UNIVERSITAT POMPEU FABRA

Automatic Classification of Musical Mood by Content Based Analysis

Cyril Laurier

TESI DOCTORAL UPF / 2011

Director de la tesi:

Dr. Xavier Serra i Casals
Dept. of Information and Communication Technologies
Universitat Pompeu Fabra, Barcelona, Spain

Dissertation submitted to the Deptartment of Information and Communication Technologies of Universitat Pompeu Fabra in partial fulfillment of the requirements for the degree of

DOCTOR PER LA UNIVERSITAT POMPEU FABRA

Music Technology Group (http://mtg.upf.edu), Dept. of Information and Communication Technologies (http://www.upf.edu/dtic), Universitat Pompeu Fabra (http://www.upf.edu), Barcelona, Spain.

To my Family and Friends.

Acknowledgements

This research was conducted from October 2006 to July 2011 at the Music Technology Group in Barcelona, supervised by Xavier Serra and Perfecto Herrera.
There are many people I would like to thank. First, I wish to thank Xavier Serra for offering me the opportunity to work at the MTG. The second person I want to thank is Perfe, I have no words to express how much he helped me in this research. During my years at the MTG, I have been working on several projects, but especially a European Project called PHAROS. I thank Pedro Cano for giving me the chance to coordinate the project, and a special thank also to Martin Blech and Mohamed Sordo for all the work and fun they brought to this project.
I also want thank my MIR colleagues and their very useful suggestions since my early works: Emilia Gómez, Enric Guaus and especially Joan Serrà.
Working in the MIR technology transfer team has been a great opportunity for me to meet very talented people such as Nicolas Wack, Roberto Toscano, Andreas Beisler, Eduard Aylon, Owen Meyers, Koppi, Jordi Funollet and Vincent Ackermann.
There are also many people that I met at the Music Technology Group that I wish to thank for many different reasons: Ricard Marxer, Amaury Hazan, Jens Grivolla, Oscar Celma, Dmitry Bogdanov, Graham Coleman, Ferdinand Fuhrmann, Martin Haro, Piotr Holonowicz, Oscar Mayor, Hendrik Purwins, Justin Salamon, Gerard Roma, Alba Rosado and Cristina Garrido.
Finally, all this would not make a lot of sense without thinking about all my family, my parents, grand parents, my sister and of course Maya.

Abstract

Digital music is becoming a major part of the user experience with computers and mobile devices. Automatically organizing this content is a huge challenge. In this work, we focus on automatically classifying music by mood. For this purpose, we propose computational models using information extracted from the audio signal. The foundations of such algorithms are based on techniques from the fields of signal processing, machine learning and information retrieval. First, by studying the tagging behavior of a music social network with dimensionality reduction techniques, we find a relevant model to represent mood. We believe that this new methodology can be applied to other domains as well. Then, we propose a method for automatic music mood classification and detail the results for different types of classifiers. We analyze the contributions of audio descriptors and how their values are related to the observed mood, trying to find explanation from psychology and musicology. We also propose a multimodal version of our algorithm using lyrics information, contributing to the field of text retrieval with a new model based on key words differentiating categories. Moreover, after showing the relation between mood and genre, we present a new approach using automatic music genre classification. We demonstrate that genre-based mood classifiers give higher accuracies than standard audio models. Finally, we propose a rule extraction technique to explicit the strategy behind our models. This method allows to make sense of the classifiers and to understand how they can predict the musical mood. All the proposed algorithms are evaluated with user data. Our audio based approaches, adapted to the context, have been evaluated in international evaluation campaigns.

Resumen

La música en formato digital forma parte de nuestras vidas. Automatizar la organización de estos datos es un gran desafío. En esta tesis, nos centramos en la clasificación automática de música a partir de la detección de la emoción que comunica. Para conseguirlo, proponemos modelos usando informaciones extraídas de la señal de audio mediante técnicas de procesamiento de señales, aprendizaje automático y recuperación de información. Primero, estudiamos como los miembros de una red social utilizan etiquetas y palabras clave para describir la música y las emociones que evoca. Con una técnica para reducir la complejidad dimensional de este problema, encontramos un modelo para representar los estados de ánimo. Luego, proponemos un método de clasificación automática de emociones y detallamos los resultados para distintos tipos de clasificadores. Analizamos las contribuciones de descriptores de audio y cómo sus valores están relacionados con los estados de ánimo, intentando encontrar explicaciones desde un punto de vista psicológico y/o musicólogico. Proponemos también una versión multimodal de nuestro algoritmo, usando las letras de canciones con un nuevo método de clasificación basado en las palabras claves para distinguir categorías de emociones. Finalmente, después de estudiar la relación entre el estado de ánimo y el género musical, presentamos un método usando la clasificación automática por género. Mostramos que clasificadores basados en el género obtienen mejores resultados que otros métodos estándar. A modo de recapitulación conceptual y algorítmica, proponemos una técnica de extracción de reglas para entender como los algoritmos de aprendizaje automático predicen la emoción evocada por la música. Nuestros algoritmos han sido evaluados con datos de usuarios y en concursos de evaluación internacionales.

Preface

Back in 2006, when I started this research, I remember Perfecto Herrera listing several possible topics, leaving it open to other ideas. There was no doubts, I was fascinated by the topic I just picked: Music, Machine Learning and Emotions ! There had been a very few works published on this at that time, and no clear path had been traced to start on a solid basis. This led to a lot of questions that made be discovering the literature about mood and emotions in many disciplines, also attending to conferences more psychology or neuroscience oriented. Even if there was some solid research done, unfortunately, there was no clear or widely adopted theory to be completely confident in. Should emotions be represented as categories, or dimensions, and which ones? Should they be represented in another way? How? One of this thesis contribution is to summarize these approaches and to offer a new perspective based on a large online community analysis.

After several years and challenges, I must say it was not an easy topic. Many times I wished I could work on a subject with reliable and completely objective basis. Nevertheless, after learning a lot about emotions and machine learning, I am still fascinated by both topics and I am proud that they could meet in this work.

What you are about to read has been published in several international conferences, a journal and a book chapter. Our approaches have been proved to be within the highest accuracies in an international evaluation campaign. Also, European projects and commercial products are using the direct outcomes of this research. I hope you enjoy the reading.

Contents

List of figures

List of tables

CHAPTER 1

Introduction

"If I were not a physicist, I would probably be a musician. I often think in music. I live my daydreams in music. I see my life in terms of music. ... I get most joy in life out of music." Albert Einstein

1.1. Motivation

Music is ubiquitous in everyday life. But why do we listen to music ? Why do people enjoy music? Why is music so important in our lives ? There are many explanations at different levels of interpretation and one of them is that music easily induces strong emotions. This emotional experience is one of the main factors explaining people's passion about music. Stating that music and emotions have a close relationship is no revelation. An obvious motivation for music composers to compose is to express their sentiments. Performers like to interpret musical pieces to induce feelings to the audience (and to themselves). People often listen to music to feel something, from a small arousal to strong emotions. The relation between music and emotion have fascinated human beings since antiquity. However the mechanisms behind are still poorly understood. With the development of machine learning algorithms and content analysis, it has become possible to automatically analyze and annotate music using audio data. For instance, automatic genre classification has been studied for several years and current algorithms are considered as satisfying (see Guaus (2009)). In that context, can we similarly develop algorithms that are capable of detecting emotions in music? How a computer can be able to "perceive" emotions? Will this reflect precisely human perception? How can we deal with the subjectivity of emotions? What can be the applications of such systems? These are the main question we start to answer in this thesis.
At first glance, studying emotions can be somehow frightening. First, because it seems very complex, and also because it involves mechanisms that can be used for undesired purposes such as intrusive advertisement. But it is also truly fascinating. It can have applications in treating people using smart music

therapy. Indeed, machines with musical abilities can select, play and even compose music conveying targeted emotions. The technology we will explore in this thesis will enable to detect emotions from raw audio material, which is directly from the digital signal. We will also investigate how to represent emotions, based on data generated by an online community of music listeners. Lyrics will also be considered as a source of emotional information and finally we will use the musical genre to build better computational models.

1.2. Outline of the thesis

This thesis aims at developing methods to automatically classify music by mood. The basic information we start with is the audio signal. Consequently, we use techniques from signal processing to extract relevant information from this signal. To make computational models of emotions in music, we employ machine learning algorithms with a supervised learning approach, meaning that we base our models on examples. While the main focus is on audio, using low-level (close to the signal) and high-level (close to human perception) features, we also make experiments using lyrics, with text retrieval techniques. For this research, and because a few has been done on this topic, we follow a general discipline of simplicity, trying to makes things work for a general purpose.

In Chapter 2, we review the literature related to this thesis. We first summarize the definitions of emotion and mood, showing perspectives from different disciplines. After clarifying the terminology, we define the mood and emotion in the musical context. We also detail how emotions can be represented, following the literature from psychology and what are the important musical features. Then, we proceed with an explanation of the basic techniques for music classification, from extracting audio features to building classifiers. Finally, we review the literature about music mood classification.

In Chapter 3, we make a study that will serve as a basis for the remainder of the thesis. We analyze mood tags from a large online music community to understand what would be a relevant mood representation to use in our computational models. From the huge amount of tags, we create a useful space that we call the "Semantic Mood Space", representing the relationships between mood tags. This is generated based on the co-occurrence of tags that users associate with musical tracks and text retrieval techniques for dimensionality reduction. We evaluate how well-known representations fit into that semantic space, and demonstrate that basic emotion clusters directly emerge from the social network data.

In Chapter 4, we propose a technique to create a dataset of examples (ground truth) based on a large community of music listeners (crowd) and few listeners that we trust (experts). This first step shows the relevance to manually validate the data with listeners, not taking the tags for granted. From this ground truth,

we study the importance of several key audio features, trying to explain why there are related to particular moods. We detail how to build a classifier that we evaluate in accuracy and robustness. As a main result, we show that we are able to build reliable mood classifiers with high accuracies, and we validate our approach achieving good results in international evaluations.
In Chapter 5, we add another source of information: lyrics. Using text retrieval techniques, we show that we can obtain higher accuracies mixing audio and lyrics data. A new method is proposed, called "Language Model Differences", interesting by its relative simplicity to implement. This approach also gives us a clear indication on the terms that are especially important to discriminate between emotions.
In Chapter 6, we first analyze the relation between mood and genre. We show a clear relation, and based on this observation, we propose a method to exploit this information. We obtain a new genre-based mood classification algorithm, significantly increasing the accuracy of the original model.
Chapter 7 provides a summary of contributions and perspectives for future research on music mood classification. It also concludes this thesis.

CHAPTER 2

Literature Review

"Everyone knows what an emotion is, until asked to give a definition. Then, it seems, no one knows" Fehr & Russell (1984).

2.1. Introduction

The literature review is divided into four main parts. First, we discuss about the definition of mood and emotion from different perspectives and we decide about what terminology to use. Then, we focus on the musical case. In particular, we detail the representation models that appear in the literature and how some musical features are related to emotions. The third section is devoted to the music classification problem and to the methods used to extract audio features and classify musical pieces. This is mainly about explaining the principles of supervised learning, signal processing for audio feature extraction, machine learning technique to build classifiers and measures to evaluate their accuracies. Finally, we give an overview of the related works in music mood classification.

2.2. What are Moods and Emotions?

What are emotions? This is the first question we will try to answer. And what about similar words such as "mood" or feeling"? The terms "emotion", "mood" or "feeling" are often employed in everyday's life. However, clear definitions of those words are rarely stated. Moreover, the perceived difference between those words is vague. If we look at the Oxford American Dictionary, emotion is a "natural state of mind deriving from one's circumstances, mood or relationship with others, any of the particular feeling that characterize such a state of mind". In the same dictionary, mood is a "temporary state of mind or feeling" and feeling is defined as an emotion. Dictionaries are, by definition, self referencing and it is hard from those explanations to differentiate between these three terms. If emotion and feeling appear to be synonyms, we identify

a slight difference with "mood". Indeed the definition of mood includes a temporal aspect, but we need a deeper semantic analysis.

2.2.1. Definitions

Emotions are difficult to define and measure. Emotion can be defined as an everyday concept ("folk-theory") but also as a scientific theory. The common assumption is that there are emotions that make us feel good and others bad. Also, it is commonly accepted that some people are more "emotional" than others, as pointed out by Sloboda (2001).
But who should be the expert to talk about emotions? Who can clearly define what it is? Looking at the scientific literature, there are many studies coming from psychology but also neuroscience (with sub-fields such as affective neuroscience), sociology, philosophy, anthropology and biology.
An emotion could be defined as an intense mental state arousing the nervous system and invoking physiological responses. According to Damasio (1994), emotions are a series of body state changes that are connected to mental images that have activated a given brain subsystem (e.g., the music processing subsystem). So emotions involve physiological reactions but also they are object-oriented and provoke a categorization of their object: "if the emotion is one of fear its object must be viewed as harmful" (Davies (2001), p. 26). Emotions also induce an attitude towards the object. Moods could be considered as lasting emotional states. They are not object oriented and take into account quite general feelings. Moods and emotions can be considered as very similar concepts in some cases, for instance happiness, sadness and anger can be seen as both moods and emotions. However some emotions can only be considered as transient, such as surprise (if we consider surprise as an emotion).
After reviewing of 92 definitions, Kleinginna & Kleinginna (1981) proposed the following one trying to reach the best agreement:
"Emotion is a complex set of interactions among subjective and objective factors, mediated by neural/hormonal systems, which can (a) give rise to affective experiences such as feelings of arousal, pleasure/displeasure; (b) generate cognitive processes such as perceptually relevant effects, appraisals, labeling processes; (c) activate widespread physiological adjustments to the arousing conditions; and (d) lead to behavior that is often, but not always, expressive, goal-directed, and adaptive.".
Studying emotions is problematic. The experimental settings are complex in order to avoid influencing the emotional experience of a subject. Moreover there is a great variability among subjects and also across time for the same person. In psychology, there are three main possibilities to gather evidence about emotions: physiological measurements, self-report and expressive behavior. The most commonly used being self-report. Results from this field yielded to different observations that we summarize here.

Emotions are functional. Although there is no direct link between feeling emotions and the achievement of goals (phenomena called *non-instrumentality* by Frijda (1986)), a certain behavior is observed when experiencing particular emotions (with a great variability among subjects and contexts). There is a common view that the key function of emotions is to guide behavior. The behavior provoked by emotions has been developed with regards to successful interaction with the environment, serving functions that are not always conscious are rarely intentional.

Emotions induces physiological changes. Findings by Pike (1972) suggest that emotional responses to music includes stable moods, transient emotions and feelings of pleasure. Other studies have shown that, like in other contexts, experiencing emotions induces physiological changes like heart rate or even triggers intense responses like 'thrills' or 'chills'.

Emotions are social. Even if emotions can be experienced alone, it seems that they provoke more intense reactions when other people are present. Emotions are contagious, others emotional expressions influences our behavior. There is an interesting paradox here, when considering the western-culture attitude when attending public performances of "serious" music (classical, baroque, contemporary, even jazz). The typical behavior is to be silent and avoid any emotional expression, although self-report studies demonstrated that strong emotions are perceived (see Gabrielsson (2001)). This is however not applicable to any music. During rock concerts, people tend to express their emotions and to share them with other people from the audience. However, in classical music or event opera (which is historically surprising), a silent behavior would be well considered or even required by the audience.

Emotions are universal and cultural. This contradictory statement reveals both aspects of emotions (common to all human being and specific to some cultures) and reflects many research results, like the ones summarized by Wierzbicka (1999). The main problem is to sort out the culture-specific from the universal.

2.2.2. Terminology

Scientists have used many different related terms to explain similar phenomena: affect, emotion, mood, feeling, arousal etc... Looking at the literature those terms are often employed interchangeably. This confusion is most probably due to the fact that there are no clear or standardized definitions of those terms. However, we can note some differences. Affect is often defined as more general than emotion (Oatley & Jenkins (1996), p124) and seems to be more related to the positive and negative aspects (usually called "valence"). Moods are considered to last longer than emotions (or to be an emotion that lasts) and to have a less clear stimuli. Davidson (2001) suggest that "emotion

bias action whereas moods bias cognition". However, in the case of music, the behavioral influence of emotions is unclear. It is not set that emotions experienced in music will directly bias actions more than moods could. Meyer (1956) talks about mood as a long lasting emotional state. Most psychology research seems to prefer the term emotion, since they usually focus on the human responses to emotion stimuli. However, if we are interested to classify music, we should focus on aspects that are less objective than what emotion the listener feels. We would rather be interested in which mood is carried by the music. This is why we would prefer using the term mood classification than emotion classification. However, as emotion and mood are very related, and as also noted by Kim et al. (2010), used interchangeably in the literature, we will simplify our terminology by using both emotion and mood for the same idea.

2.3. The Musical Case

Why does music convey emotion? Emotion and expressive properties of musical elements have been studied since the time of ancient Greece (see Juslin & Laukka (2004)). The fact that music induces emotions is evident for everyone. However we do not intuitively apprehend why. Emotions are mostly said to be complex and to involve a complicated combination of cognition, positive or negative feeling changes, appraisal, motivation, autonomic arousal, and bodily action tendency or change in action readiness.

Although emotions have been studied in psychology for decades, music was rarely mentioned. Emotional reaction to music seems to be considered as less important, probably because its potential vital role in daily life is not perceived. Also, its relation to aesthetic has not been judged as worth studying by a majority of scientists (as well as for other arts like paintings, drama, cinema etc...). Nevertheless the tendency is changing and there is a recent important growth in number of studies about emotions and music (see Sloboda (2001)).

Understanding how music conveys emotion is not trivial. Kivy (1989) gives two such hypotheses. The first might be a "hearing resemblance between the music and the natural expression of the emotion". Some musical cues can induce emotions because of their similarity to speech. One example is "anger" where the loudness and the spectral dissonance (derived from frequency ratios and harmonic coincidence in the sound spectrum and based on psychoacoustic tests) are two components we can find in both an angry voice and music. However it might not always be that simple. The second hypothesis Kivy gives is the "accumulated connotations a certain musical phenomena acquire in a culture". In that case, we learn in our culture which musical cues correspond to which feeling. Most probably, both hypotheses are valid. Frijda (1986) argues for a notion of emotions as action tendencies where "various emotions humans or animals can have - the various action readiness modes they may

experience or show - depends upon what action programs, behavior systems, and activation or deactivation mechanisms the organism has at its disposal.". As pointed out by Nussbaum (2007), this correlates with results in neuroscience from scientists such as Damasio (1994).

Grewe et al. (2007) demonstrated that the intensity of the emotion induced by music could vary depending on personal experience and musical background. If a musician knows and has studied the piece for a performance, he/she is more likely to rate the intensity of the emotion higher. This is an auto-reinforcement by training. We can also imagine that listening to a musical piece too many times can create the opposite behavior. Almost everyone has experienced the fact of being bored, or less and less sensitive to a musical piece they used to love. Consequently, the emotion induced by the same song on the same subject fluctuates a lot in time, also depending on the current mental state of the listener. Besides, it is important to notice that emotions in music are not restricted to adults or musically trained people. The emotional processing of music starts at an early age. Four-months-old children have a preference for consonant (pleasant) over dissonant (unpleasant) music (see Trainor et al. (2002)). At five years old, they can distinguish between happy and sad music using the tempo (sad = slow, happy = fast), but at six, they use information from the mode (sad = minor, happy = major) such as adults do (Dalla Bella et al. (2001)).

Studies in neuroscience, exploiting the current techniques of brain imaging also give a hint about the emotional processing of music, with some schemas of the brain functions involved (see Koelsch et al. (2006)). Gosselin et al. (2005) demonstrated that the amygdala, well established to have an important role in the recognition of fear, is determinant in the recognition of scary music. Blood & Zatorre (2001) revealed that music creating highly pleasurable experience like "shivers-down-the-spine" or "chills" activate regions in the brain involved in reward and motivation. It is worth noticing that these areas are also active in response to other euphoria-inducing stimuli like food, sex and drugs. Huron (2006) simply states that music making and listening are primarily motivated by pleasure and that the contrary is biologically implausible (p. 373). Meyer (1956) describes the importance of expectation as a tool for the composer to create emotions. This work has been continued and formalized as the ITPRA1 theory by Huron (2006). One important way to control the pleasure in a musical piece is to play with this feature by delaying expected outcomes and fulfilling our expectation.

Additional research by Menon & Levitin (2005) seems to have also found the physical connections between music and mood alteration by means of antidepressants: the latter act on the dopaminergic system which has one of its main centers in the so-called nucleus accumbens, a brain structure that also receives a dramatic degree of activation when listening to music. These results are coherent with the work from Lazarus (1991), when he argues that emotions are evolutionary adaptations, to evoke behaviors that improve chances for survival

and procreation, and with Tomkins (1980) view that emotions can be understood as "motivational amplifiers". It links music with survival related stimuli. Often, damages to emotional controls limiting the normal functionability of the emotional behavior are disastrous for people (Damasio (1994)). Moreover people who did not develop social emotions seem incapable of appreciating music (Sacks & Freeman (1994)). However, this evolutionary adaptation theory can be balanced by the fact that most emotional responses to music are neither used to achieve goals, nor practically related to survival issues. This argument is used by researchers who assume that music cannot induce basic survival emotions, but more "music-specific emotions" (Scherer & Zentner (2001), p. 381). Nonetheless, other notable researchers, in Panksepp & Bernatzky (2002),affirm about music that it is "remarkable that any medium could so readily evoke all the basic emotions of our brain". This is one of the multiple contradictions we can observe in current research on music and emotions. As pointed out by Juslin & Västfjäll (2008), the literature presents a confusing picture with conflicting views. Nevertheless there is no doubt that music induces emotion because of the related context. It evokes emotions from past events because it is associated in our memory to emotional events.

When talking about emotion and music, one important distinction to make is the difference between induced and perceived emotions (see Juslin & Laukka (2004)). That is what we could define as "emotion in music" and "emotion from music". The former represents the intended emotion and the latter the emotion felt while listening to a musical piece. One is the emotion recognized and the other the emotion felt. A typical example of differentiation between both is the expression of anger. When someone else is angry, people might recognize anger and feel scared or defensive. The induced emotion is radically different from the perceived one in this case. Different factors can influence both types, for instance the symbolic aspect or the social context of a song will influence more the induced emotion (like for a national anthem). As noticed by Bigand et al. (2005) both aspects are not strictly independent and there will always be an influence of the induced emotion on someone asked to judge the perceived one. Nevertheless it should be observed that people tend to agree more on the perceived emotion than on the induced emotion(Juslin & Laukka (2004)). This property is very important for our work as we want to reach the best consensus to build our computational models.

It is also worth noticing that a relevant part of the emotion in songs comes from the lyrics. Psychological studies have shown that part of the semantic information of songs resides exclusively in the lyrics (see Besson et al. (1998)). This means that lyrics can contain relevant information to express emotions that is not included in the audio. Indeed, Juslin & Laukka (2004) reported that 29% people mentioned the lyrics as a factor of how music expresses emotions. Although there is an increase in research about the causal links between music and emotion, there still remain many open questions (Patel (2007)). In addition to the biological substrate, there are important links related to the

musical features that are present or absent when perceiving or feeling a given music-related emotion. In Section 2.3.2, we give some results about these musical features, but first we will discuss the different representations of musical emotions that arise from psychological studies.

2.3.1. Emotion Representations

One of the first question that comes to mind to formalize emotions is how to representation them. From the literature in music psychology, there are two main paradigms to represent emotions. This distinction is quite general, it is not only about musical emotions, but studies were designed specifically to test and refine these models for music. The first one is the categorical representation that distinguishes among several emotion classes. The other one is the dimensional representation defining an emotional space. We detail here the main theories using both approaches and we make explicit the special case of musically-related emotional representations.

Categorical representation

The categorical representation aims to divide emotions in categories, where each emotion is labeled with one or several adjectives. The most canonical model is the concept of basic emotions where several distinct categories are the basis of all possible emotions (see Section 2.3.1).A lot of psychologists propose that their emotion adjective set is applicable to music. One of the most relevant works in this domain is the study by Hevner (1936) and her adjective circle shown in figure 2.1. Hevner's adjective list is composed of 67 words arranged into eight clusters. From this study each cluster includes adjectives that have a close relationship. This similarity between words of the same cluster enables one to work at the cluster level reducing the taxonomy to eight categories. Farnsworth (1954) modified Hevner's list into ten clusters. These categories were defined by conducting listening tests and subjective answers. Moreover, we should note that most of these studies were conducted using classical music from the western culture and mainly of the baroque and romantic periods. We can imagine that the emotions evoked by popular music are different. A problem of the categorical approach is that classifying a musical piece into one or several categories is rather difficult sometimes, as pointed out by Hevner (1936). For instance in one of her studies, based on a musical piece called "Reflections on the water" by Debussy was rated to belong to all the clusters unless a continuous measure was considered. Although it was argued that a word list could not describe the variety of possible emotions in music, using a reduced set helps to achieve an agreement between people (even if it gives less meaning) and offers the possibility for automatic systems to model the general consensus of musical pieces.

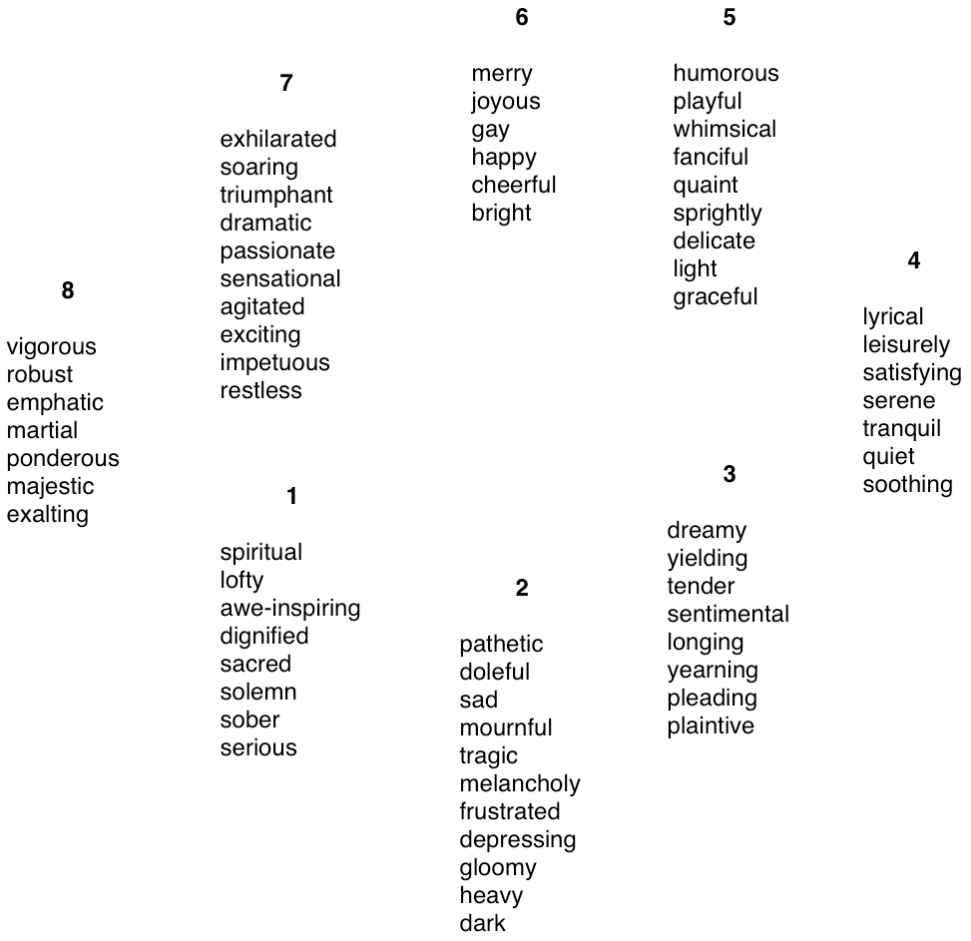

Figure 2.1: Hevner (1936) model with adjectives grouped into eight clusters.

The Basic Emotion Theory

A mentioned previously, the basic emotion theory states that there is a basic set of universal emotions. This set is considered as the basis of all possible emotions. In a similar way that primary color can compose any color, basic emotions could compose any emotion. This concept is particularly illustrated by Ekman's basic emotion theory, developed for facial expression, distinguishing between anger, fear, sadness, happiness and disgust (Ekman (1992)). Each basic emotion has a functional goal that can be defined as key appraisals reviewed in Juslin & Sloboda (2001) p 76. and presented here in Table 2.1. The basic emotion theory has been criticized, especially because researchers proposed different sets of basic emotions. Looking at the literature, we note that it mainly depends on how emotions are defined. If we consider that basic emotions should be universal, inducing distinct feelings, having different functionalities and contributing to survival triggering different physiological

Emotion	Juncture of plan	Core relational theme
Happiness	Subgoals being achieved	Making reasonable progress towards a goal
Anger	Active plan frustrated	A demeaning offense against me and mine
Sadness	Failure of major plan or loss of active goal	Having experienced an irrevocable loss
Fear	Self preservation goal threatened or goal conflict	Facing an immediate, concrete, or overwhelming physical danger
Disgust	Gustatory goal violated	Taking in or being close to an indigestible object or idea (metaphorically speaking)

Table 2.1: Key appraisals for basic emotions adapted from Sloboda (2001).

changes, there is a reasonable consensus on the set exposed in Table 2.1 : happiness, anger, sadness, fear, and disgust.

Dimensional representation

In a dimensional representation, the emotions are classified along axes. Most of the proposed representations in the literature are inspired by the Russell (1980) "circumplex model of affect", using a two-dimensional space spanned by arousal (activity, excitation of the emotion) and valence (positivity or negativity of the emotion). In Figure 2.2, we represent this bipolar model with the different adjectives placed in this emotional space. In this two-dimensional space, a point at the upper-right corner has high valence and arousal, which means happy with a high activity such as "excited". Opposite to this one, the lower-left part is negative with low activity like "bored" or "depressed". Several researchers such as Thayer (1989) applied this dimensional approach and developed the idea of an energy-stress model. Other studies propose other dimensional representations. However, they all somehow relate to the models previously presented, as in the case of Schubert (1999) two-dimensional emotion space (called 2DES), with valence on the x-axis and arousal on the y-axis with a mapping of adjectives from different psychological references. The main advantage of representing emotion in a dimensional form is that any emotion can then be mapped in that space. It allows a model where any emotion can be represented, within the limitation of these dimensions. One common criticism of this approach is that very different emotions in terms of semantic meaning (but also in terms of psychological and cognitive mechanisms involved) can be close in the emotional space. For instance, looking at the "circumplex model of affect" in Figure 2.2, we observe that the distance between "angry" and

"afraid" is small although these two emotions are quite different.

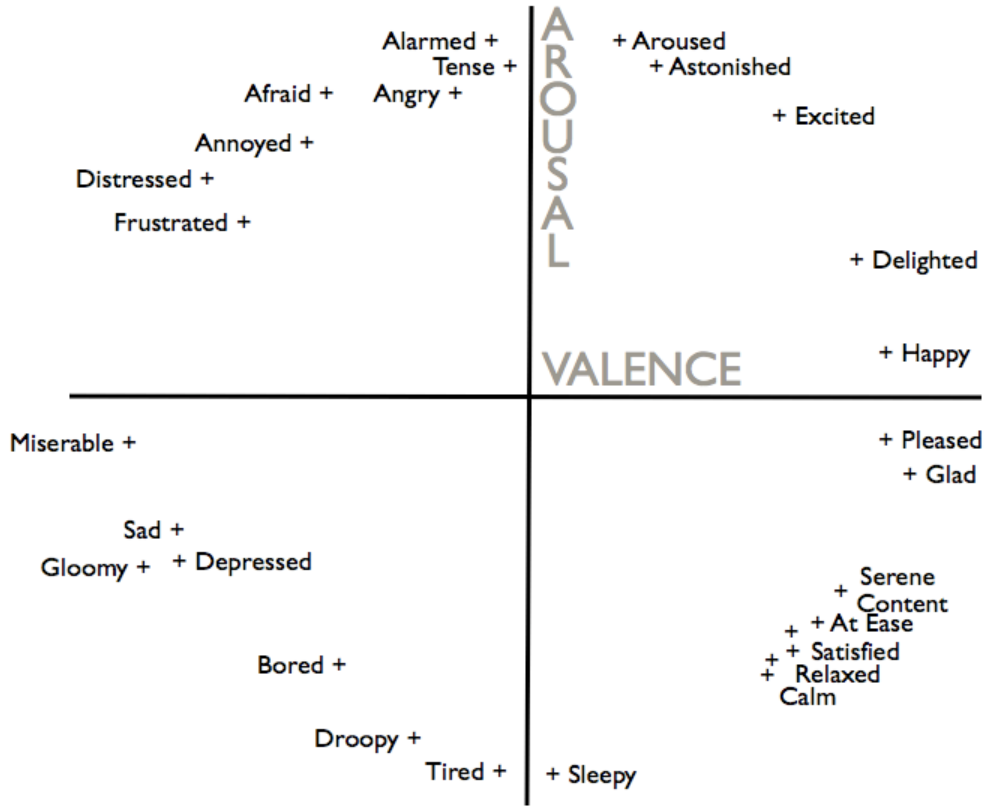

Figure 2.2: "Circumplex model of affect" with arousal and valence dimensions, adapted from Russell (1980)

2.3.2. Musical features and emotions

Several studies investigated musical features and their relations to particular emotions. However, most of the available research is centered on the western musical culture and mainly from classical music. Note that both composers and performers use these musical features. In Figure 2.3, we report on the main mapping between musical features and emotion categories found in the literature. In parenthesis is the quadrant number in Russell's dimensional space (see Figure 2.2)[1]. Each independent feature is probably not sufficient to conclude about one emotion; on the contrary this may require a rich set of musical descriptors. It is interesting to notice that these features correlate with research made on speech by Scherer (1991). Of course the comparison is limited to only a small set of attributes useful for speech like the pitch, the loudness and the

[1] 1 is positive valence and high arousal, 2 is negative valence and high arousal, 3 is negative valence and low arousal, 4 is positive valence and low arousal

tempo (which would be speed in speech). From the list shown in Figure 2.3, we observe that some features can be automatically extracted from polyphonic audio content with existing technologies developed in this document[2]. These features are marked with an asterisk. For instance the tempo can be estimated by locating the beats. Of course it would work better on music with evident tempo and prominent percussion on beats (rock or techno for example). The results are less reliable for music with a smooth and subtle rhythm (such as classical music). From audio content the reliability of these features is not always optimum but still it makes sense to use them, as they are informative. The key and the mode can also be extracted with a satisfying correctness (see Gómez (2006)) by analyzing frequency distributions and comparing with tonal profiles. Other attributes are more difficult to extract from a complex mix of instruments and would be reliable only on monophonic tracks (one instrument). They are marked with two asterisks. For example, the vibrato or the singer formant changes can be detected if we work on audio information containing just the singer's voice (such as a monophonic recording), but it becomes too complex on a mix containing all the instruments. Other musical cues should be informative about emotions, for instance a measure of the tonal induction (see Toiviainen & Krumhansl (2003)), predictability or tension (like studied by Farbood (2006); Krumhansl (1996); Lerdahl (1996); Lerdahl & Krumhansl (2007)) as it also correlates with physiological and neural responses (see Steinbeis et al. (2006)). From these results, can we seriously think about automatically predicting the emotion from music? Can machines have an emotional understanding close to ours? In the recent years, research in machine learning and signal processing has allowed one to extract relevant and robust audio and musical features with techniques we will detail in the next section.

[2]For a review on automatic extraction of audio features, see Herrera et al. (2005) and Gouyon et al. (2008)

Musical Features	Happiness (1)	Sadness (3)	Anger (2)	Fear (2)	Tenderness (4)
Tempo*	Fast, small variability	Slow	Fast, small variability	Fast, large variability	Slow
Mode*	Major	Minor	Minor	Minor	Major
Harmony*	simple and consonant	dissonant	atonality, dissonant	dissonant	consonant
Loudness*	medium-high, small variability	low, moderate variability	high, small variability	low, large level variability, rapid changes	medium-low, small variability
Pitch**	high, much variability, wide range, ascending	low, narrow range, descending	high, small variability, ascending	high, ascending, wide range, large contrasts	low, fairly narrow range
Intonation**	rising	flat, falling	accent on tonally unstable notes	-	-
Singer's formant**	raised	lowered	raised	-	lowered
Intervals**	perfect 4th and 5th	small (minor 2nd)	major 7th and augmented 4th	-	-
Articulation**	staccato, large variability	legato, small variability	staccato, moderate variability	staccato,large variability	legato, small variability
Rhythm*	smooth and fluent	ritardando	complex, sudden changes, accelerando	jerky	-
Timbre*	bright	dull	sharp	soft	soft
Tone attacks**	fast	slow	fast	soft	slow
Timing variability*	small	large (rubato)	small	very large	moderate
Vibrato**	medium-fast rate, medium extent	slow, small extent	medium-fast rate, large extent	fast rate, small extent	medium fast, small extent
Contrast between long and short notes**	sharp	soft	sharp	-	soft
Micro-structure*	regularities	irregularities	irregularities	irregularities	regularities
Others		pauses	spectral noise	pauses	accents on tonally stable notes

Figure 2.3: The most frequent musical features mapped with the emotion categories based on Juslin & Laukka (2004). An asterisk (*) means that some information can be extracted from polyphonic audio content; two asterisks (**) means that it can be extracted only from monophonic audio content (one instrument), in both cases using state-of-the-art technology.

2.4. Music Classification

The research on music classification from audio signal has been studied in the MIR field. MIR stands for "Music Information Retrieval". In this section, we want to explain the basic concept and schema of music classification commonly used in MIR. First, we want to describe the building blocks of a music classification system using supervised learning. Supervised learning is a machine learning task using training data to infer a function. In simple terms, a classifier using supervised learning is an algorithm that can learn categories from examples.
There are four main steps in supervised learning:

- Dataset Collection or Ground truth
- Audio features extraction
- Classification
- Evaluation

2.4.1. Ground Truth

The quality of the training dataset (or ground truth) is crucial. It also has to be defined in terms of taxonomy. To construct the dataset, we need to consider different aspects : the number of categories, the number of instances and the length of our audio excerpts. A good compromise needs to be found in order to be able to construct a reliable dataset of representative instances.

2.4.2. Audio Feature Extraction

Audio files are decoded as a succession of digital samples representing the waveform. From this data, audio features or descriptors can be extracted. The main objective is to have a compact representation of audio representing key facets of music. Descriptors are often divided into low-level (close to the signal) and high-level (close to human semantics). Orio (2006) summarized the most important facets of music as: timbre, orchestration, acoustics, rhythm, melody, harmony and structure. Audio features try to represent some aspect of these facets.
A digital audio signal is commonly converted to a general format (such as PCM 16 bits), with a sampling rate (from 5 to 44.1Khz). The audio signal is cut into frames. The *frame rate* is the number of frames per second. Then, a window function (Gaussian or Hanning window) is applied, to minimize the discontinuities at the beginning and end of each frame. A *hop size* is usually defined and is equal to the *frame rate* minus the overlap we want between consecutive frames (for a smoother analysis). Also, many features are derived from a spectral representation of audio. The spectrum is obtained from each

frame by applying a Discrete Fourier Transform, most often with the Fast Fourier Transform (FFT). Gouyon et al. (2008) gives a detailed explanation on how most common audio descriptors are computed. Once the features are extracted frame by frame, they are summarized using statistics such as mean, variance and derivatives. Then, in most cases, dimensionality reduction is applied as feature extraction can provide a large amount of data and part of it is not useful for classification. Indeed this can be considered as noise and decrease the performance (both in time and accuracy) of a classification algorithm. In Chapter 4, we will detail the audio features techniques and analyze those that are relevant for music mood classification.

2.4.3. Classification

Classification is a learning procedure based on the statistical learning theory. A classifier is a system that performs a mapping from a feature space X to a set of labels (also called classes) Y. A classifier assigns a pre-defined class label to a sample. In a supervised-learning context classes are pre-defined and samples from these classes are given. The classifier goal is then to model the observed data (called ground truth) to classify new instances with the highest accuracy possible. The classifier aims at discovering relationships between descriptors of samples from the ground truth and the class labels. Classifiers are trained on positive (and sometimes negative) examples of the to-be-learned class and tested on new unknown data. In the following part of this section, we present the most common classification algorithms, covering a good range of classifier types.

k-Nearest Neighbor (k-NN)

The K-Nearest neighbor method is a standard supervised classification method (Fix & Hodges (1951)). This is probably the simplest classifier. For each new observation, the k-NN algorithm looks for a number k of its closest training samples to decide on the class to predict. It classifies according to the most common class in the k nearest neighbors, where k is a positive integer.
Given each training sample x labelled as $f(x)$, given a query instance x_q, look at the k closest instances from the training examples $(x_1...x_k)$ and using a distance function d, the classification is made as follows:

$$\hat{f}(x_q) \leftarrow argmax_{v \in V} \sum_{i=1}^{k} d\big(v, f(x_i)\big) \tag{2.1}$$

where $\hat{f}(x_q)$ is an estimator of $f(x_q)$. The result relies mostly on the choice of distance function (which might not be trivial in our case), and also in the choice of k. In our experiments, we tested different values of k (between 1 and 20) with the Euclidean distance function :

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{p} (x_{ir} - x_{jr})^2} \tag{2.2}$$

with $x_i = (x_{1i}, x_{2i}, ..., x_{pi})$ is the p-dimensional feature vector.

