–177.

[2] J-J. Aucouturier and F. Pachet, "Improving Timbre Similarity : How High Is The Sky ?," in *Journal of Negative. Results in Speech and Audio Sciences*, Vol.1, No.1, 2004.

[3] M. Aly, *Survey on Multiclass Classification Methods*

[4] "AllMusic". Available : https://www.allmusic.com/ [Accessed June 2019]

[5] P. Ayush, *Introduction to Logistic Regression*, Available : https://towardsdatascience.com/introduction-to-logistic-regression-66248243c148, Jan. 22 2019 [Accessed may 2019]

[6] R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, 1999

[7] R. Banse, K. Scherer, "Acoustic profiles," in *Vocal Emotion Expression*, J. Perso. Soc. Psychol. 70 (3), 1996, pp. 614-636.

[8] D. Bassal, *La pratique du mastering en électroacoustique*, December 2002.

[9] A.S. Bhat, V.S. Amith, N.S. Prasad and D.M. Mohan, "An efficient classification algorithm for music mood detection in western and hindi music using audio feature extraction," in *Proceedings of the 5th International Conference on Signal and Image Processing*, IEEE Xplore Press, Jeju Island, Jan. 8-10 2014, pp. 359-364.

[10] L. Bergelid, "Classification of explicit music content using lyrics and music metadata", Degree Project, Stockhom institute of computer science and engineering, 2018

[11] J. Bergstra, Y. Bengio, "Random Search for Hyper-Parameter Optimization". Journal of Machine LEarning Research 13, 2012, pp. 281-305.

[12] A. Berrian, *Project on Music/Voice Separation*, August 2014. Available : https://www.math.ucdavis.edu/aberrian/research/voice_separation/index.html [Accessed July 2019]

[13] M.W. Berry, M. Brownie, *Understanding Search Engines: Mathematical Modeling and Text Retrieval (Software, Environments, Tools)*, second edition, 2005

[14] T. Bertin-Mahieux, D. P.W. Ellis, B. Whitman, and P. Lamere, "The Million Song Dataset," in *Proceedings of the 12th International Society for Music Information Retrieval Conference* (ISMIR 2011), 2011.

[15] E. Bigand, "Multidimensional scaling of emotional responses to music: The effect of musical expertise and of the duration of the excerpts," in *Cognition and Emotion*, vol. 19, no. 8, 2005, page 1113.

[16] "Binary Logarithm", *Wikipedia*. Available https://en.wikipedia.org/wiki /Binary_logarithm [Accessed July 2019]

[17] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python* (1st ed.), O'Reilly Media, Inc., 2009

[18] K. Bischoff, C.S. Firan, R. Paiu, W. Nejdi, W. Laurier, M. Sordo, "Music mood and theme classification - a hybrid approach," in *Proceedings of the International Conference on Music Information Retrieval*, 2009, pp. 657-662.

[19] C.M. Bishop, *Pattern Recognition and Machine Learning*, Springer-Verlag New York, 2006.

[20] W. Björn Schuller, "Speech emotion Recognition : two decades in a nutshell, benchmarks and Ongoing trends," in *Communications of the acm*, vol. 61, no. 5, may 2018, pp. 90-99

[21] S. Blanton, "The voice and the emotions," in *Journal of Speech 1*, 2, 1915, pp. 154–172.

[22] M. Borchert, A. Dusterhoft, "Emotions in speech - experiments with prosody and quality features in speech for use in categorical and dimensional emotion recognition environments," in *Proceedings of IEEE*, 2005.

[23] L. Bosch, "Emotions, speech and the asr framework," in *Speech Commun*, 40, 2003, pp. 213-225.

[24] S. Bou-Ghazale, J. Hansen. "A comparative study of traditional and newly proposed features for recognition of speech under the stress," in *IEEE Trans. Speech Audio Process.* 8 (4), 2000, pp. 429-442.

[25] R. Bracewell, *The Fourier Transform Its Application*, 1999

[26] M. Bréal, *Semantics : studies in the science of meaning*, 1900

[27] C. Breazeal, L. Aryananda, "Recognition of affective communicative intent in robot-directed speech," *Autonomous 2*, 2002, pp. 83-104.

[28] L. Breiman, J. Friedman, C. J. Stone, R.A. Olshen, *Classification and Regression Trees* Leo Breiman, Jerome Friedman, 1984

[29] L. Breiman, "Bagging Predictors," in *Machine Learning 24*, 1996, pp. 123-140.

[30] L. Brieman, "Random Forests," in *Machine Learning* 45, 5-32, 2001

[31] E. O. Brigham, *The Fast Fourier Transform and Its Applications*, 1988.

[32] S. Brilis, E. Gkatzou, A. Koursoumis, K. Talvis, K. Kermanidis, I. Karydis, "Mood Classification Using Lyrics and Audio: A Case-Study in Greek Music," in *8th International Conference on Artificial Intelligence Applications and Innovations*, Sep 2012, Grece, pp. 421-430.

