POLITECNICO DI MILANO
Facoltà di Ingegneria dell'Informazione
Corso di Laurea in Ingegneria Informatica
Dipartimento di Elettronica, Informazione e Bioingegneria

A music search engine based on a contextual-related semantic model

Supervisor: Prof. Augusto Sarti
Assistant supervisor: Dr. Massimiliano Zanoni

Master graduation thesis by:
Alessandro Gallo, ID 779771

Academic Year 2012-2013

POLITECNICO DI MILANO
Facoltà di Ingegneria dell'Informazione
Corso di Laurea in Ingegneria Informatica
Dipartimento di Elettronica, Informazione e Bioingegneria

Motore di ricerca di brani musicali basato su un modello semantico contestuale

Relatore: Prof. Augusto Sarti
Correlatore: Dr. Massimiliano Zanoni

Tesi di Laurea Magistrale di:
Alessandro Gallo, matricola 779771

Anno Accademico 2012-2013

Abstract

During the past years, the advent of digital audio content has drastically increased the size of available music collections. Music streaming services provide a huge amount of music content to users, much more than they can concretely listen to in an entire lifetime. Classical meta-information, such as the artist and the title of songs, have been used for years. Today they are not enough to navigate such vast collections. Therefore, it is important to develop specific approaches that allow high-level music content description.

Music Information Retrieval (MIR) is the research field that deals with the retrieval of useful information from music content. Information can provide different levels of abstraction, from a higher level to a lower level. In this work we propose an approach for music high-level description and music retrieval, that we named *Contextual-related semantic model*. Classical semantic representation models such as ontologies only provide categorical approaches for defining relations (e.g. happy is synonym for joyful, happy is antonym for sad). On the other hand, actual dimensional description models map on a unique semantic space also concepts that are not in a semantic relation. Our method defines different semantic contexts and dimensional semantic relations between music descriptors belonging to the same context.

Our model has been integrated in *Janas*[1], a music search engine based on semantic textual queries. In order to test the scalability of our model, we implemented an automatic content-based method to expand the dataset.

The retrieval performances of our model have been compared with two other approaches: the one originally used by *Janas*, that combines emotional and non-emotional description of music, and the Latent Semantic Indexing approach [2], a very common model for music recommendation applications. The system has been tested by 30 subjects. The obtained results are promising and our *Contextual-related semantic model* outperformed the other approaches.

Sommario

Negli ultimi anni l'introduzione di contenuti audio digitali ha cambiato drasticamente le dimensioni delle librerie musicali. Diversi servizi di streaming musicale forniscono enormi quantità di contenuti musicali all'utente, molto più grandi di quanto potrebbe realmente ascoltare nell'arco della sua vita. In ambito musicale per anni sono state utilizzate delle meta-informazioni classiche, come l'artista o il titolo di una canzone. Oggi tutto ciò non è più abbastanza per navigare librerie così vaste. E' importante quindi sviluppare degli approcci specifici che consentano una descrizione di alto livello del contenuto musicale.

Il *Music Information Retrieval* (MIR) è l'ambito di ricerca si occupa di recuperare informazioni utili a partire dal contenuto musicale. In questa tesi proponiamo un approccio per la descrizione musicale di alto livello, che abbiamo chiamato *Contextual-related semantic model.*

I modelli di rappresentazione classici, come ad esempio le ontologie, forniscono solamente un approccio categorico per definire delle relazioni semantiche (e.g. contento è sinonimo di felice, contento è contrario di triste). D'altro canto, i modelli di rappresentazione di tipo dimensionale mappano su un unico piano semantico anche concetti che non sono in relazione semantica fra loro. Il nostro metodo definisce dei contesti semantici e delle relazioni semantiche dimensionali tra descrittori musicali che appartengono allo stesso contesto.

Il nostro modello è stato integrato in *Janas*[1], un motore di ricerca basato su query semantiche testuali. Inoltre, abbiamo implementato un metodo content-based automatico per espandere il dataset e per verificare la scalabità del nostro modello.

Le prestazioni del nostro modello sono state confrontate con quelle di due altri approcci: quello originariamente utilizzato da *Janas*, che combina una descrizione emotiva con una descrizione non emotiva, ed un approccio di tipo Latent Semantic Indexing [2], un modello molto comune per applicazioni di raccomandazione musicale. Il sistema è stato testato da 30 soggetti. I

risultati ottenuti sono promettenti e il nostro *Contextual-related semantic model* ha ottenuto prestazioni migliori rispetto agli altri approcci.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Over the centuries of human history, music has always been present in the society and it is an important constituent of everyday life. Music is rich in content and expressivity, and when a person engages with it, different mental processes are involved. We especially enjoy this form of art for its ability to induce emotions. In fact, composers and interpreters explicitly declare their goal to communicate their sentiments and convey certain feelings through their music.

During the last two decades the introduction of digital audio formats and the advent of the Internet allowed the distribution of enormous amount of music items. In the initial stage of the digital music revolution, peer-to-peer networks allowed the exchange of music files, then online music stores such as *iTunes*[1] started to offer downloads and other services like internet radio. Thereafter, the advent of the Web 2.0 encouraged the creation of online communities and the simplification of the interaction between musicians and listeners, greatly facilitating the distribution of music content. Nowadays, the availability of music streaming services such as *Deezer*[2] and *Spotify*[3] allows to easily access a huge amount of music content, like have been never happened before in the human history.

These phenomena transformed the experience of the listening to music. Nevertheless, it is difficult to orient in massive collections of digital contents. Scientific community and music industry are working to build new systems that can help in organizing, recommend, browse and retrieve music. For their realization it is crucial to figure out how to effectively represent the music content. Some meta-information such as the artist, the track title or

[1] iTunes, `http://www.apple.com/itunes/`
[2] Deezer, `http://www.deezer.com/`
[3] Spotify, `https://www.spotify.com/`

the release year have been used for decades in the music industry in order to organize music. Unfortunately, these aspects do not properly describe the music content, thus they are not relevant in order to reach and listen to music without knowing any prior information about it. On the other hand, it is important to investigate how users understand and describe music contents. For example, people may want to search music according to a specific mood (e.g. calm, fun) [6]. They could even be interested in exploiting other non-emotional elements, such as timbral characteristics (e.g. smooth, harsh) or rhythmic cues (e.g. flowing, fast) of the music piece. In order to bridge the semantic gap between the music content and the user description it is necessary a collaboration among different research areas such as signal processing, statistical modeling, machine learning, neuro-science, music cognition and musicology [7].

Music Information Retrieval (MIR) is an emerging interdisciplinary research field that investigates the possibility to automatically understand, organize and retrieve music by analyzing the information that music itself provides. Music information can be described hierarchically from a lower level of abstraction, related to audio content, to a higher level of abstraction, related to the human perception of music. These elements are generally referred as features or descriptors. *Low-level features* (LLF) are content-based descriptors directly extracted from the audio signal. They provide an objective description by measuring some energetic and spectral characteristics of a sound, but they lack of semantics. *Mid-level features* (MLF) introduce a first level of semantics by combining LLF with musicological knowledge. They refer to structural music components such as melody, tempo and harmony. *High-level features* (HLF) bring a higher level of abstraction **??**, making them easily comprehensible to people. They describe cognitive aspects of music, such as the emotion perception related to a music piece or the genre.

The first MIR systems and the most of commercial systems today are based on a *context-based* approach, in which high-level and mid-level features are generally annotated by hand. Unfortunately, this type of annotation is oftentimes unable to adequately capture a useful description of a musical content, due to the constantly growing amount of available music and the high subjectivity of annotations. In the last years, aside the context-based approach some *content-based* paradigms have been emerging. Content-based approaches extracts information from the audio signal. The majority of conent-based systems use LLF. However, given its low level of abstraction, this method tends to produce semantically poor descriptors. Nowadays, some MIR systems combines context- and content-based approaches based on LLF, MLF and HLF in order to obtain a semantically meaningful description of music content. In Chapter 2 we provide a review of applications based on both context- and content-based approaches.