Decision Trees

The decision tree algorithm splits the training dataset into subsets based on a test attribute value. This process is repeated on each subset in a recursive manner (recursive partitioning). Decision trees classify instances by sorting then down the tree from the root to a lead node which provides the classification of the new instance, and each branch descending from that node is one of the possible values for this attribute. We can see the decision tree algorithm as a method for approximating discrete-valued target functions in which the learned function is represented by a decision tree (see Mitchell (1997)). These trees, once learned, can be easily implemented using sets of $if - then$ rules. We used an implementation of the C4.5 decision tree from Quinlan (1993) (called J48 in the Weka software[3] and in Table 4.6). To optimize the parameters of the decision tree, we performed a grid search on the two main parameters: C (the confidence factor used for pruning, i.e limiting the tree growth) from 0.1 to 0.5 in 10 steps and M (the minimum number of instances per leaf) from 2 to 20. In Figure 2.4, we show a typical example of a decision tree from Mitchell (1997).

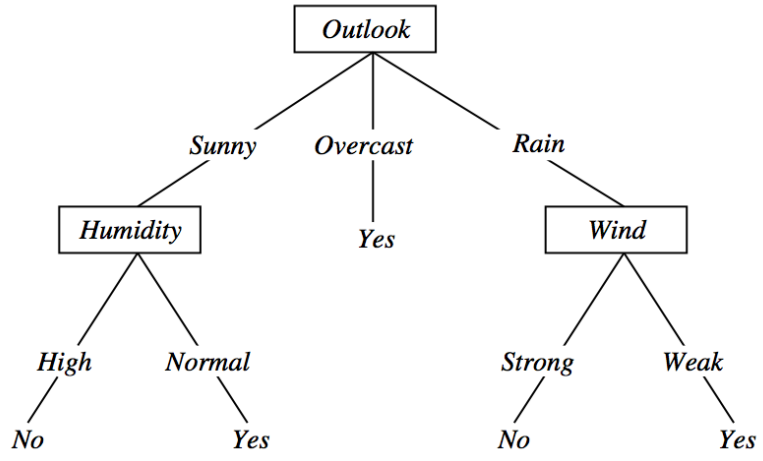

Figure 2.4: Decision tree for classifying if the day is a good day to play tennis. Figure from Mitchell (1997)

[3]See Witten & Frank (1999). http://www.cs.waikato.ac.nz/ml/weka/

Random Forests

The random forest classifier uses several decision trees in order to improve the classification rate. The basic concept behind this algorithm is common to other classifier strategies. It is the idea of combining weak learners (decision tree in this case), to build better models. As Breiman (2001) proposed, to classify a new object from an input vector, we put the input vector down each of the trees in the forest. For the kth tree, a random vector Θ_k is generated, independently of the previous vectors $\Theta_{k-1}, ..., \Theta_1$ but with the same distribution. There are different options to create these random vectors (see Breiman (2001) for more details). Then a decision tree is created using the training set and the vector Θ_k, resulting in the classifier $h(x, \Theta_k)$, where x is an input vector. The random forest is a classifier based on the collection of trees: $\{h(x, \Theta_k), k = 1, ...n\}$ Each tree returns a classification decision (considered as a "vote") and the forest chooses the classification having the most votes (over all the trees in the forest). We used the implementation in Weka for this algorithm.

Support Vector Machines (SVMs)

Support Vector Machine (Boser et al. (1992)), is a widely used supervised learning classification algorithm. It is known to be efficient, robust and to give relatively good performance in benchmarking studies (Baesens et al. (2003)). Indeed, this classifier is widely used in MIR research. In the context of a two-class problem in n dimensions, the idea is to find the "best" hyperplane separating the points of the two classes. This hyperplane can be of n-1 dimensions and found in the original feature space, in the case that it is a linear classifier. Otherwise, it can be found in a transformed space of higher dimensionality using kernel methods (non-linear), the kernel trick making it computationally feasible. The position of new observations compared to the hyperplane tells us in which class belongs the new input.
More formally, given a set of N data points $\{(x_i, y_i)\}_{i=1}^{N}$ with input data (also called features or descriptors values) $x_i \in \mathbb{R}^n$ and their corresponding binary class labels $y_i \in \{-1, +1\}$, the SVM classifier follows these conditions:

$$\begin{cases} w^T\varphi(x_i) + b \geq +1, & \text{if } y_i = +1 \\ w^T\varphi(x_i) + b \leq -1, & \text{if } y_i = -1 \end{cases} \tag{2.3}$$

which is equivalent to:

$$y_i[w^T\varphi(x_i) + b] \geq 1, i = 1, ..., N \tag{2.4}$$

where $\varphi(x_i)$ is a nonlinear funcion mapping the input space to a higher-dimensional space and b is a bias, an adjustable parameter. In this feature space, the equations 2.3 and 2.4 construct a hyperplane $w^T\varphi(x) + b = 0$ separating the two classes. The objective of the classifier is to minimize $w^T w$ so

that the margin between both classes is maximized. This problem is illustrated in figure 2.5.

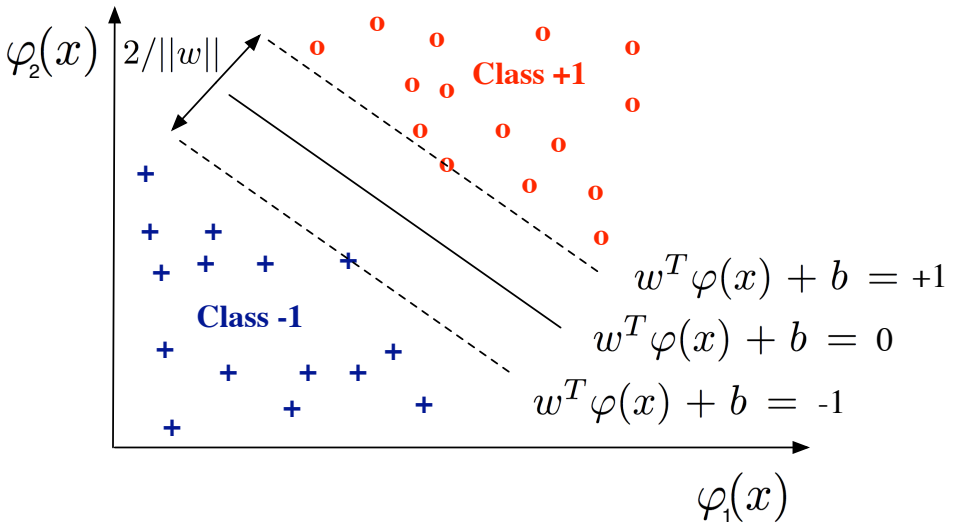

Figure 2.5: SVM optimization problem, adapted from Martens et al. (2009).

Maximizing the margin is maximizing $1/||w||$, which is equivalent to minimizing $||w||$. In primal weight space, the classifier is as shown here:

$$y(x) = sign[w^T\varphi(x) + b] \tag{2.5}$$

To evaluate the classifier, we can optimize the problem using the Lagrangian, which leads to a solution and a classifier as follows:

$$y(x) = sign\Big[\sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b\Big] \tag{2.6}$$

where $K(x_i, x) = \varphi(x_i)^T\varphi(x)$ is a kernel function satisfying the Mercer theorem (Mercer (1909)). The Lagrange multipliers α_i are calculated according to this optimization problem:

$$max_{\alpha_i} -\frac{1}{2}\sum_{i,j=1}^{N} y_i y_j K(x_i, x_j)\alpha_i\alpha_j + \sum_{i=1}^{N}\alpha_i \tag{2.7}$$

where

$$\begin{cases} \sum_{i=1}^{N} \alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C, i = 1, ..., N \end{cases} \tag{2.8}$$

with $C \in \mathbb{R}_+$ as a tuning parameter. The problem is a convex quadratic programming problem in α_i.
The kernel functions that are typically used are the following:

- Linear kernel: $K(x, x_i) = x_i^T x$
- Polynomial kernel: $K(x, x_i) = (1 + x_i^T x/c)^d$
- Radial basis function kernel (RBF): $K(x, x_i) = exp\{-||x - x_i||_2^2/\sigma^2\}$
- Sigmoid kernel: $K(x, x_i) = tanh(\kappa x_i^T x + \theta)$

For our evaluations, we tried the different kernel methods: linear, polynomial, radial basis function (RBF) and sigmoid respectively called SVM linear, SVM poly, SVM RBF and SVM sigmoid, as shown in Table 4.6. To find the best parameters in each case we used a 10 times 10-fold cross-validation method on the training data (see 2.4.4 for more details on cross-validation). For the linear SVM we looked for the best value for the cost C (penalty parameter), and for the others we applied a grid search to find the best values for the pair (C, γ) Boser et al. (1992). For C, we used the range [2-15,215] in 31 steps. For γ, we used the range [215,23] in 19 steps. In the other cases than the linear SVM, once we have the best pair of values (C, γ), we conduct a finer grid search on the neighborhood of these values. We used an implementation of the Support Vector Machines called libsvm[4] by Chang & Lin (2001).

Logistic Regression

Logistic regression can predict the probability of occurrence of an event by fitting data to a logistic curve. It is a generalized linear model used for binomial regression. It tries to model the data into the logistic function:

$$f(z) = \frac{e^z}{e^z + 1} = \frac{1}{1 + e^{-z}} \tag{2.9}$$

estimating z as:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \beta_k x_k \tag{2.10}$$

with a constant β_0 and β_1,β_2,β_3...β_k as regression coefficients of corresponding feature values: β_1,β_2,β_3...β_k. In the classification context the logistic curve models the relationship between a set of variables and a binary response expressed as a probability. This binary value is the classifier output. We use the implementation in Weka inspired from Le Cessie & Van Houwelingen (1992).

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Gaussian Mixture Models (GMMs)

A GMM is a linear combination of Gaussian probability distributions. This approach assumes that the likelihood of a feature vector can be expressed with a mixture of Gaussian distributions. GMMs are universal approximations of density, meaning that with enough Gaussians, any distribution can be estimated. In the training phase, the parameters of the Gaussian mixtures for each class are learnt using the Expectation-Maximization algorithm, which iteratively computes maximum likelihood estimates (Dempster et al. (1977)). The initial Gaussian parameters (means, covariance, and prior probabilities) used by the EM algorithm are generated via the k-means method (Duda & Hart (1973)).

2.4.4. Evaluation

To evaluate the quality of a classifier, we usually compare its prediction with ground truth data. When comparing the predicted value by the classifier to the annotated value, there are four possibilities:

- True Positive (TP), If a relevant item is expected, the system indeed classifies it as relevant.
- True Negative (TN), It occurs when a system that should classify an item as non-relevant does so.
- False Negative (FN), If the system should classify an item as relevant, but does not.
- False Positive (FP), If the system should classify an item as non-relevant, but does not.

From those term, there are several evaluation measures that can be used, in particular:

- Precision, the number of relevant items retrieved in proportion to the total number of items retrieved. $Precision = \frac{TP}{TP+FP}$
- Recall, the ratio between the number of the relevant items retrieved over the total number of relevant items. $Recall = \frac{TP}{TP+FN}$
- F-Measure, which is a weighted combination of precision and recall
- Accuracy, the ration between correctly classified items and the total number of items. $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

Cross-Validation: A standard technique to validate a classification approach is the K-fold cross-validation. This allows to evaluate the predictive power of a model without having to annotate new data. The original dataset is split into K equally distributed sub-samples from which $K-1$ are used for training and the remaining for testing. This is repeated K times, with each sub-sample used once as the testing data, and we measure the average precision, recall and accuracy over all the splits.

2.5. Mood Classification

2.5.1. State of the Art

Several studies have demonstrated that musical emotions are not too subjective or too variable to deserve a mathematical modeling approach (Bigand et al. (2005); Juslin & Laukka (2004); Krumhansl (1997); Peretz et al. (1998)). Indeed, within a common culture, the emotional responses to music can be highly consistent within and between listeners, but also accurate, quite immediate and precocious (Vieillard et al. (2008)). This stated, it opens the door to reproduce this consistent behavior with machines. Our purpose is to automatically classify music by mood.
In the literature other results are available and can be of interest, especially if the approach is different. Basically almost every scientific contribution differs in at least one key aspect. Several consider the category representation based on basic emotions (Laurier & Herrera (2007); Lu et al. (2006); Shi et al. (2006); Sordo et al. (2007)), while others treat the categories in a multi-labeling approach like Wieczorkowska et al. (2005). The basic emotion approach gives simple but relatively satisfying results with accuracies around 80-90% depending on the data and the number of categories. The lower accuracies for the MIREX approach detailed in the next section might be due to an overlap in the concepts included in the class labels like noted by Hu et al. (2008). It could also be due to a stricter evaluation on more data than the other mentioned works. The latter (multi-labeling) suffer from a difficult evaluation in general, as the annotated data needed should be much larger. Indeed if we want to use precision and recall in an appropriate way, we need to annotate all the data we evaluate with all categories (presence or absence), otherwise we might consider wrong results that are actually correct. Li & Ogihara (2003) extracted timbre, pitch and rhythm features and trained Support Vector Machines. They used 13 categories, 10 from Farnsworth (1954) and 3 additional ones. However the results were not satisfying (it was one of the very first studies of mood classification), with low precision (around 0.32) and recall (around 0.54). This might be due to the small dataset labeled by only one person, and to the large adjective set. Another similar work should be mentioned; Skowronek et al. (2007) used spectral, tempo rhythm, tonal and percussive detection features together with a quadratic discriminant analysis to model emotions. They made a mood

predictor with 12 categories, considered as binary, with an average accuracy around 85%. Other studies concentrated on the dimensional representation. Lu et al. (2006) used Thayer (1996) model based on the energy and stress dimensions and modeled the four parts of the space: contentment, depression exuberance and anxious. They modeled the different parts of the space using Gaussian Mixture Models. The system was trained with 800 excerpts of classical music and the system achieved around 85% accuracy (trained with three fourths and tested on the remaining fourth of the data). Although it was based on a dimensional system the prediction was made on the four quadrants as exclusive categories. However, another relevant study by Yang et al. (2008b) used Thayer's arousal-valence emotion plane, but with a regression approach, to model each of the two dimensions. They used mainly spectral and tonal descriptors together with loudness features. With these tools, they modeled arousal and valence using annotated data and regression functions (Support Vector Regression). The overall results were very encouraging and demonstrated that a dimensional approach is also feasible. In another work worth to be mentioned here, Mandel et al. (2006) designed a system using MFCCs and SVM. The interesting aspect of this work is the application of an active learning approach. The system learns according to the feedback given by the user. Moreover the algorithm chooses the examples to be labeled in a smart manner, hence reducing the amount of data needed to build a model achieving a similar accuracy with a standard method.
Similarly to what we can observe in psychology, there is no agreement for a common representation in the Music Information Retrieval (MIR) community. This makes arduous the comparison between mood classifiers because they all differ in many aspects including the representation model. In Table 2.2, we summarize the music mood classification literature in a chronological way. We note that most of the approaches are categorical (probably because simpler to annotate that dimensional), and Support Vector Machines seem also to give good results. We do not show accuracies in the Table because each approach uses different taxonomies, databases and even evaluation methods.

2.5.2. MIREX

With the objective of systematically evaluating state-of-the-art algorithms for Music Information Retrieval (MIR) systems, the Annual Music Information Retrieval Evaluation eXchange (MIREX) created an Audio Mood Classification (AMC) task for the first time in 2007. MIREX, as the largest evaluation event in the MIR community, built an audio dataset and ground-truth for audio mood classification to facilitate collaborations and evaluations among MIR researchers. A ground-truth set of 600 tracks distributed across five mood categories was built based on metadata analysis and human assessments. The AMC task adopted the set of five mood clusters. These clusters have been derived from data found in the popular music website AllMusicGuide.com, where the

Reference(s)	Rep.	Taxonomy	Ground Truth	Features	Classifier
Li & Ogihara (2003)	Cat	Farnsworth (1954) + mysterious,passionate,bluesy	499 of 30s from 128 albums	Spectral,Pitch,Rhythm	SVM
Liu et al. (2003)	Cat	Thayer (1996)	250 of 20s, classical/romantic	Spectral,Intensity,Rhythm	GMMs
Yang & Lee (2004)	Dim	intensity	500 of 20s	BPM, MPEG-7, Spectral, Sony EDS	SVR
Wieczorkowska et al. (2005)	Cat	happy, graceful, pathetic dramatic, sacred, bluesy	872	Spectral	kNN
Pohle et al. (2005)	Cat	happy, neutral, sad soft, neutral, aggressive	834	Spectral,ZCR,MFCCs,Rhythm, Pitch, MPEG-7	kNN, NB, SVM
Lu et al. (2006)	Cat	Thayer (1996)	800 of 20s from 250 music pieces	Intensity, Spectral, Rhythm	GMMs
Shi et al. (2006)	Cat	calm, sad, pleasant, excited	194 of western genres	Intensity, Spectral, Tempo (MDCT-LMFC)	Adaboost+GMMs
Sordo et al. (2007)	Cat	angry, happy, mysterious, sad	191 from Magnatune	MFCCs, Spectral, Rhythm, Tonal	kNN
Skowronek et al. (2007)	Cat	12 from literature	1059	Rhythm, Chroma	QDA
Yang et al. (2008b, 2006)	Dim	Arousal, Valence	195 western, chinese, japanese pop	Spectral, Tonality, Pitch, Loudness, Dissonance	MLR, SVR, AdaBoost.RT
Lin et al. (2009)	Cat	12 from AMG	1535	Spectral, Pitch, Rhythm, MPEG-7, Linear Prediction	SVM
Eerola et al. (2009)	Both	happy, sad, tender, scary, angry arousal, valence, tension	110 from soundtrack	Dynamics, Spectral, Harmony, Register, Rhythm, Articulation	MLR, PCA +PLS
Yang & Chen (2010)	Dim.	arousal, valence	1240 of 30s, Chinese Pop	Melody, Spectral, MFCCs, Rhythm	SVR, ListNet, RBFListNet
Laurier et al. (2010)	Cat.	happy, sad, relaxed, aggressive	1000 mainstream music	Spectral, MFCCs, Rhythm, Tonality, Dissonance	SVM

Table 2.2: Summary table. Music Mood Classification methods from audio content. Rep. stands for Representations, Cat for Categorical and Dim for Dimensional. In representations, Both mean that both categorical and dimensional representations were used. Abbreviations for features are MFCCs for mel cepstrum frequency coefficients, Abbreviations for classifiers are SVM for Support Vector Machine, GMMs for Gaussian Mixture Models, kNN for k-Nearest Neighbours, NB for Naive Bayes, MLR for Multiple Linear Regression, SVR for Support Vector Regression, RBF for Radial Basis Function, PCA for Principal Component Analysis, and PLS for Partial Least Squares.

reviews are labelled with 179 possible different moods. Hu & Downie (2007) grouped the mood labels by albums and songs and, with an agglomerative hierarchical clustering, could reduce the mood space into a manageable set. We list the five mood clusters in Table 2.3. The words in each cluster collectively define the "mood spaces" associated with the cluster. However, Hu et al. (2008) showed that these clusters might not be optimal as we observe some semantic overlap between categories. Also the limited agreement between human annotators while making the ground truth shows the limitation of the dataset and the taxonomy employed. Nevertheless, this work was the first and, up to redaction of this document, the only effort to compare mood algorithms in a common context. Moreover, a very few work on this topic involves more than one annotators and the means employed to involve many people (15) in the annotation process is worth noticing. Also, it has stimulated a lot of work around the topic and many researchers could tried their generic classification algorithm on this task. However, we should also note that this categorical representation has not been adopted by researchers yet, to our knowledge only a very few uses this taxonomy outside from the MIREX context (Bischoff et al. (2009)). In the next Chapters, we will detail the MIREX results and compare with our algorithms and submissions.

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 2.3: Clusters of mood adjectives used in the MIREX Audio Mood Classification task.

2.6. Conclusion

From this multidisciplinary literature review, we can make several conclusions for our research. First, we will use interchangeably the terms "mood" and "emotion". Then, we observed that there was no clear representation emerging from the literature review in music psychology, but rather a list of possible standard ones. Categorical representations (using a standard taxonomy like reviewed previously) or dimensional representations (with arousal and valence) are widely used and both proved to be effective. Moreover, Eerola & Vuoskoski (2011) demonstrated that for users, both representations give very compatible ratings. Reviewing the literature in automatic music mood classification both paradigms are used, but with a majority for the categorical approach. Finally,

to avoid confusions and optimize the consensus while annotating the data, we will focus on emotions recognized in music instead of experienced or induced emotions.

CHAPTER 3

Representation Models from Social Tags

"I just want people to be moved by the music. If you're not moved by the music, then everything else falls away ... It is the part of music that is the hardest to talk about, and I do not spend much time talking about it. But it is the bottom line." Steve Reich.

3.1. Introduction

Before making any computational model of musical emotions, we need to find a relevant representation model. This is our first main dilemma in this thesis. Indeed, it is fundamental and will highly influence all the work that will follow this crucial choice. Looking at the literature review, summarized in the previous chapter, it is ambiguous and confusing as no clear or easy decision can be taken. On the one hand, all of the previously mentioned representations are criticized and none is really widely accepted. But on the other hand, all are used and can be considered as somehow valid. Partial evidence for and against each one can be found, making our choice difficult. We tried different approaches thinking also about how people would have to describe music using each chosen representation. We made small scale experiments with our colleagues trying different categorizations. We set up a web survey that ran internally and among friends. The conclusions, to be taken as intuitive but not rigorous, were that dimensional representations were quite difficult to use and annotate. The results were not very consistent even between users themselves at different times. Using categories seemed easier, but we could not really conclude on which categories to use. Another doubt raised by the literature is the kind of music employed in the experiments. We would rather work with mainstream popular music and the experiments validating representation theories use almost exclusively classical music. Inspired by other works comparing experts representations with wisdom of the crowds (Sordo et al. (2008)), we

decided to try to extract valuable information from user generated data, and to contrast it with experts models. This lead to our first experiment reported in this thesis, finding a relevant representation based on knowledge from experts and a large amount of people.

3.2. Experiment 1: Representations Models from Social Tags

3.2.1. Objectives

This experiment aims at analyzing how do a large community of users represent their music collection with moods. We aim to understand how do people label their music by mood and to find the underlying model behind, comparing it to the existing models from the literature.
In this experiment, we want to address our problem using data collected in an "everyday life" context (not in controlled laboratory settings like in most of psychological studies), studying mood representations with a bottom-up approach, from a community point of view. With this perspective, we do not control all the variables (such as the bias of the sample) but also we do not influence the participants. From this data, we want to create a semantic space for mood. In Sordo et al. (2008), the authors studied the agreement between experts and a community for genre classification. Levy & Sandler (2007) studied how tags can be used for genre and artist similarity, and proposed a visualization of certain words in an emotion space. Both studies inspired our approach of using social tags to compare the semantics of the wisdom of crowds with expert knowledge. The goal of this experiment is to create a semantic mood space where we can represent mood and compare it with existing representations in the literature. There are two main expected outcomes for this study. First we aim to verify if the knowledge extracted from social tags and the knowledge from the experts (psychologists, musicologist or others) converges. Then, we want to decide on a mood representation that will serve as a basis for creating music mood classification models, the main goal of this thesis.

3.2.2. Dataset

Our approach is to obtain a mood representation from social tags. It can be seen as transforming a tag space into a mood space. For this purpose, we need tags related with music pieces. A music social network is the perfect dataset candidate for this experiment.

Social Network Data

For this study, we want to observe the way people use mood words in a social network. We selected words related to emotions based on the main articles in

music and emotion research. We included words from different psychological studies like Hevner (1936) or Russell (1980). We also added words representing basic emotions and other related adjectives (Juslin & Sloboda (2001)). Finally we aggregated the mood terms mostly used in MIR (Laurier & Herrera (2009)) and the ones selected for the MIREX task (Hu et al. (2008)). At the end of this process, we obtained a list of 120 mood words.

Last.fm[1] is a music recommendation website with a large community of users who are very active in associating tags with the music they listen to. With over 30 million users in more than 200 countries[2], this social network is a good candidate to study how people tag their music. We crawled 6,814,068 tag annotations from 575,149 tracks in all main genres. From those, 492,634 tags were distinct. This huge dataset contains tags of any kind. From the original 120 mood words, 107 tags were found in our dataset. However some of them did not appear very often. We decided to keep only the tags that appeared at least 100 times, resulting in a list of 80 words. We also chose to keep the tracks where the same mood tag had been used by several users. This subset contains 61,080 tracks. We observe that the mood tags mostly used are *sad*, *fun*, *melancholy* and *happy*. For instance, the tag *sad* has been used 11,898 times in our dataset. On the contrary, the least used tags are *rollicking*, *solemn*, *rowdy* and *tense*, applied in less than 150 tracks. In average, a mood tag is used in 754 tracks.

Semantic Mood Space We aim at comparing mood terms by their co-occurences in tracks. Intuitively *happy* should co-occur more often with *fun* or *joy* than with *sad* or *depressed*. This co-occurence information included in the data we crawled from *last.fm* is embodied in a document-term matrix where the columns are track vectors representing tags. The main problem we have when dealing with this matrix is its high dimensionality and its sparsity. To solve this problem, we use a classical technique in text analysis that is Latent Semantic Analysis (LSA) (Deerwester et al. (1990)), with Singular Value Decomposition (SVD). This technique allows to project the data into a space of a given lower dimensionality, while maintaining a good approximation of the distances between data points. LSA is indeed a common technique to map words and documents in a "concept space", mapping the document-term matrix (or in our case track-tag matrix) into a reduced conceptual space, that we call the "Semantic Mood Space". LSA filters out noise and imposes some simplifications:

- Concepts, the new dimensions in the mapped space, are represented as patterns of tags that appear together in tracks

[1] http://www.last.fm
[2] http://blog.last.fm/2009/03/24/lastfm-radio-announcement

- Tags are assumed to have only one meaning. Choosing only words about mood or emotions does not prevent polysemy. We should be aware of this limitation while analyzing the results

- Tracks are represented as "bag of tags", without considering how many times the were used in the document. However we did use this information to filter the tracks were a tag was not used by several users

Considering our track-tags matrix X, as for any rectangular matrix, it can be decomposed into the product of three other matrices using the singular value decomposition:

$$X = Tags * S * Tracks^T \tag{3.1}$$

where S is a diagonal matrix of eigenvalues and $Tags$ and $Tracks$ matrices are the left and right singular vectors. This is only another representation of the same matrix X using orthogonal dimensions. However, we can use reduced rank singular value decomposition, keeping k largest singular values. This results in the best k-dimensional approximation to the original matrix, in a least square sense:

$$X = Tags_k * S_k * {Tracks_k}^T \tag{3.2}$$

A schema of the whole procedure is shown in Figure 3.1. This technique has been shown to be very efficient to capture tag representations for genre and artists similarity (Levy & Sandler (2007)). We decided to use a dimension k of 100, which seems to be good trade-off for similarity tasks like pointed out by Levy & Sandler (2007). In the following experiments, we tried to change this dimension parameter (from 10 to 10 000 on a logarithmic scale), with no significant impact on the outcomes except less relevant results when selecting a too low or too high dimensionality.
One of the most interesting outcomes of this transformation, is that it enables to easily compare tags. Once we have the data into this semantic space, we can compute a common distance between terms, the cosine distance. The cosine distance d_{cos} of two vectors of terms frequencies A and B is computed using a dot product and the magnitude like follows:

$$d_{cos}(A, B) = \frac{||A||\,||B||}{A.B} \tag{3.3}$$

or more explicitly:

$$d_{cos}(A, B) = \frac{\sqrt{\sum_{i=1}^{n} {A_i}^2}\sqrt{\sum_{i=1}^{n} {B_i}^2}}{\sum_{i=1}^{n} A_i B_i} \tag{3.4}$$

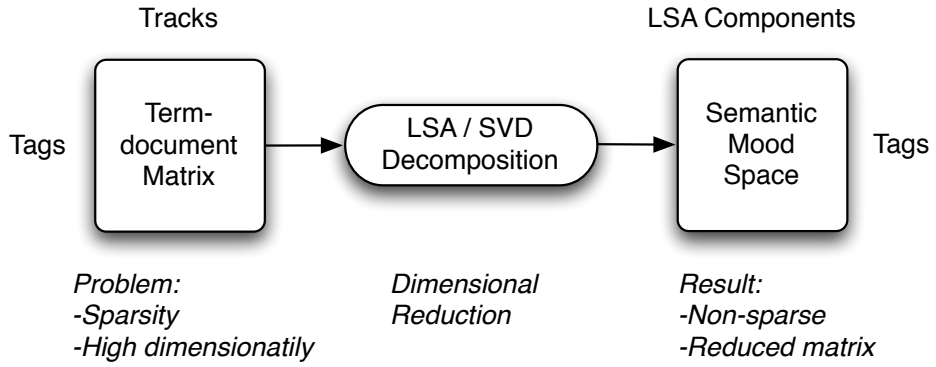

Figure 3.1: Schema of the transformation applied to the raw tag data (LSA).

where n is the number of dimensions. As the term frequency can not be negative, the angle between two vectors can not exceed 90 degrees and, consequently, the distance values are included in the range $[0, 1]$.

Here are some examples of distances between mood tags:

$d_{cos}(happy, happy) = 0$
$d_{cos}(happy, sad) = 0.99$
$d_{cos}(cheerful, sleepy) = 0.97$
$d_{cos}(restless, calm) = 0.99$
$d_{cos}(scary, fun) = 0.99$
$d_{cos}(tense, serene) = 0.98$
$d_{cos}(anger, aggressive) = 0.06$
$d_{cos}(calm, relaxed) = 0.03$
$d_{cos}(sleepy, calm) = 0.09$
$d_{cos}(melancholy, plaintive) = 0.04$
$d_{cos}(exciting, playful) = 0.04$

Obviously the distance between one term and itself is equal to 0. We observe that *happy* and *sad* are quite far from each other, as well as *cheerful* and *sleepy*. On the other hand, we note that *anger* is close to *aggressive* and that *calm* is similar to *relaxed*. Even if we show here some prototypical examples, values in the whole distance matrix intuitively make sense. This distance measure (and so similarity) is a useful tool to extract information from this semantic mood space and to compare it with other proposed representations from the literature.

3.2.3. Categorical Representations

To study the categorical mood representations, we first derive a folksonomy (community-based taxonomy) representation by means of unsupervised clustering from the social data. This will give us the "natural" clusters that emerge from the data itself. Then, we evaluate how the expert taxonomies fit into our semantic mood space.

Folksonomy representation

From our semantic space, we want to infer a relevant categorical representation modeling the distance between mood tags (and so the way people tag music by mood). To achieve this goal, we apply an unsupervised clustering method using the Expectation maximization (EM) algorithm from Dempster et al. (1977) (and its implementation in Weka by Witten & Frank (1999)).
The EM algorithm do not requite a training phase. It consists of an iterative algorithm, which aims at finding the parameters of the probability distribution that has the maximum likelihood of its attributes. The first step is the Initialization, followed by the Expectation and the Maximization, both repeated iteratively until convergence.

Initialization Considering K clusters, each class j (forming a cluster C_j) is composed by a parameter vector λ_j constituted by the mean μ_j and the covariance matrix P_j, representing the features of the Gaussian probability distribution. At the initial ($t = 0$), we generate random values for μ_j and P_j

$$\lambda(0) = \{\mu_j(0), P_j(0)\}, j \in \{1...K\} \tag{3.5}$$

Expectation (E-step) In this step, we estimate the probability of each element to belong to each cluster C_j.

Maximization (M-step) This step is responsible to estimate the parameters of the probability distribution of each class for the next iteration (going back to the E-step if the convergence test fails).

Convergence test This is the final step if the convergence is reached (or a maximum number of iteration). After each iteration of the E and M steps, a convergence test is performed. It verifies if the difference between the two last attribute vectors (at t and $t-1$) is below a certain error tolerance ξ.

$$if(||\lambda(t+1) - \lambda(t)|| < \xi) \xrightarrow{then} stop \tag{3.6}$$

The first important question to be answered is how many clusters should we consider. As we want this number to be inferred by the data itself, we used the *v-fold cross validation* algorithm. We divided the dataset in v folds, training

on $v-1$ folds and testing on the remaining one. We measure the log-likelihood computed for the observations in the testing samples. The results for the v replications are averaged to yield a single measure of the stability of the model. In Figure 3.2, we show the results of this process, displaying an average cost value (in our case twice the negative log-likelihood of the cross-validation data). Intuitively the lower is the value, the better is the cluster. To choose the "right" number of clusters, we look at the cost value while increasing the number of clusters. Practically, we stop when the mean cost value stops decreasing and select the current number of clusters.

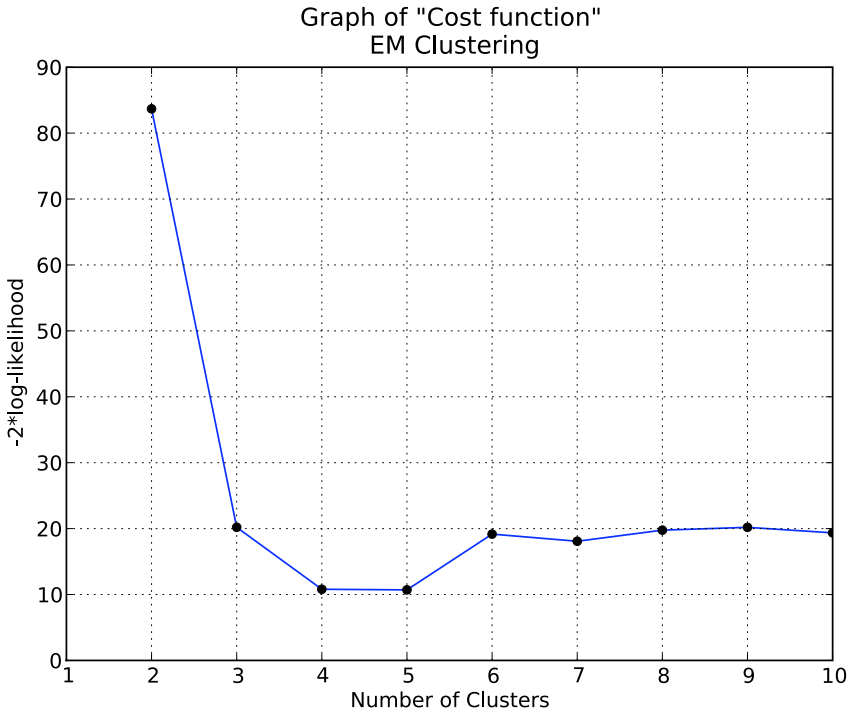

Figure 3.2: Plot of the cost values (2 times the negative log-likelihood) depending on the number of clusters.

We observe that the cost rapidly decreases with the number of clusters until four clusters. After that, it is stable and even increases, meaning that the data is overfitted. Consequently, the optimal number K of clusters is four. Using this number for the EM algorithm, we obtained the clusters listed in Table 3.1. These four clusters are very similar to the categories posed by the main basic emotion theories as reviewed by Juslin & Sloboda (2001). Moreover, these clusters represent the four quadrants of the classical arousal-valence plane from Russell like shown in Figure 3.3:

Cluster 1: angry (high arousal, low valence)

Cluster 1	Cluster 2	Cluster 3	Cluster 4
angry	sad	tender	happy
aggressive	bittersweet	soothing	joyous
visceral	sentimental	sleepy	bright
rousing	tragic	tranquil	cheerful
intense	depressing	good natured	happiness
confident	sadness	quiet	humorous
anger	spooky	calm	gay
exciting	gloomy	serene	amiable
martial	sweet	relax	merry
tense	mysterious	dreamy	rollicking
anxious	mournful	delicate	campy
passionate	poignant	longing	light
quirky	lyrical	spiritual	silly
wry	miserable	wistful	boisterous
fiery	yearning	relaxed	fun

Table 3.1: Folksonomy representation. Clusters of mood tags obtained with the EM algorithm. For space and clarity reasons, we show only the first tags.

Cluster 2: sad, depressing (low valence, low arousal)
Cluster 3: tender, calm (high valence, low arousal)
Cluster 4: happy (high arousal, high valence)

The label for the clusters have been chosen because they were close to the cluster centroid.
To summarize, the semantic space we created is relevant and coherent with existing basic emotion approaches. This result is very encouraging and assesses a certain quality of this semantic space. Moreover, it confirms that the community uses mood tags in a way that converges with some theories of basic emotions from psychology.

Agreement between experts and community

Having this semantic space modeling the social network, we can evaluate how well common representations from the literature fit into that space. In the following part, we measure the agreement between experts and our community-based representation. To do so, we performed a coarse-grained similarity, where we measured how *separable* the expert-defined mood clusters are in our semantic space. First, we computed the LSA cosine similarity among all moods within each cluster (intra-cluster similarity) and then we computed the dissimilarity among clusters, using the centroid of each cluster (inter-cluster dissimilarity). The expert representations we selected for this experiment are

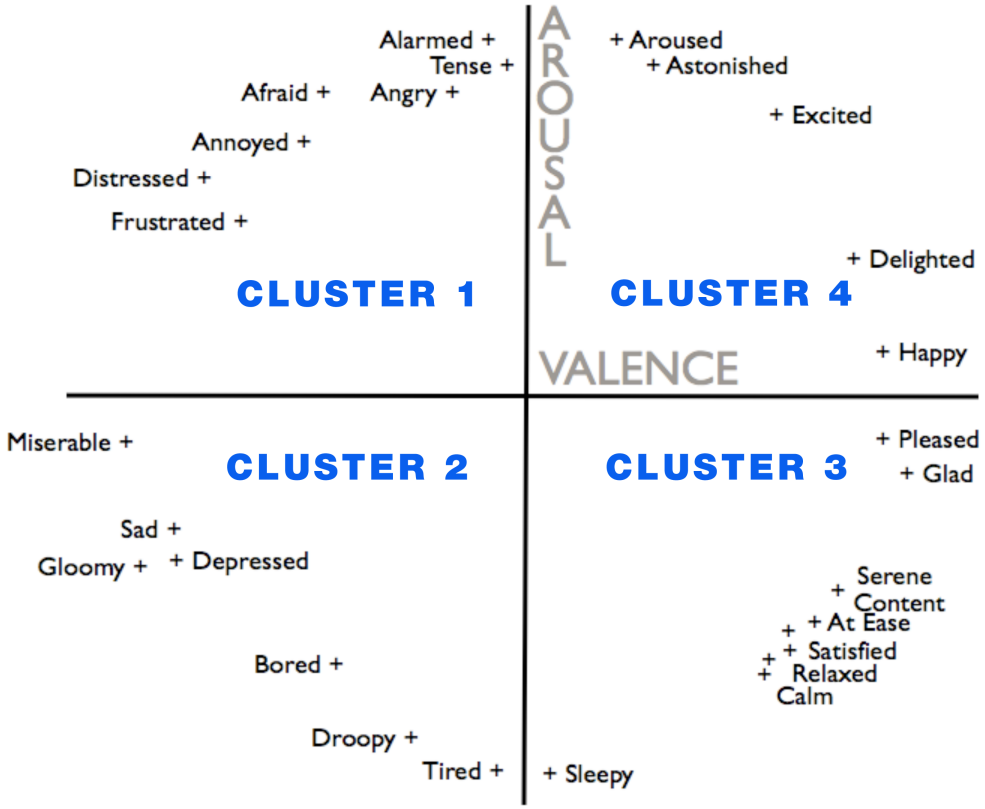

Figure 3.3: "Circumplex model of affect" with arousal and valence dimensions, adapted from Russell (1980), mapping the clusters found from the semantic mood space.

the eight clusters from Hevner where we could match more than 50% of the tags and the five clusters from the MIREX taxonomy where all 31 tags were matched (see previous chapter for more details on these taxonomies). The matched words for these representations are shown in Table 3.2 for the Hevner clusters and in Table 3.3 for the MIREX clusters.

Intra-cluster similarity For each cluster of the expert representations, we compute the mean cosine similarity between each mood tag in the cluster. Considering a cluster of size n as $X = x_1, x_2, ...x_n$, the intra-cluster similarity can be formalized as follows:

$$\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{cos}(x_i, x_j) \tag{3.7}$$

The results for intra-cluster similarity are presented in Figure 3.4 for the Hevner representation and in Figure 3.5 for the MIREX clusters.

Clusters	Hevner Mood Adjectives
Cluster 1	spiritual, sacred
Cluster 2	pathetic, sad, mournful, tragic, melancholy, depressing, gloomy, heavy, dark
Cluster 3	dreamy, tender, sentimental, longing, yearning, plaintive
Cluster 4	lyrical, serene, tranquil, quiet, soothing
Cluster 5	humorous, whimsical, playful, delicate, light
Cluster 6	merry, joyous, gay, cheerful, bright
Cluster 7	restless, passionate,exciting
Cluster 8	martial, majestic

Table 3.2: Matched mood adjectives in Hevner model

Clusters	MIREX 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 3.3: Matched mood adjectives in MIREX model

In the results for the Hevner clusters, we note a high intra-cluster similarity value for cluster 1, which is the one including *spiritual* and *sacred* (please look at Figure 2.1 for the complete list). Cluster 6 performs also quite well (*joyous*, *bright*, *gay*, *cheerful*, *merry*). However we have poor intra-cluster similarity for cluster 8, which includes *martial* and *m*ajestic. This might be because these words are also some of the less used in our dataset, but we hypothesize that they are less descriptive today than when the taxonomy was created (1936). Moreover, these words were selected for classical music which is not the main content of the *lasf.fm* music. The rest of the intra-cluster similarity values are in average quite low, meaning that this representation is not optimal in the semantic mood space.