[33] J. Brownlee, Classification Accuracy is Not Enough: More Performance Measures You Can Use. Available : https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/ [Accessed July 2019]

[34] C. Busso, S. Lee, S.Narayanan, "Analysis of emotionally salient aspects of fundamental frequency for emotion detection," in *IEEE Trans, Audio Speech Language Process*, 14 (4), pp. 582-596, 2009.

[35] J. Cahn, "The generation of affect in synthesized speech," in *J. Am. Voice Input/Output Soc. 8*, 1990, pp 1-19.

[36] D. Caims, J. Hansen, *Nonlinear analysis and detection of speech under stressed conditions*, J. Acoust. Soc. Am., 1994, pp. 3392-3400.

[37] W. Campell, "Databases of emotional speech," in *Proceedings of the ISCA*, 2000, pp. 34-38.

[38] D.M. Chandwadkar, M.S. Sutaone, "Role of features and classifiers on accuracy of identification of musical instruments," in *2nd National Conference on Computational Intelligence and Signal Processing*, 2012, pp. 66-70.

[39] A. Chanrungutai, C. A. Ratanamahatana, "Singing Voice Separation in Mono-Channel Music," in *Communications and Information Technologies*, 2008

[40] L. Chen, X. Mao, Y. Xue and L.L. Cheng, "Speech emotion recognition; Features and classification models," in *Digital Signal Processing 220*, 2012, pp. 1154-1160

[41] "Choosing a proper tolerance value in Logistic Regression (Scikit-learn)", Stack Overflow. Available : https://stackoverflow.com/questions/50599895/choosing-a-proper-tolerance-value-in-logistic-regression-scikit-learn [Accessed July 2019]

[42] G. Collier, "Beyond valence and activity in the emotional connotations of music," in *Psychol. Music* 35, 1, 2007, pp. 110–131.

[43] *Colossus Run SC. Spectral Centroid.* Available : https://yorkporc.wordpress.com/2014/03/15/colossus-run-sc-spectral-centroid/ [Accessed July 2019]

[44] J. W. Cooley, J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," in *Math. Comp.*, vol. 19, 1965, pp. 297-301

[45] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, S. Kollias, W. Fellenz, J.Taylor, "Emotion Recognition in human-computer interaction," in *IEEE Signal Processs*, Mag. 18, 2001, pp. 32-80.

[46] Danceability function, from Essentia. Available : https://essentia.upf.edu/documentation/reference/std_Danceability.html [Accessed March 2019]

[47] A.K. Datta, S.S. Solanki, R. Sengupta, S. Chakraborty, K. Mahto, A. Patranabis, "Music Information Retrieval," in *Signal Analysis of Hindustani Classical Music*, 2017, chap. 2.

[48] S. Davis, P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 28 No. 4, 1980, pp. 357-366

[49] R. Delbouys, R. Hennquin, F. Piccoli, J. Royo-Letelier, M. Moussallam, *Music Mood Detection Based on Audio and Lyrics with Deep Neural Net, Deezer.*

[50] J.R. Deller, J. H.L. Hansen, J. G. Proakis, *Discrete-Time Processing of Speech Signals*, 2000

[51] F. Dellaert, T. Polzin, A. Waibel, Alex, "Recognizing emotion in speech," in *ICSLP-1996*, 1996, pp. 1970-1973.

[52] A. della Faille de Leverghem, "Prédiction de la direction des taux de change à court terme : comparaison de méthodes", Louvain School of Management, Université catholique de Louvain, 2017. Prom. : Saerens, Marco.

[53] Deriving cost function using MLE : Why use log function ?, Mathematics. Available : https://math.stackexchange.com/questions/886555/deriving-cost-function-using-mle-why-use-log-function [Accessed July 2019]

[54] T.G. Dietterich, "Ensemble Methods in Machine Learning," in *International Workshop on Multiple Classifier Systems*, 2000, pp. 1-15.

[55] B.N. Do, *Neural Networks for Automatic Speaker, Language and Sex Identification*, Master thesis, University of Groningen, 2015.

[56] J.S. Downie, "The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research," in *Acoustical Science and Technology*, vol. 29, no. 4, 2008, pp. 247–255.

[57] P. Dupont. *INGI2262 : Decision Tree Learning (I)*, ICTeam Institute, Université Catholique de Louvain - Belgium.

[58] P. Ekman, "An argument for basic emotions," in *Cognition Emotion 6*, 3, 1992, pp. 169–200.

[59] Essentia. Available : https://essentia.upf.edu/documentation/ [Accessed February 2019]

[60] M. El Ayadi, M. S. Kamel, F. Karray, *Survey on speech emotion recognition : Features, classification schemes, and databases.*

[61] I. Engberg, A. Hansen, *Documentation of the Danish emotional speech database*, 1996.

[62] EqloudLoader function, from Essentia. Available : https://essentia.upf.edu/documentation/reference/streamingEqloudLoader.html [Accessed March 2019]

[63] J. Eulenberg, "Fundamental Frequency and the Glottal Pulse," in *Spring Semester 2011.* Available : https://msu.edu/course/asc/232/study_guides/F0_and_Glottal_Pulse_Period.html [Accessed June 2019]

[64] F. Fahy, *Sound Intensity*, 1995

[65] P.R. Farnsworth, *The social psychology of music*, The Dryden Press, 1958.