In this thesis we particularly focus on high-level descriptors. There are two possible approaches for high-level music description: *categorical* and *dimensional*. The categorical approach assumes that music can be described by a limited number of universal descriptors. For example, categorical descriptors can represent emotional states (e.g. sweet, joyful, anxious, etc.), musical genre (e.g. Rock, Pop, Jazz, etc.) or structural factors (e.g. hard, dynamic, slow, etc.). This assumption is intuitive but at the same time it lacks in expressiveness. In fact, this approach is not able to quantify the pertinence of a term to the music content. The dimensional approach copes with this problem by defining music descriptors on a continuous domain. For example, in the dimensional approach that consider a weight scale from 0 to 1, a song can be associated to the descriptor *0.8 happy*.

Emotion perception is one of the most salient features that human beings experience in every moment of their lives and its relation with music has always fascinated people. *Music Emotion Recognition* (MER) is the discipline of MIR that investigates the possibility to automatically conceptualize and model emotions perceived from music. It is very complex to computationally determine the affective content of a music piece. In fact, the perception and interpretation of emotions related to a song is strictly personal, hence the semantic descriptions of emotions are highly subjective. In the specific case of emotion-related descriptors, the most celebrated dimensional model is the Valence-Arousal (V-A) two-dimensional space [8]. It is based on the idea that all emotional states can be expressed through two descriptors: Valence, related to the degree of pleasantness, and Arousal, concerning to the energy of the emotion. For the purposes of MER, songs can be mapped in the V-A space as points corresponding to the emotion perceived from the music content. It is interesting to notice that in the last few years, emotion-based music retrieval has received increasing attention in both academia and industry applications. For example, *Stereomood*[4] is a music streaming service that generates music playlist tailored to the user's mood.

Since the high-level description paradigm is increasingly adopted in several applications, it is reasonable to exploit a high-level interactions between users and systems. In the last few years new technological paradigms are emerging in this field. In particular, many applications allow people to express their requests as most intuitively as possible. Natural Language

[4]Stereomood, `https://www.stereomood.com/`

Processing (NLP) is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human language. An example of NLP applications is *Watson*[5], a computer system developed by IBM that is capable of answering questions posed in natural language.

The purpose of this thesis is to define a music search engine that is content-based, that uses an interaction scheme based on a high-level of abstraction, and that defines HLF in a dimensional space. Similar approaches are presented in the literature. The most common approach for multimedia indexing and retrieval is the *Latent Semantic Indexing*[2]. However, this approach does not consider semantic relation between descriptors defined by humans, since they are estimated by analyzing their co-occurences in music annotation. In [9] the authors built a music search engine that considers emotional descriptors modeled in the V-A plane by considering the ANEW dataset [10], and bipolar non-emotional descriptors. This approach suffers from some drawbacks. In fact, the model maps on a unique semantic space also concepts that are not in a semantic relation.

The final goal consists in overcoming this issue. We present an innovative model for semantic description of music, that we named *Contextual-related Semantic Model*. It makes use of a music-specific approach that defines three musical context for music description: *a*) perceived emotion, *b*) timbre description and *c*) dynamicity. We defined a list of 40 common adjectives used for describing music facets and then we assigned them to these contexts through a survey. Furthermore, we built a vector space model by estimating the semantic similarity between the terms. The system describes music with high-level features and annotates music pieces with a content-based approach, making it interactive and flexible to high amount of digital items. We address the issue of music searching with our content-based approach that considers natural language queries. We compare the performances of our model with the performances of other two semantic description approaches, a common co-occurrences method and a the approach proposed in [9].

The thesis is organized in 6 chapters. In Chapter 2 we present an overview of the state of the art for Music Information Retrieval and Music Emotion Recognition. Chapter 3 provides the theoretical background needed for the implementation of our project, including machine learning, information retrieval, audio features, emotion models and natural language processing. In Chapter 4 we describe the implementation details of our model and its inte-

[5]IBM Watson, `http://www.ibmwatson.com/`

gration in a search engine. Experimental results are presented in Chapter 5. In Chapter 6 we define our conclusions and we analyze future applications.

Chapter 2

State of the Art

In this chapter we review the state of the art for Music Information Retrieval research field. We first show how digital music content is semantically described and we provide a list of application for retrieving and recommending music that make use of high-level descriptors. In the second part we describe Music Emotion Recognition, introducing models and applications developed for automatically organize and retrieve music by their affective content.

2.1 Music Description

During the analog era, music producers and distributors used meta-information such artist, title and genre meta-information for organizing their recording catalogs. Nowadays, music content has increasingly become digital and personal collections have grown enormously. In this scenario, users may want to retrieve and discover music without having any prior information about it, thus new methods for music description are needed. High-level description of music is a feasible solution, since it carries a great semantic significance for human listeners. On the other hand, high-level description introduces a high degree of subjectivity. An experiment on a heterogeneous subset of population [11] demonstrated that the description of a music piece is strongly influenced by demographic and musical background of the listener.

High-level descriptors can be used for music organizing, retrieving or browsing. In the following paragraphs we review three paradigms used for music description both from a theoretical and an applicative point of view: *a*) Social tagging, *b*) the Semantic Web and *c*) Ontologies.

2.1.1 Social Tagging

In the last decade the Internet has evolved in the so-called Web 2.0. A prominent characteristics of Web 2.0 is the diffusion of social platforms where users contribute to build communities interested on a specific topic. These communities permit the creation and exchange of member-generated content. One of their main focuses is *social tagging*, a practice that consists in collaboratively categorizing digital items by annotating them with free-text labels called tags. Unlike traditional keyword systems, where terms are selected from a vocabulary, tags are words that describe relevant facets of music with no restrictions on their make up. For example, the music track *Born to Run* by Bruce Springsteen can be described as a *classic rock* song, released in the 1970*s*, performed by a *male vocalist*, which may be perceived as *happy*. A social tagging platform could represent this item with a set of tags T:

$$T = \{classic\ rock,\ 1970s,\ male\ vocalist,\ happy\} \tag{2.1}$$

The set of tags defined by multiple users constitutes a *Folksonomy*[12] (a portmanteau of the words folk and taxonomy). The collaboration between users permits to ponder different opinions and reflect the overall perception of a resource, building what has been defined as the *wisdom of crowd* [13]. The incentives and motivations that encourage people to tag resources are disparate [14, 15], such as the facilitation of personal retrieval, the discovery of new similar resources, the sharing of personal tastes and opinions, and the contribution to the community knowledge. Since the generation of tags is uncontrolled, they are affected by various drawbacks, such as irrelevance, noisiness, bias to a personal representation of the items, malicious generation, usage of synonym terms [16].

One of the main services for music tagging is Last.fm[1]. This popular platform is used by more than 50 million of users that have built an unstructured vocabulary of free-text tags for annotating millions of songs. Last.fm provide useful information of how people describe music content. Various researchers have used annotations extracted from this social tagging platforms for building music browsing and retrieving applications. For example, in [17] the authors developed a mobile music player application for browsing music by tags (such as genre, years or other descriptors).

In social tagging platforms it is common that songs by few highly popular artists receive much more annotations than songs by the rest of artists (*long tail distribution*). This leads to recommendation and retrieving issues, since

[1] Last.fm, `http://www.last.fm/`

popular songs are more likely to be retrieved. Celma [18] developed a recommendation system that explores the music popularity curve by exploiting similarity on tags and audio content.

Social tags require manual annotation. This process is not scalable for large music collections. Different techniques have been proposed in order to automatically annotate music (*autotagging*). For example, Turnbull et al.[19] developed a content-based system that annotates novel tracks with semantically meaningful words. The system can be used for retrieving relevant tracks from a database of unlabeled audio content, given a text-based query. Autotagging also addresses the long tail distribution issue, since it annotate popular songs as well as unpopular ones.