For the MIREX clusters, we remark that the lowest intra-cluster similarity is for cluster 2 (*sweet*, *good natured*, *cheerful*, *rollicking*, *amiable*, *fun*). Maybe is it quite clear that this category is about *happy* music, however the words used are not so common and may lower this value. In average, the intra-cluster similarity value is quite high for this representation. For comparison purpose, we note that the intra-cluster similarity of the folksonomy representation has an average intra-cluster similarity value of 0.82. We summarize the results with the mean values in Table 3.7. Obviously, as the folksonomy representation was made from the semantic space itself, it has better results than the other models.

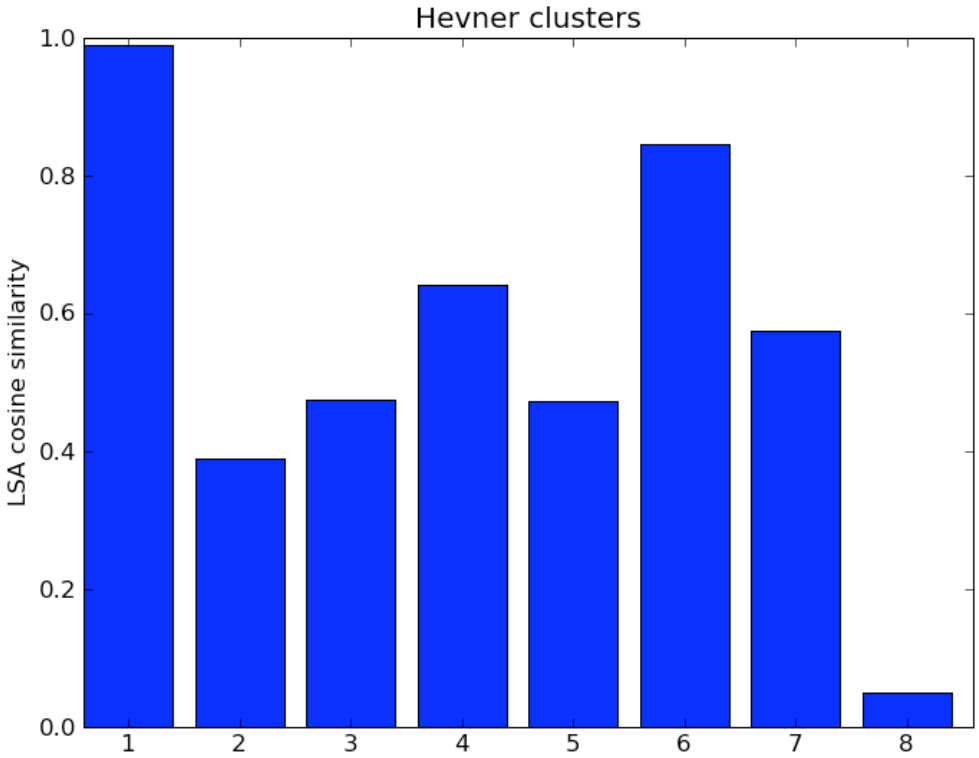

Figure 3.4: Intra-cluster cosine similarity for Hevner's representation.

It is made to optimize the cluster consistency.

Mood Taxonomy	Intra-cluster similarity
Hevner	0.55
MIREX	0.73
Folksonomy	0.82

Table 3.4: Intra-cluster similarity means for each mood taxonomy.

In this part, we have looked at the consistency inside each cluster, however it is also crucial to look at the distances between clusters to evaluate the quality of the clustering representations.

Inter-cluster dissimilarity To measure how *separable* are the different clusters, we compute the mean cosine distance from each cluster centroid (mean of all cluster points) to the other cluster centroids. If we look at our folksonomy representation clusters from Section 3.2.3, the cosine distance between centroids of clusters are all quite high (0.9 in average, see Table 3.7). This

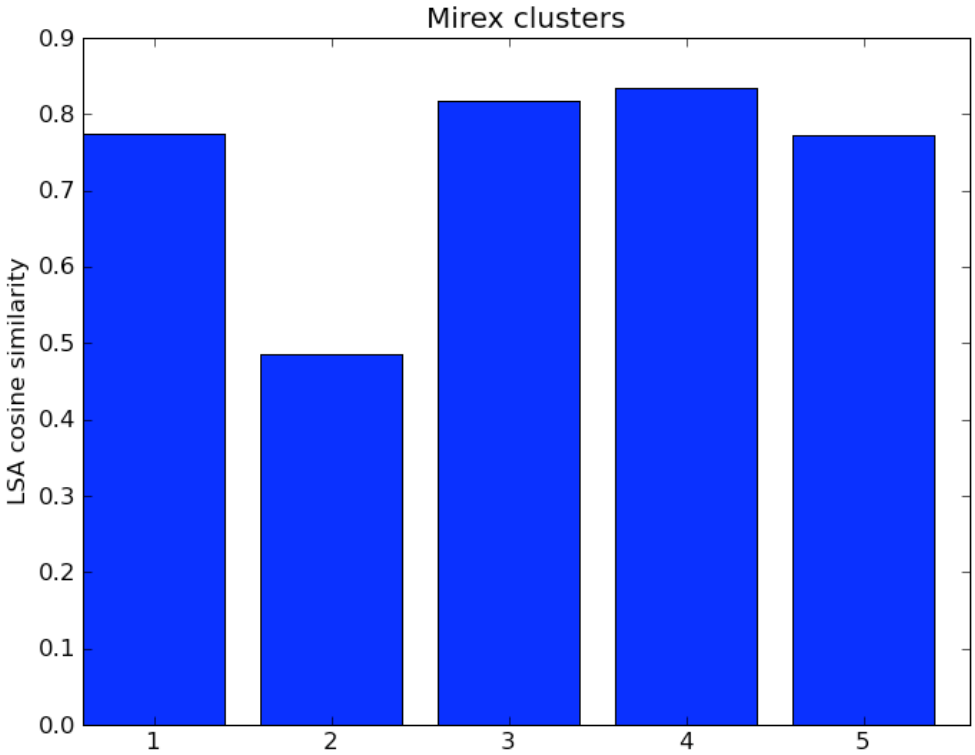

Figure 3.5: Intra-cluster cosine similarity for MIREX representation.

is not very surprising as the representation was designed with this data thus to optimize a good separation between clusters. However, to better analyze the results, we need a baseline. To measure it empirically from our data, we randomly generated clusters of mood terms and computed the inter-cluster dissimilarity. We repeated and averaged this procedure over 1000 iterations. This gives us a solid basis to compare representations to what a random one could achieve. This measure is increasing with the number of clusters like plotted in Figure 3.7.

In Table 3.5, we show the confusion matrix of the inter-cluster dissimilarity for the MIREX clusters. We notice that the lowest value is between cluster 1 and cluster 5, meaning that these clusters are quite similar. This finding correlates with the results from the MIREX task, in which the confusion between these two classes was found significant (see Hu et al. (2008)). However the confusion between clusters 2 and 4, also relevant in the analysis from Hu et al. (2008), is not reflected in our case. Additionally, we observe that the most separated clusters (5 and 2), are also the less confusing in the MIREX results. Looking at the confusion matrix for the Hevner clusters in Table 3.6, we remark many high values like highlighted in the Table (for instance between clusters 1 and 2, 1 and 6, 1 and 7, 2 and 8 etc...). On the contrary, the lowest value (0.09)

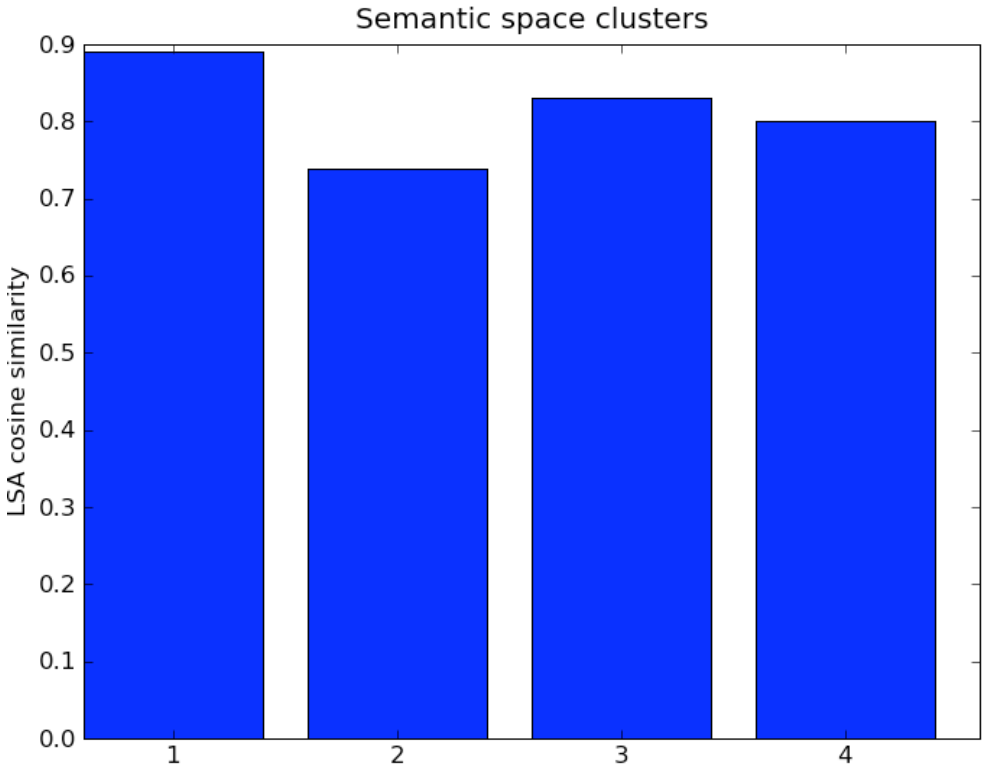

Figure 3.6: Intra-cluster cosine similarity for the mood semantic space representation

is between clusters 1 and 4. Indeed both clusters have words than can appear similar like *spiritual* (cluster 1) and *serene* (cluster 4) for instance. Also, we observe a low value (and so a high confusion) between clusters 5 and 7. Again in that case, we note several semantic similarities, such as between *playful* (cluster 5) and *restless* or *exciting* (both from cluster 7). We summarize the results of both intra and inter-cluster measures for the different taxonomies in Table 3.7 together with the random baseline averages.

	C1	C2	C3	C4	C5
C1	0	0.74	0.128*	0.204	0.108*
C2	0.74	0	0.859	0.816	**0.876**
C3	0.128*	0.859	0	0.319	0.265
C4	0.204	0.816	0.319	0	0.526
C5	0.108*	**0.876**	0.265	0.526	0

Table 3.5: Confusion matrix for the inter-cluster dissimilarity for the MIREX clusters (C1 means cluster 1, C2 cluster 2 and so on). The values marked with an asterisk are the most similar and in bold are the less similar values (below 0.2).

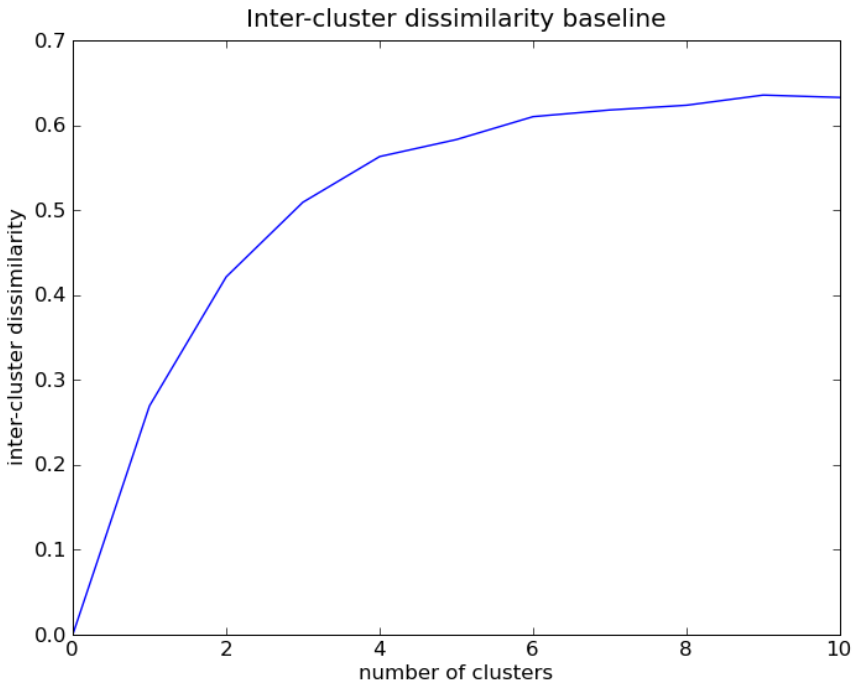

Figure 3.7: Inter-cluster dissimilarity baseline with different number of random clusters

	H1	H2	H3	H4	H5	H6	H7	H8
H1	0	0.99*	0.32	**0.09**	0.85	0.98*	0.97*	0.57
H2	0.99*	0	0.93	0.74	0.89	0.86	0.72	0.96*
H3	0.32	0.93	0	0.31	0.83	0.93	0.96*	**0.17**
H4	**0.09**	0.74	0.31	0	0.67	0.81	0.83	0.56
H5	0.85	0.89	0.83	0.67	0	0.42	**0.14**	0.75
H6	0.98*	0.86	0.93	0.81	0.42	0	0.63	0.65
H7	0.97*	0.72	0.96*	0.83	**0.14**	0.63	0	0.96*
H8	0.57	0.96*	**0.17**	0.56	0.75	0.65	0.96*	0

Table 3.6: Confusion matrix for the inter-cluster dissimilarity for the Hevner clusters. The values marked with an asterisk are the most similar (above 0.95) and in bold are the less similar values (below 0.2).

In a nutshell, the Hevner clusters are less consistent but are more separated than the MIREX ones. Indeed, even if the latter have more intra-cluster similarity, they suffer from confusions between some categories as reflected in our results.

Mood Taxonomy	Inter-cluster dissimilarity	baseline
Hevner	0.70	0.62 (8 clusters)
MIREX	0.56	0.56 (5 clusters)
Folksonomy	0.9	0.51 (4 clusters)

Table 3.7: Inter-cluster dissimilarity means for each mood taxonomy and its random baseline for comparison

3.2.4. Dimensional representation

Dimensional representation is an important paradigm in emotion studies. We provided several examples in the previous chapter. To visualize our dataset in a comparable way, we decided to project our semantic mood space into a bi-dimensional space. To achieve this goal, we used the Self-Organizing Map algorithm (SOM), also called by its inventor Kohonen map (see Kohonen (1982)). This data transformation technique reduces the dimension of an input space through the use of self-organizing neural networks. An artificial neural network is trained using unsupervised learning to construct a low-dimensional representation of the input space samples (usually between 1 and 3 dimensions to allow a visual representation). We can consider SOM as a non-linear generalization of principal component analysis (PCA).
Considering x a point from the input space, we aim at mapping it to a point (x) in the output space. Also each point I from output space will map to a point w(I) in the input space. The main characteristic of this algorithm is the use of a discrete output space made with connected neurons. Initially the connection weights are initialized with small random values. The SOM algorithm can be divided in five major steps:

Initialization: Initialize a map (called topographic map) with random values for the initial weight vectors w_j

Sampling: Select a sample training input vector x from the input space

Matching: Find the winning neuron $I(x)$ with weight vector closest to the selected input vector x

Updating: Apply the weight update equation:
$\Delta w_{ji} = (t)T_{j,I(x)}\eta(t)(x_i - w_{ji})$, where $T_{j,I(x)}$ is a Gaussian neighborhood and $\eta(t)$ is the learning rate

Continuation: Go to Sampling step until convergence

In our case, we want to visualize the semantic mood space in 2 dimensions. We decided to use SOM for its topology properties and because it stresses more on the local similarities and distinguishes groups within the data. Because less than half of the Russell's adjectives are present in our dataset, we prefer to compare qualitatively more that quantitatively the expert and the community models. We trained a SOM and mapped each tag onto its best-matching unit in the trained SOM. In Figure 3.8, we plot the resulting organization of mood tags (for clarity reasons, we show here a subset of 58 tags).

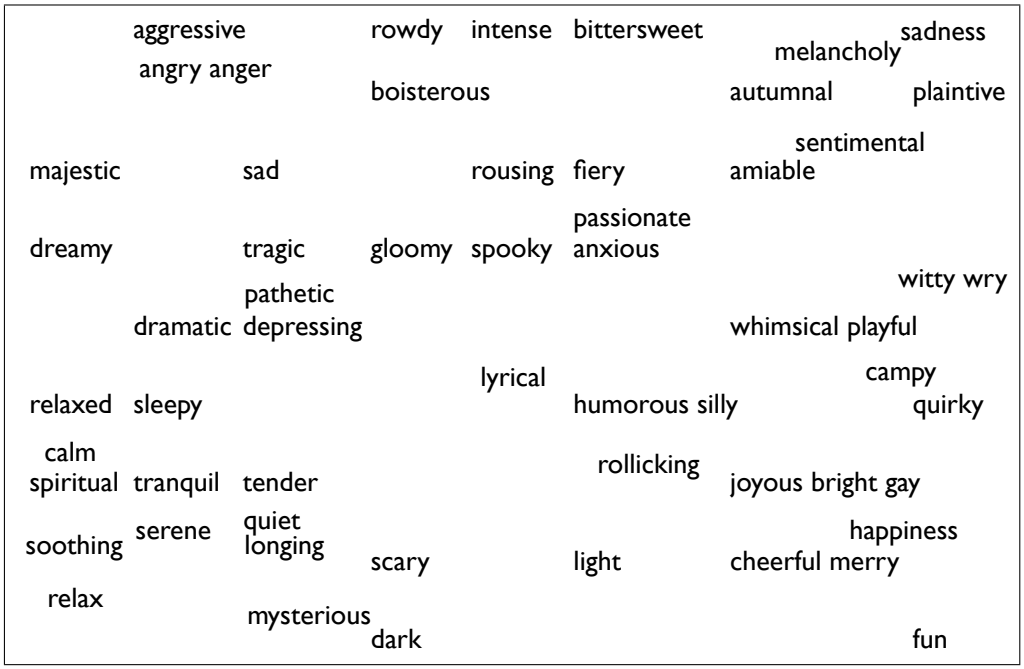

Figure 3.8: Self-Organizing Map of the mood tags in the semantic space.

We observe in the 2D projection four main parts. At the top-left, terms related to *aggressive*, below *calm* and other similar words, at the top-right tags related to *sad* and below words close to *happiness*. We notice the four clusters corresponding to the basic emotions and our folksonomy representation mentioned in Section 3.2.3. This is somehow expected as we already got these clusters from the semantic space. However, having the same results with a second technique confirms our findings. Comparing with Russell's dimensions, we find that the diagonal from top-left to bottom-right is of high arousal. On the contrary, the diagonal from top-right to bottom-left is of low arousal. The vertical axis represents the valence dimension. Even though the 2D representation is not equal, there is a correlation (although difficult to quantify) between the community and the experts when framing the problem into two dimensions.

3.2.5. Hierarchical representation

The semantic mood space can be visualized in many different ways. In this subsection we experimented hierarchical clustering techniques to produce a tree diagram (dendrogram). In hierarchical clustering the data points are not assigned to a cluster in a single step. Alternatively, the data is partitioned by aggregating or dividing sample sets. In the agglomerative method, more commonly used, the samples are grouped by series of fusions, obtaining a hierarchical tree representation also called dendrogram. The divisive method reaches the same result by successively separating the sample sets into smaller groups. There are several ways to define distance between clusters. The single linkage, or nearest neighbors, defines the cluster distance as the distance between the closest pair of samples. The complete linkage technique, also called farthest neighbor, is the opposite of the single linkage. Cluster distances are defined as the distance between the most distant pair of objects of each group. Taking two clusters X and Y, with x and y respectively from cluster X and Y, the distance with the complete linkage technique is defined as:

$$D(X, Y) = max(d(x, y)) \tag{3.8}$$

We applied a common agglomerative hierarchical clustering method with a complete linkage (Xu & Wunsch (2008)) and the same cosine distance applied previously. We used the hcluster[3] implementation to conduct this analysis. With the 20 most used tags in our dataset, we computed the clustering and plot the resulting dendrogram in Figure 3.9 .

Although there exists some dendrogram representation of emotions in the psychology literature (see Juslin & Sloboda (2001)), the comparison is complex because many of the terms employed there are not present in our dataset and also because finding the right metric to measure the similarity between both is not trivial.

The hierarchical clustering starts with two branches. Looking at the tags of this first branching, we observe a very clear separation in arousal with *dreamy* and *calm* on the left and *angry* and *happy* on the right. Then the two following branching (resulting in four clusters) represents the four basic emotions also found as the best categorical representation in Section 3.2.3 (in order in the dendrogram: *calm*, *sad*, *angry* and *happy*, highlighted in Figure 3.10). This confirms another time our findings about the relevance of these four clusters. Moreover, we notice that the clearest separation is related to arousal, often considered as the most important dimension. The remaining branches group together similar terms like *angry* and *aggressive* or *sad* and *depressing*.

[3] http://code.google.com/p/scipy- cluster

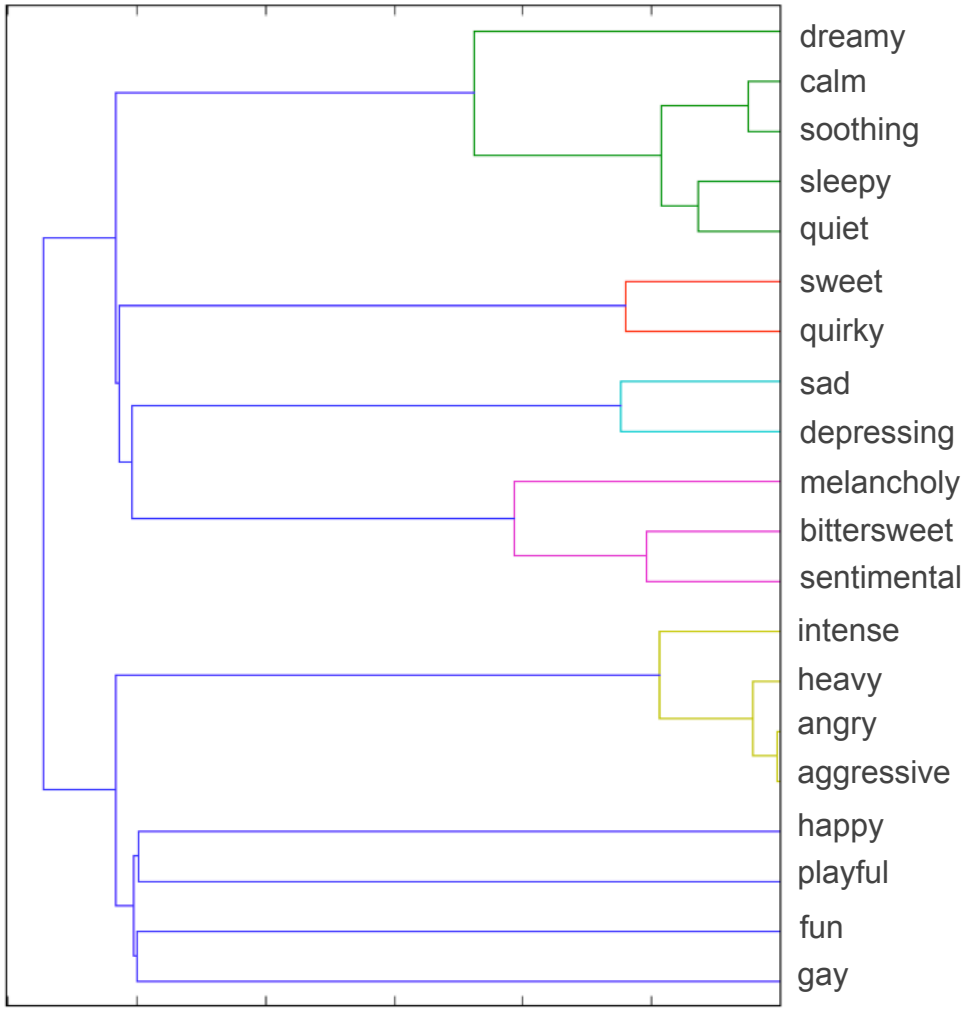

Figure 3.9: Dendrogram of the 20 most used tags.

3.3. Conclusion

This chapter presented convergent evidence about mood representations by means of analyzing data from a social network. We created a semantic mood space based on a community of users from *last.fm*. We derived different representations from this data and compared them to the expert representations taken from literature on psychology and emotions. We demonstrated that the basic emotions, that can be summarized as: *happy*, *sad*, *angry* and *tender*, are very relevant to the social network. We also found that the arousal and valence dimensions are pertinent both using a dimensional reduction and a hierarchical clustering. Moreover we have shown that both Hevner's and MIREX representations have advantages and limitations when evaluated in the seman-

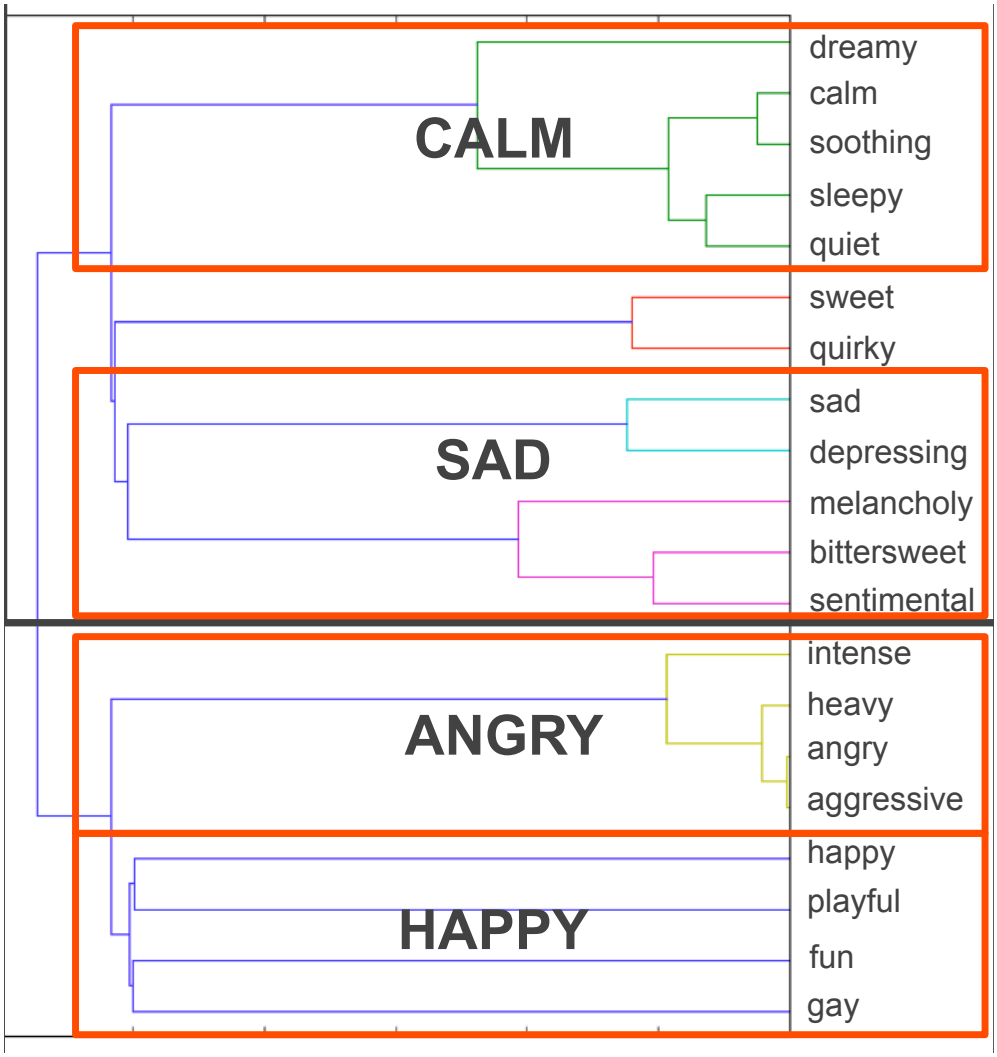

Figure 3.10: Dendrogram of the 20 most used tags. We highlighted the branching corresponding first to arousal and then valence. The result after the second branching are shown as clusters very related to previously found basic emotions

tic mood space. The former having better separated clusters and the latter having more consistent clusters. Observations on the confusion and similarity between MIREX clusters confirmed results from previous analysis (Hu et al. (2008)), and support the idea that it may not be an optimal representation. We also presented a dendrogram visualization using hierarchical clustering, validating again the basic emotion postulate (with *happy*, *sad*, *angry* and *calm*), and offering a new representation of the mood space. All these findings show the relevancy of using a mood semantic space derived from social tags. We believe that this approach can be generalized to find other domain-specific rep-

resentations and would allow to find validation for other theories. We should however keep in mind the limitations of this approach. Even if we benefit from the huge amount of data a social network like *last.fm* provides (especially compared to traditional studies), we must be aware of the bias it introduces. The community contributing to this social network is made of young people (but also many psychological studies are conducted on students only), technology and music enthusiasts, mostly listening to western mainstream music. This is also a explanation why Hevner's approach does no fit into the semantic mood space. It is not a recent study and it was conducted using classical music. In our goal to find a representation model, we are confident in using the basic emotions naturally found from the data of our experiment. However, we prefer not to limit the model to mutually exclusive categories and to consider each mood category against its complementary in a binary approach.

CHAPTER 4

Mood Classification from Audio

"The advantage of the emotions is that they lead us astray, and the advantage of science is that it is not emotional" Oscar Wilde.

4.1. Introduction

Can an algorithm decode the emotions a music contains? This is the main question we want to answer here. This Chapter is about adapting existing approaches in music classification (reviewed in Chapter 2) to our case of mood classification from audio. As deduced from the literature review, we will constraint our approach focusing on the emotion evoked, not the emotion induced. We define the problem as classifying the mood, or lasting emotion contained in a musical piece. With this approach, we make the problem more objective taking into account the intended emotion and not the induced emotion. In this part of the thesis, we propose to explore the possibilities of using the raw audio signal a unique source of information.
To classify music by mood, we frame the question as an audio classification problem using a supervised learning approach as explained in Chapter 2 (Section 2.4). In Figure 4.1, we present a schema of the approach. This method is divided into 4 main components:

- Ground truth creation: selecting examples to train our classifier.
- Audio Feature extraction: extract audio descriptors from the signal that will be mapped to the class we want to predict with the classifier.
- Training and Classification: choose the classification method to obtain the trained model.
- Evaluation.

We base our learning approach on examples and consequently our first phase is to build a ground truth (a dataset of examples of the positive and negative categories to be learned by a classifier). From the previous chapter, we find that four main emotions clusters are particularly relevant: *happy*, *sad*, *angry* and *relaxed*[1]. This approach is following both the basic emotion theory and the folksonomy that emerged from the music social network tags. We decide to consider these unambiguous categories to allow for a greater understanding and agreement between people (both human annotators and end-users). We build the ground truth to train machine learning models on both social network knowledge (wisdom of crowds) and experts validation (wisdom of the few), as explained in the next section of this chapter (Section 4.2). Then we extract a rich set of audio features that we describe in Section 4.3, analyzing the distribution of their values according to the mood categories. We apply classification techniques that we evaluate in Section 4.4. Once the best algorithm is chosen, we evaluate the contribution of each descriptor type in 4.4.2 and the robustness of the model as reported in Section 4.5.

4.2. Ground Truth

For this study we use a categorical approach to represent the mood. We focus on the following categories: happy, sad, angry, and relaxed. The reason for this choice has been motivated by the results detailed in the previous chapter. As we do not want to be restricted to exclusive categories, we consider the problem as a binary classification task for each mood. One song can be *happy* or *not happy*, but also independently *angry* or *not angry* and so on.
The main idea of the present method to collect ground truth is to exploit information extracted from both a social network and several experts validating the data. To do so, we have pre-selected the tracks to be annotated using last.fm[2] tags (textual labels). Last.fm is a music recommendation website with a large community of users (30 million active users in more than 200 countries[3]) that is very active in associating tags with the music they listen to. These tags are then available to all users in the community. In Figure 4.2, we show an example of a "tag cloud", which is a visualization of the tags assigned to one song with the font size weighted by the popularity of the tag for this particular song.
In the example shown in 4.2, we can see that *happy* is present and quite highly weighted (which means that many people have used this tag to describe the song). In addition to *happy*, we also have "cheerful", "joy", "fun" and "upbeat". To gather more data, we need to extend our query made to last.fm with more words related to mood. For the four chosen mood categories, we gener-

[1] We will use synonym of *calm*, *relaxed* in the remainder of this document
[2] http://www.last.fm
[3] March 2009. http://blog.last.fm/2009/03/24/lastfm-radio-announcement

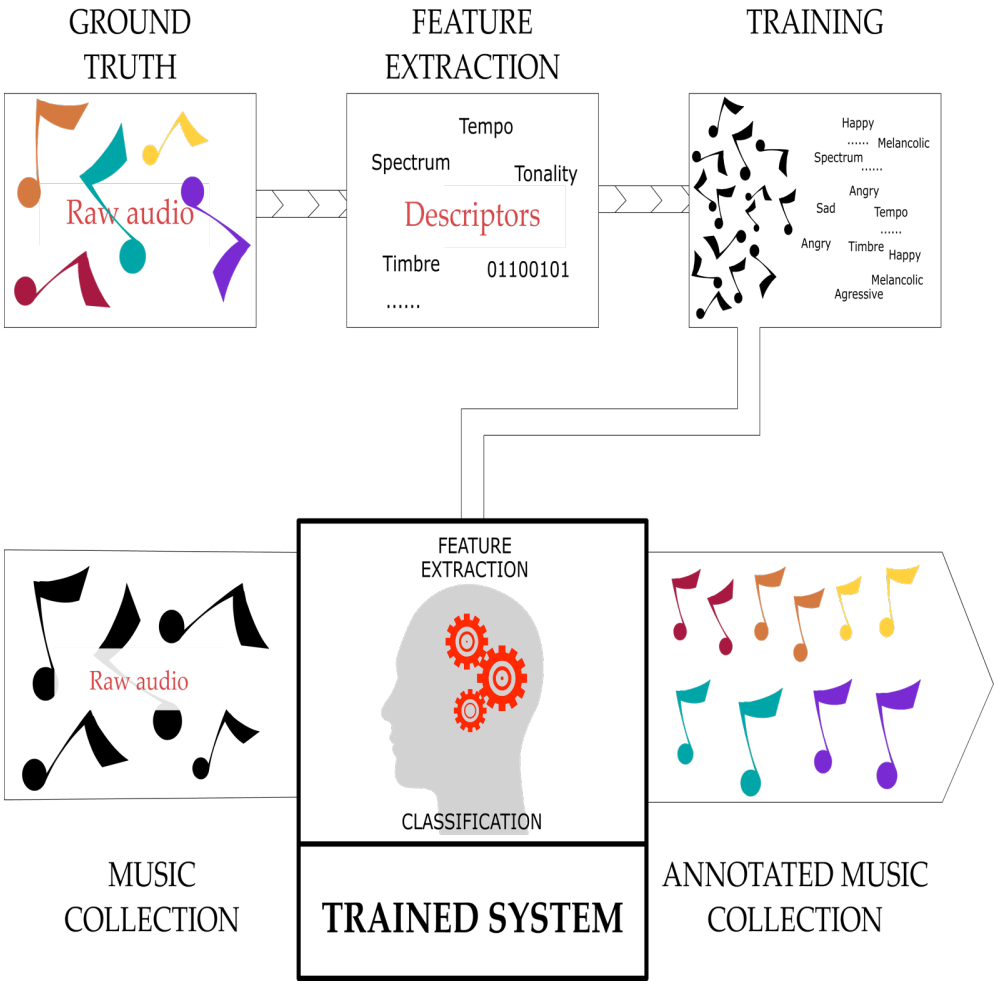

Figure 4.1: Method for Music Mood classification: Supervised learning

ated a set of related semantic words with the help of Wordnet[4], a large lexical database of English words with semantic relationships (such as synonyms), and looked for the songs frequently tagged with these terms. For instance "joy", "joyous", "cheerful" and "happiness" are grouped under the *happy* category to generate a larger result set. We query the social network to acquire songs tagged with these words and apply a popularity threshold to select the best instances (we keep the songs that have been tagged by many users).

Note that the music for the "not" categories (like *not happy*) was evenly selected using both music tagged with antonyms (also using Wordnet), and a random selection to create more diversity. Afterwards, we asked 17 listeners

[4] http://wordnet.princeton.edu

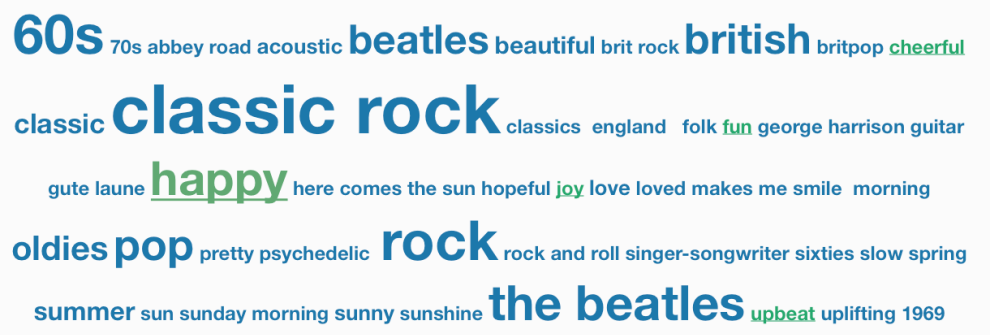

Figure 4.2: Tag cloud of the song "Here comes the sun" from the Beatles. The tags recognized as mood tags are underlined. The bigger the tag is, more people have used it to define that song.

(mainly students and researchers at the Music Technology Group) to validate this selection. We considered a song to be valid if the tag was confirmed by, at least, one listener, as the pre-selection from last.fm granted that the song was likely to deserve that tag. We included this manual tag confirmation in order to exclude songs that could have received the tag by error, to express something else, or by a "following the majority" type of effect. The listeners were given only 30 seconds of the songs to avoid, as much as possible, changes in the mood and to speed up the annotation process. Consequently, only these 30 second excerpts have been included in the final dataset. In total, 17 different evaluators participated and an average of 71% of the songs originally selected from last.fm was included in the training set. We observe that the *happy* and *relaxed* categories have a higher validation rate (ratio of musical pieces validated over discarded ones) than the *angry* and *sad* categories. This might be due to confusing terms in the tags used in the social networks for these latter categories or to a better agreement between people for "positive" emotions (*happy* and *relaxed* are considered of positive valence (See Russell (1980))). These results indicate that the validation by experts is a necessary step to ensure the quality of the dataset. If we would have just blindly followed the tags assigned by the community of last.fm users, around 29% of errors, on average, would have been introduced. This should be considered as an important advice for other research. It is not safe to blindly trust tags, mainly because we do not know the context and the intention of the person who attached it to a particular track. Our method is relevant to pre-selecting a large number of tracks that potentially belong to one category. In Figure 4.3, we present a schema summarizing the overall ground truth creation process.

At the end of the song selection process, the database was composed of 1000 songs excerpts divided between the 4 categories of interest plus their complementary categories ("not happy", *not sad*, *not angry* and *not relaxed*), i.e. 125

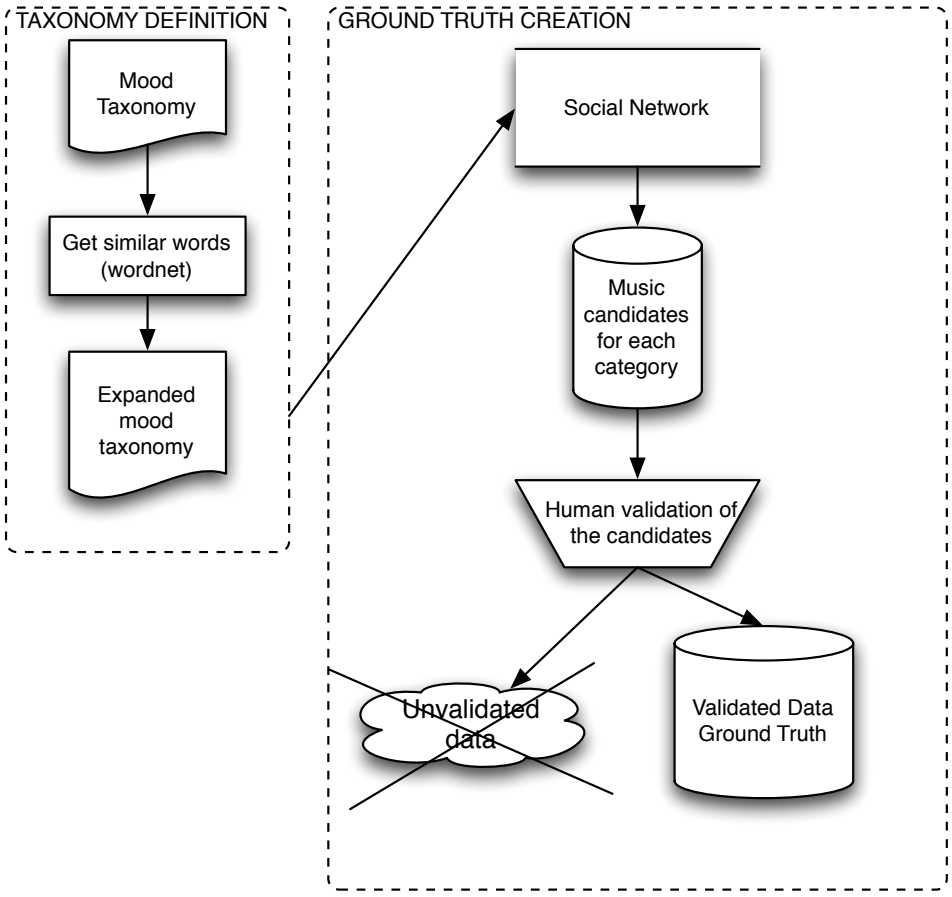

Figure 4.3: Schema of the method employed to create the ground truth

songs per category. The audio files are 30-second stereo clips at 44khz in a 128kbps mp3 format.