[66] G.T. Fechner, G. Howes, E.D. Boring, *Elements of psychophysics*,vol. 1, United States of America, 1966

[67] M. Feurer, F. Hutter, "Hyperparameter Optimization," in *Automated Machine Learning : Methods, Systems, Challenges*, Barcelona, 2018

[68] R. Feldman, "Techniques and Applications for Sentiment Analysis," in *ACM* 56, 2013, pp. 82-83.

[69] L. Feng, *Speaker Recognition*, 2004.

[70] Y. Feng, Z. Huang, Y. Pan, "Popular music retrieval by detecting mood," in *International Conference on Information Retrieval*, 2003, pp. 375-376

[71] J. Foote, S. Uchihashi, "The beat spectrum : a new approach to rhythm analysis," in iIEEE International COnference on Multimedia, 2001

[72] S. Furui, "Robush Methods in Automatic Speech Recognition and Understnading," in *Proc EUROSPEECH 2003*, Genova, 2003, pp. 1993-1998.

[73] J. Gall and V. Lempitsky, "Class-specific hough forests for object detection," in *Decision Forests for Computer Vision and Medical Image Analysis*, 2013, pp. 143–157

[74] S. Garcia, J. Luengo, F. Herrera, *Data Preprocessing in Data Mining*, 2015

[75] B.W. Gawali, S. Gaikwad, P. Yannawar and S.C. Mehrotra, "Marathi isolated word recognition system using MFCC and DTW features," in *Int. J. Inform. Technol.*, 2001, pp. 21-24.

[76] R. Genuer, J.-M. Poggi, C. Tuleau-Malot, "Variable selection using Random Forests".*Pattern Recognition Letters*, Elsevier, 2010,

[77] A. Gerniers, "Maximum entropy method for multi-label classification", Master degree thesis, EPL, Université Catholique de Louvain, 2018. Prom. : Saerens, Marco.

[78] C. Gobl, A.N. Chasaide, "The role of voice quality in communicating emotion, mood and attitude," in *Speech Commun.* 40 (1-2), 2003, pp. 189-212.

[79] M. Goto, "A real-time music-scene-description system: predominant f0 estimation for detecting melody and base line in real world audio signals," in *Speech Communication*, vol. 43, 2004, pp. 311-329.

[80] M. Goudbeek, J.P. Goldman, K. R. Scherer, Brighton, September 6 2009, pp. 1575-1578.

[81] S. Gupta, J. Jaafar, F.W. Ahmad, A. Bansal, "Feature extraction using MFCC," in *SIPU 4*, 2013

[82] I. Guyon, "A scaling law for the validation-set training-set ratio size," in *AT T Bell Laboratories*, 1997.

[83] P. Harrington, *Machine Learning in Action*, Manning Publications Co, Greenwich, USA, 2012

[84] T. Hastie, R. Tibshirani, J. Friendman, *The Elements of Statistical Learning: DAta Mining, Inference, and Prediction*. Springer-Verlag New York, second edition, 2003.

[85] K. Hevner, "Experimental studies of the elements of expression in music," in *American Journal of Psychology*, vol. 48, no. 2, 1936, pp. 246–267.

[86] K. Hevner, "Expression in music : A discussion of experimental studies and theories," in *Psychol. Review 48*, no. 2, 1935, pp. 186-204.

[87] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the ACM International Conference of Information Retrieval*, 1999, pp. 50-57

[88] X. Hu, J. Downie, C. Laurier, M. Bay, and A. Ehmann, "The 2007 MIREX audio mood classification task: Lessons learned," in *Proc. of the Intl. Conf. on Music Information Retrieval*, Philadelphia, PA, 2008.

[89] H. Hu, X. Xu, W. Wu, "Fusion of global statistical and segmental spectral features for speech emotion recognition," in *International Speech Communication Assocition-8th Annual Conference of the International Speech Communication Association*, Interspeech, vol. 2, 2007, pp 1013-1016.

[90] X. Hu, *Who knows the relationship among pitch, fundamental frequency and tone, intonation of voice*. Available : https://www.researchgate.net/post/Who_knows_the_relationship_among_pitch_fundamental_frequency_F0_and_tone_intonation_of_voice [Accessed June 2019]

[91] Y. Hu, X. Chen, D. Yang, "Lyric-based song emotion detecton with affective lexicon and fuzzy clustering method," in *Proceedings of the International Conference on Music Information Retrieval*, 2009

[92] X. Huang, A. Acero. and H. Hon, *Spoken Language Processing - A Guide to Theory, Algorithm, and System Development*, Prentice Hall PTR, New Jersey, 2001

[93] J.L. Flanagan, *Speech Analysis, Synthesis and Perception*, Springer- Verlag, New York, 1972

[94] H. Fletcher, W. A. Munsin, "Loudness, Its Defintion, Measurement and Calculation," in *Jour. Acous. Soc. Amer.*, 1939, pp. 82-108

[95] Fundamental frequency, available : https://medical-dictionary.thefreedictionary.com/fundamental+frequency [Accessed July 2019]

[96] R.W. Hamming, *The Art of DOing Science and Engineering : Learning to learn*, CRC Press, 1997

[97] J. Han, M. Kamber, Data Mining: Concepts and Techniques, San Francisco, 2006.