A folksonomy does not define any semantic relation between tags. This implies that, given a user request, a folksonomy-based system is not able to retrieve songs that are annotated with tags with similar semantics. Latent Semantic Indexing (LSI) is a popular technique that estimates interconnections between tags by analyzing their co-occurrences. Several studies have been undertaken in this direction[20, 21, 22, 23]. Furthermore, in [24] the authors compared a folksonomy-based LSI search engine with classical search engines and they observed similar performances. In this work, we compare our model with a solution based on LSI.

Nevertheless, LSI does not exploit the actual tags semantics. In the next sections we introduce the Semantic Web and ontologies, two approaches that are used to model the semantics of high-level description.

2.1.2 Semantic Web

The Semantic Web, sometimes referred also as Linked Data Web, represents the next major evolution of Web 2.0. It aims to provide a common framework that allows data to be shared and reused across application, enterprise and community boundaries [25]. Berners-Lee introduced this concept in [26], outlining the possibility to create a web of data that can be processed by machines. The basic goal of Semantic Web is to publish structured data in order for them to be easily used and combined with other similar data. It consists primarily of three technical standards:

- the Resource Description Framework (RDF) that specifies how to store and represent information

- SPARQL (SPARQL Protocol and RDF Query Language) that indicates how to query data across various systems

- the Web Ontology Language (OWL), a knowledge representation language that enable to define concepts

Exploiting Semantic Web for music description is very appealing, in fact having a common, structured, interlinked format to describe all the web knowledge about music could facilitate the gathering of information: a distributed and democratic knowledge environment can act as a data hub for many music-related applications and data sources.

Since still few platforms have switched to Semantic Web, this powerful idea has remained largely unrealized. Some music retrieval systems that are based on this paradigm have been implemented. For example, in [27, 28] the authors combined Semantic Web data with users' listening habits extracted from social media to create a music recommendation system. Other researchers attempted to exploit information from *DBPedia* [2][29, 30], a project aiming to extract semantically structured content from Wikipedia.

2.1.3 Ontologies

In computer science and information science, an ontology is a structural framework for organizing information and representing knowledge as a set of concepts. It also provides a shared vocabulary in order to denote the types, properties and relations of those concepts [31].

Various ontology languages have been developed in the last decades. *WordNet* [32] is a popular lexical database for English language that groups words into sets of synonyms and records multiple semantic relations between them. Despite it was initially developed as a database, it can be interpreted and used as a lexical ontology for knowledge representation [32].

Exploiting the properties of an ontology could be very useful in order to enrich the semantic description of items, in particular by analyzing the relations between conceptual categories. In the musical field, the most popular ontology is the *Music Ontology*[3], that provides a model for publishing structured music-related data on the Semantic Web. Other researchers defined other music-based ontologies for music recommendation [33, 34, 35].

Like ontologies, our model represents concepts and their relations but at the same time it also provides a numeric degree of semantic relatedness. Further details of our model are presented in Chapter 4.

[2]DBPedia, `http://dbpedia.org/About`
[3]Music Ontology, `http://musicontology.com/`

2.1.4 Music Semantic Retrieval

Classical music retrieval systems are based on a keyword search. In the last few years, various systems have been proposed in order to exploit the semantic description of the user request [36].

One of the first attempts was implemented by Slaney more than a decade ago [37], in which he developed a retrieval model based on a probabilistic representation of both acoustic features and semantic description. In [38] the authors created a system of Gaussian Mixture Models of tracks for music retrieval based on semantic description queries. In order to overcome the lack of music collection semantically labeled, they collected annotations for 500 tracks for capturing semantic association between music and words. This dataset, named CAL500, is currently available online[4]. A more recent attempt has been made in [9], where the authors developed a search engine based on semantic query. They defined emotional descriptors, mapped in the V-A plane (see section 3.4) and non-emotional descriptors, defined in a bipolar way. A natural language processing tool parses the user semantic query and the search engines retrieves music content with a similar description.

The model implemented in this work partially makes use of the system proposed in [9], and we aim at comparing its performances when different semantic models are considered.

2.2 Music Emotion Recognition

The relationship between music and emotions has been studied by psychologists for decades, well before the widespread availability of music recording. The research problems faced by psychologists include whether the everyday emotions are the same as emotions that are perceived in music, whether music represents or induces emotions, how musical, personal and situational factors affect emotion perception, and how we should conceptualize music emotion. From a psychological perspective, emotions are often divided into three categories: *expressed emotions*, *perceived emotions* and *felt emotions*[4]. Expressed emotions are referred to the emotions that the composer and the performer try to express to the listener, while perceived and felt emotions refer to the affective response of the listener. In particular, perceived emotion refers to the emotion expressed in music, while felt emotion is related to the individual emotional response. Felt emotions are especially

[4]CAL500 Dataset, `http://cosmal.ucsd.edu/cal/`

complicated to interpret because they depend on an interplay between musical, personal and situational factors. For this reason, MER has mainly focused on perceived emotions.

Engineering point of view of the problem dates back only to the 2000s [4]. We aim at developing a computational model of music emotion and facilitating emotion-based music retrieval and organization.

In this chapter *a*) we analyze how people describe emotions, *b*) we discuss different computational approaches that have been proposed in order to qualify and quantify emotions related to music, and finally *c*) we show various retrieval applications based on this paradigm.

2.2.1 Emotion Description

In the study of emotion conceptualization, researchers oftentimes utilize people's verbal reports of emotion responses. This approach suffers from the imperfect relationship between emotions and the affective terms that denote emotions, introducing a bias connected to the way in which people describe and communicate their feelings. This issue is called ambiguity, or fuzziness, and it is a characteristic of natural language categories in general, with a specific highlight whit emotions. As claimed by J.A. Russell[8],

> "*a human being usually is able to recognize emotional state but has difficulties with its proper defining*".

In [39] the authors developed a communicative theory of emotions from a cognitive point of view. They assumed that emotions have a two-fold communicative function: externally, amongst members of the species, and internally, within the brain, in order to bypass complex inferences. Their theory defines a small set of terms related to basic signals that can set up characteristic emotional modes within the organism, roughly corresponding to *happiness*, *sadness*, *fear*, *anger* and *disgust*. These basic emotions have no internal semantics, since they cannot be analyzed into anything more basic. They assume that each emotional signal is associated with a specific physiological pattern, implying that the organism is prepared to act in certain ways and to communicate emotional signals to others. Their theory considers that the mental architecture consists in a hierarchy of separate modules processing in parallel: an emotion can be set up by a cognitive evaluation occurring at any level in this hierarchy; in particular, each module is connected to one of the basic modes. From this theory, basic emotion words cannot be analyzed semantically because they denote primitive subjective experiences (they are experienced without the experience knowing their cause). All other words associated to complex emotions have a highly specific propositional content that is strictly related to their experience, thus they are found combining a basic term in a context that captures this content. The theory specifies that the most important concepts about semantic of emotional words are the intensity, the emotion relations, the distinction between caused and causatives emotions, the emotional goal and the intrinsic composition of complex emotions.

Moreover, the semantic structure of emotion terms in different languages appears to be universal. A research compared English, Chinese and Japanese speaking subjects and found that they share similar semantic structure with respect to fifteen common emotion terms [40].

2.2.2 Emotion Computational Model

Automatic emotion recognition in music needs a computational model in order to represent emotion data. Two types of approaches have been proposed, *a*) the categorical description and *b*) the parametric model. In the next sections we describe their characteristics.

2.2.2.1 Categorical Representation

Categorical representation considers that humans experience emotions as limited universal categories. Nevertheless, different researchers have come up with different sets of basic emotions.

One of the first researches in this field was undertaken by Hevner in 1936. She initially used 66 adjectives related to affective description of music, which were arranged into eight clusters [41] (figure 2.1). In a more recent study, Zenter et al. [42] defined a set of 146 terms for representing moods, founding that their interpretation varies between genres of music.