4.3. Audio Feature Extraction

We call audio features or audio descriptors those variables we extract from the audio signal describing some aspect of the information it contains. This is a key phase in audio classification in general, as these descriptors are what the classifiers will use as input. Indeed, we extracted a rich set of audio features based on temporal and spectral representations of the audio signal. This has been done using our library called Essentia (Wack (2010)). It contains most of the standard audio descriptors (please refer to Peeters (2004) and Guaus (2009) for more details on particular descriptors). To extract these descriptors, for each audio excerpt, we merge the stereo channels into a mono mixture. Then its frame-based features are summarized with their component-wise statistics across the whole song. In Table 4.1, we present an overview of the extracted descriptors by category.

Type	Features
Timbre	Bark bands, MFCCs, pitch salience, hfc, loudness, spectral: flatness, flux, rolloff, complexity, centroid, kurtosis, skewness, crest, decrease, spread
Tonal	dissonance, chords change rate, mode, key strength tuning diatonic strength, tristimulus
Rhythm	bpm, bpm confidence, zero-crossing rate, silence rate, onset rate

Table 4.1: Overview of the audio features extracted by category. See Peeters (2004), Gouyon et al. (2008), Logan (2000) and Guaus (2009) for a detailed description of the mentioned features.

Following this procedure, for each excerpt of the ground truth, we obtain a total of 200 feature statistics (minimum, maximum, mean, variance and derivatives), that we standardize across the whole music collection values. In the next sections, we describe some of the most relevant descriptors for this mood classification task, with results and figures based on the training data.

4.3.1. Experiment 2: Correlation between audio features and mood categories

We would like to know which audio descriptors are relevant and correlates with the mood of a song. When possible, we also aim at finding explanations or hypotheses justifying these relations. In this section, we present comparisons of descriptor distributions for different categories. We have chosen here to show the most discriminative descriptors and the clearest results we obtain analyzing our ground truth. Note that with our approach, we have always two categories: c and not_c (for instance *happy* and *not happy*). The selection technique to decide which descriptor to consider for this analysis is based on the following rule: considering descriptor values x_c for n songs of class c as:

$x_c = [x1_c, x2_c, ...xn_c]$, we compute means and standard deviations for x_c and x_{not_c} as μ_c, μ_{not_c}, σ_c and σ_{not_c}. Then we consider the descriptor as relevant if:

$$|\mu_c - \mu_{not\_c}| > \sigma_c + \sigma_{not\_c} \tag{4.1}$$

This criterion ensures that the descriptor distributions are quite different between the category and its complementary. While observing the comparison between descriptors in the next part, we also explicit how each feature is computed.

Mel Frequency Cepstral Coefficients (MFCCs)

MFCCs Logan (2000) are widely used in audio analysis, and especially for speech research and music classification tasks. The method employed is to divide the signal into frames. For each frame, we take the Cepstrum, defined as the Inverse Fourier Transform of the logarithm of the amplitude spectrum:

$$c[n] = \frac{1}{N} \sum_{k=0}^{N-1} log_{10}|X[k]|^{j\frac{2\pi}{N}kn}, 0 < n < N-1 \tag{4.2}$$

where $X[k]$ is the spectrum of the input signal and N its length in samples. Then we divide it into bands and convert it to the perceptually-based Mel spectrum (based on the *mel* scale). The *mel* scale tries to map the perceived frequency of a tone onto a linear scale that approximates the frequency resolution of our hearing:

$$mel_{frequency} = 2595 \cdot log_{10}\left[1 + \frac{f}{700}\right] \tag{4.3}$$

Finally we take the discrete cosine transform (DCT). The number of output coefficients of the DCT is variable, and is often set to 13, as we did in the present study. Intuitively, lower coefficients represent spectral envelope, while higher ones represent finer details of the spectrum but most of them are not directly interpretable. In Figure 4.4, we show the mean values of the MFCCs for the *sad* and *not sad* categories. We note a difference in the shape of the MFCCs. This indicates a potential usefulness to discriminate between the two categories. This is also the case for other mood categories and in particular *angry* versus *not angry* (also in Figure 4.4). It is probably less clear for the others, *happy* and *relaxed* in Figure 4.5. However, this way of visualizing the data is not so informative to intuitively understand and interpret the differences.

Bark bands

The Bark band algorithm computes the spectral energy contained in a given number of bands, which corresponds to an extrapolation of the Bark band

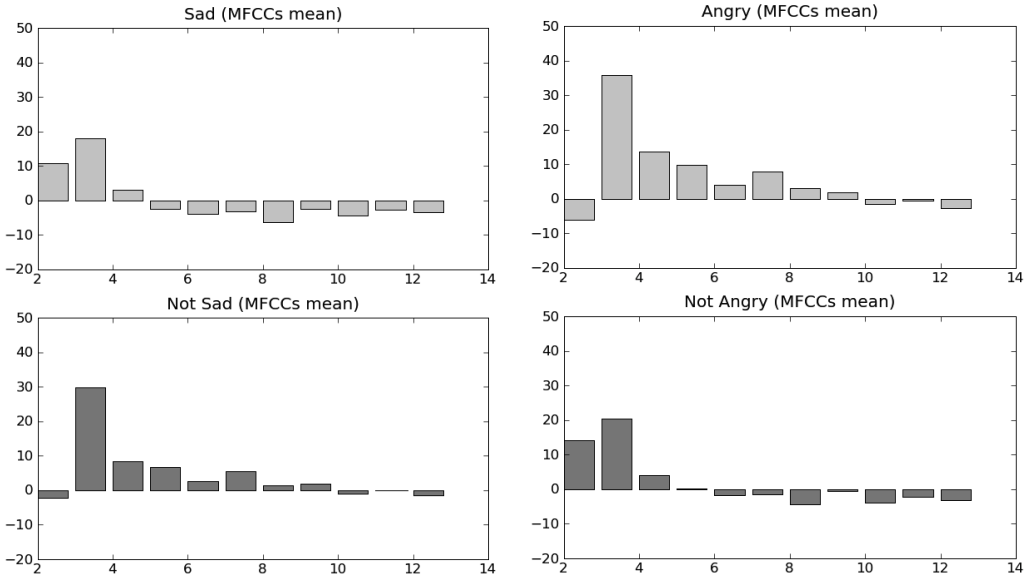

Figure 4.4: MFCC mean values for coefficients between 2 and 13 for the *sad* and *angry* categories of our annotated dataset.

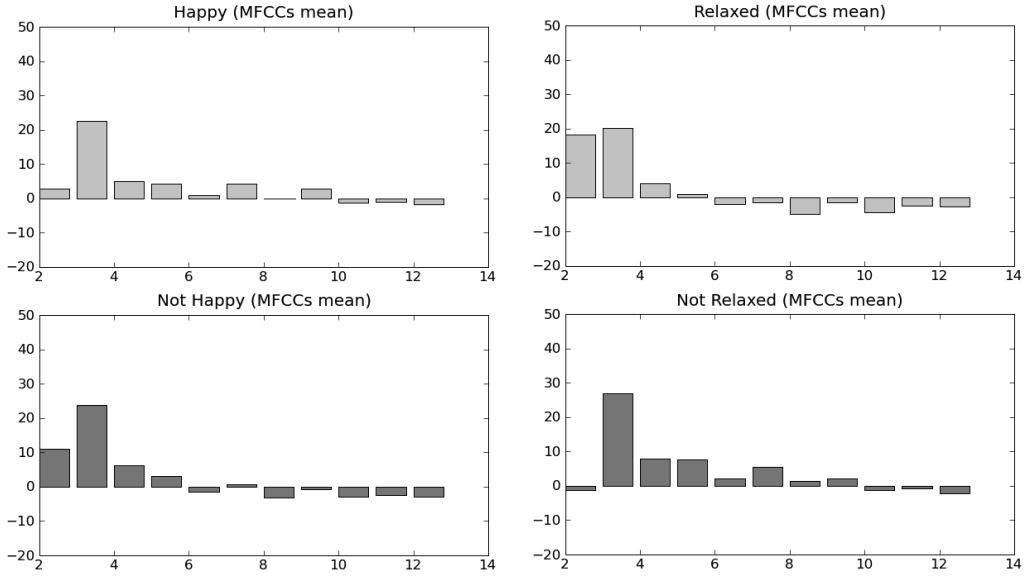

Figure 4.5: MFCC mean values for coefficients between 2 and 13 for the *happy* and *relaxed* categories of our annotated dataset.

scale (see Peeters (2004) and Smith & Abel (1999)). Barks correspond to bandwidths of human auditory filters (Zwicker & Terhardt (1980)). Here is how to convert frequency into Bark:

$$bark = 13 \cdot arctan\left(\frac{0.76}{100} f\right) + 3.5 \cdot arctan\left(\left(\frac{f}{7500}\right)^2\right) \tag{4.4}$$

For each Bark band (27 in total) the power-spectrum is summed. In Figure 4.6, we show an example of the Bark bands means for the *sad* category.

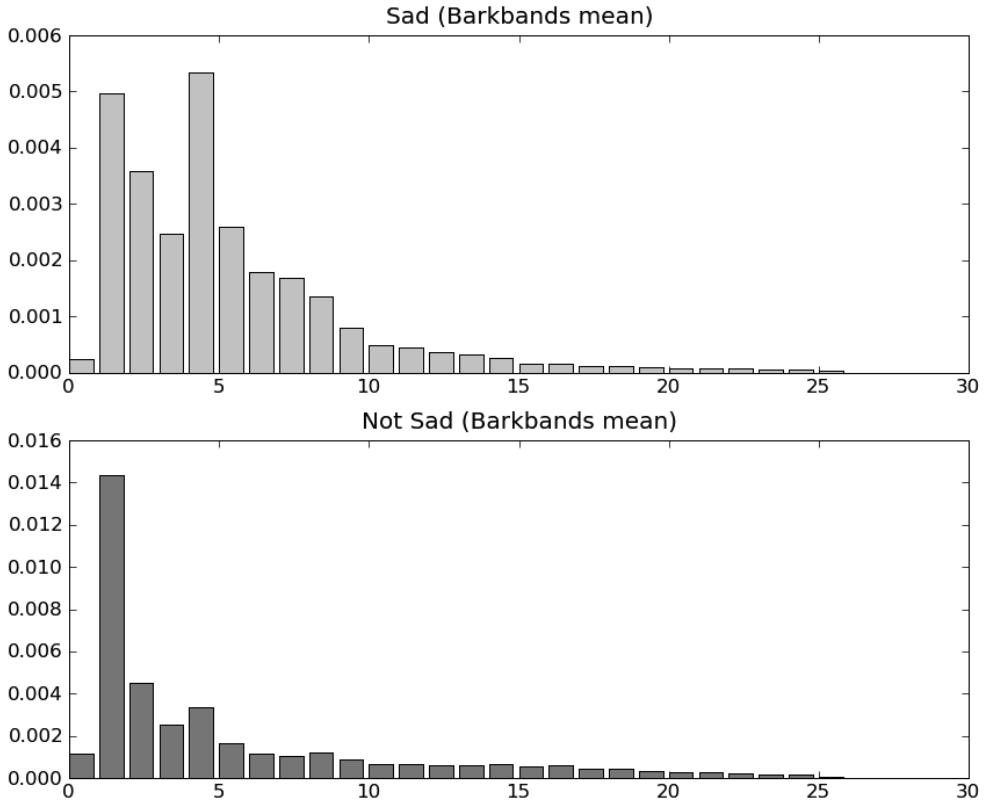

Figure 4.6: Bark band mean values for coefficients between 1 and 27 for the *sad* and *not sad* categories of our annotated dataset.

As with the MFCCs (these approaches are similar), the Bark bands appear to have quite different shapes for the two categories, indicating a probable utility for classification purposes. Again, except its potential to classify *sad* against *not sad* (and the same for other categories) there is no particular interpretation we could deduce from these results. For this reason too, we do not plot the other categories.

Zero Crossing Rate

This one of the simplest time domain descriptor. It measures the rate in which the waveform changes its sign (crossing zero). Kedem (1986) and Saunders (1996) defined it as a measure of the weighted average of the spectral energy distribution.

$$ZCR = \frac{1}{2}\sum_{n=1}^{N} |sign(x[n]) - sign(x[n-1])| \tag{4.5}$$

This descriptor has been used a lot in music information retrieval (but also in speech recognition) because it is easy to compute and partially reveals the noisiness of an audio excerpt. Figure 4.7 shows the box-and-whisker plots of the zero crossing rate standardized means for the *relaxed*, *not relaxed*, *angry* and *not angry* categories. These results are based on the entire training dataset (this is true also for all the similar plots of this section). We plot these two comparison because the separation is not that clear for the other categories. A key to interpret the values is that this descriptor would give higher values for noisy sounds and lower values for more periodic sounds. Looking at our results, this would mean that *angry* music is more noisy than *relaxed* music. We will study better these differences using more precise descriptors of the spectral shape in 4.3.1. We will also observe similar results with the spectral complexity descriptor in the next paragraph.

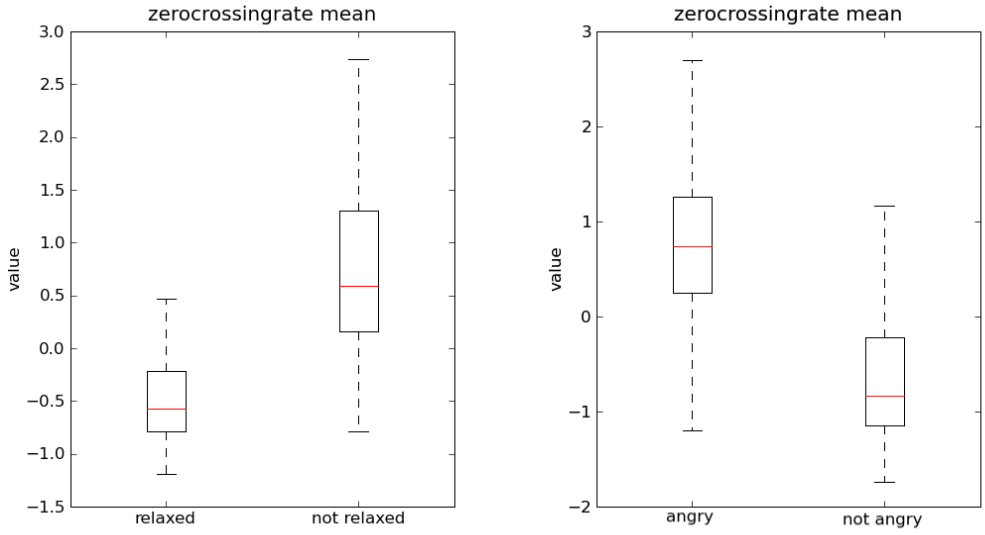

Figure 4.7: Box-and-whisker plot of the standardized zero crossing rate mean value for *relaxed* / *not relaxed*, and *angry* / *not angry*.

Spectral Complexity

The spectral complexity descriptor is based on the number of peaks in the input spectrum. We apply peak detection on the spectrum (between 100Hz and 5Khz) and we count the number of peaks. This feature describes the complexity of the audio signal in terms of frequency components. In Figures 4.8, we show the box-and-whisker plots of the spectral complexity descriptor's standardized means for the *relaxed*, *not relaxed*, *happy* and *not happy* categories. In Figures 4.9, we plot the same results but for the *angry* and *sad* categories. These plots illustrate the intuitive result that a relaxed song should be less "complex" than a non-relaxing song. Moreover, Figure 4.8 tells us that happy songs are on average spectrally more complex. On the contrary, we observe that *angry* song are more "complex" which is also a very intuitive result (and the logical opposite for *relaxed*). We note that *sad* and *relaxed* have similar distributions in terms of spectral complexity.

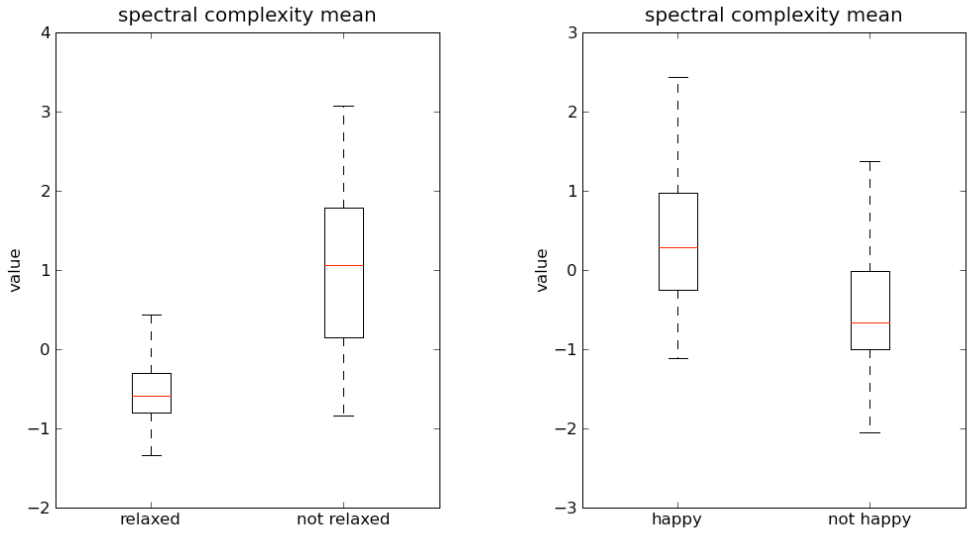

Figure 4.8: Box-and-whisker plot of the standardized spectral complexity mean feature for *relaxed* / *not relaxed*, and *happy* / *not happy*.

Spectral Shape

The spectral centroid, skewness, kurtosis and rolloff descriptors, as reported by Peeters (2004) are descriptions of the spectral shape (as well as others such as spread or decrease). MFCCs and Barkbands detailed before could also be considered in that category.

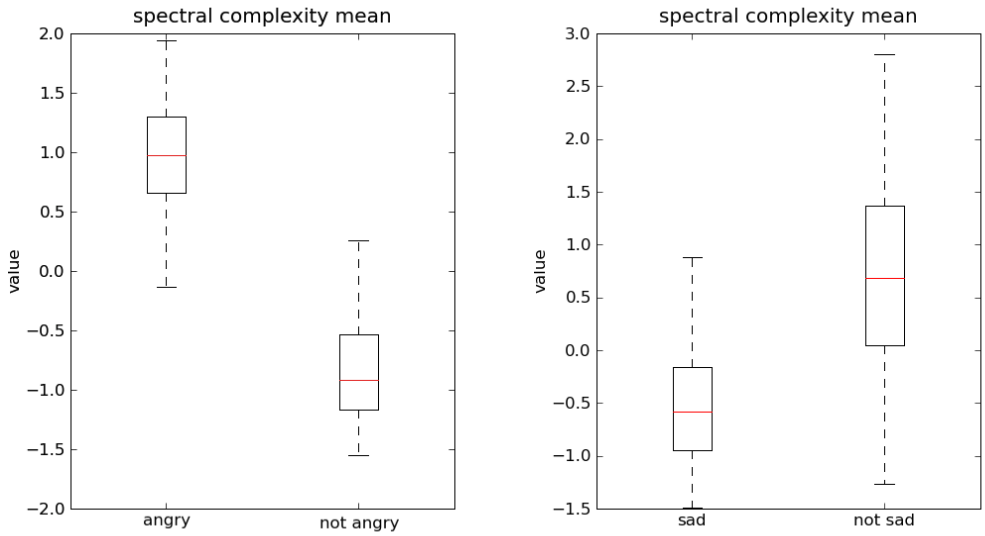

Figure 4.9: Box-and-whisker plot of the standardized spectral complexity mean feature for *angry* / *not angry*, and *sad* / *not sad*.

Spectral Centroid: the spectral centroid is the barycenter of the spectrum, which considers the spectrum as a distribution of frequencies. It can be expressed as:

$$SpectalCentroid = \frac{\sum f_i a_i}{\sum a_i} \tag{4.6}$$

where f_i is the frequency value of each FFT bin and a_i its amplitude.

Spectral Skewness: the spectral skewness measures the asymmetry of the spectrum's distribution around its mean value. If the skewness is negative, more data is on the left of the mean than on the right. If it is positive, more data is on the right of the mean. The skewness of a distribution is expressed as follows:

$$Skewness = \frac{E(x-\mu)^3}{\sigma^3} \tag{4.7}$$

where x are the observed data (normalized amplitude values), μ is the mean of x, σ its standard deviation and $E(x)$ the expected value of x.

Spectral Kurtosis: the kurtosis of a distribution is a measure its flatness around its mean value. It is defined as:

$$Kurtosis = \frac{E(x-\mu)^4}{\sigma^4} \tag{4.8}$$

where x are the observed data (normalized amplitude values), where μ is the mean of x, σ its standard deviation and $E(x)$ the expected value of x. A normal distribution has a kurtosis of 3. Values lower than 3 indicated a flatter distribution and values greater than 3 a peaker distribution.

Spectral Roll-Off: the spectral roll-off is the frequency that splits the signal energy in two parts using a threshold in energy. It can be computed as defined by Tzanetakis & Cook (2002):

$$SpectralRF_t = max\{f | \sum_{n=1}^{f} M_t[n] < TH \cdot \sum_{n=1}^{N} M_t[n]\} \tag{4.9}$$

where $M_t[n]$ the magnitude of the Fourier transform at frame t and frequency bin n. TH is the energy threshold, typically 0.95, meaning that the spectral roll-off is the frequency point where 95% of the signal energy contained is below. In some sense, this descriptor is correlated to the harmonic/noise cutting frequency like pointed out by Peeters (2004). In Figure 4.10 we plot the spectral centroid's box-and-whisker plot for *angry* and the spectral skewness for *sad*. It shows a higher spectral centroid mean value for *angry* than *not angry*, which intuitively means more energy in higher frequencies. For the spectral skewness, the range of mean values for the *sad* instances is bigger than for the *not sad* ones. This probably means that there is a less specific value for the centroid. In any case, it seems to have on average a lower value for the *not sad* instances.
In Figure 4.11 we plot the same analysis for the spectral kurtosis and roll-off descriptors respectively for the *relaxed* and *sad* categories. It is worth noticing the peaky distribution of *not relaxed* category compared to *relaxed* which is much broader. This descriptor is most probably very useful to discover *not relaxed* instances. *Sad* songs seems to have a lower spectral roll-off, with more energy in the lower part of the spectrum like we can also see in figures from the analysis of the MFCCs and Bark bands (Figures 4.4 and 4.6).

Spectral Flatness dB

This descriptor is based on the mean of the power spectral density components, in each critical band, for the input signal. It is defined by the ratio of its geometrical mean to the arithmetical mean as follows (converted to decibels, as detailed in Johnston (1998)):

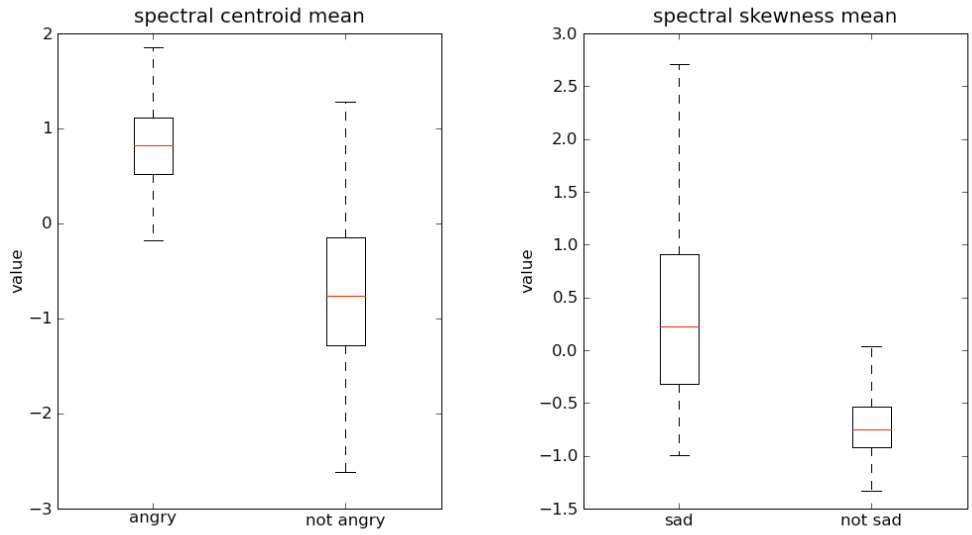

Figure 4.10: Box-and-whisker plot of the standardized spectral centroid mean for *angry* and *not angry* and of the standardized spectral skewness mean for *sad* and *not sad*.

$$SpectralFlatnessDB = 10 \cdot log_{10} \frac{G_m}{A_m} \tag{4.10}$$

where G_m is the geometrical mean and A_m the arithmetical mean. Izmirli (1999) describes how to compute this descriptor. We can interpret its value as a sort of "tonality" in the sense that a high value would describe a signal that is tone-like, and a low value a signal that is totally noise-like.
In Figure 4.12, we compare the *relaxed* and *angry* moods by means of spectral flatness. As expected, *angry* is more noisy and *relaxed* more tonal. This also correlates with the results shown about the spectral complexity descriptor. However we do not see anything significant for *happy* (which is only a bit more noisy that the *not happy* category) and sad which has spread values and a distribution with no particularity and hence is not plotted.

Dissonance

The dissonance feature (also known as "roughness", see Sethares (1998)) is defined by computing the peaks of the spectrum and measuring the spacing of these peaks. Consonant sounds have more evenly spaced spectral peaks and, on the contrary, dissonant sounds have more irregularly spaced spectral peaks.

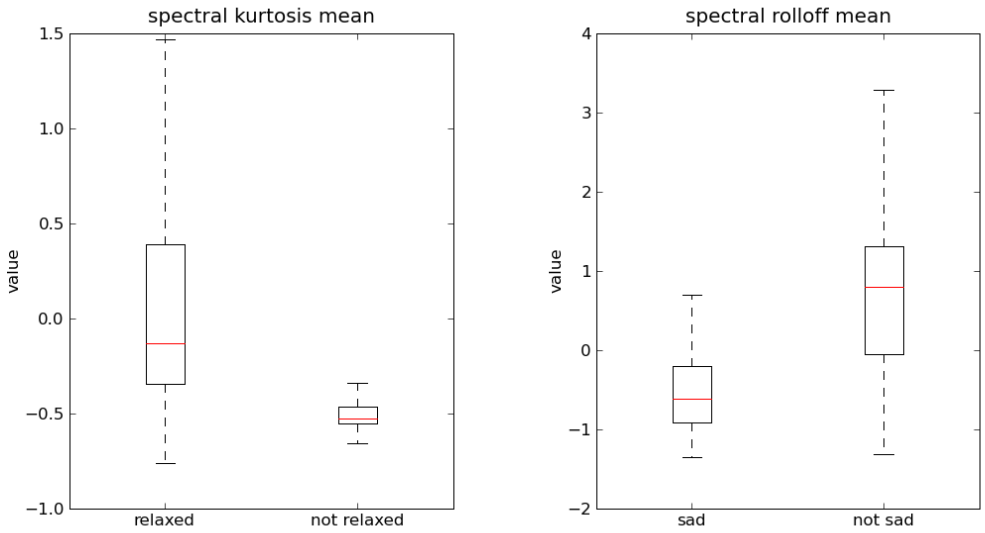

Figure 4.11: Box-and-whisker plot of the standardized spectral kurtosis mean for *relaxed* and *not relaxed* and of the standardized spectral roll-off mean for *sad* and *not sad*.

This descriptor is also described as "Harmonic Spectral Deviation" by Peeters (2004):

$$Dissonance = \frac{1}{H} \sum_{h} a(h) - SE(h) \tag{4.11}$$

where H is the number of harmonics, $a(h)$ the amplitude of the harmonic h and $SE(h)$ the amplitude of the spectral envelope at frequency $f(h)$.

In Figure 4.13, we compare the dissonance distributions for the *relaxed* and *angry* categories. These figures show that *angry* is clearly more dissonant than *not angry*. Listening to the excerpts from the training data, we noticed many examples with distorted sounds like electric guitar in the *angry* category, which seems to be captured by this descriptor. Moreover this observation correlates with the spectral flatness and spectral complexity descriptor results. These findings also relate to psychological studies stating that dissonant harmony may be associated with anger, excitement and unpleasantness (Hevner (1936), Wedin (1972)). Our analysis goes in the same direction and confirms these results from psychology.

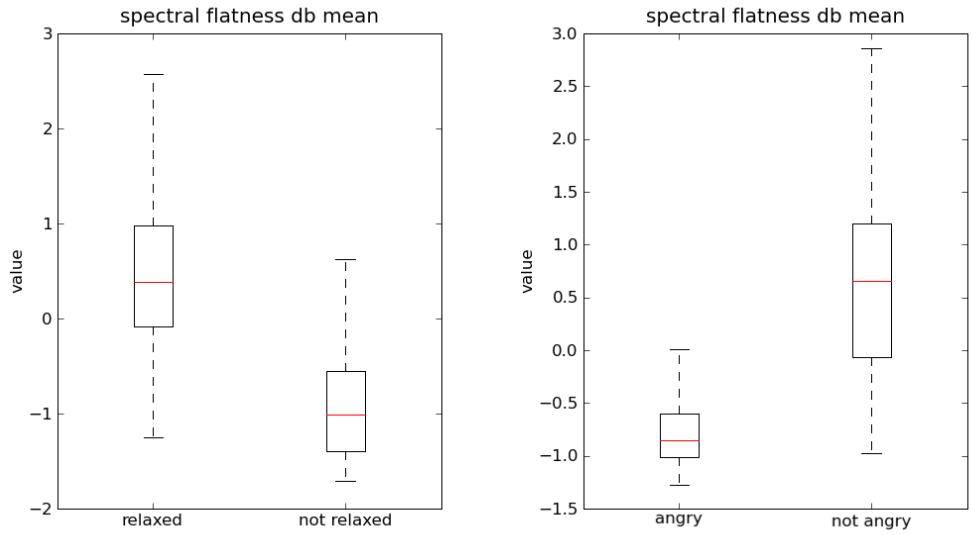

Figure 4.12: Box-and-whisker plot of the standardized spectral flatness mean value for *relaxed* / *not relaxed*, and *angry* / *not angry*.

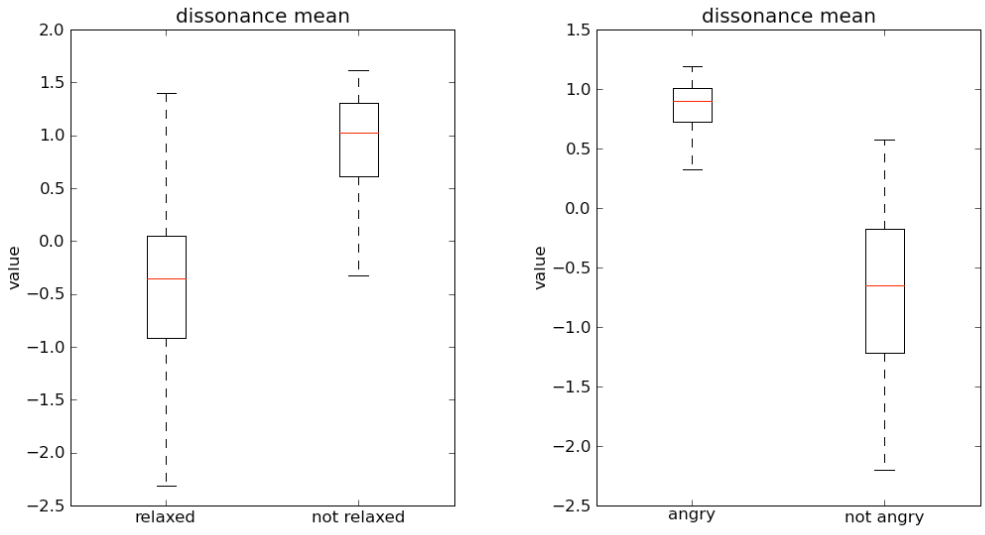

Figure 4.13: Box-and-whisker plot of the standardized dissonance mean for *relaxed* and *not relaxed*, and for the *angry* and *not angry* categories

Mode

In Western music theory, there are two basic modes: major and minor. Each of them has different musical characteristics regarding the position of tones and semitones within their respective musical scales. Gómez (2006) explains how to compute an estimation of the mode from raw audio data. The signal is first pre-processed using the direct Fourier transform (DFT), filtering frequencies between 100 Hz and 5000 Hz and locating spectral peaks. The reference frequency (tuning frequency) is then estimated by analyzing the frequency deviation of the located spectral peaks. Next, the Harmonic Pitch Class Profile (HPCP) feature is computed by mapping frequency and pitch class values (musical notes) using a logarithmic function (Gómez (2006)):

$$HPCP(n) = \sum_{i=1}^{nPeaks} w(n, f_i) \cdot a_i^2 \qquad n = 1...N \tag{4.12}$$

The global HPCP vector is the average of the instantaneous values per frame, normalized to [0,1] to make it independent of dynamic changes. The resulting feature vector represents the average distribution of energy among the different musical notes. Finally, this vector is compared to minor and major reference key profiles based on music theory (Krumhansl (1997)). The profile with the highest correlation with the HPCP vector defines the mode. In Figure 4.14, we represent the percentages of estimated major and minor music in the *happy* and *not happy* categories. We note that there is more major music in the *happy* than in the *not happy* pieces. In music theory and psychological research, the link between valence (positivity) and the musical mode has already been demonstrated (Juslin & Laukka (2004)). Having empirical data from an audio feature automatically extracted showing the same tendency is an interesting result. We note also that the proportion of major music is also high in the *not happy* category, which is related to the fact that the majority, 64%, of the whole dataset is estimated as major. However, a strange fact is that *sad* music is not especially minor as one could expect. On the contrary *sad* music seems more major than *not sad* music (Figure 4.15), we do not find any explanation for this but the reader should note that this difference is low. Another noticeable result is what we can observe for the *angry* category: it is definitely more minor than major (Figure 4.17). This confirms the theory that the mode is related with the valence as *angry* is a more a "negative" mood. The *relaxed* category (Figure 4.16), confirms this theory, being more major than the *not relaxed category.*

Onset rate, Chords change rate

From psychological results, Juslin & Laukka (2004) cite rhythm as an important musical feature when expressing different mood types (generally, faster means more arousal). The basic measure/element of rhythm is the onset,

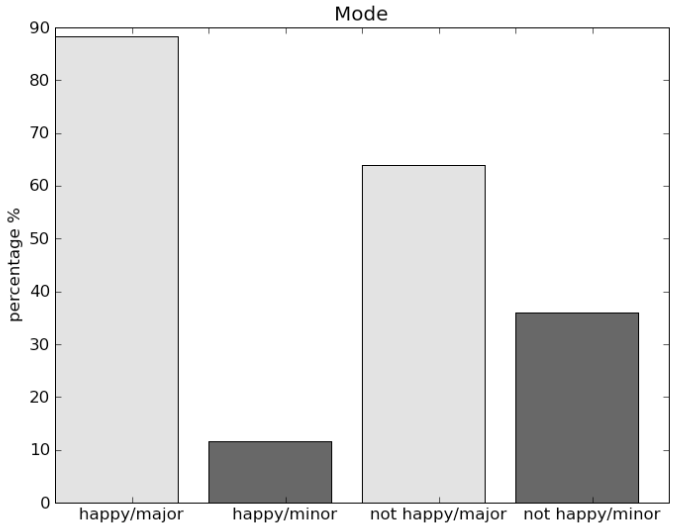

Figure 4.14: Bar plot of the estimated mode proportions (in percentage) for the *happy* and *not happy* categories.

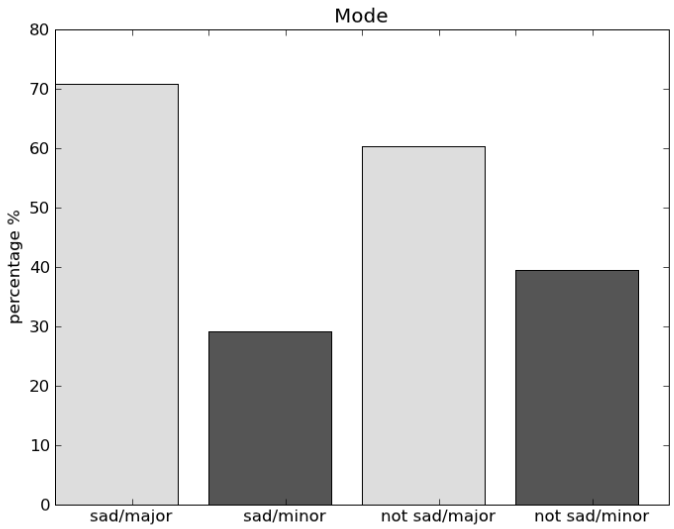

Figure 4.15: Bar plot of the estimated mode proportions (in percentage) for the *sad* and *not sad* categories.

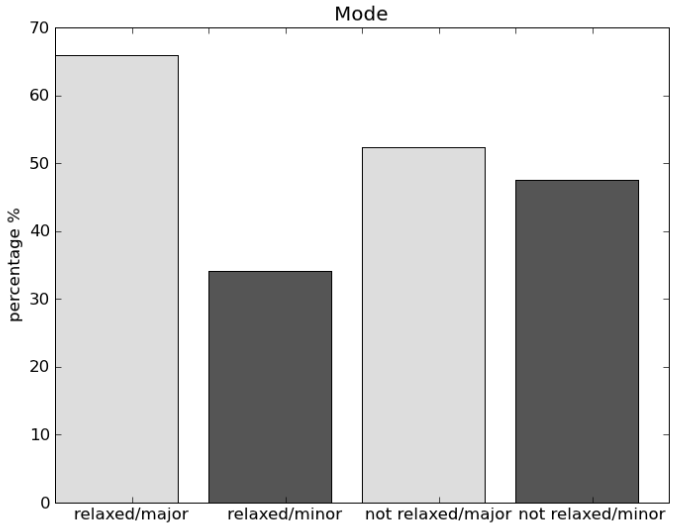

Figure 4.16: Bar plot of the estimated mode proportions (in percentage) for the *relaxed* and *not relaxed* categories.

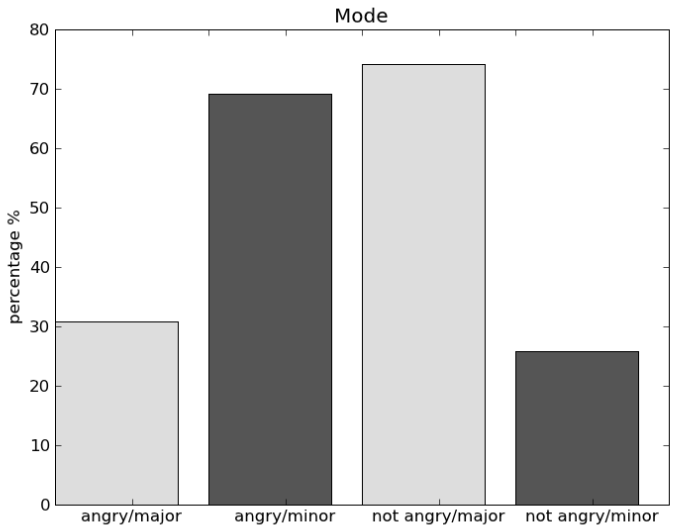

Figure 4.17: Bar plot of the estimated mode proportions (in percentage) for the *angry* and *not angry* categories.

which is defined as "changes" in the audio. Onsets are properties of notes, noises, or any other acoustic event. They are often the beginning of notes or attacks. The onset times are estimated by looking for peaks in the amplitude envelope (see Gouyon (2003)). First, the energy of each non-overlapping frames is calculated. The onset will be detected when the energy of the current frame is superior to a specific percentage (i.e. 200%) of a fixed number (i.e. 8) of the previous frames energy average. It is assumed that there is a minimum gap of 60ms between onsets and a weighting factor is applied to each onset according to the number of consecutive onsets whose energy satisfies the mentioned threshold. The onset rate is simply the number of onsets in one second. This gives us an estimation of the number of events occurring per second, which is related to a perception of the speed.
Another related descriptor is the chords change rate. It is an estimator of the number of chord changes per second. Chords are estimated using the above mentioned $HPCP$ descriptors. In Figure 4.18, we compare the onset rate values for the *happy* and *not happy* categories. It shows that *happy* songs have higher values for the onset rate, which confirms the observation made by Juslin & Laukka (2004) from psychological studies that *happy* music is fast. In Figure 4.18, we look at the chords change rate, which is higher for *angry* than for *not angry*. This is also a confirmation of the studies previously mentioned, associating a higher arousal with faster music.