[98] R. Hasan, M. Jamil, G. Rabbani, S. Rahamn, "Speaker identification using mel frequency cepstral coefficients," in *ICECE 2004*, Dhaka, 28-30 December 2004.

[99] H. Hui, J. Jianming, X. Yuhong, C. Bo, S. Wu, and Z. Ling, "Language Feature Mining for Music Emotion Classification via Supervised Learning from Lyrics," in *ISICA 2008: Advances in Computation and Intelligence*, 2008, pp. 426–435.

[100] X. Hu, J.S. Downie, "Improving mood classification in music digital libraries by combining lyrics and audio," in *Proceedings of the 10th annual joint conference on Digital libraries*, pp.159-168.

[101] X. Hu, J.S. Downie , A. F. Ehmann, "Lyric text mining in music mood classification," in *10th International Society for Music Information Retrieval Conference*, International Music Information Retrieval Systems Evaluation Laboatory University of Illinois at Urbana-Champaign, 2009.

[102] D. Huron, *Sweet Anticipation: Music and the Psychology of Expectation*, MIT Press, Cambridge, 2006.

[103] D.W. Hosmer, S. Lemeshow, *Applied logistic regression*, 2nd Edition, 2000.

[104] "Infinite impulse response", *Wikipedia*. Available : https://en.wikipedia.org/wiki/Infinite_impulse_response [Accessed July 2019]

[105] Izotope RX7. Available : https://www.izotope.com/en/products/repair-and-edit/rx.html [Accessed March 2019]

[106] A.K. Jain, Fundamentals of Digital Image Processing, Prentice Hall, 1989.

[107] N. Japjowicz, M. Shah, *Evaluating Learning Algorithms: A Classification Perspective*, Cambridge University Press New York, NY, USA, 2011

[108] J.H. Jensen, M.G. Christensen, M. Murthi, S.H. Jensen, "Evaluation of MFCC estimation techniques for music similarity," in *Proceedings of the European Signal Processing Conference*, 2006, pp. 926-930

[109] D. Jurafsky, "Feature Extraction and Acoustic Modeling," in *Speech Recognition and Synthesis*, 2007.

[110] P. Juslin, P. Luakka, "Expression, perception, and induction of musical emotions: A review and questionnaire study of everyday listening", in *Journal of New Music Research*, vol. 33, no. 3, 2004, page 217.

[111] P. Juslinn J. Sloboda, *Music and Emotion: theory and research*. Oxford Univ. Press, 2001.

[112] T. Johnstone, K.R. Scherer, *Vocal Communication of Emotion, second ed., Guilford*, New York, 2000, pp. 226-235.

[113] S. Joglekar, *Cross Validation and the Bias-Variance tradefoff (for Dummies*. Available : https://codesachin.wordpress.com/2015/08/30/cross-validation-and-the-bias-variance-tradeoff-for-dummies/ [Accessed July 2019]

[114] Y. E. Kim, E. M. Schmidt, R. Migneco, B. G. Morton, P. Richardson, J. Scott, J. Speck, D. Turnbull, "Music emotion recognition : a state of the art review," in *11th International Society for Music Information Retrieval Conference*, 2010, pages 255-266.

[115] J.F. Kenney, E.S. Keeping, "Root Mean Square," in *Mathematics of Statistics*, Princeton, 1962, pp. 59-60

[116] S.G. Koolagudi, K.S. Rao, "Emotion recognition from speech: A review, in *Intern. J. of Speech Technology 15*, 2012, pp. 99–117.

[117] N. Kulkarni, V. Bairagi, "Use of Complexity Features for Diagnisos of Alzheimer Disease," in *EEG-Based Diagnosis of Alzheimer Disease*, 2018.

[118] M. Kumar, *Performance Analysis of LPC and MFCC Techniques in Automatic Speech Recognition*

[119] T. Kinnunen, "Spectral Features for Automatic Text-independent Speaker Recognition", University of Joensuu, Department of Computer Science, Dec. 2003

[120] C. Laurier, J. Grivolla, P. Herrera, "Multimodal music mood classification using audio and lyrics," in *Proc. of the Int. Conf. on Machine Learning and Applications*, 2008.

[121] C. Laurier, M. Sordo, J. Serra, and P. Herrera. "Music mood representation from social tags," in *Proc. of the Intl. Society for Music Information Conf.*, Kobe, Japan, 2009.

[122] C. Laurier, "Automatic Classification of Musical Mood by Content-Based Analysis," PhD thesis, Universitat Pompeu Fabra, Barcelona, Spain, 2011.

[123] R.S. Lazarus, *Emotion and Adaptation*, Oxford University Press, Oxford, UK, 1991.

[124] C. Lee, R. Pieraccini, "Combining acoustic and language information for emotion recognition," in *Proceedings of the ICSLP 2002*, 2002, pp. 873-876.