The Music Information Research Evaluation eXchange (MIREX), a community based framework for formally evaluating MIR systems, uses the five mood clusters shown in table 2.1 in order to evaluate automatic music mood classification algorithms [43].

Hobbs and Gordon described a process for defining a list of most frequent words about cognition and emotion from a computational linguistic point of view [44]. They considered *WordNet*, an ontology that contains tens of thousands of synsets referring to highly specific categories, from which they developed a lexicon of emotions, further divided in 33 categories. Another attempt to computationally map emotions to their linguistic expressions starting from *WordNet* was undertaken by Strapparava and Valitutti [45].

Merry
Joyous
Gay
Happy
Cheerful
Bright

Exhilarated
Soaring
Triumphant
Dramatic
Passionate
Sensational
Agitated
Exciting
Impetuous
Restless

Humorous
Playful
Whimsical
Fanciful
Quaint
Sprightly
Delicate
Light
Graceful

Vigorous
Robust
Emphatic
Martial
Ponderous
Majestic
Exalting

Lyrical
Leisurely
Satisfying
Serene
Tranquil
Quiet
Soothing

Spiritual
Lofty
Awe-inspiring
Dignified
Sacred
Solemn
Sober
Serious

Dreamy
Yielding
Tender
Sentimental
Longing
Yearning
Pleading
Plaintive

Pathetic
Doleful
Sad
Mournful
Tragic
Melancholy
Frustrated
Depressing
Gloomy
Heavy
Dark

Figure 2.1: Hevner Adjective Clusters

Cluster	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 2.1: Mood clusters and adjectives used in the MIREX Audio Mood Classification task

Using information coming from the lexicon and the semantic relations between synsets, they developed a linguistic resource for lexical representation of affective knowledge named *WordNet-Affect*.

However, the categorical approach uses a limited emotion vocabulary. Furthermore, these approach does not consider the intensity of emotions and the relations among emotion terms. These limitations are overcome by parametric approaches.

2.2.2.2 Parametric Representation

While the categorical approach focuses mainly on the characteristics that distinguish emotions from one another, parametric models assume that emotions can be represented effectively with a multi-dimensional metric.

Russell proposed to organize emotion descriptors by means of low-dimensional models [8]. In his work he introduced the two-dimensional Valence-Arousal (V-A) space, where emotions are represented as points on a plane with two independent axes: Arousal, that represent the intensity of an emotion, and Valence, that indicates an evaluation of polarity (ranging from positive to negative emotions). In section 3.4 we provide a more detailed overview of this approach.

Other researches asserted that three dimensions are needed to describe emotions [46, 47], but there is no agreement about the semantic nature of the third component, which has been defined as tension, kinetics and dominance. Fontaine et al. [48] proposed a four-dimensional space that includes: evaluation-pleasantness, potency-control, activation-arousal and unpredictability-surprise. Nevertheless, additional dimensions increase the cognitive load for annotating emotions.

Bradley et al. collected the Affective Norms for English Words (ANEW), which consists of a set of 1034 words annotated with values of pleasure (valence), arousal and dominance (dominant/submissive nature of the emotion). [10].

2.2.3 Implementations

In the MER field several researches have been undertaken in order to recognize the emotions perceived from a music piece and different methods have been proposed, both from a context-based and a content-based perspective.

Context-based MER applications Various attempts have been made in order to obtain the affective description of music content with a context-based approach. For example, the online music guide AllMusic [5] used professionals to annotate their music database with high-quality emotion tags. Since manual labeling is a time consuming task, a recent approach consists in using collaborative online games to collect affective annotations, for example *MajorMiner* [49], *Listen Game* [50] and *TagATune* [51]. Other researchers collected music mood annotations by exploiting social tagging platforms [52, 53, 54].

Context-based MER systems are not scalable, thus a content-based approach for emotion recognition is preferable.

Content-based MER applications Several music emotion recognition applications are based on a content-based approach. However, the relationship between acoustic facets and emotion perception of a listener is still far from well understood [55]. The performance of conventional methods that exploit only the low-level audio features seems to have reached a limit. The MIREX audio mood classification task is a contest for music emotion recognition algorithms that aims to classify emotions in five clusters: passionate, rollicking, literate, humorous and aggressive. Despite various low-level audio features and their combinations have been used, the best classification systems of 2013[6] obtained an accuracy of 69%.

In order to overcome this issue, other researchers exploited content-based methods that consider also high-level features. For example in [56, 57, 58, 59] the authors developed different systems employing both audio features and lyrics analysis, while Bischoff et al.[52] combined social tag information from Last.fm and audio content-based analysis for mood classification.
In this work we used a content-based approach on both emotional-related and non emotional-related descriptors, and we consider a computational model for defining the semantic relations among descriptors.

[5] AllMusic, `http://www.allmusic.com/`

[6] MIREX 2013 Mood Classification Results, `http://www.music-ir.org/nema_out/mirex2013/results/act/mood_report/index.html`

Chapter 3

Theoretical Background

In this chapter we present the theoretical background needed for the development of our work. We analyze the machine learning tools used in the development of our system, then we provide an overview on information retrieval, audio features, music emotion models and natural language processing.

3.1 Machine Learning Systems

Machine Learning is a research field that aims at developing systems capable of acquiring and integrating the knowledge automatically. The capability of the systems to learn from experience while looking for patterns in the data allows continuous adjustments in order to self-improve and thereby exhibit efficiency and effectiveness. There is a wide variety of machine learning tasks and successful applications, such as optical character recognition or genetic algorithms.

Machine learning algorithms can be divided in *supervised* and *unsupervised* learning. Supervised learning algorithms are trained on labeled examples (input and output of the system are shown) and attempt to generalize a mapping from inputs to outputs. Unsupervised learning algorithms operate on unlabeled examples (the output is unknown) and aim at discovering patterns in the data.

3.1.1 Regression Models

Given a training set composed by N pairs, related to the input and the output of the system:

$$(\mathbf{x_i}, y_i), \quad i \in \{1, ..., N\} \tag{3.1}$$

where $\mathbf{x_i}$ is a $1 \times P$ feature vector and y_i is the real value to predict, a *regressor* $r(\cdot)$ is defined as the function that minimizes the error ε between the expected and the predicted values. A typical measure for the prediction error is the mean squared error (MSE).

Regression analysis belongs to supervised learning algorithms and it is widely used for prediction and forecasting. Many techniques have been proposed in order to implement regression analysis, including linear regression, ordinary least squares,and non-parametric regression.

In general a regressor is estimated by two steps: during the *training phase*, the training set is used to estimate a regression function, while in the *test phase* a test set is used to estimate its performances by comparing the output with the correct outcome. The block diagram of a generic regression model is shown in figure 3.1.

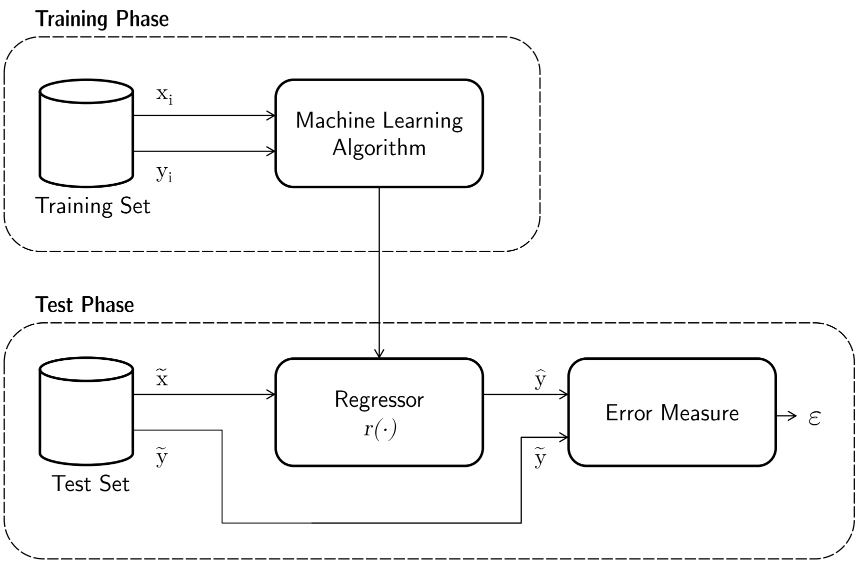


Figure 3.1: Block diagram of training and test phases for a supervised regression problem

3.1.2 Neural Networks

A neural network is a multi-stage regression or classification model based on nonlinear functions [60]. It is typically represented by a network diagram, that consists of the input stage composed by X, the output stage composed by Y, and the hidden layers in-between input and output that are composed by the so-called derived features Z. The most widely used architecture for neural network is composed by one single hidden layer, as shown in figure 3.2. Each node of the network is referred as *neuron*.