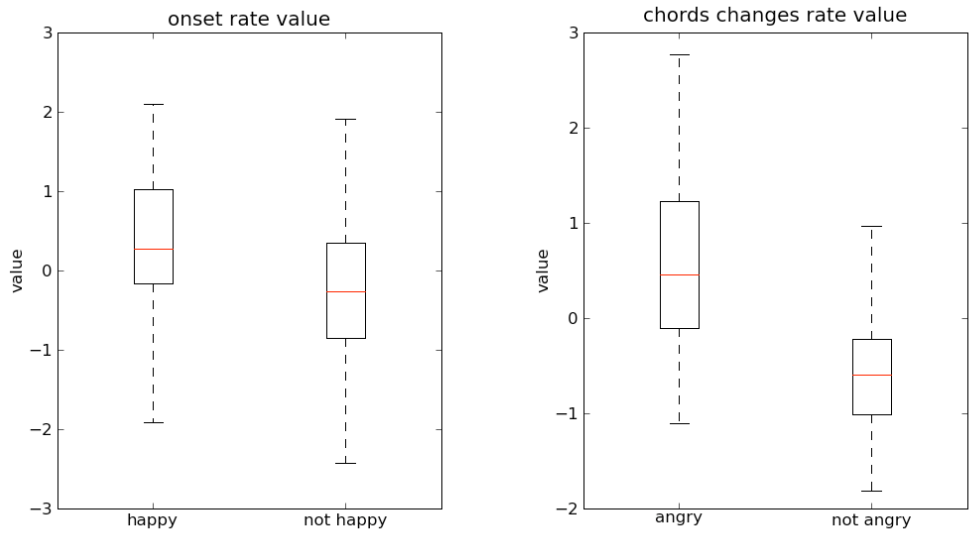

Figure 4.18: Box-and-whisker plot of the standardized onset rate value mean for the *happy* and *not happy* categories. Box-and-whisker plot of the chords change mean for the *angry* and *not angry* categories.

Summary of results

In Table 4.2, 4.3, 4.4 and 4.5, we summarize the results previously mentioned for the categories *angry*, *sad*, *relaxed* and *happy*. In these table, we only mention the descriptors that show a significant difference between the category and its complementary, "$>$" meaning that the descriptor values are positively correlated with the category, and "$<$" positively correlated with its complementary.

	angry / not angry
Zero Crossing Rate	$>$
Spectral Complexity	$>$
Spectral Centroid	$<$
Spectral Flatness dB	$<$
Dissonance	$<$
Mode = Minor	$>$
Onset Rate	$>$

Table 4.2: Summary of the descriptor correlation with a category or its complementary (angry).

	sad / not sad
Spectral Complexity	$<$
Spectral Skewness	$>$
Spectral Roll-Off	$<$
Mode = Major	$>$

Table 4.3: Summary of the descriptor correlation with a category or its complementary (sad).

	relaxed / not relaxed
Zero Crossing Rate	<
Spectral Complexity	<
Spectral Kurtosis	>
Spectral Flatness dB	>
Dissonance	<
Mode = Major	>

Table 4.4: Summary of the descriptor correlation with a category or its complementary (relaxed).

	happy / not happy
Spectral Complexity	>
Mode = Major	>
Onset Rate	>

Table 4.5: Summary of the descriptor correlation with a category or its complementary (happy).

4.4. Classification

Classification is a learning procedure based on the statistical learning theory. We reviewed the basic concept, models and applications in music classification in Section 2.4.3.
We have mentioned in the previous section some of the most relevant features showing their potential to individually discriminate between categories. However, we keep all the descriptors in our "bag-of-features"; those that are not obviously useful could be significant when combined with others in a linear or non-linear way. To capture these relationships, we use feature selection algorithms and build the model, trying several kinds of classification algorithms that we evaluated. A common technique to validate a classification approach is the K-fold cross-validation (see Section 2.4.4). This allows to evaluate the predictive power of the classifier without having to validate the classification on new unknown data.

4.4.1. Classifiers

Once the ground truth created and the features extracted, we performed a series of tests with 8 different classifiers. We evaluated the classifiers using 10 runs of 10-fold cross-validation to account for the chance of splitting the data in easier ways than others. Next, we list the different classifiers we employed. We chose the most typical algorithms, capturing the diversity of criteria to

build classifiers: k-Nearest Neighbor, Decision trees (J48, also known as C4.5), Random Forests, Support Vector Machines, Logistic Regression and Gaussian Mixture Models. Details about these classifiers can be found in the Literature Review, Section 2.4.3.

4.4.2. Experiment 3: Mood classification, comparison of classifiers and audio features

After independent parameter optimization for each classifier, the evaluation was made with 10 runs of 10 fold cross-validation. For comparison purposes, we show the mean accuracies obtained for each mood category and algorithm configuration separately in Table 4.6. Each value in a cell represents the mean value of correctly classified data in the test set of each fold. Considering that each category is binary (for example, *angry* vs. *not angry*), the random classification accuracy is 50%. The SVM algorithm with different kernels and parameters, depending on the category, achieved the best results. The accuracies we obtained using audio-based classifiers are quite satisfying and even exceptional when looking at the *angry* category with 98%. All four categories reached classification accuracies above 80%, and two categories ("angry" and *relaxed*) peaked above 90%. Even though these results might seem surprisingly high, this is coherent with similar studies by Skowronek et al. (2007).

	Angry	Happy	Relaxed	Sad	Mean
SVM linear	95.79%	**84.57%**	90.68%	87.31%	89.58%
SVM poly	**98.17%**	84.48%	**91.43%**	**87.66%**	**90.44%**
SVM RBF	95.19%	84.47%	89.79%	87.52%	89.24%
SVM sigmoid	95.08%	84.52%	88.63%	87.31%	88.89%
J48	95.51%	80.02%	85.25%	85.87%	86.66%
Random Forest	96.31%	82.55%	89.47%	87.26%	88.90%
k-NN	96.38%	80.89%	90.08%	85.48%	88.21%
Logistic Reg	94.46%	73.60%	82.54%	76.38%	81.75%
GMMs	96.99%	79.91%	91.13%	86.54%	88.64%

Table 4.6: Mean classification accuracy with 10 runs of 10-fold cross-validation, for each category against its complementary. In bold is the highest accuracy for each category.

Audio feature contribution

Here, we evaluated the contribution of the audio features described in 4.3. To achieve this goal, we chose the best overall classifier for each category and we made 10 runs of 10-fold cross-validation with only one descriptor type statistic. We show in Table 4.7 the resulting mean accuracies for each configuration

compared to the best accuracy obtained with all the features in the first row. We observe that most of the descriptors give the worst results for the *happy* category. This reflects also the results with all features, with a lower accuracy for *happy*. Moreover, some descriptors like the spectral centroid and the chords change rate do not seem to contribute positively for this category. In general, with the lowest accuracy, this *happy* category seems difficult to model. Nevertheless, we note that the mode helps to discriminate between *happy* and *not happy* (at 64.73%), like also seen in Figure 4.14. It is even more relevant for the *angry* category (at 71.43%). It is also worth noticing that individual descriptors can give relatively high accuracies. For instance, very simple descriptors such as the zero crossing rate give surprisingly high results. We should keep in mind that this is a binary classification tasks, so the random baseline is 50%. A result below 80% is relatively not that high. Finally, we find that spectral complexity, spectral centroid, dissonance and spectral flatness help a lot for categorizing the *angry* and *relaxed* categories. In particular, spectral complexity or dissonance alone can classify *angry* with an accuracy above 90%. Anyhow, the global model combining all the features is the most accurate.

	Angry	Happy	Relaxed	Sad
All features	98.17%	84.57%	91.43%	87.66%
ZCR	84.03%	71.53%	80.73%	77.41%
MFCCs	89.47%	57.59%	83.87%	81.74%
Bark bands	90.98%	59.82%	87.10%	83.48%
Spectral complexity	95.86%	55.80%	88.71%	86.52%
Spectral centroid	89.47%	50%	85.48%	83.04%
Spectral skewness	77.44%	52.23%	73.38%	73.48%
Spectral flatness	85.32%	72.58%	84.58%	74.59%
Spectral roll-off	85.55%	71.84%	79.68%	77.93%
Spectral kurtosis	58.79%	63.53%	64.73%	66.26%
Dissonance	91.73%	62.05%	82.66%	79.57%
Onset rate	52.63%	60.27%	63.31%	72.17%
Chords change rate	74.81%	50%	69.35%	68.26%
Mode	71.43%	64.73%	52.82%	52.08%

Table 4.7: Mean classification accuracy with 10 runs of 10-fold cross-validation, for each category against its complementary with feature sets made of one descriptor statistic.

4.5. Robustness

The concept of robustness refers to those factors that can impact on the accuracy of a classifier. A typical example would be a modification of the quality

(and especially a degradation), compared to the expected quality. In this part we want to evaluate the quality of these models estimating their robustness to audio quality degradation. The objective is to know how well they would behave, even if we had audio files with less quality.

4.5.1. Method

The best models we have found previously are SVM optimized models for each mood category. Since the goal is also to use the models in real-world conditions, they should be able to deal with different audio quality, especially different encodings artifacts. In particular, we want to test the robustness of the mood models to low quality encodings. The original encodings of the training set were mp3 at 128 kbps (kilobits per second). We generated two modified versions of the dataset, lowering the bit rate to 64 kbps and 32kbps. We then measured the accuracy degradation of the classifier trained with the entire dataset and test on the same one with the previously mentioned low-rate encodings. We decided to train and test with full datasets to test the model trained with the maximum data.

4.5.2. Results

Please note that the accuracies are different from the cross-validation results simply because in this case we are not performing cross-validation (this because we train and test with the entire dataset). In Figure 4.19, we represent the accuracy degradation of the classifier trained with the entire dataset and tested on the same one with the previously mentioned low-rate encodings.
We observe degradation due to encoding at a lower bit rate. However, in all cases, this does not seem to have a strong impact. The degradation, in percentage, compared to the original version at 128 kbps is acceptable. For instance, we observe that for the *angry* category, at 32 kbps, only 0.7 percent points of the dataset is no longer correctly classified as before. We also notice that the highest percentage of degradation is 3.6 percent points obtained for the *relaxed* category (with 32 kbps). Even though there is a slight drop in the accuracy, the classification still gives satisfying results and can be used also with audio files with strong encodings.

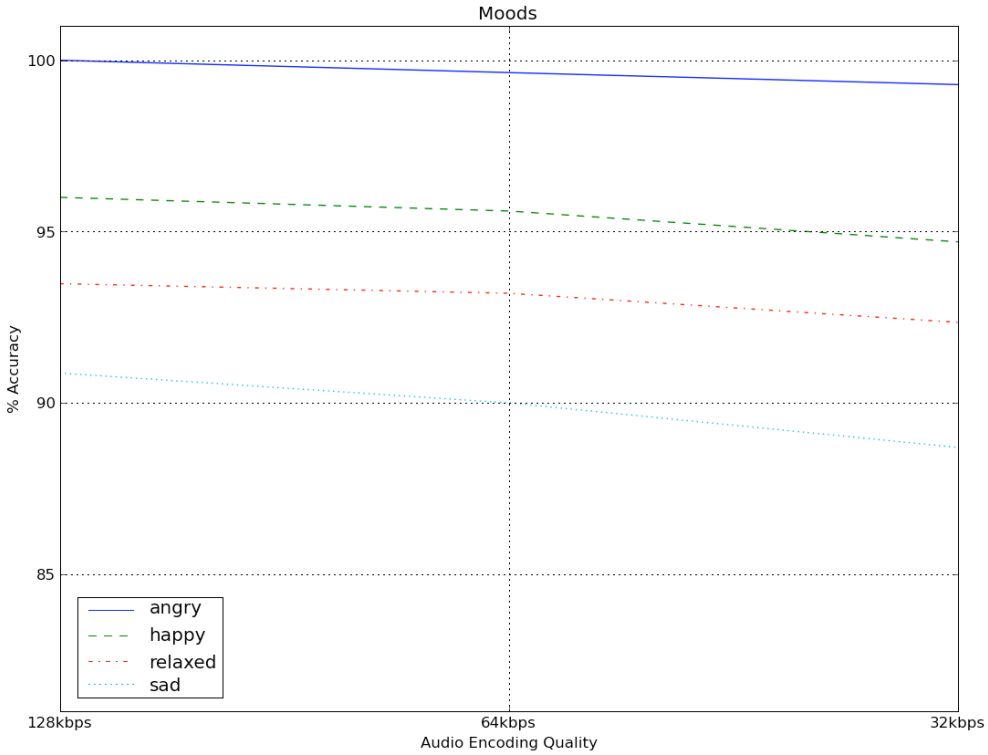

Figure 4.19: Robustness of mood models.

4.6. Evaluations: Audio Mood Classification at MIREX

4.6.1. Description

With the goal of systematically evaluating state-of-the-art algorithms for Music Information Retrieval (MIR) systems, the Annual Music Information Retrieval Evaluation eXchange (MIREX) included an Audio Mood Classification (AMC) task for the first time in 2007. MIREX, as the largest evaluation event in the MIR community, is a good venue to build an available audio dataset and ground-truth for AMC and to facilitate collaborations among MIR researchers around the world. A ground-truth set of 600 tracks distributed across five mood categories was built based on metadata analysis and human assessments (see Hu et al. (2008)). A small dataset, not included in the evaluation ground-truth, has been published in order to illustrate the categories and give examples to the researcher working to build a submission to this task. It also served as a basis for the human assessors of the ground-truth set. The AMC task adopted the set of five mood clusters proposed in Hu & Downie (2007) which effectively reduce the mood space into a manageable set. For clarity purposes, we reproduce the

mood clusters in Table 4.8. The words in each cluster collectively define the "mood spaces" associated with the cluster.

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 4.8: Clusters of mood adjectives used in the MIREX Audio Mood Classification task.

In the rest of this section, we present our different algorithms submitted to the evaluation. We should notice that the models we proposed to be evaluated are not directly the one described before in this thesis. Indeed, the mood taxonomy and the clusters differ from our approach. Consequently, the submitted models have been trained using a different ground-truth that is not available to the public (to avoid overfitting and allow a fair comparison between algorithms). The evaluation was performed using a 3-fold cross-validation. Nevertheless, these evaluations show the high potential of our features and classification methods compared to other state-of-the-art systems. The dataset used to evaluate the algorithms remained the same across the years.

4.6.2. MIREX 2007

Feature Extraction

The system submitted to the first MIREX Audio Music Mood Classification task in 2007 used a set of 133 descriptors and a Support Vector Machine to predict the mood clusters. All the features used in this submission were selected based on results obtained empirically with the exemplar set provided and our databases. Using several feature selection methods in WEKA (PrincipalComponent, InfoGainAttributeEval, CfsSubsetEval, SVMAttributeEval) we sorted out 133 features of different kinds:

- Spectral: spectral centroid, crest, flux, rolloff, skewness, strong peak, high frequency content, MFCC, Bark bands, energy band ratio, flatnessDB
- Loudness: RMS, loudness from Bark bands, dynamic complexity
- High level: tonality, mode, key strength
- Temporal: zero crossing rate, onset rate, beats per minute

Most of these features are extracted using windowing. Afterward we compute statistics of these values (min, max, mean, variance, derivative variance, second-derivative variance). The decision to keep or not each value is made using feature selection methods in WEKA as previously mentioned.

Classification

Once the features extracted and normalized, we trained a Support Vector Machine model. We used the libsvm library by Chang & Lin (2001). According to preliminary tests, the best results were achieved by the C-SVC method with the RBF kernel (Radial Basis Function). Consequently we used this configuration in our algorithm. Then to decide which values to choose for the cost C and the γ of the kernel function, we implemented a grid search algorithm like one suggested in Hsu et al. (2003). We kept the parameters that obtained the best accuracy using a 10-fold Cross Validation on the training set. Finally when the optimal parameters were found, we trained a SVM model and used it to predict the mood categories.

Results and Discussion

Overall classification: in Table 4.9 , we show the results of the different submissions in terms of mean accuracy over the 3-fold cross validation performed by the MIREX team.

Participant	Accuracy
George Tzanetakis	**61.50%**
Cyril Laurier, Perfecto Herrera	**60.50%**
Lidy, Rauber, Pertusa, Iñesta	59.67%
Michael Mandel, Dan Ellis	57.83%
IMIRSEL M2K svm	55.83%
Michael Mandel, Dan Ellis	55.83%
Kyogu Lee 1	49.83%
IMIRSEL M2K knn	47.17%
Kyogu Lee 2	25.67%

Table 4.9: Classification average accuracies over the three train/test folds.

Confusion Matrix: to better understand the strong and weak points of the algorithm, Table 4.10 shows the confusion matrix.
We notice that the best predictable categories are cluster 3 and 5, which correspond roughly to sad and aggressive. The other clusters were more difficult to predict as one can expect by listening to the examples provided. Consequently

all the algorithms perform better with this two clusters. The category with the worse accuracy is cluster 1 often predicted as cluster 5. This makes sense as there are some acoustic similarities. Both are energetic, loud and many of both use electric guitar. Looking at the other submissions the same confusion appears. Moreover there is a clear confusion between cluster 2 and 4. Looking at the mood adjectives of these clusters, we can notice a possible semantic overlap. For example, using Wordnet[5] , we find that fun (from cluster 2) and humorous (from cluster 4) share the synonym : amusing. Besides humorous is a synonym of funny. We can observe this confusion also in the other algorithms results.

Truth/Predicted	1	2	3	4	5
Cluster 1	**45.8%**	11.7%	5.0%	17.5%	20.0%
Cluster 2	10.8%	**50.0%**	11.7%	27.5%	0.0%
Cluster 3	1.7%	11.7%	**82.5%**	4.1%	0.0%
Cluster 4	10.0%	31.7%	4.2%	**53.3%**	0.8%
Cluster 5	18.3%	1.7%	2.5%	6.7%	**70.8%**

Table 4.10: Confusion matrix with mean values over the 3 cross-validation folds for our algorithm.

To sum up we can argue that there are three main shortcomings in the proposed clusters :

- Cluster 3 and 5 are the most predictable
- There is a problem to predict Cluster 1 because it is close to Cluster 5 (acoustic similarities)
- There is a confusion between Cluster 2 and 4 (semantic similarities)

This first participation at MIREX confirmed we were ranked among the best classification approaches with most probably state-of-the-art audio descriptors and a good classification method. At least, with the above mentioned limitation of the dataset, we can consider our algorithm as state-of-the-art. We can criticize the limitations of the MIREX approach openly because we actively participated in its creation (for this Mood task). We believe that, to cope with the confusion between clusters, the organizers should revise the taxonomy and build another dataset. However, even if not perfect, the MIREX Audio Mood Classification task is the best available tool for comparing mood classification approaches.

[5]http://wordnet.princeton.edu/

4.6.3. MIREX 2009

We did not submit any algorithm to the MIREX 2008 evaluation, where the same conclusion than for MIREX 2007 remain valid. The best approach reached an accuracy of 63,67% with an original approach modeling the classes in two steps: a frame-statistical model followed by a track-statistical model (Peeters (2008)). For MIREX 2009, we had more features and different approaches to try and compare.

Feature Extraction

The algorithms submitted to this task in 2009 were coded in C++ and python. For the feature extraction part, we used an internal library of the Music Technology Group already mentioned before called Essentia (Wack (2010)). This library contains many descriptors mentioned previously in this thesis and summarized below. All frame-based statistics were aggregated using: mean and derivatives until second order, variance and derivatives until second order, minimum and maximum.
In Table 4.11, we list the set of features that performed the best in our preliminary experiments made on our genre, artist and mood databases. Many of them were already present in our set submitted in 2007 but had been revised and pre-selected.

Type	Features
Low level	barkbands spread, skewness, kurtosis, dissonance, hfc pitch and confidence, pitch salience, spectral complexity spectral crest, spectral decrease, energy, spectral flux spec spread/skewness/kurtosis, spec rolloff, strong peak ZCR, barkbands, mfcc
Rhythm	bpm, beats loudness, onset rate
Sound FX	inharmonicity, odd2even, pitch centroid, tristimulus
Tonal	chords strength (frame), key strength(global), tuning freq

Table 4.11: Feature set for all our classifiers.

Classification

The three classification algorithms are also coded in C++ and python. They are implemented using Gaia, a library for manipulating dataset and computing similarity distances (Wack (2010)). Each algorithm has the option to look for its best parameters with a grid-search cross-validation approach on the training data. We submitted different algorithms using: Support Vector Machine, Soft

Independent Modeling of Class Analogies, and Relevant Component Analysis (described bellow).

Soft Independent Modeling of Class Analogies (SIMCA): the Machine Learning (ML) and Music Information Retrieval (MIR) communities have developed a pool of applications: Weka (Witten & Frank (1999)), Marsyas (Tzanetakis (2007)), jMIR (Mckay (2010)), MIRtoolbox (Lartillot & Toiviainen), R[6], etc...). Focusing on the supervised approach, most of the techniques implemented in these packages try to find a unique classifier that predicts all the categories proposed in a taxonomy, or a pool of classifiers that individually propose an output to the whole problem and apply a policy to gather all the obtained results by the different techniques (voting schema, grading, etc.). In some cases, the presented problem requires independent analysis for their categories (p.e. different families of descriptors or different temporal scope). The use of binary classifiers are a first approach to solve these problems. They set m specific classifiers, being m the number of categories, in a 1-against-all architecture, or $C_m^2 = \frac{m!}{(m-2)!2!}$ classifiers in a 1-against-1 architecture. By using these configurations, each classifier is independently trained providing a specific classifier for each category.
In the case of music, the extracted descriptors can be grouped into different families, related to the musical facet they represent. In a specific problem (p.e. genre or mood classification), categories may be described by different families of descriptors. The 1-against-all architecture seems adequate to gather these particularities, but we need to include an automatic feature selection algorithm at the input stage of each classifier that properly selects the audio descriptor that best represents that specific category. Principal Component Analysis creates a linear combination of the existing descriptors (eigenvectors) ordered in descending order according to the covered percentage of variance for each combination (eigenvalues). By selecting only the most representative combinations, we ensure that only the representative audio descriptors will be used, and the other ones will be discarded. Our proposed methodology consists in a set of m 1-against-all classifiers, being m the number of categories, with a PCA analysis at the input stage of each individual classifier.
This approach for classification is known as the SIMCA method (Soft Independent Modeling of Class Analogies proposed by Wold (1976). More information can be found in Vanden & Hubert (2005) and the first application to MIR problems in Guaus (2009).

Relevant Component Analysis and Nearest Neighbours: Relevant Component Analysis (RCA) is a supervised transformation which aims at maximizing the global variance of a dataset while reducing the intra-class variance (representing unwanted variability). The algorithm is split in two parts: the

[6]http://www.r-project.org/

first part is the dimensionality reduction that consists in applying a modified version of the Fisher Linear Discriminant (FLD) where we only use part of the classified vectors for training. This transformation amounts to resolving the following estimator:

$$\max_{A \in M_{P \times Q}} \frac{A^t S_t A}{A^t S_w A} \tag{4.13}$$

transforming from a space with P dimensions to a space with Q dimensions where A is the searched transformation matrix, $M_{P \times Q}$ is the space of all transformations, S_t is the total covariance matrix and S_w is the inner-class covariance matrix.
The second part consists in applying the actual RCA transformation, which scales down those dimensions that have great variability within our classes by whitening the resulting feature space. We first calculate the covariance for all the centered data-points:

$$\hat{C} = \frac{1}{p} \sum_{j=1}^{k} \sum_{i=1}^{n_j} (x_{ji} - \overline{x}_j)(x_{ji} - \overline{x}_j)^t \tag{4.14}$$

where p is the total number of points in the chunklets and $\overline{x}_j$ is the mean of the data-points of the chunklet j. Finally we obtain the whitening matrix:

$$W = \hat{C}^{-\frac{1}{2}} \tag{4.15}$$

so the new feature space is given by:

$$x_{new} = Wx \tag{4.16}$$

Our classification algorithm is made of a K-nn classifier using a weighted distance based on two distances. One is from the reduced space mentioned previously where we use the euclidean distance. The other is the Kullback-Leibler distance applied to MFCCs.

$$Dist(a, b) = \alpha(KL_{MFCC}(a, b)) + (1 - \alpha)(Euclidean_{RCA}(a, b)) \tag{4.17}$$

We optimize the weight α between both distances with a cross-validation technique on the training set.

Results and Discussion

In Table 4.12, we present the accuracy of our different submissions. There were 33 submissions from 16 different research groups (including our 5 submissions). In Table 4.13, we show the comparison of the accuracies of the best submission of each group (in decreasing order).
The best results is obtained with our submission with the SVM classifier, without any grid search to optimize the parameters. It seems that the grid search is

ID	Algorithm	Grid search	Accuracy
1	RCA	No	57.67%
2	RCA	Yes	57.5%
3	SIMCA	No	59.83%
4	SIMCA	Yes	59.33%
5	**SVM**	**No**	**62.83%**
5	SVM	Yes	59.5%

Table 4.12: MIREX2009: Accuracies of our submissions.

Submission	Accuracy
CL1	65.67%
GP	63.67%
MTG5	**62.83%**
HW2	61.67%
LZG	61.67%
GLR1	60.83%
FCY1	60.33%
VA2	60.17%
XZZ	60.00%
BP2	59.67%
GT1	59.33%
SS	58.83%
HNOS1	58.67%
XLZZG	57.00%
TAOS	56.83%
RK1	53.17%
ANO	50.67%

Table 4.13: MIREX 2009: Comparison with other submissions.

not efficient in this case because it does not perform better either with the other classifiers. This might be due to a problem when implementing it, but we were not able to identify the reason at the moment of submitting the algorithms. Another possibility is that the grid search could have overfit the parameter space, it could be advisable to select some near but not optimal value to be protected over that. In any case, we note that the best result is using SVM, same conclusion than with our database presented in this thesis. Compared to our first submission in 2007, we increased the accuracy to 62.83% (2.3 percent points), which is ranked in the top results with no statistical significance with the other top-ranked results.

4.7. Conclusion

In this chapter, we have presented our approach to build a pure audio content-based mood classifier. We employed an original approach to build the ground truth, taking advantage of the wisdom of crowds as well as experts. This method is recommended for other problems because it eases a lot the annotation phase to create a dataset of examples: we involved directly less people (which is often the costly and limiting part of the method), we coped with the cultural and social bias in the sample (keeping in mind the bias of the social network itself), and finally, we combined the validation from many people and the accuracy of experts. An important conclusion from the dataset creation method is that tags should not be trusted as ground truth. Indeed the manual validation is an necessary step to verify that the tags are not wrong and above all that we are understanding and using them in the correct way for our purpose. In our case, this process allowed to avoid 29% of error. We have demonstrated the importance of some of the audio descriptors, comparing their value distributions across mood categories, and their influence in the classification task. We conclude from these experiments that surprisingly, single descriptors can already classify quite well some categories. The most typical example being the spectral complexity classifying at 95.86% the *angry* category. A similar result is found with the dissonance supporting the idea that arousal is related somehow to how dissonant or complex the spectrum is. Also, results about the mode and its relation to valence are also worth noticing even if the *sad* category is not following that theory. Anyhow, we also proved that the sum of all these descriptors give the best results, which are quite high with accuracies above 84% for all and reaching 98.17% for *angry* music. To balance this conclusion, the *happy* category (84.57% of accuracy) is the most difficult to classify, showing us that happiness, even in music, is difficult to predict or notice.

Also, with the goal of proving the stability of the classification models, we tested the robustness of the classifiers (and consequently the audio descriptors) towards audio quality degradation. This has been done applying strong encoding and on the audio signals. A very small drop in accuracy was observed, meaning that our mood classifiers are robust to these transformations.

Finally, we reported on external evaluations and comparisons with other approaches at MIREX, showing that our method is state-of-the-art and ranked among the best results. Nevertheless, we believed that this approach can be improved using additional information from the audio content (with high-level features such as genre) and also from other sources of information (such as lyrics). This is what we will study in the following chapters.

CHAPTER 5

Mood Classification with Lyrics

"The important thing about lyrics is not exactly what they say, but that they lead you to believe they are saying something." Brian Eno (1981).

5.1. Introduction

The mood of a song is expressed by means of musical features of different type such as rhythm, tonality or timbre such as detailed in Chapter 4. Even if instrumental music convey strong emotions, we should not discard the power of words in songs. Indeed, a relevant part of the sentiments also seems to be conveyed by the lyrics. In this chapter, we study how adding lyrics analysis to our system helps to categorize music by mood. Of course, in this context, we discard instrumental music and concentrate on songs. With this idea of studying the importance of lyrics, we derive a bit from our original goal which is to use only audio signals as a source of information. However, this hybrid approach, using both audio and text data, evaluates the potential of a possible future system that would extract the lyrics directly from the audio signal. Maybe it would not be that accurate, but at least knowing how lyrics would help in the best scenario is a very valuable information. Moreover, it can also be seen as a real use case for music classification, as lyrics for known songs can be found automatically online. In the following experiments, we demonstrate that, as one could expected, lyrics help to classify by mood, adding useful information to models based only on audio signals.

5.2. Experiment 4: Mood Classification using Audio and Lyrics

5.2.1. Summary

In this experiment, we study music mood classification using audio and lyrics information. We evaluate each factor independently and explore the possibility to combine both, using Natural Language Processing and Music Information Retrieval techniques. We show that using standard method such as Latent Semantic Analysis (LSA) (see Deerwester et al. (1990) and Chapter 3 for more details) is able to classify the lyrics significantly better than random, but the performance is still quite inferior to that of audio-based techniques. We then introduce a method based on differences between language models that gives performances closer to audio-based classifiers. Moreover, integrating this in a multimodal system (audio+text) allows an improvement in the overall performance. We demonstrate that lyrics and audio information are complementary, and can be combined to improve a classification system.

5.2.2. Related Work

Although there is existing work dealing with mood detection in text (Alm et al. (2005); Cho & Lee (2006)), until recently, only very little has been done to address the automatic classification of lyrics according to their mood. Fewer works addressed the problem of combining both audio and lyrics information. Prior to our contribution, an early work from Yang & Lee (2004) combined audio and lyrics bag-of-features to disambiguate emotion categories when classifying, but with a rather small dataset of 145 songs. Another early work from Mahedero et al. (2005) reported promising results in using lyrics for thematic categorization suggesting that a mood classification from lyrics is feasible. Neumayer & Rauber (2007) have shown the complementarity of audio and lyrics in the context of genre classification. Later, Mayer et al. (2008a,b) introduced several new lyrics features such as rhyme, style, word per minute, to be used for the same purpose of genre classification. Logan et al. Logan et al. (2004) have investigated the properties of lyrics using Latent Semantic Analysis. They discovered natural genre clusters and their conclusion was also that lyrics are useful for artist similarity searches but the results were still inferior to those achieved using acoustic similarity techniques. However, they also suggested that both systems could profitably be combined as the errors of each one were different.

More recently, van Zaanen & Kanters (2010) experimented td.idf (in a comparable way as we do in the first experiment of this Chapter) with promising results. Another work on lyrics only, by He et al. (2008), compared bag-of-words features and found that the combination of word unigrams, bigrams and trigrams together with tf.idf weighting gave the best results (see 5.2.5 for

a description of tf.idf). The particularity of this approach was also the use of only two categories: "lorn" and "lovelorn" which means respectively of "lonely and abandoned" and "unhappy because of unrequited love" . Nonetheless, Hu et al. (2009b) showed a method to classify songs in Russell's bidimensional plane (Russell (1980)) using an affective lexicon and a clustering method.
Finally, from another field, studies in cognitive neuropsychology from Peretz et al. (2004) demonstrated the independence of both sources of information and so the potential complementarity of both melody and lyrics in the case of emotional expression can be foreseen.

5.2.3. Database

For this study we use our categorical approach to represent the mood, as justified in Chapter 3, with the following categories: *happy*, *sad*, *angry*, *relaxed*. Our collection is based on the one described in Chapter 4. We have pre-selected the tracks using last.fm[1] tags, generated a synonym set using Wordnet[2] and looked for the songs mostly tagged with these terms. Then we asked 17 listeners to validate this selection by mood. We considered a song to be valid if the tag was confirmed by at least one listener, as the pre-selection from last.fm granted that the song was likely to deserve that tag. We included this manual tag confirmation in order to exclude songs that could have gotten the tag by error, to express something else, or by a "following the majority" type of effect. The annotators listened to 30 seconds excerpts, first to avoid as much as possible changes in the mood, and then to speed up the annotation process. Therefore they could not listen to the whole lyrics and some excerpts may not contain lyrics all length, all the time, thus their judgment had to be biased toward an analysis of the audio. This might influence negatively the results if the mood of the lyrics is not coherent with the mood expressed by the music. In many cases both would match, in other cases it would introduce some error in the system that we cannot detect. All lyrics are in English and extracted from LyricWiki[3]. The database is composed of 1000 songs divided between 4 categories of interest plus their complementary categories (i.e “not happy”, “not sad”, “not angry” and “not relaxed”). We have used an equal distribution of these binary classes.

5.2.4. Audio Classification

To classify music by mood, we used the supervised learning approach described previously (see Chapter 4 for more details).
In order to classify the music from audio data, we first extracted audio features of different kinds: timbral (for instance MFCC, spectral centroid), rhythmic

[1] http://www.last.fm
[2] http://wordnet.princeton.edu
[3] http://lyricwiki.org

(for example tempo, onset rate), tonal (like Harmonic Pitch Class Profiles) and temporal. All these features are standard and derived from state-of-the-art research in Music Information Retrieval and described in Chapter 4. For each song, their 200 frame-basis extracted features were summarized with their average and variance.
As mentioned previously, we use Support Vector Machines (SVM) to classify music by mood. SVM are known to be efficient in many classification tasks and we have shown in Chaper 4 that they were the best classifier for this purpose. We obtained the results shown in Table 5.1 using Weka (Witten & Frank (1999)) and 10 runs of 10-fold cross-validation. We report here the accuracies obtained using Support Vector Machines, the best kernel between linear, polynomial and RBF, and the optimized parameters using a grid search. We also tried other classifiers shown here for comparison with optimized parameters, but Support Vection Machines performed better than others.

	Angry	Happy	Relaxed	Sad
SVM	**98.17%**	**84.57%**	**91.43%**	**87.66%**
J48	95.51%	80.02%	85.25%	85.87%
Random Forest	96.31%	82.55%	89.47%	87.26%
k-NN	96.38%	80.89%	90.08%	85.48%
Logistic Reg	94.46%	73.60%	82.54%	76.38%
GMMs	96.99%	79.91%	91.13%	86.54%

Table 5.1: Classification accuracy using audio features, for each category against its complementary

The performances we obtained using audio-based classifiers are quite satisfying and even exceptional when looking at the "angry" category with more than 98%. All four categories reached classification accuracies above 80%, and two categories ("angry" and "relaxed") even above 90%. As we deal with binary comparisions on a balanced dataset, the random baseline is 50%. Even though these results can seem surprisingly high, this is coherent with other similar studies (Skowronek et al. (2007)) and the fact that we are using relatively "simple" categories, with which a consensus is either to obtain. Also our approach considering the emotion intentions (evoked, not perceived) makes the problem much more objective.

5.2.5. Lyrics classification

In addition to the results from the audio analysis, lyrics can provide valuable information about the mood of a song. In this section we report three experiments. In the first one we used similarity between lyrics, in the second feature vectors based on Latent Semantic Analysis dimensional reduction, and

in the third one we propose a technique to select the most discriminative terms looking at the differences between language models.
The first two approaches treat the text in an unsupervised way, where the representation in vector space is independent of the categories we are interested in. In the third approach, we use our categories (in a supervised process) to select an appropriate representation of the lyrics before addressing the classification task.

Experiment 4.1: Classification based on similarity using Lucene

Our first approach was based on the assumption that songs that are "similar",in a general sense, are most likely similar for specific relevant aspects, such as genre, mood, etc.
We defined the similarity between different songs in a way commonly used in document retrieval tasks. The representation of the songs is reduced to a bag of words, i.e. the set of words or terms used in a song as well as their frequency. This is then used, with the help of the Lucene document retrieval system[4], to rank documents by their similarity. The similarity measure used by Lucene essentially corresponds (with some performance tweaks) to the very common vector model of information retrieval (see Salton (1971)), with tf.idf weighting in order to attribute more importance to those terms that are frequent in the given song, but less frequent overall in the collection. tf stands for "term frequency" and idf for "inverse document frequency". idf is defined as:

$$\text{idf}(t) = \log \frac{|D|}{|\{d : t \in d\}|} \tag{5.1}$$

where $|D|$ is the total number of documents and $|\{d : t \in d\}|$ the number of documents where the term t appears. Note that this is defined only for a corpus where t appears. Then, the tf.idf weight of a term t in a document d is given by:

$$\text{tf-idf}(t, d) = \text{tf}(t, d) \times \text{idf}(t) \tag{5.2}$$

We decided against using some of other techniques frequently used in document retrieval such as stemming, given that our focus is quite different from typical retrieval tasks. Word stemming (reducing a word to its root) would fail or produce unpredictable results in many cases as song lyrics often contain colloquial versions of words (with spellings that do not fit well with the rules used). In addition, we are not interested in uniformization of the language, but we want to exploit the information provided by linguistic particularities. Document retrieval systems focus more on extracting the words that are relevant for the general thematic of a document and therefore they tend to neglect aspects relating to the personal communication aspect.

[4] http://lucene.apache.org/

The most classic approach for using similarity to classify is the k-NN classifier (described in 2.4.3). Based on a source item (in our case a song) for which the class is unknown, the k most similar items from the annotated collection are retrieved. Each of these provides a class label, and the majority label (the most represented one) is chosen as the predicted class of the source item.
It is important to note that this approach is very sensitive to unbalanced collections, i.e. overrepresentation of one class over the others. The increased a-priori probability that an item belongs to that class greatly increases the probability that many of the k retrieved items belong to that class, and may therefore excessively reduce the chance of predicting one of the minority classes. All experiments reported here were conducted with balanced datasets to avoid such bias.

Results: we conducted experiments with varying numbers of similar documents (k) to be taken into account. In general, a low k provides less stability, as the predicted label depends strongly on individual examples from the collection. Large ks on the other hand can mean that examples are taken into account that are not actually very similar (and thus representative) of the one that is to be classified. The optimal k value depends on the application and the distribution of the datapoints and can not be easily predicted a-priori.
While better than the random baseline for most of the moods (the baseline is 50%), the prediction power of the similarity-based approach for lyrics remains limited, with averaged accuracy around 60% as shown in Table 5.2. The most predictable category is "angry" and the least predictable is "sad".

	k=3	k=5	k=7	k=9	k=11
Angry	69.5%	67.5%	69.0%	68.5%	67.0%
Happy	55.9%	57.4%	60.9%	64.5%	64.1%
Sad	55.0%	52.8%	58.9%	54.5%	55.0%
Relaxed	61.8%	65.8%	61.0%	59.8%	59.1%
Mean	60.5%	60.9%	62.5%	61.8%	61.3%

Table 5.2: Classification accuracies using k-NN with a tf.idf-based distance on lyrics for different values of k

Limitations: it is difficult to directly integrate the results from both approaches as similarities for audio and lyrics are calculated in different ways. While on the audio side, the feature vectors can be used with different classification algorithms, this is not as easily the case for the lyrics. The typical sparse vector-of-terms representation of the lyrics generates a very high dimensionality, as the length of the vector is the full size of the vocabulary used in

the entire collection. On our relatively small annotated collection the vocabulary size already reached over 7000 words, while more complete collections (e.g. the full LyricWiki) reach vocabulary sizes of several hundred thousand distinct words.

Experiment 4.2: Classification using Latent Semantic Analysis (LSA)

One approach to deal with the dimensionality problem is to project the lyrics into a lower-dimensional space that is manageable by generic classifiers. The most common method for this is Latent Semantic Analysis (LSA) (see 3.2.2), which, similar to approaches like Principal Component Analysis (PCA), projects the data into a space of a given dimensionality, while maintaining a good approximation of the distances between data points.
In combination with tf.idf weighting, LSA allows us to obtain a low-dimensional representation of the data. The resulting dimensions tend to relate to clusters of similar documents, and the most significant terms contributing to those dimensions typically reflect the common vocabulary of groups of semantically related documents.
We calculated the LSA projection using the full LyricWiki collection (approximately 400,000 songs at that time) as that should provide a more accurate model. We note, however, that significantly smaller subsets yielded very similar results in our experiments.
We conducted experiments to determine the impact of the number of dimensions used in the Latent Semantic Analysis on classification performances. As it could be expected, performance (using lyrics alone) is very low for extremely low dimensionality and tends to improve with a greater number of dimensions. The peak performance (which remains quite moderate) is obtained at different numbers of dimensions for the different categories.

Results: in Table 5.3 we show the results from this experiment. The accuracies presented here are averaged over the 10 runs of 10-fold cross-validation. The use of LSA does not dramatically improve performance compared to our first experiment, depending on the category it can even be worse. The reduction in dimensionality does, however, provide more flexibility, as different types of classifiers can be used on the resulting representation. The results shown here use a reduction to 30 dimensions.

Experiment 4.3: Classification using Language Model Differences (LMD)

While distances between songs based on lyrics cannot separate our mood categories very well, lyrics convey other types of information to be exploited in

	SVM	Logistic	RandForest
Angry	**62.1% (9.1)**	62.0% (10.2)	61.3 (11.5)
Happy	**55.2% (10.3)**	54.1% (12.5)	54.8 (10.7)
Sad	**66.4% (9.7)**	65.3% (11.0)	56.7 (12.1)
Relaxed	**57.5% (8.2)**	57.3% (9.1)	56.8 (9.79)
Mean	**61.3% (9.3)**	59.7% (10.7)	57.4% (11.0)

Table 5.3: Classification accuracies using LSA (30 dimensions) on lyrics (with standard deviation)

pursuing their separation according to mood. In order to assess that potential, we analyzed the language models corresponding to the different categories (Ponte & Croft (1998)). Figure 5.1 shows document frequencies (i.e. the proportion of documents containing a given term) for the 200 most frequent terms in the "angry" category, compared to the frequencies in the "not angry" class (results are similar for the other mood categories). However, there are very important differences for quite a number of other terms.