[125] C. Lee, S. Yildrim, M. Bulut, A. Kazemzadeth, C. Busso, Z. Deng, S. Lee, S. Narayanan, "Emotion recognition based on phoneme classes," in *Proceedings of ICSLP*, 2004, pp. 2193-2196.

[126] L. Leinonen, T. Hiltunen, *Expression of emotional-motivational connotations with a one-word uterance*, J. Acousti. Soc. Am. 102 (3), 1997, pp. 1853-1863.

[127] T. Li, M. Ogihara. "Detecting emotion in music," in *ISM*, Johns Hopkins University, 2003, pp. 239-240.

[128] T. Li, M Ogihara, "Semi-supervised learning from different information sources," in *Knowledge and Information Systems*, 7, pp. 289-309

[129] L. Muda, M. Begam, I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques," in *Journal of Computing*, Volume 2, issue 3, march 2010.

[130] Z.T. Liu, M. Wu, W.H. Cao, J.W. Mao, J.P. Xu, "Speech Emotion Recognition based on features selection and extreme learning machine decision tree," *Neurcomputing*,2017

[131] B. Liu, *Sentiment Analysis and Opinion Mining*, Morgan Claypool Publishers, May 2012.

[132] "Logistic Regression implementation, Scikit-learn". Available : https://scikit-learn.org/stable /modules/generated/sklearn.linear_model.LogisticRegression.html [Accessed May 2019]

[133] *Loudness* function, Essentia. Available : https://essentia.upf.edu/documentation/reference /streaming_Loudness.html [Accessed April 2019].

[134] L. Lu, D. Liu, H-J. Zhang, "Automatic Mood Detection and Tracking of Music AUdio Signals," in *IEEE transactions on audio, speech, and language processing*, Vol 14, 2006?

[135] "LyricWiki". Available : https://lyrics.fandom.com/wiki/LyricWiki [Accesed March 2019]

[136] C. Manliguez, *Generalized Confusion Matrix for Multiple Classes*, 2016.

[137] C. D. Manning, P. Raghavan, H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[138] S. Marsella, J. Gratch, "Computationally modeling human emotion," *ACM 57*, December 12 2014, pp. 56-67.

[139] R. Mayer, R. Neumayer, A. Rauber. "Combination of Audio and Lyrics Features for Genre Classification in Digital Audio Collections," in *Proceeding MM '08 Proceedings of the 16th ACM international conference on Multimedia*, 2008, pp. 159–168.

[140] C. McKay, J. A. Burgoyne, J. Hockman, J. B. L. Smith, G. Vigliensoni, I. Fujinaga, "Evaluating the genre classification performance of lyrical features relative to audio, symbolic and cultural features," in *Proceedings of the 11th International Society for Music Information Retrieval*, Québec, 2010

[141] A. Mehrabian, J. A. Russell, *An Approach to Environmental Psychology*, MIT Press, 1974.

[142] O. C. Meyers, "A Mood-Based Music Classification and Exploration System," Ph.D. dissertation, Massachusetts Institute of Technology, 2007.

[143] G. A. Miller, *WordNet: A Lexical Database for English*, 1995

[144] F. Miller, M. Stiksel, R. Jones. "Last.fm". Available : https://www.last.fm/, 2002, [Accessed February 2019]

[145] F. Miller, M. Stiksel, R. Jones, "Last.fm in numbers," in *Last.fm press material*, February 2008.

[146] T. Mitchell, *Machine Learning*, McGraw Hill, 1997.

[147] M. B. Mokhsin, N. B. Rosli, S. Zambri, N. D. Ahmad, S. R. Hamidi, "Automatic Music Emotion Classification using Artificial Neural Network based on Vocal and Instrumental sound timbres," in *Journal of Computer Science 10*, 2014.

[148] L. Muda, M. Begam and I. Elamvazuthi, "Voice recognition algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) techniques," in *J. Comput.*, 2010, pp. 138-143

[149] I. Murray, J. Arnott, "Toward a simulation of emotions in synthetic speech: A review of the litterature on human vocal emotion," in *J. Acoust. Soc. Am.* 93 (2), 1993, pp. 1097-1108.

[150] "MusicXMatch". Available : https://www.musixmatch.com/fr [Accessed March 2019]

[151] R.K. Nawasalkar, *Extracting Melodic Patter of 'Mohan Veena' from Polyphonic Audio Signal of North Indian Classical Music*

[152] I. Nedeljkovic, "Imagine classification based on fuzzy logic," in *Int. Arch. Photogrammetry* 34, 1994

[153] P. Norvig, S. Russel, *Artificial Intelligence: A Modern Approach*, (3rd Edition), 1994

[154] M. Nuzzolo. *Music Mood Classification.* Available : https://sites.tufts.edu/eeseniordesignhandbook/2015/music-mood-classification/ [Accessed may 2019]

[155] T. Nwe, S. Foo, L. De Silva, "Speech emotion recognition in speech using Markov models," in *Speech Commun* 41, 2003, pp. 603-623.