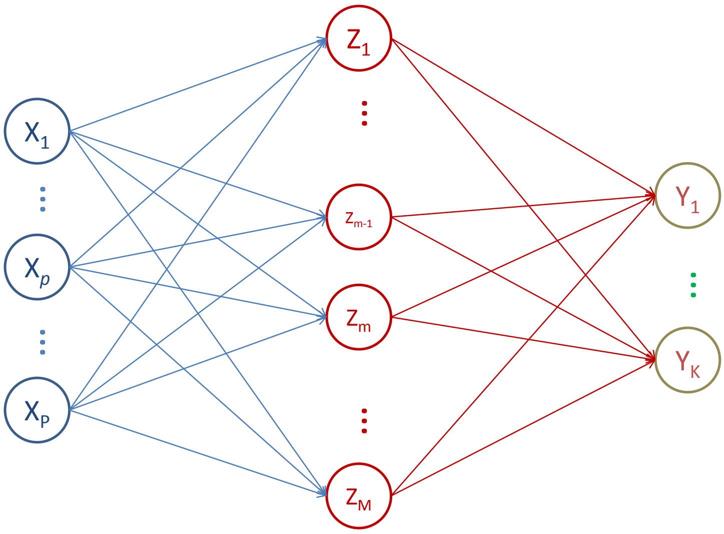


Figure 3.2: Single hidden layer neural network

The derived features Z_m are created from linear combinations of the inputs X:

$$Z_m = \sigma(\alpha_0 + \alpha_m^T X), \quad \text{with } m = 1, ..., M \tag{3.2}$$

where:

- $\alpha_m \in \mathbb{R}^P$ are the weights relative to the neuron Z_m,
- $X \in \mathbb{R}^P$ is a single input sample
- P is the number of feature of the input
- α_{0m} is the bias for the intercept
- M is the amount of neurons in the hidden layer
- $\sigma(\cdot)$ is is the so-called *activation function*

The activation function is traditionally chosen to be the *Sigmoid function*:

$$\sigma(v) = 1/(1 + e^{-v}). \tag{3.3}$$

Each output node Y_k is modeled as a function of linear combinations of the hidden neurons Z:

$$\begin{aligned} T_k &= \beta_{0k} + \beta_k Z, \quad \text{with } k = 1, ..., K, \\ f_k(x) &= g_k(T), \quad \text{with } k = 1, ..., K, \end{aligned} \tag{3.4}$$

where:

- $T = (T_1, T_2, ..., T_K)$ is the linear combination of the hidden neurons
- $\beta_k \in \mathbb{R}^M$ are the weights associated to the output Y_k
- K is the number of outputs
- β_{0k} is the bias for the intercept
- $f_k(\cdot)$ is the prediction of the output
- $g_k(\cdot)$ is the transformation

Possible choices for $g_k(\cdot)$ are the Identity function $g_k(T) = T_k$, or the Sigmoid function again, applied to the k-th linear combination $g_k(T) = \sigma(T_k)$.

Given a training set (x_i, y_i), the purpose of a regression based on neural network is to minimize the sum-of-squared errors:

$$R(\theta) \equiv \sum_{i=1}^{N} R_i(\theta) = \sum_{i=1}^{N}\sum_{k=1}^{K}(y_{ik} - f_k(x_i))^2, \tag{3.5}$$

where:

- θ is the complete set of weights: $\{\alpha_{0m}, \alpha_1, ..., \alpha_M\} \cup \{\beta_{0k}, \beta_1, ..., \beta_K\}$
- $f_k(x_i)$ is the prediction of the k-th output for x_i

The optimal weights in θ are computed by means of the back-propagation algorithm, that consists in an implementation of the gradient descent. After assigning a random values to all the weights, the algorithms involves two steps:

1. a *forward stage*, in which the hidden layer Z and output Y are computed

2. a *backward stage*, in which the prediction error is computed and then used for correcting the weights

Forward and backward steps are repeated for a certain amount of iterations, before approaching the global minimum, in order to avoid the model to be overfitted to the training set. The algorithm computes the prediction error for each step, and then it uses it for computing the partial derivative of $R(\theta)$:

$$\begin{aligned} \frac{\partial R_i}{\partial \beta_{km}} &= -\, 2(y_{ik} - f_k(x_i)) g'_k(\beta_k^T z_i) z_{mi}, \\ \frac{\partial R_i}{\partial \alpha_{ml}} &= -\sum_{k=1}^{K} 2(y_{ik} - f_k(x_i)) g'_k(\beta_k^T z_i) \beta_{km} \sigma'(\alpha_m^T x_i) x_{il}. \end{aligned} \tag{3.6}$$

The gradient descent update at the $(r+1)$-th iteration is formalized as:

$$\begin{aligned} \beta_{km}^{(r+1)} &= \beta_{km}^{(r)} - \gamma_r \sum_{i=1}^{N} \frac{\partial R_i}{\partial \beta_{km}^{(r)}}, \\ \alpha_{ml}^{(r+1)} &= \alpha_{ml}^{(r)} - \gamma_r \sum_{i=1}^{N} \frac{\partial R_i}{\partial \alpha_{ml}^{(r)}}, \end{aligned} \tag{3.7}$$

where γ_r is the *learning rate*, a constant chosen in order to minimize the error function.

Moreover, a weight decay term can be added to the error function in order to prevent the model from overfitting:

$$R(\theta) + \lambda J(\theta), \tag{3.8}$$

where

- $J(\theta) = \sum_k \|\beta_k\|^2 + \sum_m \|\alpha_m\|^2$ is a penalty function that works as a limiter for the size of weights
- $\lambda \geq 0$ is a tuning parameter

In this work we use neural network for annotating new songs in the dataset, as described in Chapter 5.

3.2 Multimedia Information Retrieval

Multimedia Information Retrieval is a research discipline that deals with the representation, storage, organization of, and access to multimedia items.

It aims at automatically extracting meaningful information from sources of various nature, in order to satisfy the user information need. Differently from data retrieval, information retrieval uses unstructured queries for producing a ranking of relevant results. An information retrieval model (IRM) can be defined as:

$$IRM =< D,\, Q,\, R(q_i, d_j) >, \tag{3.9}$$

where:

- D is the set of documents in the collection: they could be text files, audio files, videos, pictures, etc.
- Q is the set of queries that represent the user need.
- $R(q_i, d_j)$ is a ranking function that associates a real number to a document representation d_j with a query q_i. Such ranking defines an ordering among the documents with regard to the query.

The relevance is subjective, dynamic, multifaceted and is not known to the system prior to the user judgment.

3.2.1 Information Retrieval Models

Retrieval models assign a measure of similarity between a query and a document. In general, the more often query and document shares terms[1], the more relevant the document is deemed to be to the query.

A retrieval strategy is an algorithm that takes a query q and a set of documents $d_1, ..., d_N$ and then identifies the Similarity Coefficient $SC(q, d_j)$ for each document in the collection.