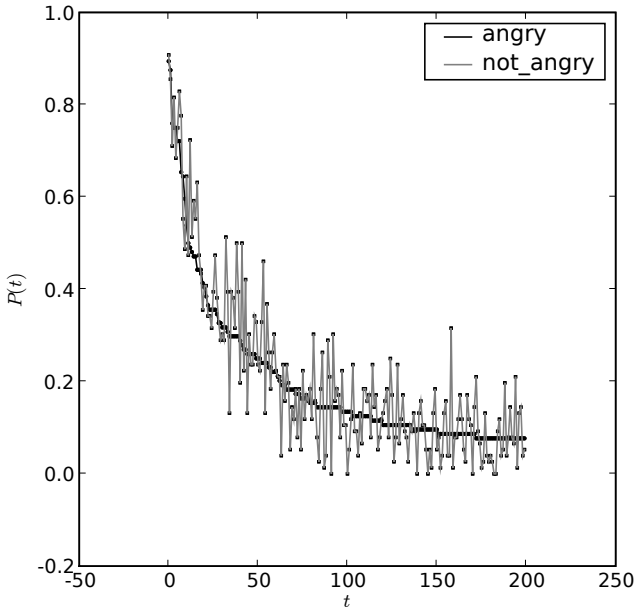

Figure 5.1: Document frequencies ($P(t)$) of terms in "angry" and "not angry" category where t is the term id.

Due to the very high dimensionality of the language models, some classifiers or feature selection techniques can have difficulties in exploiting this information.

We therefore decided to extract a reduced list of relevant terms externally, while using the Weka framework to perform the classification. This is done by comparing the language models generated by the different categories and choosing the most discriminative terms from this comparison.
When comparing two language models (in our case one from the category and one from its complementary), the simplest approach is to calculate the difference in document frequency for all terms. This can be computed either as an absolute difference, or as a relative change in frequency. Both of these, however, have important drawbacks. The absolute difference favors high-frequency terms, even when the relative difference in frequency is not very big. The relative difference on the other hand tends to favor low-frequency terms, especially those that do not occur at all in one of the language models (which results in a difference of 100%).
Example of terms ranked by absolute difference (extracted from the lyrics):

- *angry:* world, die, death, control, ...
- *not angry:* me, love, i'm, can, could, so, but, ...

Example of terms ranked by relative difference (extracted from the lyrics):

- *angry:* realms, dissolution, bear, four, thirst, perverted, evermore, ...
- *not angry:* chillin, nursery, hanging, scheming, attentive, lace, buddy, sweetest, endings, ...

We are interested in terms with a large relative difference (document frequency in one class being multiple times that in the other class), but that are quite frequent in order to cover a large amount of songs. Therefore, we need to find a measure that provides a good mixture of absolute and relative difference. This also has the effect of providing stable results for the selected top-ranked terms, as their frequency is sufficiently high to reduce to effect of chance variations in occurrence counts.
The measure (5.5) we settled on is a compromise between absolute difference (5.3) and relative difference (5.4).

$$\Delta_{abs}(t) = abs(P(t|LM_1) - P(t|LM_2)) \tag{5.3}$$

$$\Delta_{rel}(t) = \frac{abs(P(t|LM_1) - P(t|LM_2))}{max(P(t|LM_1), P(t|LM_2))} \tag{5.4}$$

$$\Delta_{mixed}(t) = \frac{abs(P(t|LM_1) - P(t|LM_2))}{\sqrt{(max(P(t|LM_1), P(t|LM_2)))}} \tag{5.5}$$

where $P(t|LM_i)$ is the probability of term t occurring in a document represented by the language model LM_i, which is estimated as the document frequency of the term in the corresponding category (normalized by the number of documents).

Using this measure Δ_{mixed} gives us a nice list of terms that cover a good percentage of the songs, with very different distribution between the two categories, and that clearly make sense semantically:

- *angry:* death, control, die, dead, god, evil, hell, world, pain, fate, ...
- *not angry:* love, could, heart, can, i'm, were, blue, today, then, need, ...

Results: for each category, we selected the n terms with the highest Δ_{mixed}. We obtained a vector representation with n dimensions that can be used with different classifiers. We made 10 runs of 10-fold cross-validation (this includes the creation of the model and the term selection, to avoid overfitting) and tried different values n. Depending on the categories the accuracy dropped under a certain value of n. For $n = 100$, we had relatively good results with no significant increase by changing its value for any of the categories. In Appendix B, we list the 100 terms for each mood category. Classification performance is significantly better than with the distance based approaches, with accuracies in the 80% range using SVM as shown in Table 5.4. These results are also closer to those obtained using audio based descriptors. We ran the tests with several other classifiers (decision trees, kNN, logistic regression, random forest ...), some of which obtained good results also, but SVMs performed best overall. We therefore used the SVM classifier with this kind of data for our further experiments.

	SVM	Logistic	RandForest
Angry	**77.9%(10.3)**	60.6%(12.0)	71%(11.5)
Happy	**80.8%(12.1)**	67.5%(13.3)	70.8%(11.4)
Sad	**84.4%(11.2)**	83.9%(7.0)	75.1%(12.9)
Relaxed	**79.7%(9.5)**	71.3%(10.5)	78.0 (9.5)
Mean	**80.7%(10.8)**	70.8%(10.7)	73.7%(11.3)

Table 5.4: Classification performances using the 100 most discriminant terms (see Appendix B for the complete list), in parenthesis is the standard deviation

5.2.6. Combining Audio and Lyrics information

Both audio and lyrics can help in estimating the mood of a song. As these two modalities are quite different and potentially complementary, we chose to combine them in order to create a hybrid classification system. We used two approaches to integrate these two information sources. The first one used separate predictions for audio and lyrics and combined them through voting (using the mean of the two classifiers probabilities to take a decision). The second approach was to combine all features in the same space, having a vector

composed of both audio and lyrics features. This allowed to use audio and lyrics information within one classifier.
As Table 5.5 shows, the combination of the language model differences with the audio descriptors yielded relatively good results. For each category we show the accuracy of the SVM classifier for the audio analysis, for the lyrics analysis, and for the multimodal approach combining both. *Mixed* stands for the technique of having a vector with both audio and lyrics features mixed and *Voting* is for taking the classification decision based on the mean probability of the lyrics and audio classifiers. As in the previous experiments, the accuracies shown in Table 5.5 are averages over the 10 runs of 10-fold cross-validation.

	Audio	Lyrics	Mixed	Voting
Angry	98.2%(3.8)	77.9%(10.3)	**98.3%(3.7)**	95.0% (4.3)
Happy	84.6%(11.5)	80.8%(11.2)	**86.8%(10.6)***	86.5% (10.8)
Sad	87.7%(11.0)	84.4%(11.2)	92.8%(8.7)*	**95.6% (8.2)***
Relaxed	91.4%(7.3)	79.7%(9.5)	91.7%(7.1)	**93.4% (6.7)***

Table 5.5: Classification accuracies using audio features, lyrics with language model differences, the voting and mixed feature approach for merging both. We used SVM and in parenthesis is the standard deviation. '*' means that the increase of a hybrid approach compared to the best of the individual methods (Audio or Lyrics) is statistically significant ($p < 0.05$)

These hybrid methods give significant improvements over both individual approaches, leveraging the complementary information available from audio and lyrics, at least for three of the four categories: "happy", "sad" and "relaxed", with all a significant ($p < 0.05$ using a Paired t-Test) overall increase. For the angry categories there is a slight increase in classification performance. However, the extremely high baseline of over 98% accuracy on audio alone limits the benefits of using a hybrid method. We should also notice that the multimodal approach reduces the standard deviation of the accuracies between folds, which means that the systems are more robust.

5.3. Conclusion

The results obtained with the different methods presented above are very encouraging, and the level of performance is more than sufficient for many practical applications. Our multimodal approach increases the performances for all the mood categories. We note very interesting results particularly for the "happy" , "sad" and "relaxed" categories, in which the complementarity of lyrics and audio significantly increases the overall accuracy. Performance using audio purely is already very high for the "angry" category, limiting the potential

impact of a multimodal approach. These results show that audio and lyrics information combined led to a better music mood classification system.
We should also comment that we have obtained the same trend in our results as Cho & Lee (2006) who were working on affect recognition but using a technique based on a manually-built affect lexicon. They reported better results on "happy" and "sad" lyrics than on "violent" (which could be related to our "angry" category). The results we presented here confirm the relevance of the lyrics to convey emotions or at least that the mood expressed in music and acoustical data is correlated with information contained in the text.
Posterior results from our contribution showed comparable results where lyrics enhanced the results. Yang et al. (2008a) evaluated unigram and bigram bag-of-features lyrics features with three methods for combining lyrics and audio information on 1,240 songs divided into four categories (the same as ours) and also in two dimensions, arousal and valence (based on Russell (1980)). The conclusion of this work was that lyrics could improve classification accuracy over audio-only classifiers. Hu et al. (2009a) validated our approach with another dataset. Their study described a method to build a ground truth of songs classified by mood based on last.fm tags. The approach is rather similar to the one we exposed in Chapter 4, except that they do not manually listen and validate each song. They compare different approaches for classification and lyrics description using text features such including bag-of-words (like we used in the first experiment of this Chapter), part-of-speech (Pang & Lee (2008)) and functions words (also called stopwords). The audio features were extracted using MARSYAS (Tzanetakis (2007)). This work was conducted with 18 mood categories. The results showed that our hybrid method based on Language Model Differences (LMD) outperforms lyric and audio features in nine categories among all combined feature sets. It also outperforms all other methods in five mood categories and achieves significantly better results than spectral features in three other mood categories. A worth noticing result from this work, more detailed in Hu & Downie (2010a,b) is also that in some categories (romantic, cheerful, hopeful, angry, aggressive, exciting and anxious), using lyrics alone gives better results than the classification with only audio features. However, we should keep in mind that this comparison may not be fair for audio systems, as they compare advanced text analysis against a rather basic audio classification with only spectral features.
In the future it will be interesting to compare our results to other approaches in affect recognition from text, like the methods based on common-sense or affective lexicons (Cho & Lee (2006)), and to investigate more advanced multimodal techniques. Besides the lyrics, the fine grain analysis of the voice itself using source separation techniques would be a very interesting research topic. And if we can then extract the lyrics from the voice signal (like proposed in Hosoya et al. (2005)), the technique we exposed in this chapter could be used directly on the transcription from the audio. It would also be worth studying the way the lyrics are related to the beat. Like discussed by Levitin (2006),

in the same vein that note syncopation is an important concept related to expectation and surprise, the words' rhythm and position in the measure should affect the way people perceive the lyrics and thus the emotion they convey. Moreover, supposing we could extract the lyrics and the melody, studying the position of the words in the measure (like in the work by Nichols et al. (2009)) would most probably enrich our system.

CHAPTER 6

Mood Classification using Genre

"It is with the heart that one sees rightly; what is essential is invisible to the eye" Antoine de Saint-Exupéry. Le petit prince.

6.1. Introduction

In cognitive science, there is a debate about the universality of emotional expressions in all aspects, including music. Some of the important musical features used by humans to categorize emotions (see Chapter 2 for more details), could be considered as universal. However, it is very complicated to prove this universality as listeners not exposed to western musical culture are very rare on the planet and also because most of the work on emotions are in a common language (english) and imposes a western type of analysis (see Wierzbicka (1999)). Fritz et al. (2009) report on cross-cultural studies with participants from a native African population (Mafa) and Western participants. It shows that the Mafa listeners can detect basic emotions better than random but not at the level of Western listeners. For people heavily exposed to western music as the author (and probably most of the readers) of this thesis, it is difficult to differentiate between innate cues that could be "built-in" our cognitive process, and cultural cues that are learned by exposure. Nevertheless, one very clear example to illustrate this learning of emotional expression is the Indian musical culture. In Indian *raagas* of South-Indian classical (Carnatic) music, emotions (called *rasas*) are formalized in the music with very particular cultural codes that only experts and highly educated indian people can apprehend and enjoy completely (Koduri & Indurkhya (2010)). Moreover, even in what we call the western musical culture, we believe that these codes could be dependent on the genre. The verification of this hypothesis goes beyond the scope of this thesis. Despite that, we want to verify if genre information is somehow associated with

mood and if it could help to classify music by mood. For instance, "ambient" music may be mostly *relaxed* and "heavy metal" most probably *angry*.

Actually, the first work experimenting mood classification from audio includes a related preliminary observation from their data. Li & Ogihara (2003) noticed that segmenting their dataset by genre and training on these subgroups, they obtained better accuracies than training with mixed genres. They mentionned that the "use of genre information might improve emotion detection". In opposition to this idea, Hu & Downie (2007) analyzed the co-occurences of mood and genre labels in music review and stated that there is "strong evidence that genre and mood are independent of each other". However they also provided some associations that suggest a possible relation. Lin et al. (2009) showed a correlation between emotions and genre labels but, according to our analysis, with an error-prone approximation, applying mood labels from an album to all its tracks. These works leads us towards a first step which is to verify and validate if there is a correlation between mood and genre. Then, a second step would be to exploit this information for mood classification.

This Chapter is divided into four main parts. We first explore the relation between genre annotations and our mood models, using reliable genre annotations (Section 6.2). After demonstrating a significant correlation between genre and mood, we design a Genre-based Mood Classifier and compare its accuracy to our previous results (Section 6.3). Then, we analyze mood classifiers to extract rules and to interpret their behavior (Section 6.4). Finally, in Section 6.5, we detail our related submission to MIREX 2010, which exploited the idea being discussed here and the results obtained.

6.2. Experiment 5: Association Mood / Genre

6.2.1. Objectives

In this experiment, we want to study the relationship between mood and genre. Hu & Downie (2007) explored these relationships based on datasets from AllMusicGuide[1] (AMG) and showed some association between mood and genre. In Table 6.1, we summarize the associations found in this study. Our first observation from this table is the rich set of mood adjectives employed, which is hard to compare with our models based on basic emotions (*happy*, *sad*, *angry*, *relaxed*). Our objective is to analyze the relationship between mood and genre by selecting a reliable genre dataset that we will then classify with our mood models. We will analyze the correlation between our mood classifications and a reliable genre annotation.

[1] http://www.allmusicguide.com. AllMusicGuide is a reference in music review, it claims to be "the most comprehensive music reference source on the planet"

Genre	Moods
RnB	sensual
Rap	street smart, witty
Jazz	fiery
Electronica	hypnotic, fun
Blues	gritty, rollicking
Vocal	sentimental
Country	sentimental
Gospel	spiritual, joyous
Comedy	silly
Folk	earnest, wistful
Latin	spicy, rousing
World	hypnotic, confident
Reggae	outraged, druggy
Soundtrack	atmospheric, theatrical
Easy listening	soothing, fun

Table 6.1: Summary of the significant genre-mood pairs found in the AMG dataset. Adapted from Hu & Downie (2007).

6.2.2. Dataset

The dataset is originally composed of 84677 song excerpts of 30 seconds annotated with genre. The reference for the genre annotation is the iTunes music store[2]. This dataset was chosen for two main reasons: i) because we had enough audio data to make an experiment at a large scale, ii) the genre annotation is made by professional of the music business (probably not musicologists, but we believe they take care about a useful categorization for their sells). The complete list of genre includes: 'Rock', 'Jazz', 'RnB/Soul', 'Alternative', 'Pop', 'Country', 'Blues', 'Vocal', 'Reggae', 'Latin', 'Hip Hop/Rap', 'Folk', 'Electronic', 'Classical', 'Soundtrack', 'Holiday', 'Dance', 'World', 'Inspirational', 'Disney', 'Children's Music', 'New Age', 'Opera', and 'Spoken Word'. However, in this long list, some labels appear in a very few cases only. To obtain more reliable results, and to avoid genres not used in enough cases, we decided to keep only the genres appearing in at least 1% of the songs (meaning at least in more than 900 songs). Our final dataset is composed of 81749 songs excerpts divided in 15 genres shown in Table 6.2. From table, we see that 34.49% is *Rock*, which is a standard issue in genre labeling, this genre category having very debatable limits (see Guaus (2009)).

[2]http://www.apple.com/itunes/

Genre	Songs	Percentage
Rock	28196	34.49%
Jazz	11986	14.66%
RnB/Soul	6663	8.15%
Alternative	5516	6.75%
Pop	5409	6.62%
Country	5171	6.33%
Blues	4526	5.54%
Vocal	3495	4.28%
Reggae	2293	2.80%
Latin	2022	2.47%
Hip Hop/Rap	1623	1.99%
Folk	1543	1.89%
Electronic	1443	1.77%
Classical	950	1.16%
Soundtrack	913	1.12%

Table 6.2: Distribution of genres in our dataset.

6.2.3. Method

To study the relationship between mood and genre, we apply our mood models to classify the songs from this genre dataset. For each mood, we use the best classifier obtained in Section 4.4.2 (i.e. Support Vector Machines with the best parameters for each category). At the end of the process, each song has an annotated genre and a set of mood classes. For each genre, we want to observe if each predicted mood occurs significantly more or less than in other genres, looking for some relevant relationships. To test for significance, we chose the Fisher's Exact Test (FET) like in Hu & Downie (2007). FET is a test used to examine the statistical significance of association/dependency between two variables and can work with small sample size and with unbalanced data (Fisher (1922)). This test is useful to verify if two classifications are associated. In our case, we want to verify if a genre is more significantly of a certain mood than the other genres. For instance, if we take the genre *rock* and the mood *angry*, we want to test if *rock* is "angrier" than other genre categories. Considering the 2x2 contingency table shown in Table 6.3:
Fisher demonstrated that the probability p of obtaining these values is:

$$p = \frac{(a+b)!(c+d)!(a+c)!(b+d)!}{a!b!c!d!n!} \tag{6.1}$$

If the p-value p is lower than 0.05 ($p < 0.05$), it means that the hypothesis "*rock* and *angry* are independent" is less than 5% likely. We use this threshold to

	Rock	Others	Total
Angry	a	b	a+b
Not Angry	c	d	c+d
Total	a + c	b + d	n

Table 6.3: 2x2 contingency table for the genre *rock* and the mood *angry*.

consider a statistical significance. Consequently in this example, if $p < 0.05$, *rock* and *angry* have a high probability of being related, we would consider them significantly correlated. To quantify better how strong this association is, we choose to apply another measure of the distribution divergence. The odds ratio describes the strength of the association between two data values. The closer it is to one, the more independent are the values (see Cornfield (1951)). In our case, if we want to calculate it on Table 6.3, we would have the following value:

$$oddsratio = \frac{\frac{a}{c}}{\frac{b}{d}} \tag{6.2}$$

$$\begin{cases} if oddsratio \geq 1 & signedoddsratio = oddsratio \\ if oddsratio < 1 & signedoddsratio = \frac{-1}{oddsratio} \end{cases} \tag{6.3}$$

6.2.4. Results

In Table 6.4, for each genre we show if each mood is significantly more representative with "+" and with "-" if the complementary (the "not" category) is significantly more representative. For instance *rock* songs are significantly more *angry* significantly less "sad" than the rest of songs. If the result is not statistically significant ($p >= 0.05$), it is shown by a 0.
One of the first comments we can make looking at the result is the strong association between genre and mood. Almost all genres are significantly correlated with mood categories (either positively of negatively). Moreover most of the associations seem intuive. For instance *rock* is mostly *angry* and *happy*, *blues* is *sad* and *relaxed*, *pop* and *reggae* are mainly *happy*. The only genre that have one mood category uninformative is *latin* which is not significantly more sad or not sad than the rest of the dataset. Also, some particular cases are interesting. *Rap* for instance, is mostly *angry*, *not sad*, *not relaxed and not happy*. Moreover *country* music, like *blues* or *jazz* is *not angry*, *sad* and *relaxed* but, on the other hand, it also has a particularity with a significant component of *happiness*. In a nutshell, we note here a clear association between mood and genre, with quite intuitive and logical results. However, we would like to study this phenomenon in more details with a quantitative measure. In Tables 6.5, 6.6 and 6.7, for each genre we show the signed odds ratio for each mood category. These plots can be understood as mood profiles for each

Genre	Angry	Sad	Relaxed	Happy
Rock	+	-	-	+
Alternative	+	-	-	+
Rap	+	-	-	-
Electronic	+	-	-	-
Blues	-	+	+	-
Folk	-	+	+	-
Jazz	-	+	+	-
Classical	-	+	+	-
Soundtrack	-	+	+	-
Vocal	-	+	+	-
Country	-	+	+	+
Reggae	-	-	-	+
Pop	-	-	-	+
RBSoul	-	-	-	+
Latin	-	0	-	+

Table 6.4: Significant mood for each genre.

genre. In fact, the greater the value of the odds ratio, the more correlated the moods with the genre. If the odds ratio is negative, the mood that is mostly associated is the "not" category. For instance, *rap* is the most "not relaxing" genre. The "happiest" are *pop*, *reggae* and *country*, while the saddest is definitely *classical*. "Angriest" genres are *rock* and *alternative*, genre categories where most probably we can find *metal* or *hard rock* styles that do not have a representative category in this taxonomy (see Section 6.4 for more insights). The most neutral genre in mood are *soundtrack* and *pop*. The former is most probably a mix of very different genres but used in a movie context. The same conclusion for the latter, which is also quite logical, *pop* music having a very vague meaning can represent very different music and, as it seems, all kind of moods. An analysis would not be fair without pointing out counter-intuitive results such as for *reggae*, observed mostly *not relaxed*. Again, this is relative, so this means to the whole music database is seen by the mood classifier as more relaxed than the particular case of *reggae*. This quite evident relation between genre and mood is motivating the next part, in which we will use genre to make a better mood classifier than the previous model (using mostly low-level features) detailed in Chapter 4.

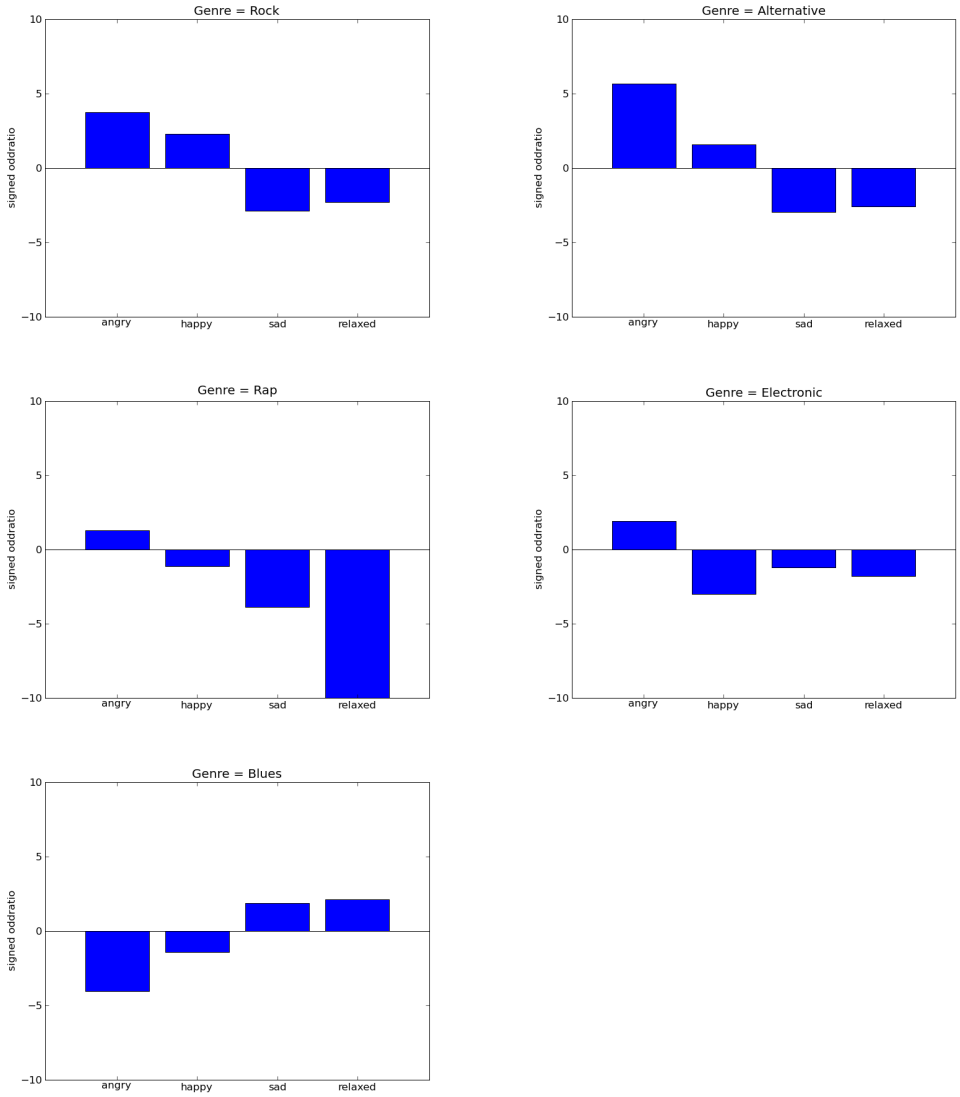

Table 6.5: Signed odds ratio of mood categories for *rock*, *alternative*, *rap*, *electronic* and *blues*.

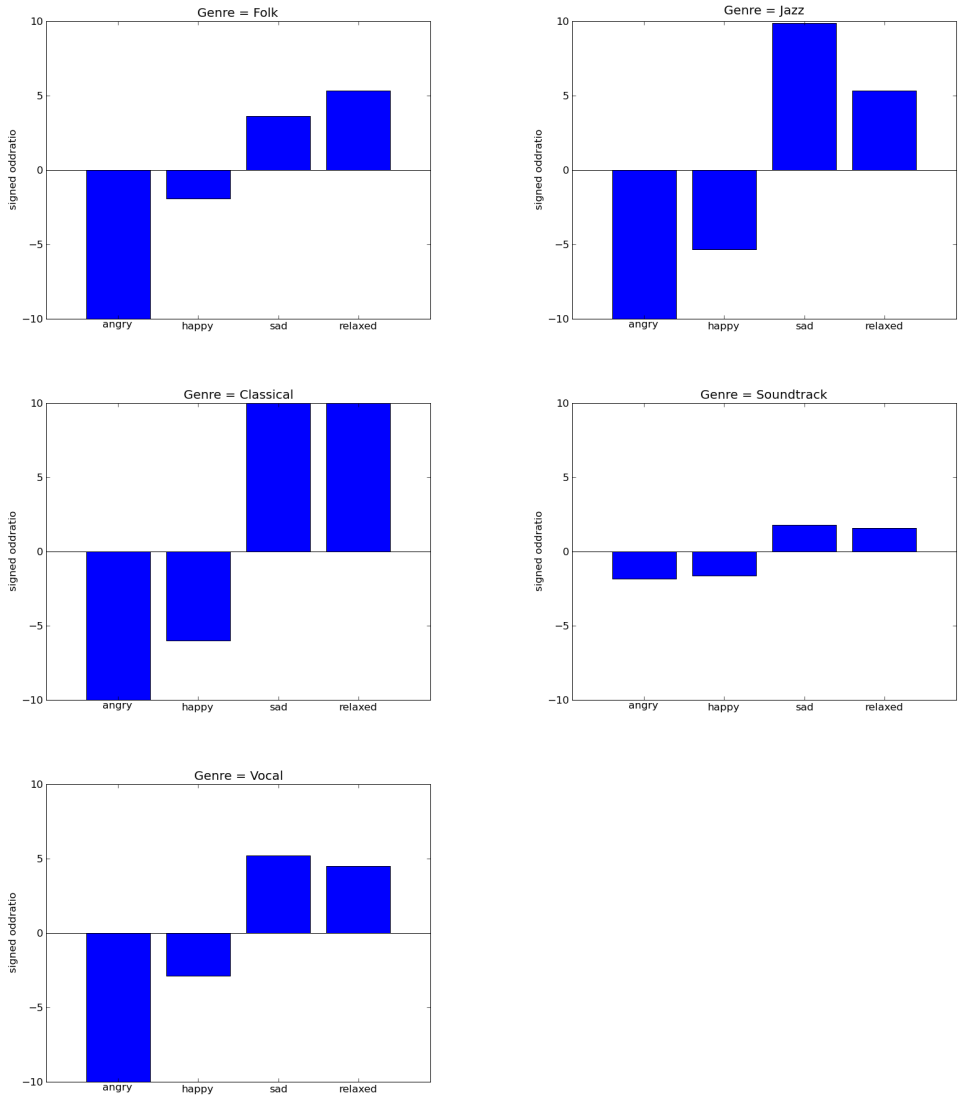

Table 6.6: Signed odds ratio of mood categories for *folk*, *jazz*, *classical*, *soundtrack* and *vocal*

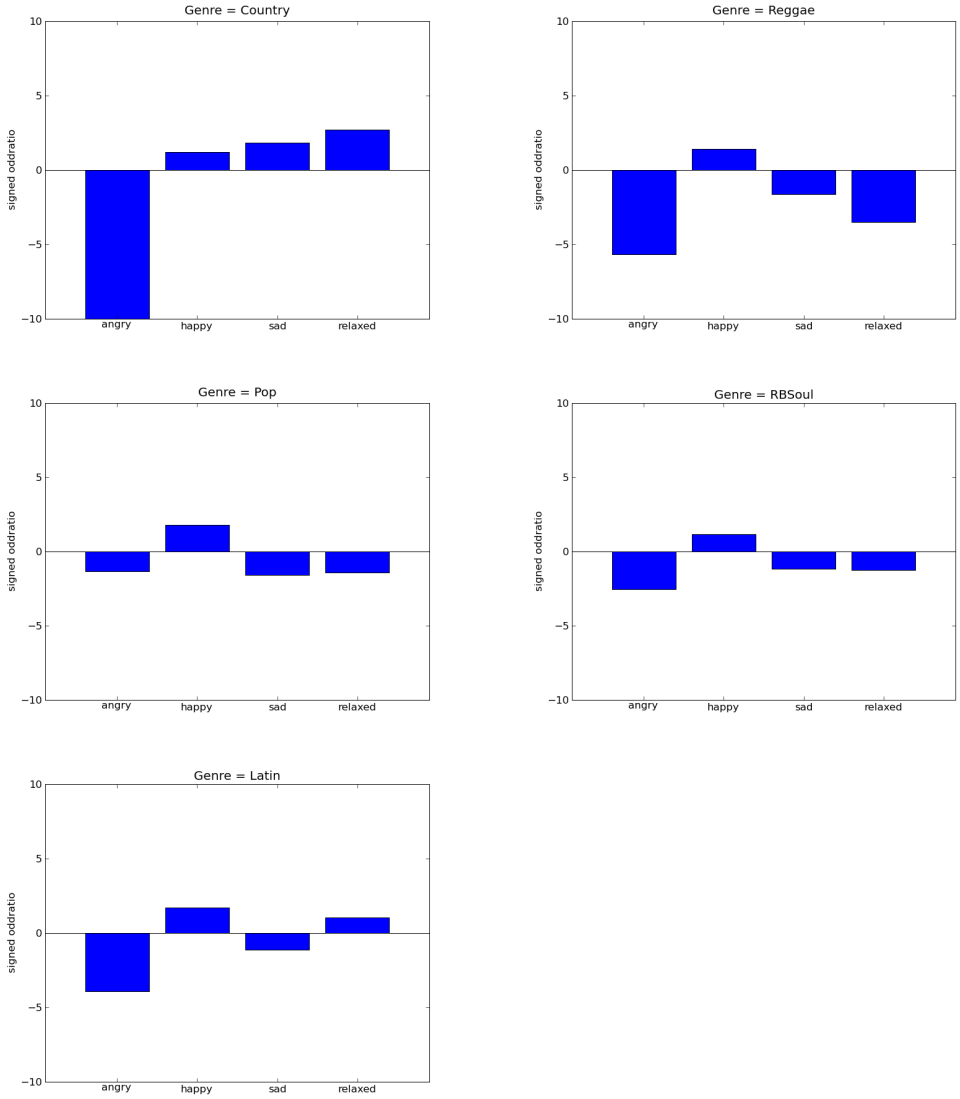

Table 6.7: Signed odds ratio of mood categories for each *country*, *reggae*, *pop*, *RnB/Soul*,*latin*

6.3. Experiment 6: Genre-based Mood Classifier (GMC)

6.3.1. Objectives

In the previous Section, we have shown that there is a significant association between mood and genre. Our hypothesis is that genre information would be very useful to classify music by mood. The objective of this experiment is to verify this hypothesis, defining a method to exploit genre information and to obtain better classification results than with the standard method (described in Chapter 4).

6.3.2. Method

Based on the evidence that genre and mood are related (see results in 6.2), we propose a Genre-based Mood Classifier (GMC). The main objective is to exploit genre information for classification. Keeping the objective of using audio data only, we need to produce genre information from the data itself. Consequently, we aim at extracting genre probabilities from the audio data using genre classification methods (see Guaus (2009) for more details). A schema of the GMC is shown in Figure 6.1.

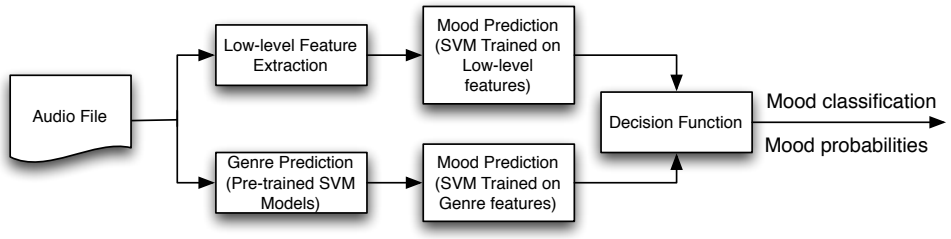

Figure 6.1: Schema of the Genre-based Mood Classifier (GMC)

The principle is to base the classification on two models: a standard model based on "low-level" features and a genre model based on genre features. Both models give a prediction in terms of probability that a decision function merges into one. Note that the upper branch follows the method from Chapter 4 that will serve as a baseline to compare with. Lin et al. (2009) proposed a comparable approach but only taking into account the genre predictions to build the mood classifiers. It is encouraging that this strategy already led to better results. In our method, we also consider, as important, other information from the audio descriptors not encoded by genre classifiers. For this experiment, we use our same mood dataset as described in Chapter 4, but

before evaluating our approach we will specify the new blocks: *genre prediction* with the genre descriptors and the *decision function*.

Genre prediction

In the next paragraphs, we detail the genre datasets that we used to train our genre classifiers.

Tzanetakis: This dataset was created by Tzanetakis & Cook (2002). It contains 1000 audio excerpts of 30 seconds distributed in 10 musical genres (blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, rock). This dataset has been used by many authors (Li & Ogihara (2005),Holzapfel & Stylianou (2007)).

Dortmund: We call "Dortmund" a dataset based on Garageband [3]. Garageband is an online community that allows to download free music from artists that upload their work to the platform. Users download music, rate it and write comments to the artists. Students manually classified part of the music available (1886 songs) into 9 musical genres (alternative, electronic, funk/soul/rnb, pop, rock, blues, folk/country, jazz, rap/hiphop) and computed descriptors to perform audio classification experiments (Homburg et al. (2005); Mierswa & Morik (2005)). Information on the detailed dataset is available online[4].

Electronica: The Electronica dataset is based on a work by Sesmero (2008). It is specific to eletronic music and aims a classifying its sub-genres. Nevertheless, this type of music has very eclectic styles and this can help at a more general level. It consist of 250 audio files distributed in 5 genres: ambient, drum and bass, house, techno and trance.

Rosamerica: This dataset was created internally at the Music Technology Group by a musicologist (see Guaus (2009) for more details) and is made of 400 audio songs for 8 genres: classical, dance, hiphop, jazz, pop, rhythm and blues, rock, and speech (which is more a type of audio than a musical genre).

For each collection, we build the best SVM classifier using the same approach as described in Chapter 4 for the mood classifiers. In Table 6.8, we summarize the databases used to train SVM models and show the accuracy of the best models, obtained using cross-validation. Even if, compared to the mood accuracies, they seem quite lower, this is not really the case, as these genres are considered as mutually exclusive. For instance, this means that the baseline of a random

[3] http://garageband.com
[4] http://www-ai.cs.uni-dortmund.de/audio.html

classifier for the Tzanetakis collection is around 10%. In that context, 77.74% is a satisfying result.

Genre DB	Categories	Accuracy
Genre (Tzanetakis)	blues, classical, country, disco, hip-hop jazz, metal, pop, reggae, rock	77.74%
Genre (Dortmund)	alternative, electronic, funk/soul/rnb, pop rock, blues, folk/country, jazz, rap/hiphop	60.29%
Genre (Electronica)	ambient, drum and bass, house techno, trance	89.33%
Genre (Rosamerica)	classical, dance, hiphop, jazz, pop, rhythm and blues, rock, speech	88.22%

Table 6.8: Genre ground-truth collections used to train our genre classifiers and the accuracy, in percentage, of the genre model we trained for each.

Genre descriptors

We call collection-based genre descriptors each prediction probability for each genre category. For instance, given a song, its probability to be *blues* given by the Tzanetakis-based genre model is one descriptor (between 0 and 1). Thus, from these collections of genre ground truths, we obtain 32 classifiers and consequently a set of 32 descriptors. Here is an example of a genre descriptor vector for a given song:
Please note that we extract the genre descriptors for new songs applying the pre-computed SVM models on the whole audio files (in our case 30s excerpts).

Decision function

To merge the decision between the two models (low-level model and genre-based model), we applied a simple multi-criteria analysis method: the weighted sum model. This model makes sense in our case because our data (output probabilities of two classifiers) are comparable (same unit and same range). The general model of the weighted-sum method is defined as follows:

$$score = \sum_{j=1}^{n} w_j a_j \tag{6.4}$$

where w_j is a weight coefficient and a_j a value, in our case a classifier probability. In our particular case of two elements it becomes:

$$score = w_{genre} P_{genre} + w_{std} P_{std} \tag{6.5}$$

genre tzanetakis	genre electronica
blues: 0.0203340500593	ambient: 0.971560716629
classical: 0.0280432011932	dnb: 0.00250834110193
country: 0.100025169551	house: 0.0124133452773
disco: 0.0134060159326	techno: 0.000919656595215
hiphop: 0.00809990148991	trance: 0.0125979110599
jazz: 0.238386839628	
metal: 0.00484063988551	
pop: 0.368437618017	
reggae: 0.0964034497738	
rock: 0.122023105621	

genre dortmund	genre rosamerica
alternative: 0.0254560224712	cla: 0.023480694741
blues: 0.0193506479263	dan: 0.00756007013842
electronic: 0.00889345258474	hip: 0.00603456888348
folkcountry: 0.238772079349	jaz: 0.520091176033
funksoulrnb: 0.139724195004	pop: 0.0914856642485
jazz: 0.403022617102	rhy: 0.33792090416
pop: 0.134895712137	roc: 0.0078586442396
raphiphop: 0.0042907949537	spe: 0.00556829059497
rock: 0.0255944803357	

Table 6.9: Example for a collection-based genre descriptors vector.

where w_{genre} is the weight applied to the genre-based model output probability and w_{std} the same for the standard model. Note that for normalization purposes and to have *score* in the same range as a probability value (between 0 and 1), we have $w_{genre} = 1 - w_{std}$, obtaining:

$$score = w_{genre}P_{genre} + (1 - w_{genre})P_{std} \tag{6.6}$$

Other decision functions could have been considered but would have needed parameter tuning. Indeed, we were afraid of adding here a possibility of overfitting disguised as another parameter optimization. For this reason, we applied this simple model and a balanced weight value $w_{genre} = \frac{1}{2}$.

6.3.3. Results

In Table 6.10, we show the results of the 10 run of 10-fold cross validation evaluation made with our GMC model and compared with:

- STD: The standard model from Chapter 4, "bag-of-feature" approach with the best SVM classifier. This is the upper branch of our GMC model. It is also equivalent to the GMC model using $w_{genre} = 0$ in the decision function.

- GOM: Genre-only model, classifying also with a SVM but only using genre descriptors. This is the lower branch of our GMC model. It is also equivalent to the GMC model with $w_{genre} = 1$.
- GMIX: instead of separating especially genre and other features, we "mix" the genre features adding them to the "bag-of-feature" from the STD model.

Category	STD	GOM	GMIX	GMC
Angry	98.17%	97.18%	97.51%	**99.12%***
Relaxed	91.43%	87.02%	89.21%	**93.54%***
Happy	84.57%	80.43%	85.32%	**88.51%***
Sad	87.66%	82.21%	87.73%	**91.30%***

Table 6.10: Accuracies of 10 runs of 10-fold cross validation for the different methods: standard (STD), genre-only (GOM), genre descriptors mixed with other features (GMIX) and our genre model (GMC).'*' means that the increase in accuracy compared to the other results is statistically significant.

We note that the GMC method gives the best results and that the accuracies obtained are statistically significantly greater than the standard models (at $p < 0.05$). This is also the case comparing with the model mixing all features including the genre descriptors. These results were obtained using a decision function for the GMC model with $w_{genre} = \frac{1}{2}$, but trying other values for this weight did not lead to significant changes (no significant higher values than the accuracies in Table 6.10). This confirms that genre information, even if extracted automatically (and thus not perfect) is a very valuable information for mood classification. Conceptually, this is also an interesting result, because the genre features are computed from the same low-level features. However, as they contain an expert knowledge, this step adds information related to human categorization. This is quite encouraging and we can make the hypothesis that the more high level feature swe add, the better we will be able to automatically classify music in general, and by mood in particular, supposing these high level features are related to mood.