[156] J. D. O'Connor, G. Arnold, *Intonation of colloquial English*, London, 1973

[157] "Online Audio Converter". Available : https://online-audio-converter.com/fr [Accessed March 2019]

[158] "The origin of constants in mel-scale". Available https://dsp.stackexchange.com/questions/46209/the-origin-of-constants-in-mel-scale-formula [Accessed July 2019]

[159] C.E. Osgood, G.J. Suci, P.H. Tannebaum, *The Measurement of Meaning*, University of Illinois Press, 1957.

[160] J. Osmalsky, M. Van Droogenbroeck, J.J. Embrechts, 2014, "Performances of low-level audio classifiers for large-scale music similarity," in *Proceedings of the International COnference on Systems, Signals and Image Processing*, Dubrovnik, May 15, pp. 91-94.

[161] A. Oster, A. Risberg, "The identification of the mood of a speaker by hearing impaired listeners," in *Speech Transmission Lab. Quarterly Progress Status report 4*, Stockholm, 1986, pp. 79-90.

[162] G. Peeters, "A generic Training and Classification System for MIREX08 Classification Tasks: Audio Music Mood. Audio Music Mood, Audio Genre, Audio Artist and AUdio Tag," in *MIREX*, Philadelphia, United States, September 2008.

[163] C.K. Peng, S.V. Buldyrev, S. Havlin, M. Simons, H.E. Stanley, A.L. Goldberger, "Mosaic organization of DNA nucleotides," in *Phys. Rev. E 49*, 1994, pp. 1685-1689

[164] R.W. Picard, R. Picard, *Affective Computing*, vol. 252, MIT Press Cambridge, 1997.

[165] Pitch Melodia, Essentia. Available : https://essentia.upf.edu/documentation/reference/std_PitchMelodia.html [Accessed in June 2019]

[166] T. Plapinger, "Tuning a Random Forest Classifier". Available https://medium.com/taplapinger/tuning-a-random-forest-classifier-1b252d1dde92 [Accessed April 2019]

[167] R. Plutchik, *Emotion: A Psychoevolutionary Synthesis*, New York, 1980.

[168] G.E. Poliner, D. P. W. Ellis, F. Ehmann, E. Gomez, S. Steich, and B. Ong, "Melody transcription from music audio : Approaches and evaluation," in *IEEE Trans. on Audio, Speech and Language Processs*, vol. 15, no. 4, 2007, pp. 1247-1256.

[169] M. Porter, F. Martin, *An algorithm for suffix stripping*, 1980, pp. 130-137.

[170] M. Porter, F. Martin,*Snowball: A language for stemming algorithms*, 2001, pp. 130-137.

[171] P. Boersma, D. Weenink, Praat: doing phonetics by computer [Computer program]. Version 6.1, available : http://www.praat.org/

[172] K.M.M. Prabhu, *Window Functions and Their Applications in Signal Processing*, CRC Press, October 12 2017.

[173] PredominantPitchMelodia function, from Essentia. Available: https://essentia.upf .edu/documentation/reference/std_PredominantPitchMelodia.html [Accessed february 2019]

[174] J.R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993

[175] J.R. Quinlan, "Induction of Decision Trees". *Mach. Learn. 1, 1*, March 1986, pp. 81-106

[176] L. Rabiner, R. Schafer, *Digital Processing of Speech Signals*, first ed., Perason Education, 1978.

[177] F.H. Rachman, R. Sarno, C. Fatichah, "Music Emotion Classification based on Lyrics-Audio using Corpus based Emotion," in *International Journal of Electrical and Computer Engineering*, vol. 8, No. 3, June 2018, pp. 1720-1730.

[178] T. Ramalingam, P. Dhanalakshlmi, *Speech/music classification using wavelet based feature extraction techniques*, 2013, pp. 34-44.

[179] K.R. Rao, P. Yip, *Discrete Cosine Transform: Algorithms, Advan- tages, Applications*. MA: Academic Press, 1990

[180] S. Raschka, *Confusion Matrix*, available : http://rasbt.github.io/mlxtend/user_guide/evalu ate/confusion_matrix/ [Accessed June 2019]

[181] S. Raschka, *Music Mood Classification Using Random Forests*. Available : https://github.com/rasbt/musicmood/blob/master/code/classify_lyrics/random_forests.ipynb [Accessed March 2019]

[182] N.A. Remington, L.R. Fabrigar, P.S. Visser, "Reexamining the corcumplex model of affect," in *J. Personality Social Psychol 79*, pp. 289-300.

[183] RhythmDescriptors function, from Essentia. Available : https://essentia.upf.edu/docu mentation/reference/std_RhythmDescriptors.html [Accessed March 2019]

[184] R. Malheiro, R. Panda, P.R. Gomes, R.P. Paiva, *Classification and Regression of Music Lyrics: Emotionally-Significant Features*, 2016.

[185] Ridge and Lasso Regression: A Complete Guide with Python Scikit-Learn, Towards data Science. Available : https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b [Accessed July 2019]

[186] D. Robinson, *Replay Gain - A proposed standard*. Available : http://replaygain .hydrogenaud.io/proposal/ [Accessed June 20119]

[187] D. Robinson, "Equal Loudness filter," in *Replay Gain - A proposed standard*. Available : http://replaygain.hydrogenaud.io/proposal/ [Accessed June 20119]

[188] J.A. Russell , "A circumspect model of affect," in *Journal of Psychology and Social Psychology*, vol. 39, no. 6, 1980, page 1161.