Every document is represented by a set of keywords called index terms, that are used to index and summarize the document content:

$$\mathbf{d}_j = [w_{1j}, ..., w_{Mj}]^T, \tag{3.10}$$

where w_{ij} represents the weight of the term t_i in the document d_j.

3.2.2 Vector Space Model

The Vector Space Model (VSM) represents documents and queries as vectors in the term space. The index term significance is represented by real valued weights associated to every pair (t_i, d_j):

$$w_{ij} \geq 0. \tag{3.11}$$

[1]The word *term* is inherited from text retrieval, but in general it indicates any relevant feature of the multimedia document.

Each document is represented by a vector in a M-dimensional space, where M is the number of index terms:

$$\mathbf{d}_j = [w_{1j}, ..., w_{Mj}]^T. \tag{3.12}$$

Each term is identified by a unit vector pointing in the direction of the i-th axis:

$$\mathbf{t}_i = [0, 0, ..., 1, ..., 0]^T. \tag{3.13}$$

The set of vectors $\mathbf{t}_i$, for $i = 1, ..., M$ forms a canonical basis for the Euclidean space $\mathbb{R}^M$.

Any document vector d_j can be represented by its canonical basis expansion:

$$\mathbf{d}_j = \sum_{i=1}^{M} w_{ij} t_i. \tag{3.14}$$

Documents that are close to each other in the vector space are similar to each other.

The query is also represented by a vector:

$$\mathbf{q} = [w_{1q}, ..., w_{Mq}]^T. \tag{3.15}$$

In figure 3.3 we show an example of Vector Space for three index terms.

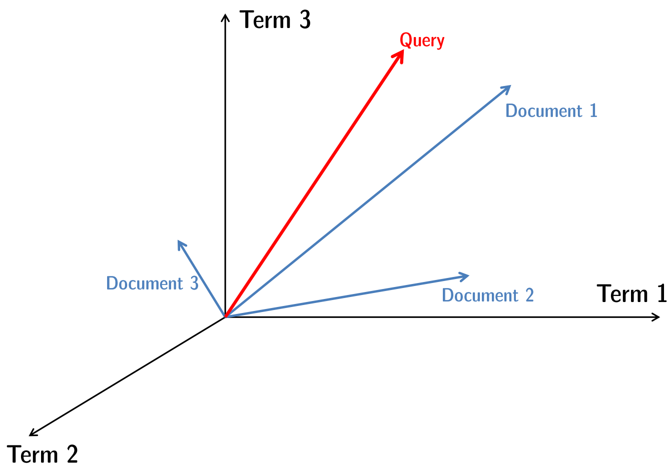


Figure 3.3: Vector Space for three index terms

The VSM computes the Similarity Coefficient $SC(\mathbf{q}, \mathbf{d}_j)$ between the query and each document, and produces a ranked list of documents. There are various measures that can be used to assess the similarity between documents (Euclidean distance, Cosine similarity, inner product, Jaccard similarity, etc.).

Index weights should be made proportional to its importance both in the document and in the collection. In order to address this issue, a popular method called term frequency - inverse document frequency is often applied in text mining. The weights are computed as:

$$w_{ij} = tf_{ij} \times idf_i, \tag{3.16}$$

where

- tf_{ij} is the frequency of the term t_i in the document d_j
- idf_i is the inverse document frequency of term t_i

Different strategies have been proposed in order to compute term frequency and inverse document frequency in text collections. This approach enables the weight w_{ij} to increase with the number of occurrences within a document and with the rarity of the term across the whole corpus.

3.2.3 Latent Semantic Indexing

Latent Semantic Indexing (LSI) is an indexing and retrieval method that uses Singular Value Decomposition (SVD) in order to identify patterns in the relationships between the terms and concepts contained in an unstructured collection of documents [2]. The basic idea behind LSI consists in assuming that terms that co-occur in the same context tend to have a similar meaning.

In order to implement LSI, a term-document matrix is constructed. The same weights w_{ij} defined by the VSM for quantifying the relation between the term t_i and the document d_j are used:

$$\mathbf{A} = [w_{ij}] = [\mathbf{d}_1, ..., \mathbf{d}_n] = [\mathbf{t}_1, ..., \mathbf{t}_m]^T. \tag{3.17}$$

Since $\mathbf{A}$ is typically a sparse matrix, the first step of LSI consists in finding its low-rank approximation by applying Singular Value Decomposition (SVD):

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T. \tag{3.18}$$

LSI uses a truncated version of the SVD, keeping only the k largest singular values and their associated vectors:

$$\mathbf{A}_k = \mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{V}_k^T. \tag{3.19}$$

This procedure finds k uncorrelated concepts, where each concept gathers co-occurrent terms. Documents and terms can be now described as a linear combination of these concepts and their similarity can be computed in this new reduced space.

For example, a user query can be represented as a vector in the k-dimensional concept space as:

$$\mathbf{q}_k = (\mathbf{U}_k \mathbf{\Sigma}_k^{-1})^T \mathbf{q}. \tag{3.20}$$

The distance between the query vector and the documents in the reduced space is proportional to their similarity. LSI can be interpreted as following:

- two documents are similar when they share terms that co-occur in many other documents

- two terms are similar when they co-occur with many of the same words

This approach is then able to capture term similarity in the k-dimensional concept space: synonym terms are mapped to the same concept.

3.3 Audio Features

Every sound can be represented by a set of features extracted from the physical signal. This kind of features are often referred to as Low-level Features (LLFs) or Audio Features, and they are able to characterize different audio signals by describing specific acoustic cues.

LLFs can be used in order to measure the energy and the spectral characteristics in the audio signal, or temporal aspects related with tempo and rhythm. In the following section we illustrate the audio features employed in this work and summarized in table 3.1, as described in [61].

	Low-level
Spectral	MFCC, Spectral Centroid, Zero Crossing Rate, Spectral Skewness, Spectral Flatness, Spectral Entropy
	Mid-level
Rhythmic	Tempo

Table 3.1: Low and mid-level features used in this work

3.3.1 Mel-Frequency Cepstrum Coefficients

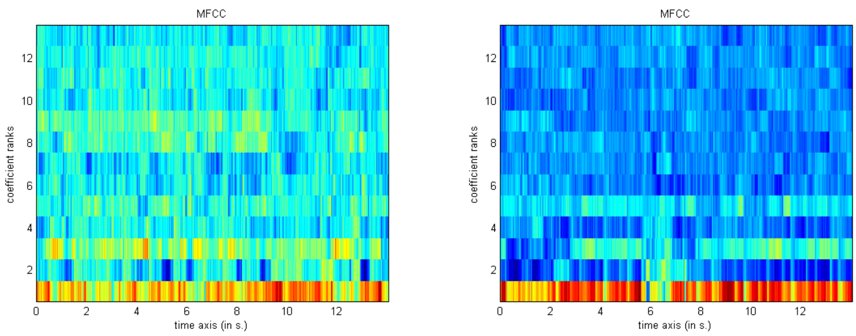
Mel-Frequency Cepstrum Coefficients (MFCCs) originated from automatic speech recognition but then they evolved into one of the standard techniques in most domains of audio retrieval. They are spectral low-level features based on the Mel-Frequency scale, a model that considers the human auditory system's perception of frequencies.

Mel-Frequency Cepstrum is a representation of the short-term power spectrum of a sound, and the coefficients are obtained from the Discrete Cosine Transform (DCT) of a power spectrum on a nonlinear Mel-Frequency scale (computed by a mel-filter bank). The mathematical formulation is:

$$c_i = \sum_{k=1}^{K_c} \{log(E_k) cos[i(k - \tfrac{1}{2}) \tfrac{\pi}{K_c}]\} \quad \text{with} \quad 1 \leq i \leq N_c, \tag{3.21}$$

where c_i is the $i - th$ MFCC component, E_k is the spectral energy measured in the critical band of the $i - th$ mel-filter, N_c is the number of mel-filters and K_c is the amount of cepstral coefficients c_i extracted from each frame.