6.4. Experiment 7: Making sense of the classifiers: Rules extraction

6.4.1. Objective

In this experiment, we want to understand our classification models better. SVMs are quite complex to interpret, however we are interested in understanding their behavior. In particular, we want to grasp the reasons why our

models would consider one song to belong to a certain mood or not. Using our genre-based model, we want to extract rules from low level audio features and genre descriptors. This would help the comprehensibility of the SVM classification criteria and could give some insights about specificities of mood classification.

6.4.2. Method

For this experiment, we will perform rule extraction to understand the classification made by the support vector machine models (employed previously mainly as a black-box). Rule extraction from Artificial Neural Networks (ANNs) has been investigated in many studies (see Huysmans et al. (2006) for a full overview). The common techniques use the trained models as an oracle to classify new training examples that are later used by a symbolic learning algorithm. The main idea behind this technique is the assumption that the trained model can better represent the data than the original dataset (Martens et al. (2009)). Consequently, instead of using our ground truth, we will use the predictions made from our models on our dataset. Once the prediction on our data is performed, we will apply a tree induction technique (C4.5) to extract rules. C4.5 induces decision trees in a top-down approach, based on information theoretic concepts. We can summarize our rule extraction technique as follows:

SVM Rule Extraction Algorithm:

- 1 - Tune SVM with a grid search using cross-validation
- 2 - Train SVM with the best parameters on all the data
- 3 - Change class label of the training data to the SVM predicted class
- 4 - Tune a C4.5 decision tree varying the M parameter (minimum number of instances per leaf) using cross-validation
- 5 - Train a C4.5 decision tree with the best parameter M on all the data

The rules extracted from the SVMs are then summarized in the trees produced in the last step of the algorithm (step 5). The accuracy obtained in step 4, tells us how well the tree represents the SVM.

6.4.3. Results

We applied our SVM Rule Extraction Algorithm for each of our mood category. In the following figures (from 6.2 to 6.5 for low-level descriptors-based models and from 6.6 to 6.7 for genre models), we show the trees induced by the C4.5 algorithm for each mood. M is the parameter of the decision tree algorithm,

used to defined the minimum number of instances per leaf. The accuracy is the accuracy of the decision tree (using the same M parameter), on the data labeled by the SVMs, in a 10-folds cross validation manner.

Rules extracted from Audio-based Classifiers

In this part, we only consider audio descriptors. We predict the mood with our trained genre-based SVM model (the one performing better for each class). For each mood we apply a decision tree algorithm to observe the underlying rules that could make sense. Obviously, these results are a simplification of the problem, but they help to understand better the way our classifier behaves. We can deduce from these results similar conclusions about the correlations between audio descriptors and mood categories than in Chapter 4 (Section 4.3). For instance the *angry* predictions of our SVM classifier are described with 95.49% accuracy by a tree shown in Figure 6.2. It confirms the importance of descriptors such as dissonance and spectral complexity (if a music is dissonant or spectrally complex, it is mostly considered as *angry*). For *relaxed*, we also observe the same similarities. It needs a low spectral flux or dissonance mean to be classified as *relaxed*. *Sad* can also be expressed at a high percentage (88.76%) with a few descriptors. But, in that case, it includes not only spectral features but also the onset rate: *sad* is slower (or with less events per seconds) than *not sad*. *Happy* is more complex to model with an accuracy of this tree of 78.95%. However it is interesting to note that in order to be *happy*, a music should have a high spectral centroid, low dissonance, major key and high spectral changes (spectral flux). Having explained in Chapter 2 and 4 the details of these audio descriptors, the extracted rules seem quite reasonable and logical.

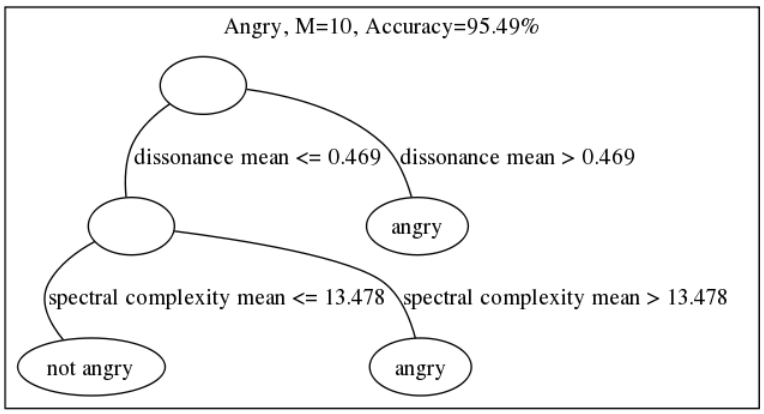

Figure 6.2: Audio descriptors rules extracted for the *angry* category

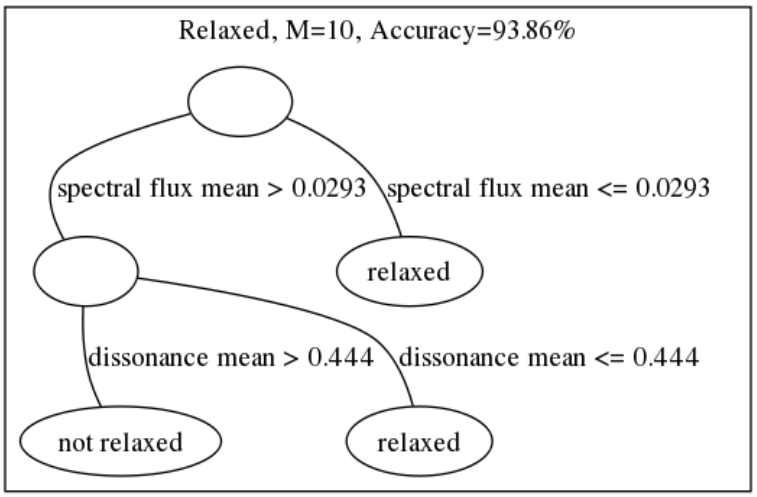

Figure 6.3: Audio descriptors rules extracted for the *relaxed* category

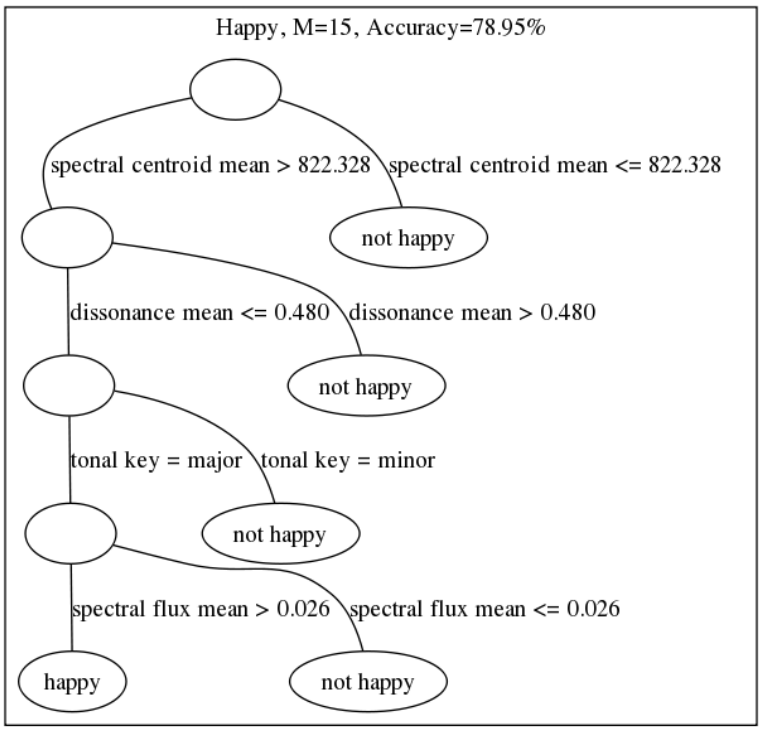

Figure 6.4: Audio descriptors rules extracted for the *happy* category

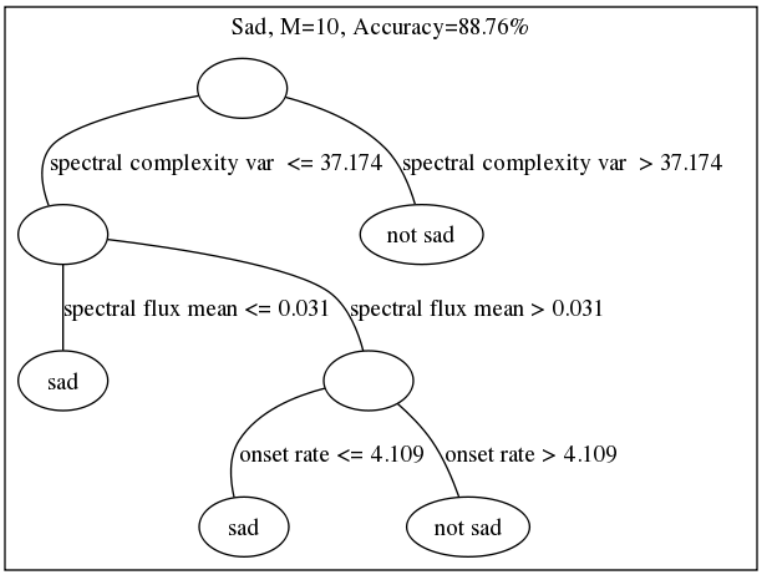

Figure 6.5: Audio descriptors rules extracted for the *sad* category

Rules extracted from Genre-only Classifiers

In this part, we only use genre descriptors that are prediction probabilities made with pre-trained SVM models. The goal is to express mood only in terms of genre and to have results that can be easily interpreted in that context. Please note that, in this case, we are inverting the logical order of causality (genres are probably not used to cause moods). The results are shown in the following figures and confirm previous observations about association between moods and genres. One striking example is the *angry* category, we observe that it is very much related to the genres *rock* and *metal*. With only one of these genres we cover 95.49% of the Support Vector Machine model results. Another simple result (more surprising) is what we observe for *sad*. It is very much related to the electronic music genre descriptors called *ambient*. This descriptor alone explains 92.70% of the SVM predictions. We could explain this because most of the *not sad* cannot be classified as *ambient*, but most of the *sad* music can. We also observe the strong relation between *jazz* and *sad*. The *relaxed* category is a bit more complex. In order to be relaxed, a music has to be mostly *jazz* or neither *metal* nor *dance*. Finally *happy* is again the most difficult to represent, with a tree only explaining 77.23% of the SVM predictions. However it is still interesting to see that, in order to be *happy*, a music should not be *jazz*, *electronic* nor *metal*.

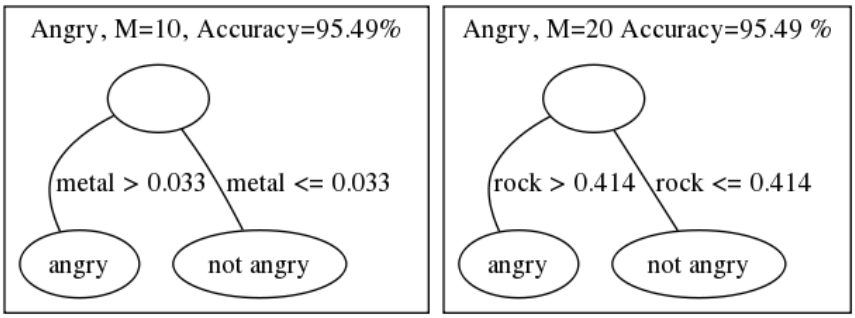

Figure 6.6: Genre descriptors rules extracted for the *angry* category

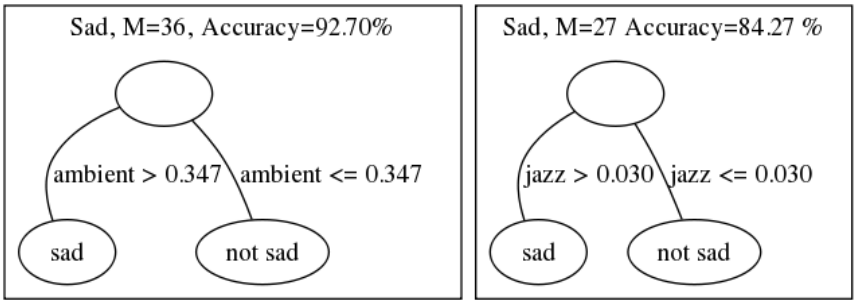

Figure 6.7: Genre descriptors rules extracted for the *sad* category

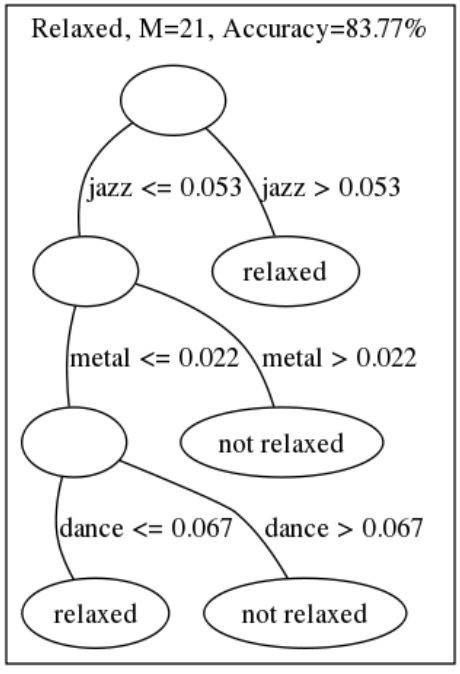

Figure 6.8: Genre descriptors rules extracted for the *relaxed* category

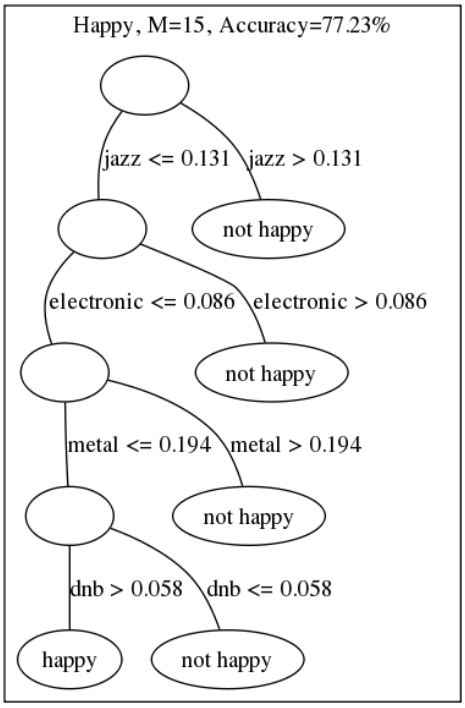

Figure 6.9: Genre descriptors rules extracted for the *happy* category

6.5. Evaluation: MIREX 2010

In 2010, the MIREX team ran the same evaluation for audio mood classification described in Chapter 4 (Section 4.6). We detail here the algorithms we submitted, one of them based on the findings previously presented.

Features

We divide our features in two main categories. The "base" or low-level features which are state-of-the-art MIR features (see MIREX 2009 in Section4.6 of Chapter 4 for the detailed "base" features) and the "high-level" features that we detail in the next paragraph. These high-level features are divided into 3 groups: genres, moods and other high level descriptors (such as gender, acoustic, perceptual speed etc...).

High-Level features: one of the originality of our approach is the integration of high-level (or semantic) descriptors. High-level concepts encapsulate different pattern of low-level descriptors into a compact representation that can add useful information. Based on this idea, we added high level features of different categories. This approach is similar to the one explained previously with genres as high-level descriptors. We added here more descriptors using trained classifiers on other categorizations. Indeed, these models are pretrained algorithms using Support Vector Machines that are added to the "bag of features". They are trained on curated ground truth databases (detailed in Bogdanov et al. (2011)). We consider them as other features with value between 0 and 1 corresponding to the SVM model prediction probability. In Table 6.11 we list the different models used. The first mood collection (Mood 5 classes) is a dataset we built on the same model as the MIREX collection (using the same taxonomy). The other mood collections are the ones we described in this thesis with: *angry*, *relaxed*, *happy* and *sad*. The genre datasets are those that we presented in Section 6.3.2[5], except Genre (5) which is a collection of ballroom styles. The rest of the ground-truth comes from collections developed internally. More information about the *culture* descriptor can be found in the related work by Gómez (2008).

Classification

For classification, we used different Support Vectors Machine classifiers for four descriptors sets: moods, genre, low-level, high-level. In a similar fashion that for our Genre-based Mood Classifier (GMC), we used a decision function with a weighted-sum approach like the one presented in Section 6.3.2. At the end we have implemented three configurations, MTG-Mood, MTG-Genre MTG-

[5] (1) Tzanetakis, (2) Dortmund, (3) Electronic, (4) Rosamerica.

Type	Classes
Mood (5 classes)	5 classes similar to the mirex clusters Hu & Downie (2007)
Mood (Happy)	happy, not happy
Mood (Sad)	sad, not sad
Mood (Relaxed)	relaxed, not relaxed
Mood (Angry)	angry, not angry
Genre (1)	blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, rock
Genre (2)	alternative, electronic, funk/soul/rnb, pop, rock, blues, folk/country, jazz, rap/hiphop
Genre (3)	ambient, drum and bass, house, techno, trance
Genre (4)	classical, dance, hiphop, jazz, pop, rhythm and blues, rock, speech
Genre (5)	cha cha cha, quickstep, rumba-international, rumba-american, rumba-misc, tango, waltz, samba, viennese waltz, jive
High-Level Perceptual Speed	fast, medium, slow
High-Level Timbre	bright, dark
High-Level Culture	western, non western
High-Level Gender	male, female
High-Level Acoustic	acoustic, not acoustic
High-Level Electronic	electronic, not electronic

Table 6.11: High-level features including Mood and Genre. Types and classes of the SVM models are trained on reference databases (see Bogdanov et al. (2011)).

Mood-Baseline and their respective decision functions, with weights defined during cross-validation experiments:

$$score_{MTG-Mood} = 0.5P_{mood} + 0.25P_{genre} + 0.125P_{lowLevel} + 0.125P_{highLevel} \tag{6.7}$$

$$score_{MTG-Genre} = 0.143P_{mood} + 0.57P_{genre} + 0.143P_{lowLevel} + 0.143P_{highLevel} \tag{6.8}$$

$$score_{MTG-Mood-Baseline} = P_{mood} \tag{6.9}$$

We also submitted another approach (MTG-RCA) using Relevant Component Analysis (RCA) and a custom distance measure combining a Kullback-Leibler distance applied to MFCCs and an Euclidean distance on the RCA components. The RCA technique was the same submitted to MIREX 2009 and further explained in section in Chapter 4, Section 4.6.3.

Results and Discussion

Submission	Accuracy
WLJW2	64.17%
SSPK1	63.83%
CH3	63.50%
GP1	63.17%
MTG-Genre	**63.00%**
MTG-Mood	**63.00%**
CH1	63.00%
CH2	63.00%
CH4	62.67%
RRS1	61.67%
TS1	61.00%
FE1	60.83%
GR1	60.67%
FCY1	60.17%
FCY2	59.50%
BRPC2	59.00%
BRPC1	58.67%
MTG-Mood-Baseline	57.67%
TN4	57.50%
MTG-RCA	55.55%
TN1	55.55%
RJ1	54.83%
RK1	54.83%
BMPE2	54.67%
HE1	54.17%
MBP1	54.00%
MW1	54.00%
WLJW1	53.83%
JR4	51.17%
JR2	51.17%
RJ2	50.17%
RK2	50.17%
TN2	48.58%
JR3	46.83%
JR1	46.33%
MP2	36.17%

Table 6.12: MIREX 2010: Comparison with other submissions.

In Table 6.12, we report the results of the evaluation[6]. Our submission ranks among the topmost accuracy results, with no statistically significant difference with the first one and followers. Again in this 2010 evaluation, like in previous editions, all the submissions with the best accuracy use Support Vector Machines. A surprising fact is the dramatical drop for our baseline submission (MTG-Baseline), with classifiers having much lower accuracies than in previous years. In our different algorithms, we note a slight increase in accuracy between the algorithm with genre and high-level descriptors compared with our submissions from 2009, using the same low-level features (62.83% of accuracy). This difference is rather small but still shows a potentially better classifier. It is worth noticing the results of the mood-only classifier (MTG-Mood-Baseline). This classifier, trained only with our mood descriptors, achieved a relatively high result of 57.67%. This means that these mood descriptors are quite relevant even used in a different context. It is however true that the database emulating the MIREX classes must have helped to get this result (even if quite small with only 218 files in total). The distance-based method (MTG-RCA), using a K-NN classifier, gave worst results that the others. If we look back at the first edition of this evaluation task, the best results achieved were already quite comparable to the 2010 best results with 61.5% of accuracy. This remains to be calculated, but improvements do no seem to be significant and this "glass-ceiling" may be due, on one hand, to the limitation produced by the model and the dataset (Hu et al. (2008)) but also on the other hand to a more general glass-ceiling problem in Music Information Retrieval (Aucouturier & Pachet (2004)). In a nutshell, we improved our results with a voting approach on the high-level descriptors such as genre and shown that our mood classifiers are quite relevant achieving a satisfying accuracy while used alone. In Figure 6.10, we plot the MIREX results we obtained in 2007, 2009 and 2010 together with the best and worst accuracies, a random baseline and the mean of the best half submissions.

[6]see the MIREX website for more details: http://nema.lis.illinois.edu/nema_out/9b11a5c8-9fcf-4029-95eb-51ed561cfb5f/results/evaluation/summary.html

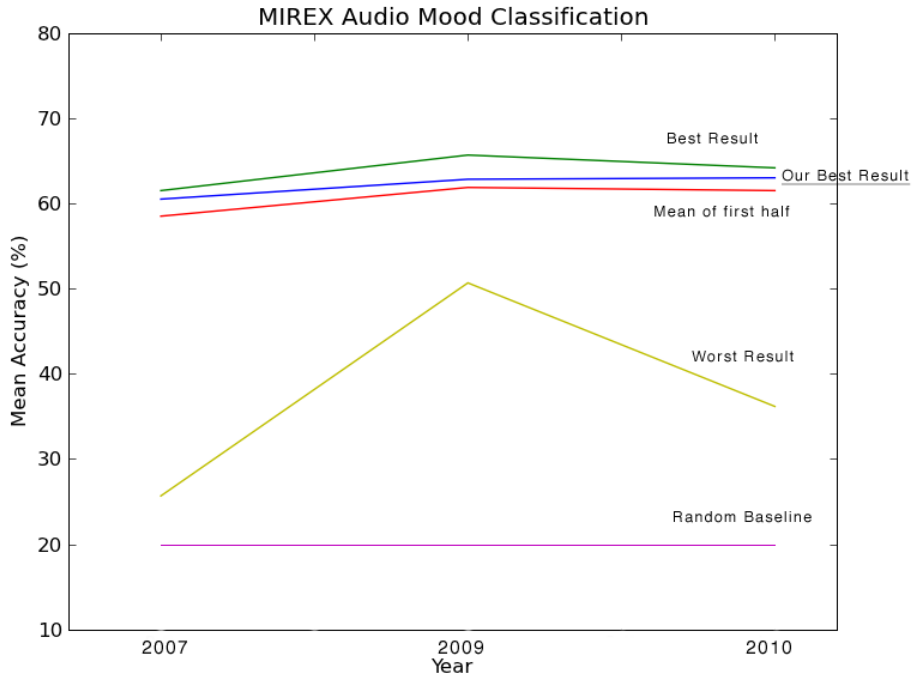

Figure 6.10: Summary of results for the Audio Mood Classification task at MIREX 2007, 2009, 2010. We plot our best results together with the lowest and highest accuracies, the random baseline and the mean of the best first half accuracies.

6.6. Conclusion

To the best of our knowledge, no previous work have studied the correlation and usefulness of genre information to classify music by mood as deeply as this work. In this Chapter, we demonstrated a clear relation between mood and genre and shown that genre helps to better classify music by mood. We proposed a method that achieved statically significantly better results. This also has been validated by high accuracy results in the MIREX mood evaluation. We also proved in 2010 MIREX evaluation that using only our mood models, we can achieve good results on a another database with a different taxonomy and mood representation. We can make the hypothesis that the better genre classification, the better results we can get. However, we believe we are reaching the limit of the approach we took by choosing a basic emotion type of representation and a generic ground truth. Indeed, the accuracies obtained by our classifier reach very high values meaning that we have succeeded in modeling simple mood categories, but challenges remain like we will discuss in the concluding chapter.

CHAPTER 7

Conclusions and suggestions for further work

"We know too much and feel too little. At least, we feel too little of those creative emotions from which a good life springs" Bertrand Russell.

7.1. Introduction

When we started this research, only a few preliminary works were available with no particular evidence that mood classification could work at the level we reached in this thesis. First, in Chapter 2, we reviewed the literature about emotions and previous work in music mood classification. In Chapter 3, we studied how people in an online music community tag music by mood, helping to understand which representation could be used. In Chapter 4, we demonstrated that we could design algorithms to automatically classify music by mood from the audio signal, explaining the contribution of individual audio features. In Chapter 5, we showed more advanced methods using lyrics. In Chapter 6, we analyzed the relation between mood and genre and proposed a model reaching higher accuracies with genre information, also automatically extracted from the audio signal. Finally, can we say that computers could feel emotions while listening to music? No, but we proved that computers are starting to recognize emotions in music.

7.2. Summary of contributions

- It is, to the best of our knowledge, the first publicly available thesis focusing on automatic audio music mood classification analyzing the contribution of high-level audio features.
- It exposes the complexity of the emotion problem and explains how to simplify several aspects to make generic music mood classification possible.

- It analyzes how a large online community uses emotion labels, compares it with well-known models and, from this results, proposes a useful representation.

- It shows a new method to create reliable music collections based on both the wisdom of crowds and experts.

- It provides a new method to use lyrics information obtaining higher accuracies and contributing to text retrieval.

- It demonstrates the relation between mood and genre and details a new algorithm using automatic genre descriptions for mood classification.

We should also notice that our approaches, adapted to the MIREX evaluation, were ranked among the best results. The mood classifiers presented in this thesis have been implemented in the European project PHAROS and several demonstration prototypes (see Appendix A). Please also note that our mood algorithms are part of products commercialized by BMAT[1]. The outcomes of the research detailed in this thesis have been published in the form of several papers in international conferences, journal and a book chapter. We list these publications in Appendix C.

7.3. Future perspectives

Obviously, we are still far from being able to model all the subtleties of emotions, especially because many aspects are not contained in the audio signal or in the lyrics. Also, we are conscious about the limitations of our methods limited to a generic approach, focused on the consensus and biased to a mainstream type of western music. But to conclude, we want to mention specific research directions for future works.

Personalized Models. The algorithms we designed are for a general purpose, working with most of people, with the drawback of being sometimes prototypical. In order to go into more details, we would need to put the user in the loop. This is the most exciting perspective as a continuation of the present work. Now that we have a solid basis for a simple emotion taxonomy, we can start to personalize these models and to adapt them to each user. We believe that with active learning, we can tune the current mood models to be more precise or to create new personalized models from scratch. The subjective part of emotions we tried to avoid with a generic purpose needs to be tackled in future works.

[1]Barcelona Music and Audio Technologies. http://www.bmat.com

More and Better Audio Descriptors. There are many musical cues associated with emotions that we cannot detect because of a lack of descriptors. Looking at the literature review in Chapter 2 (Section 2.3.2), we can easily identify audio features that would help a lot in mood classification: pitch (for each instrument, or at least the main melody), singer's formant, intervals or vibrato to name a few. We hope that results from source separation will enable to isolate different parts of a track and in particular the main melody and the singer's voice. Actually, this would probably be useful for any audio music classification task. Moreover, there is a need to consider the time variations of audio features. Descriptors such as the tonal evolution of a musical piece should be very informative.

Improved Machine Learning Algorithms. Even if very complex and advanced machine learning algorithms are available, we believe that there will be some improvements in this domain too. Especially, we expect models as accurate as Support Vector Machines but that are more flexible to changes in time, that easily allow active learning and that are not too sensitive to unbalanced datasets. Another improvement would be to have learning algorithms that could easily manage feature vectors time series.

Complex Representations. By complex representations, we mean that there is a theoretical need to formalize emotion representations in a way that is more related to human perception. We believe that we are using simplistic representation when considering only a few categories or dimensions. This also helps to get a consensus. But, more music-specific or even user-specific representations should be investigated.

Cognitive models. Related to the previous point, we can investigate representations that are based in how we perceive emotions. But moreover, the way we build our current models is not close to the way we listen to, perceive or categorize music. Using computational models inspired by cognitive studies could lead to better results.

I hope you enjoyed reading this thesis.

Bibliography

Alm, C. O., Roth, D., & Sproat, R. (2005). Emotions from Text: Machine Learning for Text-based Emotion Prediction. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 579–586. Vancouver, British Columbia, Canada: Association for Computational Linguistics.

Aucouturier, J. J. & Pachet (2004). Improving Timbre Similarity: How high is the sky? *Journal of Negative Results in Speech and Audio Sciences*, *1*(1).

Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., & Vanthienen, J. (2003). Benchmarking State-of-the-Art Classification Algorithms for Credit Scoring. *The Journal of the Operational Research Society*, *54*(6), 627–635.

Besson, M., Faita, F., Peretz, I., Bonnel, A. M., & Requin, J. (1998). Singing in the Brain: Independence of Lyrics and Tunes. *Psychological Science*, *9*(6), 494–498.

Bigand, E., Vieillard, S., Madurell, F., Marozeau, J., & Dacquet, A. (2005). Multidimensional scaling of emotional responses to music: The effect of musical expertise and of the duration of the excerpts. *Cognition & Emotion*, *19*(8), 1113–1139.

Bischoff, K., Firan, C., Paiu, R., Nejdl, W., Laurier, C., & Sordo, M. (2009). Music Mood and Theme Classification a Hybrid Approach. In *Conference of the International Society for Music Information Retrieval (ISMIR)*. Kobe, Japan.

Blood, A. J. & Zatorre, R. J. (2001). Intensely pleasurable responses to music correlate with activity in brain regions implicated in reward and emotion. *Proceedings of the National Academy of Sciences*, *98*(20), 11818–11823.

Bogdanov, D., Serra, J., Wack, N., Herrera, P., & Serra, X. (2011). Unifying Low-Level and High-Level Music Similarity Measures. *Multimedia, IEEE Transactions on*, *13*(4), 687–701.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152. New York, NY, USA: ACM.

Breiman, L. (2001). Random Forest. *Machine Learning*, *45*, 5–32.

Chang, C.-c. & Lin, C.-J. (2001). LIBSVM: a Library for Support Vector Machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Cho, Y. H. & Lee, K. J. (2006). Automatic Affect Recognition Using Natural Language Processing Techniques and Manually Built Affect Lexicon. *IEICE - Trans. Inf. Syst.*, *E89-D*(12), 2964–2971.

Cornfield, J. (1951). A method of estimating comparative rates from clinical data; applications to cancer of the lung, breast, and cervix. *Journal of the National Cancer Institute*, *11*(6), 1269–1275.

Dalla Bella, S., Peretz, I., Rousseau, L., & Gosselin, N. (2001). A developmental study of the affective value of tempo and mode in music. *Cognition*, *80*(3), 1–10.

Damasio, A. (1994). *Descartes' Error: Emotion, Reason, and the Human Brain*. New York: Harper Perennial.

Davidson, R. J. (2001). *On Emotion, mood, and related affective constructs*. Oxford University Press.

Davies, S. (2001). *Philosophical perspectives on music's expressiveness*. Oxford University Press.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, *41*, 391–407.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, *39*(1), 1–38.

Duda, R. O. & Hart, P. E. (1973). *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc.

Dunker, P., Nowak, S., Begau, A., & Lanz, C. (2008). Content-based mood classification for photos and music: a generic multi-modal classification framework and evaluation approach. In *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, MIR '08, pp. 97–104. New York, NY, USA: ACM.

Eerola, T., Lartillot, O., & Toiviainen, P. (2009). Prediction of Multidimensional Emotional Ratings in Music from Audio using Multivariate Regression Models. In *Proceedings of ISMIR 2009*, pp. 621–626.

Eerola, T. & Vuoskoski, J. K. (2011). A comparison of the discrete and dimensional models of emotion in music. *Psychology of Music*, *39*(1), 18–49.

Ekman, P. (1992). An argument for basic emotions. *Cognition & Emotion*, *6*(3), 169–200.

Farbood, M. M. (2006). *A quantitative, parametric model of musical tension.* Ph.D. thesis, MIT Media Lab.

Farnsworth, P. R. (1954). A study of the Hevner adjective list. *The Journal of Aesthetics and Art Criticism*, *13*(1), 97–103.

Fehr, B. & Russell, J. A. (1984). Concept of emotion viewed from a prototype perspective. *Journal of Experimental Psychology: General*, *113*(3), 464–486.

Fisher, R. A. (1922). On the Interpretation of X2 from Contingency Tables, and the Calculation of P. *Journal of the Royal Statistical Society*, *85*(1), 87–94.

Fix, E. & Hodges, J. L. (1951). Discriminatory analysis, nonparametric discrimination: Consistency properties. *US Air Force School of Aviation Medicine*, *Technical Report 4*.

Frijda, N. H. (1986). *The emotions.* Cambridge University Press ; Editions de la Maison des Sciences de l'homme.

Fritz, T., Jentschke, S., Gosselin, N., Sammler, D., Peretz, I., Turner, R., Friederici, A. D., & Koelsch, S. (2009). Universal recognition of three basic emotions in music. *Current biology : CB*, *19*(7), 573–576.

Gabrielsson, A. (2001). *Emotions in strong experiences with music*, pp. 431–449. Oxford University Press.

Gómez, E. (2006). *Tonal description of music audio signals.* Ph.D. thesis, Universitat Pompeu Fabra.

Gómez, E. (2008). Comparative Analysis of Music Recordings from Western and Non-Western traditions by Automatic Tonal Feature Extraction.

Gosselin, N., Peretz, I., Noulhiane, M., Hasboun, D., Beckett, C., Baulac, M., & Samson, S. (2005). Impaired recognition of scary music following unilateral temporal lobe excision. *Brain*, *128*(3), 628–640.

Gouyon, F. (2003). Towards Automatic Rhythm Description of Musical Audio Signals. Representations, Computational Models and Applications.

Gouyon, F., Herrera, P., Gómez, E., Cano, P., Bonada, J., Loscos, A., Amatriain, X., & Serra, X. (2008). *Content Processing of Music Audio Signals*, chap. 3, pp. 83–160. Berlin: Logos Verlag Berlin GmbH.

Grewe, O., Nagel, F., Kopiez, R., & Altenmüller, E. (2007). Emotions over time: Synchronicity and development of subjective, physiological, and facial affective reactions to music. *Emotion*, *7*(4), 774–788.

Guaus, E. (2009). *Audio content processing for automatic music genre classification: descriptors, databases, and classifiers*. Ph.D. thesis, Universitat Pompeu Fabra.

He, H., Jin, J., Xiong, Y., Chen, B., Sun, W., & Zhao, L. (2008). Language Feature Mining for Music Emotion Classification via Supervised Learning from Lyrics. In *Proceedings of the 3rd International Symposium on Advances in Computation and Intelligence*, ISICA '08, pp. 426–435. Berlin, Heidelberg: Springer-Verlag.

Herrera, P., Bello, J., Widmer, G., Sandler, M., Celma, O., Vignoli, F., Pampalk, E., Cano, P., Pauws, S., & Serra, X. (2005). SIMAC: Semantic interaction with music audio contents. In *Proceegins of the 2nd European Workshop on the Integration of Knowledge, Semantics and Digital Media Technologies*, pp. 399–406. London, UK.

Hevner, K. (1936). Experimental studies of the elements of expression im music. *The American Journal of Psychology*, *48*(2), 246–268.

Holzapfel, A. & Stylianou, Y. (2007). A Statistical Approach to Musical Genre Classification using Non-Negative Matrix Factorization. pp. II–693–II–696.

Homburg, H., Mierswa, I., Morik, K., Möller, B., & Wurst, M. (2005). A Benchmark Dataset for Audio Classification and Clustering. In *Proc. ISMIR*, pp. 528–531.

Hosoya, T., Suzuki, M., Ito, A., Makino, S., Smith, L. A., Bainbridge, D., & Witten, I. H. (2005). Lyrics Recognition from a Singing Voice Based on Finite State Automaton for Music Information Retrieval. In *in Proc. ISMIR, 2005*, pp. 532–535.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003). A Practical Guide to Support Vector Classification. Tech. rep., Department of Computer Science, National Taiwan University.

Hu, X. & Downie, J. (2007). Exploring mood metadata: Relationships with genre, artist and usage metadata. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pp. 23–27.

Hu, X. & Downie, J. S. (2010a). Improving mood classification in music digital libraries by combining lyrics and audio. In *Proceedings of the 10th annual joint conference on Digital libraries*, JCDL '10, pp. 159–168. New York, NY, USA: ACM.

Hu, X., Downie, J. S., & Ehmann, A. F. (2009a). Lyric Text Mining in Music Mood Classification. In *10th International Society for Music Information Retrieval Conference (ISMIR 2009)*, pp. 411–416.

Hu, X. & Downie, S. (2010b). When lyrics outperform audio for music mood classification: A feature analysis. In *Proceedings of the11th International Society for Music Information Retrieval Conference (ISMIR 2010)*.

Hu, X., Downie, S. J., Laurier, C., Bay, M., & Ehmann, A. F. (2008). The 2007 MIREX audio mood classification task: Lessons learned. In *Proceedings of the 9th International Conference on Music Information Retrieval*, pp. 462–467. Philadelphia, PA, USA.

Hu, Y., Chen, X., & Yang, D. (2009b). Lyric-Based Song Emotion Detection with Affective Lexicon and Fuzzy Clustering Method. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR 2009)*.

Huron, D. (2006). *Sweet Anticipation: Music and the Psychology of Expectation*. Cambridge: The MIT Press, 1 edn.

Huysmans, J., Baesens, B., & Vanthienen, J. (2006). Using Rule Extraction to Improve the Comprehensibility of Predictive Models .

Izmirli, O. (1999). Using Spectral Flatness Based Feature for Audio Segmentation and Retrieval. Tech. rep., Department of Mathematics and Computer Science, Connectucut College.

Johnston, J. D. (1998). Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on Selected Areas in Communications*, *6*(2), 314–323.

Juslin, P. N. & Laukka, P. (2004). Expression, Perception, and Induction of Musical Emotions: A Review and a Questionnaire Study of Everyday Listening. *Journal of New Music Research*, *33*(3), 217–238.

Juslin, P. N. & Sloboda, J. A. (2001). *Music and Emotion: Theory and Research*. Oxford: Oxford University Press.

Juslin, P. N. & Västfjäll, D. (2008). Emotional responses to music: The need to consider underlying mechanisms. *Behavioral and Brain Sciences*, *31*(5).

Kedem, B. (1986). Spectral Analysis and Discrimination by Zero-Crossings. *Proc. of the IEEE*, *74*.

Kim, Y. E., Schmidt, E. M., Migneco, R., Morton, B. G., Richardson, P., Scott, J., Speck, J. A., & Urnbull, D. (2010). Music Emotion Recognition: a State of the Art Review. In J. S. Downie & R. C. Veltkamp (Eds.) *11th International Society for Music Information and Retrieval Conference*.

Kivy, P. (1989). *Sound Sentiment: An Essay on the Musical Emotions*. Temple University Press.

Kleinginna, P. R. & Kleinginna, A. M. (1981). A categorized list of motivation definitions, with a suggestion for a consensual definition. *Motivation and Emotion*, *5*(3), 263–291.

Koduri, G. K. & Indurkhya, B. (2010). A behavioral study of emotions in south indian classical music andits implications in music recommendation systems. In *Proceedings of the 2010 ACM workshop on Social, adaptive and personalized multimedia interaction and access*, SAPMIA '10, pp. 55–60. New York, NY, USA: ACM.

Koelsch, S., Fritz, T., Cramon, D. Y. V., Müller, K., & Friederici, A. D. (2006). Investigating emotion with music: an fMRI study. *Human Brain Mapping*, *27*(3), 239–250.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, *43*, 59–69.

Krumhansl, C. L. (1996). A perceptual analysis of Mozart's piano sonata K. 282: Segmentation, tension, and musical ideas. *Music Perception*, *13*, 401–432.

Krumhansl, C. L. (1997). An exploratory study of musical emotions and psychophysiology. *Canadian journal of experimental psychology*, *51*(4), 336–353.

Lartillot, O. & Toiviainen, P. (). Mir in matlab (ii): A toolbox for musical feature extraction from audio.

Laurier, C. & Herrera, P. (2007). Audio music mood classification using support vector machine. In *Proceedings of the 8th International Conference on Music Information Retrieval*. Vienna, Austria.

Laurier, C. & Herrera, P. (2009). *Automatic Detection of Emotion in Music: Interaction with Emotionally Sensitive Machines*, chap. 2, pp. 9–32. IGI Global.

Laurier, C., Meyers, O., Serrà, J., Blech, M., Herrera, P., & Serra, X. (2010). Indexing Music by Mood: Design and Integration of an Automatic Content-based Annotator. *Multimedia Tools and Applications*, *48*, 161–184.

Laurier, C., Sordo, M., Bozzon, A., Brambilla, M., & Fraternali, P. (2009a). Pharos: An Audiovisual Search Platform using Music Information Retrieval Techniques. In *Conference of the International Society for Music Information Research (ISMIR), Demo session*. Kobe, Japan.