[189] M. Ryynänen, A. Klapuri, "Automatic transcription of melody, bass line, and chords in polyphonic music," in *Computer Music J.*, vol 32, no. 3, 2008, pp. 72-86.

[190] L. Rokach. "Ensemble-based classifiers," in *Artificial Intelligence Review*, 33.1 , 2010, pp. 1–39.

[191] M. Saerens, slides from the course SINF2275 - *Data mining*, Université Catholique de Louvain.

[192] J. Salamon, E. Gomez, "Melody Extraction from Polyphonic Music Signals using Pitch Contour Characteristics," in *IEEE Transactions on audio, speech and language processing*, 2010.

[193] J. Salamon, E. Gomez, D.P.W. Ellis, G. Richard, "Melody Extraction from Polyphonic Music Signals : Approaches, applications, and challenges," in *IEEE Signal Processing Magazine*, Vol. 31, March 2014;

[194] G. Saltonand, C. Buckley, *Term weighting approaches in automatic text retrieval,Information*, 1988, pp. 513-523.

[195] L. Sbatellan, "Speech analysis 1/3 The Human Voice : prosody and voice descriptors in digital audio," in *Natural Language Processing*.

[196] H.E. Schaefern, "Music-Evoked Emotions-Current Studies," in *Frontiers in neuroscience*, vol. 11, November 24 2017

[197] E.M. Schmidt, D. T. Urnbull, K. IM, Y. E, "Feature selection for content-based, time-varying musical emotion regression," in *Proceedings of the ACM International Conference on Multimedia Information Retrieval*, 2010, pp. 267–274.

[198] E. Schubert, "Update of the Hevner adjective checklist," in *Perceptual and Motor Skills*, vol. 96, 2003, pp. 1117–1122.

[199] M.T. Shami, M.S. Karnel, "Segment-based approach to th recognition of emotions in speech," in *IEEE International Conference on Multimedia and Expo 2005*, 2005.

[200] Pedregosa et al, "Scikit-learn: Machine Learning in Python," in *JMLR 12*, 2011, pp. 2825-2830.

[201] F. Sebastiano, "Machine Learning in automated text categorization, in *ACM Comput. Surveys 34*," 2005, pp. 1- 47

[202] S. Shalev-Shwartz, S. Ben-David, *Understanding Machine Learning : From Theory to algorithms*, Cambridge, 2014

[203] M. Shaneh, A. Taheri, "Voice command recognition system based on MFCC and VQ algorithms," in *World Acad. Sci. Eng. Technol.*, vol. 33, pp. 534-538.

[204] C.E. Shannon, "A Mathematical Theory of Communication," in *Bell System Technical Journal*, vol. 27, July 1948, pp. 379-423.

[205] "Introduction to Signal Levels", *Discovery of Sound in the Sea*. Available : https://dosits.org/science/advanced-topics/introduction-to-signal-levels/ [Accessed in July 2019]

[206] S. Singh, P. Gupta, "Comparative study ID3, CART and C4.5 Decision Tree Algorithm : A survey," in *International Journal of Advanced Information Science and Technology (IJ*, July 2014.

[207] J. Six, "Digital Sound Processing and Java," in *Documentation for the TarsosDSP Audio Processing Library*, Ghent, August 30 2012.

[208] J. Skowronek, M. McKinney, and S. Van De Par, "Ground truth for automatic music mood classification, in *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR 2006*, Victoria, Canada, October 8-12 2006, pp. 396-396.

[209] D. Ek, M. Lorentzon. "Spotify". Available : https://www.spotify.com, 2006 [Accessed March 2019]

[210] J. Shotton, A. Fitzgibbon, C. Mat, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. "Real-time human pose recognition in parts from single depth images," in *Computer Vision and Pattern Recognition (CVPR)*, 2011

[211] C. Smith, M. Koning, *Decision Trees and Random Forest : A visual Introduction For Beginners : A simple Guide to Machine Learning with Decision trees*, 2017

[212] C. Strapparava and A. Valitutti, "WordNet-Affect : an Affective Extension of WordNet", LREC, 2004, pp. 1083-1086.

[213] S. Streich, P. Herrera, "Detrended Fluctuation Analysis of Music Signals: Danceability Estimation and further Semantic Characterization," in *Proceedings of the AES 118th Convention*, Barcelona, Spain, 2005

[214] L. Su, C Yeh, J. Liu, J. Wang, Y. Yang, "A systematic evaluation of the bag-of-frames representation for music information retrieval," in *IEEE Trans. Multimedia, 16*, 2014, pp. 1188-1200.

[215] M. Slaney, "Auditory Toolbox. Version 2," in *Technical Report*, n. 10, Interval Research Corporation, 1998

[216] R. Sonar, P.R. Deshmukh, "Multiclass Classification: A Review," in *International Journal of Computer Science and Mobile Computing*, Vol. 3 Issue 4, April 2014, pp. 65-69

[217] S.S. Stevens, J. Volkmann, "The relation of pitch to frequency: A revised scale," in *The American Journal of Psychology*, vol. 53, 194, pp. 329–353

[218] S.S. Stevens, "On the psychophysical law" in *Psychological Review*, chap. 64, 1957, pp. 53–181.