An example of MFCCs related to two songs is shown in figure 3.4.



(a) Bon Jovi - "Livin' on a Prayer" (b) Eric Clapton - "Kind Hearted Woman"

Figure 3.4: MFCC for two songs, computed with [3]

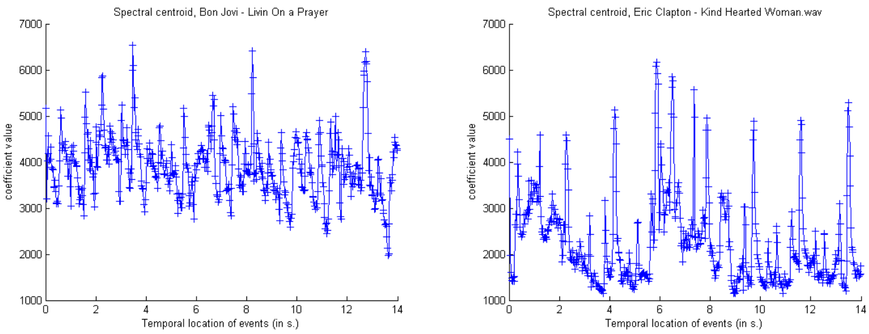
3.3.2 Spectral Centroid

Spectral Centroid (SC) is defined as the center of gravity of the magnitude spectrum (first momentum). It determines the point in the spectrum where most of the energy is concentrated and it is directly correlated with the dominant frequency of the signal. Given a frame decomposition of the audio signal, the SC is computed as:

$$F_{SC} = \frac{\sum_{k=1}^{K} f(k) S_l(k)}{\sum_{k=1}^{K} S_l(k)}, \tag{3.22}$$

where $S_l(k)$ is the Magnitude Spectrum at the $l-th$ frame and the $k-th$ frequency bin, $f(k)$ is the frequency corresponding to $k-th$ bin and K is the total number of frequency bins. Spectral Centroid can be used to check whether the magnitude spectrum is dominated by low or high frequency components. It is often associated with the brightness of the sound.

Spectral Centroids for two sample songs are shown in figure 3.5.



(a) Bon Jovi - "Livin' on a Prayer" (b) Eric Clapton - "Kind Hearted Woman"

Figure 3.5: Spectral Centroids for two songs, computed with [3]

3.3.3 Zero Crossing Rate

Zero Crossing Rate (ZCR) is defined as the normalized frequency at which the audio signal $s(n)$ crosses the zero axis, changing from positive to negative or back. It is formalized as:

$$F_{ZCR} = \frac{1}{2}\left(\sum_{n=1}^{N-1}|sgn(s(n)) - sgn(s(n-1))|\right)\frac{F_s}{N}, \qquad (3.23)$$

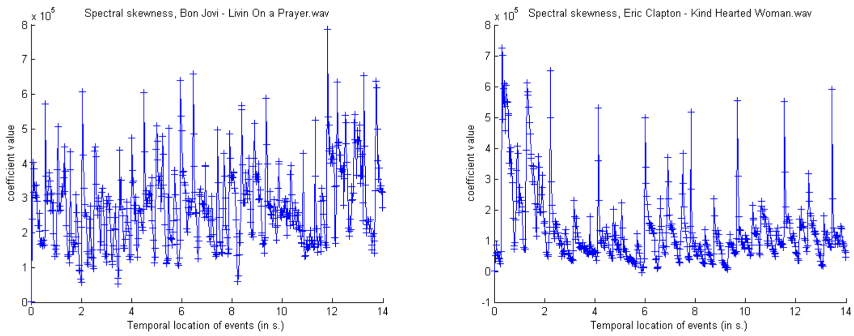
where N is the number of samples in $s(n)$ and F_s is the sampling rate. This measure is associated to the signal noisiness.

3.3.4 Spectral Skewness

Spectral Skewness (SSK) is the third moment of the distribution and it gives an estimation on the symmetry of the magnitude spectrum values. A positive value of Spectral Skewness represents an asymmetric concentration of the spectrum energy on higher frequency bins, while negative coefficients represent a distribution with a higher concentration on lower frequency bins. The perfect symmetry corresponds to the zero Spectral Skewness value. It is computed as:

$$F_{SSK} = \frac{\sum_{k=1}^{K}(S_l(k) - F_{SC})^3}{KF_{SS}}, \tag{3.24}$$

where $S_l(k)$ is the Magnitude Spectrum at the l-th frame and the k-th frequency bin, K is the total number of frequency bins, F_{SC} is the Spectral Centroid at the l-th frame (eq.3.3.2) and F˙SS is the Spectral Spread at the l-th frame (second moment of the distribution). We show the Spectral Skewness of two songs in figure 3.6.



(a) Bon Jovi - “Livin’ on a Prayer” (b) Eric Clapton - “Kind Hearted Woman”

Figure 3.6: Spectral Skewness for two songs, computed with [3]

3.3.5 Spectral Flatness

Spectral Flatness (SFlat) provides a way to measure how much an audio signal is noisy, estimating the flatness of the magnitude spectrum of the signal frame. It is calculated by dividing the geometric mean of the power spectrum by the arithmetic mean of the power spectrum:

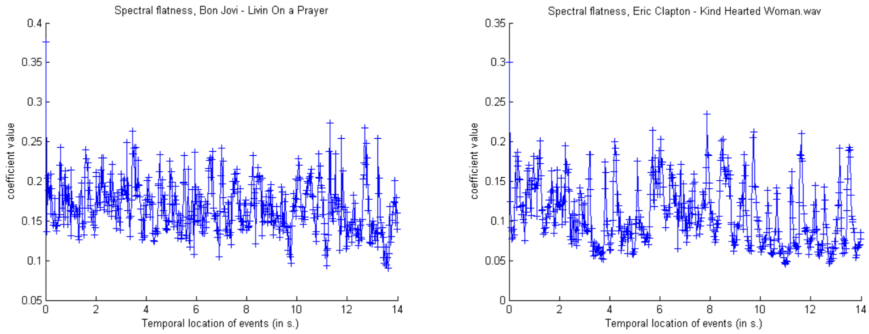
$$F_{SFlat} = \frac{\sqrt[K]{\prod_{k=0}^{K-1} S_l(k)}}{\sum_{k=1}^{K} S_l(k)}, \tag{3.25}$$

where $S_l(k)$ is the Magnitude Spectrum at the $l-th$ frame and the $k-th$ frequency bin, K is the total number of frequency bins.

The Spectral Flatness related to two sample song is displayed in figure 3.7.

3.3.6 Spectral Entropy

Spectral Entropy (SE) is a measure of the flatness of the magnitude spectrum by the application of Shannon’s entropy commonly used in information



(a) Bon Jovi - "Livin' on a Prayer" (b) Eric Clapton - "Kind Hearted Woman"

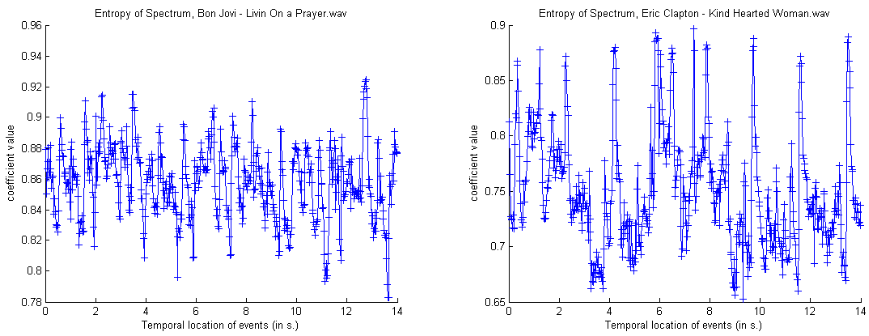
Figure 3.7: Spectral Flatness for two songs, computed with [3]

theory context:

$$F_{SE} = \frac{\sum_{k=1}^{K} S_l(k)\ logS_l(k)}{logK}, \tag{3.26}$$

where $S_l(k)$ is the Magnitude Spectrum at the $l-th$ frame and the $k-th$ frequency bin, K is the total number of frequency bins. A totally flat magnitude spectrum corresponds to the maximum uncertainty and the entropy is maximal. On the other hand, the configuration with the spectrum presenting only one very sharp peak and a flat and low background corresponds to the case with minimum uncertainty, as the output will be entirely governed by that peak.