Laurier, C., Sordo, M., & Herrera, P. (2009b). Mood Cloud 2.0: Music Mood Browsing based on Social Networks. In *International Society for Music Information Research Conference (ISMIR)*. Kobe, Japan.

Laurier, C., Sordo, M., Serrà, J., & Herrera, P. (2009c). Music Mood Representations from Social Tags. In *International Society for Music Information Retrieval (ISMIR) Conference*, pp. 381–386. Kobe, Japan.

Lazarus, R. S. (1991). *Emotion and Adaptation*. Oxford: Oxford University Press.

Le Cessie, S. & Van Houwelingen, J. C. (1992). Ridge Estimators in Logistic Regression. *Applied Statistics*, *41*(1), 191–201.

Lerdahl, F. (1996). Calculating Tonal Tension. *Music Perception*, *13*(3), 319–364.

Lerdahl, F. & Krumhansl, C. L. (2007). Modeling Tonal Tension. *Music Perception*, *24*(4), 329–366.

Levitin, D. J. (2006). *This Is Your Brain on Music: The Science of a Human Obsession*. Dutton Adult.

Levy, M. & Sandler, M. (2007). A Semantic Space for Music Derived from Social Tags. In *8th International Conference on Music Information Retrieval (ISMIR 2007)*.

Li, T. & Ogihara, M. (2003). Detecting emotion in music. In *Proceedings of the 4th International Conference on Music Information Retrieval*, pp. 239–240. Baltimore, MD, USA.

Li, T. & Ogihara, M. (2005). Music genre classification with taxonomy. pp. 197–200.

Lin, Y.-C., Yang, Y.-H., Chen, H. H., Liao, I.-B., & Ho, Y.-C. (2009). Exploiting genre for music emotion classification. In *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, pp. 618–621.

Liu, D., Lu, L., & Zhang, H. J. (2003). Automatic Mood Detection from Acoustic Music Data.

Logan, B. (2000). Mel Frequency Cepstral Coefficients for Music Modeling. In *Proceeding of the 1st International Symposium on Music Information Retrieval*. Plymouth, MA, USA.

Logan, B., Kositsky, A., & Moreno, P. (2004). Semantic Analysis of Song Lyrics. In *IN PROC IEEE INTL CONF ON MULTIMEDIA AND EXPO*, pp. 827–830.

Lu, L., Liu, D., & Zhang, H.-J. (2006). Automatic mood detection and tracking of music audio signals. *Audio, Speech, and Language Processing, IEEE Transactions on*, *14*(1), 5–18.

Mahedero, J. P. G., MartÍnez, A., Cano, P., Koppenberger, M., & Gouyon, F. (2005). Natural language processing of lyrics. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pp. 475–478. New York, NY, USA: ACM.

Mandel, M., Poliner, G., & Ellis, D. (2006). Support vector machine active learning for music retrieval. *Multimedia Systems*, *12*(1), 3–13.

Martens, D., Baesens, B., & Van Gestel, T. (2009). Decompositional Rule Extraction from Support Vector Machines by Active Learning. *Knowledge and Data Engineering, IEEE Transactions on*, *21*(2), 178–191.

Mayer, R., Neumayer, R., & Rauber, A. (2008a). Combination of Audio and Lyrics Features for Genre Classification in Digital Audio Collections. In *ACM Multimedia*.

Mayer, R., Neumayer, R., & Rauber, A. (2008b). Rhyme and style features for musical genre classification by song lyrics. In *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR 2008)*.

Mckay, C. (2010). Automatic Music Classification with jMIR.

Menon, V. & Levitin, D. J. (2005). The rewards of music listening: response and physiological connectivity of the mesolimbic system. *Neuroimage*, *28*(1), 175–184.

Mercer, J. (1909). Functions of positive and negative type and their connection with the theory of integral equations. *Philos. Trans. Roy. Soc. London*.

Meyer, L. B. (1956). *Emotion and Meaning in Music*. Chicago: University Of Chicago Press.

Mierswa, I. & Morik, K. (2005). Automatic Feature Extraction for Classifying Audio Data. *Machine Learning Journal*, *58*, 127–149.

Mitchell, T. (1997). *Machine Learning*. The McGraw-Hill Companies, Inc.

Neumayer, R. & Rauber, A. (2007). Integration of text and audio features for genre classification in music information retrieval. In *ECIR'07: Proceedings of the 29th European conference on IR research*, pp. 724–727. Berlin, Heidelberg: Springer-Verlag.

Nichols, E., Morris, D., Basu, S., & Raphael, C. (2009). Relationships between lyrics and melody in popular music. In *Proceedings of ISMIR 2009 10th International Conference on Music Information Retrieval*.

Nussbaum, C. O. (2007). *The Musical Representation: Meaning, Ontology, and Emotion*. Cambridge: The MIT Press, 1 edn.

Oatley, K. & Jenkins, J. M. (1996). *Understanding emotions*. Blackwell Publishers.

Orio, N. (2006). Music retrieval: a tutorial and review. *Found. Trends Inf. Retr.*, *1*(1), 1–96.

Pang, B. & Lee, L. (2008). *Opinion Mining and Sentiment Analysis*. Now Publishers Inc.

Panksepp, J. & Bernatzky, G. (2002). Emotional sounds and the brain: the neuro-affective foundations of musical appreciation. *Behavioural Processes*, *60*(2), 133–155.

Patel, A. D. (2007). *Music, Language, and the Brain*. Oxford: Oxford University Press, 1 edn.

Peeters, G. (2004). A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Tech. rep., IRCAM.

Peeters, G. (2008). A Generic Training and Classification System for MIREX08 Classification Tasks: Audio Music Mood, Audio Genre, Audio Artist and Audio Tag.

Peretz, I., Gagnon, L., & Bouchard, B. (1998). Music and emotion: perceptual determinants, immediacy, and isolation after brain damage. *Cognition*, *68*(2), 111–141.

Peretz, I., Gagnon, L., Hébert, S., & MacOir, J. (2004). Singing in the Brain: Insights from Cognitive Neuropsychology. *Music Perception*, *21*(3), 373–390.

Pike, A. (1972). A Phenomenological Analysis of Emotional Experience in Music. *Journal of Research in Music Education*, *20*(2), 262–267.

Pohle, T., Pampalk, E., & Widmer, G. (2005). Evaluation of Frequently Used Audio Features for Classification of Music into Perceptual Categories. In *Proceedings of the Fourth International Workshop on Content-Based Multimedia Indexing (CBMI'05*.

Ponte, J. M. & Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pp. 275–281. New York, NY, USA: ACM.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, *39*(6), 1161–1178.

Sacks, O. & Freeman, A. (1994). An Anthropologist on Mars. *Journal of Consciousness Studies*, *1*(2), 234–240.

Salton, G. (1971). *The SMART Retrieval System—Experiments in Automatic Document Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Saunders, J. (1996). Real-Time Discrimination of Broadcast Speech/Music. *Proc. ICASSP*, pp. 993–996.

Scherer, K. R. (1991). *Emotion expression in speech and music*, pp. 146–156. London: MacMillian.

Scherer, K. R. & Zentner, M. R. (2001). *Emotional Effects of Music: Production Rules*, pp. 361–392. Oxford: Oxford University Press.

Schubert, E. (1999). *Measurement and Time Series Analysis of Emotion in Music*. Ph.D. thesis, University of New South Wales.

Sesmero, J. (2008). Electronic dance music genre classification.

Sethares, W. A. (1998). *Tuning Timbre Spectrum Scale*. Springer, 1 edn.

Shi, Y.-Y., Zhu, X., Kim, H.-G., & Eom, K.-W. (2006). A Tempo Feature via Modulation Spectrum Analysis and its Application to Music Emotion Classification. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1085–1088. Toronto, Canada.

Skowronek, J., McKinney, M., & van de Par, S. (2007). A Demonstrator for Automatic Music Mood Estimation. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pp. 345–346. Vienna, Austria.

Sloboda, J. A. (2001). *Psychological perspectives on music and emotion*, chap. 4, pp. 71–105. Oxford: Oxford University Press.

Smith, J. O. & Abel, J. S. (1999). Bark and ERB bilinear transforms. *Speech and Audio Processing, IEEE Transactions on*, *7*(6), 697–708.

Sordo, M., Celma, O., Blech, M., & Guaus, E. (2008). The Quest for Musical Genres: Do the Experts and the Wisdom of Crowds Agree? In *Proceedings of the 9th International Conference on Music Information Retrieval*. Philadelphia, USA.

Sordo, M., Laurier, C., & Celma, O. (2007). Annotating Music Collections: How content-based similarity helps to propagate labels. In *Proceedings of the 8th International Conference on Music Information Retrieval*, pp. 531–534. Vienna, Austria.

Steinbeis, N., Koelsch, S., & Sloboda, J. A. (2006). The Role of Harmonic Expectancy Violations in Musical Emotions: Evidence from Subjective, Physiological, and Neural Responses. *J. Cogn. Neurosci.*, *18*(8), 1380–1393.

Thayer, R. E. (1989). *The biopsychology of mood and arousal*. Oxford: Oxford University Press.

Thayer, R. E. (1996). *The Origin of Everyday Moods: Managing Energy, Tension, and Stress*. Oxford: Oxford University Press.

Toiviainen, P. & Krumhansl, C. L. (2003). Measuring and modeling real-time responses to music: The dynamics of tonality induction. *Perception*, *32*, 741–766.

Tomkins, S. S. (1980). *Affect as amplification: some modifications in theory*. New York: Academic Press.

Trainor, L. J., Tsang, C. D., & Cheung, V. H. W. (2002). Preference for sensory consonance in 2- and 4-month-old infants. *Music Perception*, *20*(2), 187–194.

Tzanetakis, G. (2007). Marsyas-0.2: a case study in implementing music information retrieval systems. In Intelligent Music Information Systems. *Intelligent Music Information Systems*.

Tzanetakis, G. & Cook, P. (2002). Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, *10*(5).

van Zaanen, M. & Kanters, P. (2010). Automatic mood classification using tf*idf based on lyrics. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*.

Vanden & Hubert, M. (2005). Robust classification in high dimensions based on the SIMCA method. *Chemometrics and Intelligent Laboratory Systems*, *79*, 10–21.

Vieillard, S., Peretz, I., Gosselin, N., Khalfa, S., Gagnon, L., & Bouchard, B. (2008). Happy, sad, scary and peaceful musical excerpts for research on emotions. *Cognition & Emotion*, *22*(4), 720–752.

Wack, N. (2010). Essentia & Gaia: audio analysis and music matching C++ libraries developed by the Music Technology Group. http://mtg.upf.edu/technologies/essentia.

Wedin, L. (1972). A multidimensional study of perceptual-emotional qualities in music. *Scandinavian journal of psychology*, *13*(4), 241–257.

Wieczorkowska, A., Synak, P., Lewis, R., & Raś (2005). Extracting Emotions from Music Data. In M.-S. Hacid, N. V. Murray, Z. W. Raś, & S. Tsumoto (Eds.) *Foundations of Intelligent Systems*, *Lecture Notes in Computer Science*, vol. 3488, chap. 47, pp. 456–465. Berlin, Heidelberg: Springer-Verlag.

Wierzbicka, A. (1999). Emotions across languages and cultures diversity and universals.

Witten, I. H. & Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (The Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, 1st edn.

Wold, S. (1976). Pattern recognition by means of disjoint principal components models. *Pattern Recognition*, *8*(3), 127–139.

Xu, R. & Wunsch, D. (2008). *Clustering (IEEE Press Series on Computational Intelligence)*. Wiley-IEEE Press, illustrated edition edn.

Yang, D. & Lee, W. (2004). Disambiguating music emotion using software agents. In *Proc. Int. Conf. Music Information Retrieval*, pp. 52–58.

Yang, Y. H. & Chen, H. (2010). Ranking-Based Emotion Recognition for Music Organization and Retrieval. *IEEE Transactions on Audio, Speech, and Language Processing*.

Yang, Y.-H., Lin, Y.-C., Cheng, H.-T., Liao, I.-B., Ho, Y.-C., & Chen, H. (2008a). Toward Multi-modal Music Emotion Classification. In Y.-M. Huang, C. Xu, K.-S. Cheng, J.-F. Yang, M. Swamy, S. Li, & J.-W. Ding (Eds.) *Advances in Multimedia Information Processing - PCM 2008*, *Lecture Notes in Computer Science*, vol. 5353, chap. 8, pp. 70–79. Berlin, Heidelberg: Springer Berlin / Heidelberg.

Yang, Y. H., Lin, Y. C., Su, Y. F., & Chen, H. H. (2008b). A Regression Approach to Music Emotion Recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, *16*(2), 448–457.

Yang, Y. H., Liu, C. C., & Chen, H. H. (2006). Music emotion classification: a fuzzy approach. In *Proceedings of the 14th annual ACM international conference on Multimedia*, MULTIMEDIA '06, pp. 81–84. New York, NY, USA: ACM.

Zwicker, E. & Terhardt, E. (1980). Analytical expressions for critical-band rate and critical bandwidth as a function of frecuency. *J. Acoust. Soc. Am*, *68*(5), 1523–1525.

Appendix A: Demonstrations

Introduction

In this appendix, we present our work in devising demonstrations using the mood classification techniques detailed previously in this thesis. First of all, we explain how we integrated our mood classifier in the context of a search-engine project, funded by the European Union (PHAROS[2]). Then, we describe our two Mood Cloud demonstrations of real time prediction of mood, followed by another integration in a personalized music television prototype (MyMTV).

PHAROS: Mood annotation in a search engine

One of the most important application of our mood classifiers has been their integration into the PHAROS platform (Laurier et al. (2009a)). We developed a webservice delivering mood annotation (classification and probability values) for indexing music in an audiovisual search platform.

The PHAROS project

PHAROS was an Integrated Project funded by the European Union under the Information Society Technologies Programme (6th Framework Programme) with a strategic objective defined as "Search Engines for Audiovisual Content". PHAROS aimed to advance audiovisual search from a point-solution search engine paradigm to an integrated search platform paradigm. One of the main goals of this project was to define a new generation of search engine, developing a scalable and open search framework that lets users search, explore, discover, and analyze contextually relevant data. Part of the core technology included automatic annotation of content using integrated components of different kinds (visual classification, speech recognition, audio and music annotations, etc...). In our case, we implemented and integrated an automatic music mood annotator based on the work previously described in this thesis.

Integration of the mood annotator

As a search engine prototype[3], PHAROS uses automatic content annotation to index audiovisual content. However, there is a clear need to make the content

[2]Platform for searcHing of Audiovisual Resources across Online Spaces. http://www.pharos-audiovisual-search.eu, http://mtg.upf.edu/research/projects/pharos

[3]Unfortunately, no version of the PHAROS search engine is available since the project ended

analysis as efficient as possible (in terms of accuracy and time). To integrate our mood annotator into the platform, we first created an implementation in C++, based on the MTG feature extraction library Essentia (see Wack (2010)) to extract audio features together with the libsvm library for Support Vector Machines (Chang & Lin (2001)). The SVMs were trained with full ground truth datasets and optimal parameters. The representation format was an XML standard candidate defined during the project and based on a MPEG-7 derivative called DAVP[4]. We wrapped our implementation into a webservice compatible with the framework, which could be directly accessed by other modules of the PHAROS platform. Furthermore, exploiting the probability output of the SVM algorithm, we provided a confidence value for each mood classifier. This probability value has been used for ranking the results of a query by the annotation probability (for instance from the least to the most happy). This has also been employed to increase the precision of the system by showing only the results with high confidence values. Finally, it allowed to make hybrid annotators called "fusion annotators" mixing different annotators to get multimodal annotations. For instance, we implemented an audiovisual mood annotator, using both music and images of a video clip to predict its mood. We built this prototype merging our prediction with an image mood annotator by Dunker et al. (2008). We presented a demonstration of the results at ISMIR 2009 (Laurier et al. (2009a)).

The resulting annotator extracts audio features and predicts the music mood at a sufficient speed to index the content used in the project, with the same performance level than what was presented in the previous chapters (using the exact same ground truth). This annotator contributes to the overall system by allowing for a flexible and distributed usage. In our tests, using a cluster of 8 quad-core machines, we could annotate 1 million songs (using 30-seconds of each) in around 10 days. The mood annotation is used to filter automatically the content according to the needs of users and helps them to find the content they are looking for. This integrated technology can lead to an extensive set of new tools to interact with music, enabling users to find new pieces that are similar to a given one, providing recommendations of new pieces, automatically organizing and visualizing music collections, creating playlists or personalizing radio streams. Indeed, the commercial success of large music catalogs nowadays is based on the possibility of allowing people to find the music they want to hear.

User evaluation

In the context of the PHAROS project, user evaluations have been conducted with Orange Labs. The main goal of these evaluations was to assess the us-

[4]DAVP: Digital Audiovisual Profile, http://iiss039.joanneum.at/cms/index.php?id=119

ability of the PHAROS platform and, in particular, the utility of several annotations.

Protocol

26 subjects participated in the evaluation. They were from the general public, between 18 and 40 years old (27 in average), all of them self-declared eager music listeners and last.fm users. The content processed and annotated for this user evaluation was made of 2092 30-second music videos. After a presentation of the functionalities on site, the users were then directly using an online installation of the system accessible from their home. During 4 weeks, they could test it with several tasks they were asked to do every two days. The task related to our algorithm was to search for some music and to refine the query using a mood annotation. One query example could be to search for "music" in the whole system and then to refine the search with the content-based mood annotation "relaxed". They had to answer a questionnaire at the end of the study:

- "Do you find it interesting to use the mood annotation to refine a query for music?"
- "Do you find the "mood" annotation innovative?"
- "Does the use of the mood annotation correspond to your way of searching for audiovisual information?"

Results

As a general comment, there is a difficulty for users to understand directly a content-based annotation. Some effort and thinking has to be done to make it intuitive and transparent. For instance what does "sad=0.58" (music annotated sad with a confidence of 0.58) mean? Is it really sad? Is it very sad? The confidence, or probability, value of one annotation is quite relative to other instances and most of all to the training set. This can be used for ranking or filtering the results but should not be shown to the end-user directly. They would prefer nominal values like "very sad" or "not sad" for instance. Another important point seen when analyzing the comments from the users is the need to focus on precision. Especially in the context of a search engine, people will only concentrate on the first results and may not go to the second page (like the vast majority of google users). Instead, they are more likely to change their query. Several types of musical annotations were proposed to the user (genre, excitement, instrument, color, mode and key). From this list, mood was ranked as the second best in utility, just after musical genre (which is often given as metadata). Users had to rate on a scale from 0 to 10 their answer to several

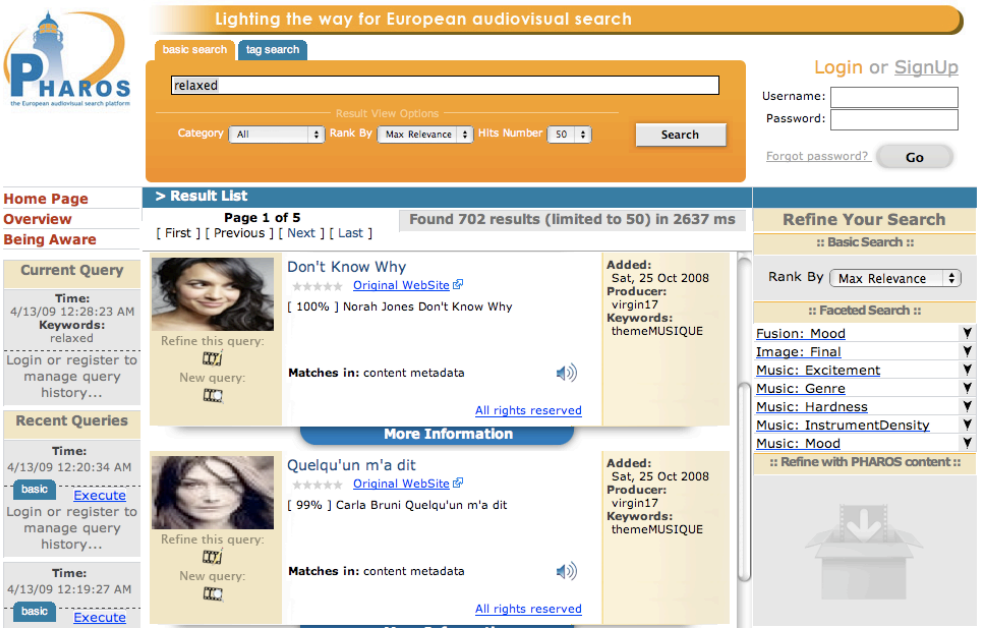

Figure 1: Screenshot of the PHAROS interface used for the user evaluation.

questions (0 would be "I strongly disagree" and 10 "I strongly agree"). We summarize here the answers to the questions related to the mood annotation:

- "Do you find it interesting to use the mood annotation to refine a query for music?" Users answered positively with a mean of 8.66, standard deviation of 1.85, showing a great interest to use this annotation.
- "Do you find the "mood" annotation innovative?" The mean of answers was also positive with 6.18 in average (standard deviation 3.81).
- "Does the use of the mood annotation correspond to your way of searching for audiovisual information?" Here users agreed with an average of 6.49 (standard deviation 3.47).

In all cases the mood annotation and its integration into the PHAROS platform was greatly appreciated and highly considered by users. They also rated it as the most innovative musical annotation overall. In Figure 1, we show a screenshot of the version of the PHAROS platform running the user evaluation. In this screenshot, the user is searching for "relaxed" music. They enter "relaxed" as a keyword and are browsing the musical results. The ones shown here were rated as "relaxed" (respectively 100% and 99%) thanks to the automatic music mood annotator we describe in this article.

Mood Cloud

Presentation

The objective of this work is twofold. Firstly, to make people understand how well an automatic model could work. Secondly, to verify empirically if predicting mood on shorter segments that the one used to train would make sense. Mood Cloud is therefore a demonstrator of automatic music mood prediction from audio content visualized in real time. While playing a song, we visualize the prediction probabilities of five mood categories : "happy", "sad", "aggressive", "relax" and "party". Each mood is represented by a colored bar graph with text, dynamically resized according to the mood probability. Consequently, each bar represents a mood model and its prediction. The resulting application is a dynamic visualization of the mood predictions, demonstrating the performance of current state of the art techniques in automatic mood classification. The "party" category, not mentioned previously in this document, was added even though it is rather situational than emotional. Party is not considered as a mood but as a scenario (a party) where you would want a particular kind of music, mainly upbeat and danceable. Nevertheless it has been created and trained using the same approach described in this thesis for the other mood categories. This visualization tool pre-computes the mood evolution by means of a supervised learning approach, using a feature set designed for this task and a SVM algorithm. To estimate the mood probabilities, we used SVM models trained on our ground truth data. The predictions are computed within windows of several seconds to show the evolution of the mood prediction. While playing a music, bar graphs representing each mood are resized according to the predicted probability value on the current audio segment. Empirically, a chunk size of 3 to 5 seconds gave relevant results while having a sufficient change frequency.

Technical details

The Mood Cloud demo is divided in two parts. The first part is the processing module (back-end) that extracts the features, classifies and outputs probabilities for each segments. It uses the libsvm library with SVM models precomputed on our ground truth. The processing module is made in Python and C++ and is cross-platform. The second part is the visualization module, created with Adobe Flash[5]. It can be run on any platform if an Internet browser with Flash player is installed. The interaction between both modules is achieved via XML.

[5] http://www.adobe.com/products/flash/

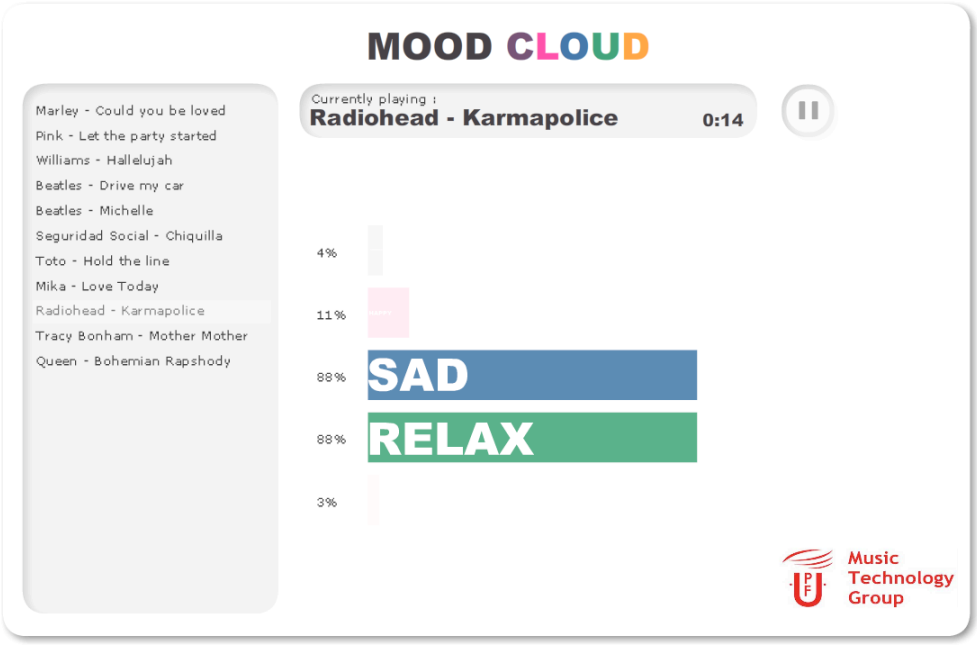

Figure 2: Screenshot of Mood Cloud for the song "Karmapolice" by Radiohead.

Conclusion

The Mood Cloud application provides an intuitive and understandable visualization of automatic mood classification algorithms. This tool helps to understand the potential and the limitations of such techniques. It would be interesting to integrate it as a plug-in for music players for desktop or mobile devices. No evaluation was made using this demo although it would be interesting to compare the time changes of the model predictions with human perception.

Mood Cloud 2.0

Presentation

Mood Cloud 2.0 is a demonstration that allows to visualize and browse music by mood. With the first version of Mood Cloud (see citeLaurier:MoodCloud), we could visualize at playing time the mood prediction of different Support Vector Machine models (one for each 'basic' mood). This helped to understand how accurate can the mood evolution be predicted. Mood Cloud 2.0 enables a new 2D visualization based social network data (see Laurier et al. (2009c) for more details) and adds retrieval features. In this representation, we can visualize one's collection, observe the mood evolution of a song in time, and

draw a path to make a playlist or retrieve a song based its time evolution. This 2D space is flexible, one can choose between different templates. the most interesting one probably being the representation extracted from social networks called semantic mood space (see Chapter 3 and Laurier et al. (2009c)). The 2D semantic mood space was obtained using Self-Organizing Maps on tag data from last.fm. Each song of one's collection is mapped into the semantic mood space using its tags. Other modes and representations are proposed. If the tags are not available, we can use the autotagger function, which automatically adds tag to the piece and so place it in the semantic space. This technique is also used to evaluate the mood evolution of one song dividing it in segments of a few seconds. Additionally, pre-computed audio mood models are available (the updated models from Mood Cloud 1.0), which are state-of-the-art mood classification algorithms. For these models, the 2D representation can be changed using different axis. We allow the user to change the two dimensions, selecting between the existing audio models in Mood Cloud 1.0 (happy, sad, aggressive, relax and party). One can visualize his collection in the agressive/sad or relaxed/happy spaces for instance. With both the autotagger and the mood models, any collection can be mapped and browsed into a 2D space. By analyzing the songs in windows of a few seconds and keeping the trace of the result, we can visualize, in the same space, the instantaneous mood and its evolution during the song.

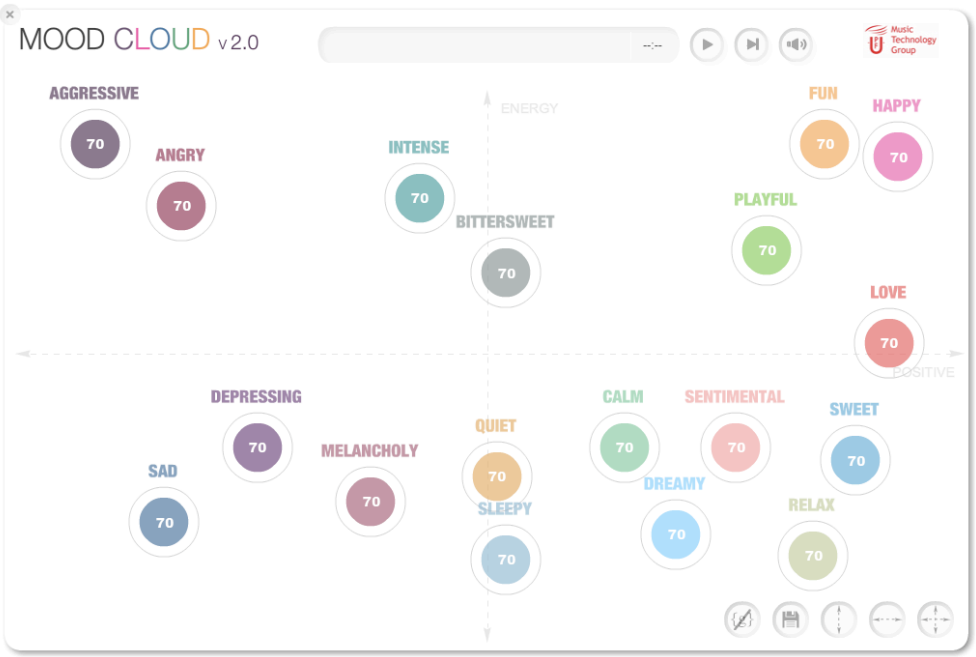

Figure 3: Screenshot of Mood Cloud 2.0 with the different tags in the 2D space.

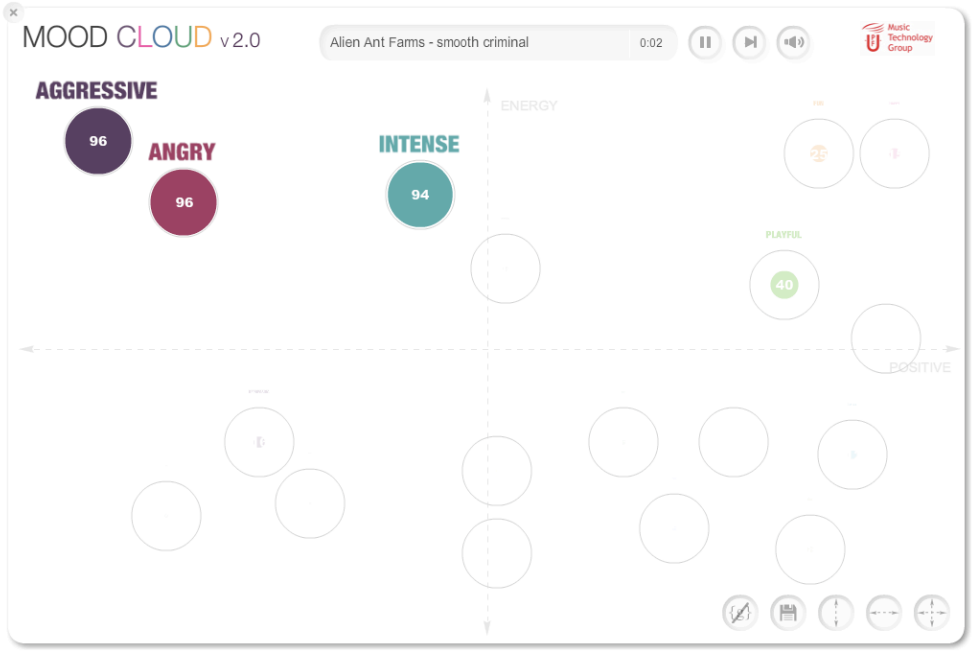

Figure 4: Screenshot of Mood Cloud 2.0 for the song "Smooth Criminal" by Alien Ant Farm.

Technical Details

This demo is coded in C++ and python for the back-end using libraries of the Music Technology Group (Essentia & Gaia). The analysis of the collection is stored in XML files. The front-end GUI is made with Macromedia Flash. As it is a Flash application, it can be either local or online. This demo was presented at ISMIR 2009 (See Laurier et al. (2009b)).

MyMTV

MyMTV has been developed with the objective of making full integrated application to demonstrate how would look a personalized and smart television, using both content-based and collaborative filtering techniques. MyMTV is an Interactive TV which adapts to one's habits, tastes and moods: A music recommendation channel, tailored to the user's tastes. Users create a personalized channel of music videos by selecting a song or an artist they like. The system identifies and log what music video the user is watching. Based on this information, the system builds a user profile to improve the quality of future recommendations. Both audio features and collaborative filtering are used for recommendation. Users can rate the songs and this information is employed to improve the recommendation quality. This prototype was developed during the EU ITEA project CANTATA [6]. Our mood models, presented in this thesis, are integrated into this prototype as mood annotations, which helps the user to find the content he would like and that fit to his mood.

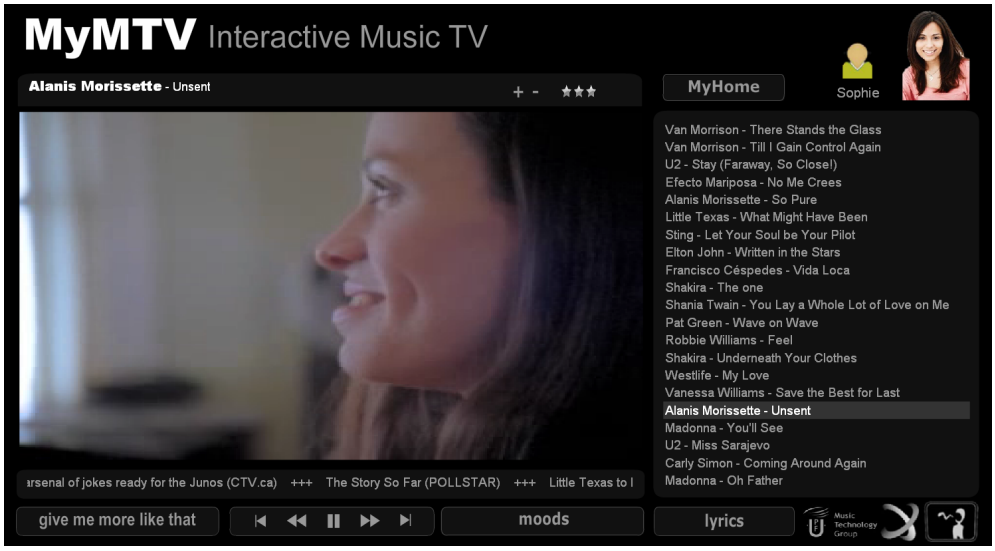

Figure 5: MyMTV Flash Interactive GUI

[6] CANTATA: Content Aware Networked systems Towards Advanced and Tailored Assistance. European Research Program sponsored via Eureka under the ITEA programme. http://www.itea-cantata.org/.

Appendix B: Lyrics differentiating mood categories

We list here the term that our Language Model Differences (LMD) model (presented in Chapter 5, Section 5.2.5) detects as important to discriminate between the mood, in each case the category against its complementary, for instance *happy* against *not happy*. These results are given when applying the methods on all our training data. We note that if some of these words are obviously informative others are artifacts that could be filtered out in future improvements of our technique.

angry
love, could, heart, death, control, die, dead, can, i'm, were, god, evil, hell, world, pain, blue, today, fate, then, need, tell, rain, baby, there, so, much, change, fast, trees, stars, sun, fucking, killing, things, lies, me, , lost, but, too, had, knew, seems, sweet, oh, come, was, say, beyond, days, mine, breathe, truth, sorry, walked, places, half, worry, though, singing, wind, someone, mind, how, hate, fuck, soon, sing, talk, soul, blood, free, path, eternal, scream, burns, til, power, knows, whole, choose, took, my, door, we, night, long, watch, stop, give, song, miles, yours, guess, kind, fly, hours, lovers, girls, both

relaxed
out, get, die, got, wanna, this, keep, try, made, lies, hell, for, going, rain, yeah, control, eyes, oh, , face, evil, fate, god, better, am, mind, our, you've, flow, alright, hatred, earth, fly, though, half, yourself, dead, fine, flesh, getting, death, forget, chorus, truth, years, free, skin, good, voices, silent, kiss, feel, four, spit, deadly, filled, fuck, cannot, killing, sight, step, eleven, pushing, disease, stars, worth, whole, used, love, a, he, mean, care, show, wonder, spell, lie, man, an, find, tell, about, hide, please, meet, little, comes, stop, ain't, on, each, morning, singing, hands, second, shining, kissing, strike, split, kingdom

sad
get, own, lonely, rest, coming, up, hurt, tu, hold, here, give, street, holding, forget, soon, she's, kiss, world, down, then, were, young, boy, fight, beyond, train, remember, en, seems, te, sometimes, le, que, should, man, live, wanna, repeat, watching, love, without, find, got, her, first, tomorrow, money, tonight, la, alive, mind, an, driving, listen, cuz, everybody's, x2, knows, ready, lips, toast, before, eyes, could, ground, put, looking, lies, hey, hit, set, back, right, away, blue, poison, we'd, ghosts, spinning, bitter, de, somehow, juntos, sign, mal, name, flowers, angel, estas, memories, shadow, isn't, para, sweet, y, un, beneath, empty, every, far

happy
out, got, live, well, again, chorus, yeah, gonna, oh, sleep, pain, getting, world, phone, dead, fine, you've, want, end, gone, can, looking, gotta, sky, tell, down, looks, em, bright, calls, two, done, yesterday, words, things, been, better, together, friends, keep, lost, best, verse, whole, repeat, but, up, something, worth, work, fit, fields, rock, living, everyday, longer, u, alright, pass, lie, sometimes, half, with, baby, we'll, then, hand, going, man, little, always, her, why, desire, dying, 2, knows, ask, once, own, fall, street, call, breathe, room, we'd, message, ten, wasted, twice, against, three, mother, shoulder, bury, presence, lies, mold, holding, none

Appendix C: Publications by the author and related to the dissertation

Journal Article

Laurier, C., Meyers O., Serrà J., Blech M., Herrera P., & Serra X. (2010). Indexing Music by Mood: Design and Integration of an Automatic Content-based Annotator. Multimedia Tools and Applications. 48(1), 161-184.

Book Chapter

Laurier, C., & Herrera P. (2009). Automatic Detection of Emotion in Music: Interaction with Emotionally Sensitive Machines. (Dr. Vallverdu, Dr. Casacuberta, Ed.).Handbook of Research on Synthetic Emotions and Sociable Robotics: New Applications in Affective Computing and Artificial Intelligence. 9-32.

Conference Proceedings

Laurier, C., Sordo M., & Herrera P. (2009). Mood Cloud 2.0: Music Mood Browsing based on Social Networks. International Society for Music Information Research Conference (ISMIR).

Bischoff, K., Firan C., Paiu R., Nejdl W., Laurier C., & Sordo M. (2009). Music Mood and Theme Classification a Hybrid Approach. Conference of the International Society for Music Information Retrieval (ISMIR).

Laurier, C., Sordo M., Serrà J., & Herrera P. (2009). Music Mood Representations from Social Tags. International Society for Music Information Retrieval (ISMIR) Conference. 381-386.

Mayor, O., Meyers O., Laurier C., & Koppenberger M. (2009). MyMTV: A personalized and interactive music channel. Conference of the International Society for Music Information Research (ISMIR), Demo session..

Laurier, C., Sordo M., Bozzon A., Brambilla M., & Fraternali P. (2009). Pharos: An Audiovisual Search Platform using Music Information Retrieval Techniques. Conference of the International Society for Music Information Research (ISMIR), Demo session.

Bozzon, A., Laurier C., Pihlajamaa O., Aichroth P., Nejdl W., Debald S., et al. (2009). Pharos: an audiovisual search platform. International Conference on Special Interest Group on Information Retrieval (SIGIR). 841.

Laurier, C., Lartillot O., Eerola T., & Toiviainen P. (2009). Exploring Relationships between Audio Features and Emotion in Music. ESCOM, Conference of European Society for the Cognitive Sciences of Music.

Laurier, C., Meyers O., Serrà J., Blech M., & Herrera P. (2009). Music Mood Annotator Design and Integration. 7th International Workshop on Content-Based Multimedia Indexing.

Laurier, C. (2009). Estimating Tonal Tension from Audio Content. Workshop on Music, Emotions, and Brain plasticity.

Hu, X., Downie S. J., Laurier C., Bay M., & Ehmann A. F. (2008). The 2007 MIREX Audio Mood Classification Task: Lessons Learned. 9th International Conference on Music Information Retrieval.

Laurier, C., Grivolla J., & Herrera P. (2008). Multimodal Music Mood Classification using Audio and Lyrics. International Conference on Machine Learning and Applications.

Laurier, C., & Herrera P. (2008). Mood Cloud : A Real-Time Music Mood Visualization Tool. CMMR, Computer Music Modeling and Retrieval.

Sordo, M., Laurier C., & Celma Ò. (2007). Annotating Music Collections How content-based similarity helps to propagate labels. 8th International Conference on Music Information Retrieval.

Laurier, C., & Herrera P. (2007). A computational approach to classify music by emotion. European Computing Conference.

Technical Reports

Wack, N., Laurier C., Meyers O., Marxer R., Bogdanov D., Serrà J., et al. (2010). Music classification using high-level models. Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.

Wack, N., Guaus E., Laurier C., Meyers O., Marxer R., Bogdanov D., et al. (2009). Music type groupers (MTG): generic music classification algorithms. Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.

Laurier, C., & Herrera P. (2007). Audio music mood classification using support vector machine. Music Information Retrieval Evaluation eXchange (MIREX) extended abstract.