[219] M. Talbot-Smith, "Sound, speech and hearing," in *Telecommunications Engineer's Reference Book*, 1993.

[220] P.N. Tan, S. Michael, V. Kuman, *Introduction to Data Mining*, Boston, 2006.

[221] T. Jianhua, K. Yongguo, *Features Importance Analysis for Emotional Speech Classification. Affective Computing and Intelligent Interaction Proceedings*, 2005, pp. 449-457.

[222] R. Tedesco, "Speech processing, automoatic speech recognition and text-To-Speech," in *Natural Language Processing*, Politecnico di Milano, Milan.

[223] "Tf-idf Model for Page Ranking", *Geeks For Geeks*. Available : https://www.geeksforgeeks.org/tf-idf-model-for-page-ranking/ [Accessed April 2019]

[224] TfidfVectoriser function, from scikit-learn. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html [Accessed March 2019]

[225] I.S. Tiwary, T. Siddiqui, *Natural Language Processing and Information Retrieval*, Oxford University Press, 2007.

[226] R.E. Thayer, *The Biopsychology of Mood and Arousal*, Oxford University Press, Oxford, 1989.

[227] M. Tolos, R. Tato, T. Kemp, "Mood-based navigation through large collections of musical data," in *Proceedings of the IEEE Consumer Communications Network Conference*, 2005, pp. 71–75

[228] R. Typke, "Music retrieval based on melodic similarity," Ph.D. dissertation, Utrecht University, Netherlands, 2007

[229] G. Tzanetakis, "Marsyas submissions to MIREX 2007," in *IMSIR*, 2007.

[230] S. Vaishnav, S. Mitra, "Speech Emotion Recognition: A review," in *Internatonial Research Journal of Engineering and Technology*, vol. 3, April 2016, pp. 313-316

[231] C.J. Manning, P. Raghavan and H. Schütze, *Introduction to Information Retrieval Cambridge University Press*, 2008.

[232] A. Verikas, A. Gelzinis, and M. Bacauskiene. "Mining data with random forests: A survey and results of new tests". *Pattern Recognition*, 44(2), 2011, pp. 330–349

[233] D. Ververidis, C. Kotropoulos, "Emotional speech classification using Gaussian mixture models and the sequential floating forward selection algorithm," in *IEEE International Conference on Multimedia and Expo 2005*, July 2005, pp. 1500-1503.

[234] E. Vincent, T. Virtanen, S. Gannot, *Audio Source Separation and Speech Enhancement*, Wiley, 2018.

[235] "Voiced and Unvoiced Speech Overview". *Samueli School Of Engineering*. Available : http://www.seas.ucla.edu/dsplab/vus /over.html [Accessed July 2019]

[236] J. Volkmann, S.S. Stevens, E.B. Newman, "A scale for the measurement of the psychological magnitude pitch," in *The Journal of the Acoustical Society of America. 8 (3)*, 1937.

[237] Understanding min_df and max_df in scikit CountVectorizer.

[238] M.Y. Wang, N.Y. Zhang, H.C. Zhu, "User-adaptive music emotion recognition," in *Proceedings of the IEEE International Conference on Signal Processing*, 2004, pp. 1352-1355.

[239] "What are the differences between Logistic Function and Sigmoid Function ?", StackExchange. Available : https://stats.stackexchange.com/questions/204484/what-are-the-differences-between-logistic-function-and-sigmoid-function [Accessed July 2019]

[240] "What exactly is tol (tolerance) used as stopping criteria in sklearn models?", StackExchange. Available : What exactly is tol (tolerance) used as stopping criteria in sklearn models? [Accessed July 2019]

[241] A. Wieczorkowska, "Towards extracting emotions from music," in *Proceedings of the International Workshop on Intelligent Media Technology for Communicative Intelligence*, 2004, pp. 228–238.

[242] C. Williams, K. Stevens, "Emotions and speech: some acoustical correlates," in *J. Acoust. Soc. Am. 52* (4 Pt 2), 1972, pp. 1238–1250.

[243] S. Wood, "What are formants ?" in *Praat for beginners*, 15 Jan 2005

[244] Wordnet. Available : https://wordnet.princeton.edu/ [Accessed March 2019]

[245] T. L. Wu, S.K. Jeng, Regrouping of expressive terms for musical qualia, *Proceedings of the International Workshop on Computer Music Audio Technology*, 2007

[246] Y. Yang, Y. Lin, H. Chen, "A regression approach to music emotion recognition," in *IEEE Transactions on audio, speech and language processing*, vol. 16, No. 2, pp. 448-457.

[247] Y.H. Tang, H.H. Chen, "Machine recognition of music emotion : A review," in *ACM Trans. Intell. Syst. Technol. 3*, pp. 40-40.

[248] E. Kim. Youngmoo, E. M. Schmidt, R. Migneco, "Music emotion recognition : a state of the art review," in *11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, 2010.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**