In figure 3.8 we show the Spectral Entropy of two songs.

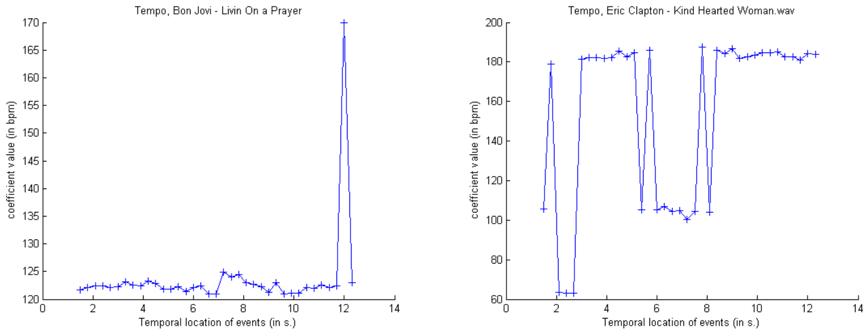


(a) Bon Jovi - "Livin' on a Prayer" (b) Eric Clapton - "Kind Hearted Woman"

Figure 3.8: Spectral Entropy for two songs, computed with [3]

3.3.7 Tempo

Tempo is a mid-level feature that represents the speed of a given piece. It is specified in beats per minute (BPM), i.e., how many beats are played in a minute. Different techniques for tempo estimation have been proposed, from simple statistical models based on sound energy to complex comb filter networks. An example of detected tempo from two songs is shown in figure 3.9.



(a) Bon Jovi - “Livin’ on a Prayer” (b) Eric Clapton - “Kind Hearted Woman”

Figure 3.9: Tempo Detection for two songs, computed with [3]

3.4 Music Emotion Models

Music Emotion Recognition (MER) is the field of MIR that studies how music and emotions are connected. As mentioned in Chapter 2, two approaches have been proposed in order to represent their relationship: the *categorical* and the *parametric* methods.

Categorical methods consider emotions as categories belonging to a limited number of innate and universal affective experiences. This approach aims at highlighting the factors that distinguish emotions from one another. Various categorical methods that describe music with a fixed set of emotion terms have been proposed, but there is no agreement between the terms that describe univocally basic emotions.

Parametric methods argue that emotion description can be organized into low-dimensional models. The most popular model is the Valence-Arousal (V-A) emotion plane, that defines two basic emotion components [8]. Given a certain emotion, valence indicates how much the feeling is positive or negative, while arousal represents its intensity. Operating in a dimensional space ease the computation of similarity between resources described by emotion terms, such as music pieces. Various researchers attempted to map terms into parametric spaces. The most popular solution has been proposed by Bradley et al.[10], that developed the Affective Norms for English Words (ANEW), which consists of 2476 affective words labeled in a Valence-Arousal-Dominance space. A simplified mapping of mood terms into the V-A space is shown in figure 3.10.

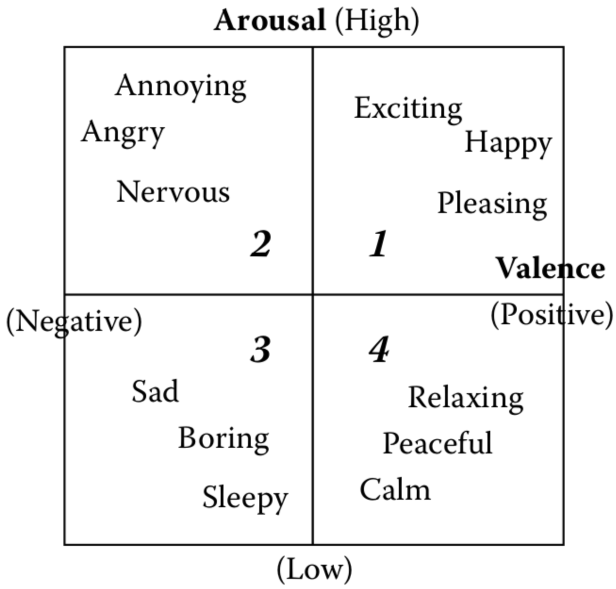


Figure 3.10: The 2D Valence-Arousal emotion plane, with some mood terms approximately mapped [4]

3.5 Natural Language Processing

Natural Language Processing (NLP) is a branch of artificial intelligence that aims to develop computational systems able to interact with humans by using natural language. NLP algorithms are usually based on machine learning and they cope with various challenges, such as discourse analysis, machine translation, speech recognition, natural language generation, question answering. In this section we review two basic natural language processing tools for phrase parsing: part-of-speech taggers and context-free grammars.

3.5.1 Part-of-Speech Tagging

Parts-of-Speech (POS) are the linguistic categories of words, generally defined by their grammar role in a sentence. English basic parts-of-speech are: *nouns*, *verbs*, *adjectives*, *adverbs*, *pronouns*, *conjunctions*, *prepositions* and *interjections*.

POS tagging is the process of assigning part-of-speech categories to words in a corpus. It is a complex task since some words can represent more than one part of speech (e.g. is *building* a name or a verb?). POS tagging represents the first step of a vast number of practical assignments, such as speech synthesis, parsing and machine translation.

A POS Tagger is a software that automatically analyzes and assigns parts-of-speech to words in a collection. In general POS tagging algorithms belong to two different groups: *rule-based* and *stochastic*. Rule-based tagging uses disambiguation rules to infer the POS tag for a term: it starts with a dictionary of possible tags for each word and uses hand-written rules to identify the correct tag when there is more than one possible tag. Stochastic tagging assign the most probable tag for each word in a sentence by choosing the tag sequence that maximizes the following probability:

$$P(\text{word—tag}) \times P(\text{tag—previous } n \text{ tags}). \tag{3.27}$$

POS rules and probabilities are computed or inferred from previously annotated sentence corpus, such as the *Penn Treebank*[62].

In order to represent a sentence, it is necessary to define some kind of formal structure of parts-of-speech.

3.5.2 Context-Free Grammars

A group of words within a sentence could act as a single unit, named *constituent*. Given a certain language, constituents form coherent classes that behave in similar ways. For example, the most frequently occurring phrase type is the *noun phrase* (NP), which has a noun as its head word (i.e. all words of the sentence are linked to the noun).

A formal grammar consists of a set of rules that indicates the valid syntactical combination among lexical elements. In formal grammar we define two types of lexical elements:

- *terminal symbols*, which in general correspond to words
- *non-terminal symbols*, which consists in the constituents of a language (e.g. noun phrase, verb phrase, etc.)

In context-free grammars (CFG) every production rule is of the form:

$$V \to w \tag{3.28}$$

where V is a single non-terminal symbol, and w is a set of terminals and/or non-terminals symbols. For example, a noun phrase can be defined in a CFG with the following rules:

$$NP \to Det\ Nominal \tag{3.29}$$

$$NP \to ProperNoun \tag{3.30}$$

A *nominal* can be defined as well as:

$$Nominal \to Noun|Noun\ Nominal, \tag{3.31}$$

A determiner (Det) and a $Noun$ could be described by the following rules:

$$Det \to a; \tag{3.32}$$

$$Det \to the; \tag{3.33}$$

$$Noun \to flight. \tag{3.34}$$

The sequence of CFG rules applied to a phrase is commonly represented by a parse tree 3.11.

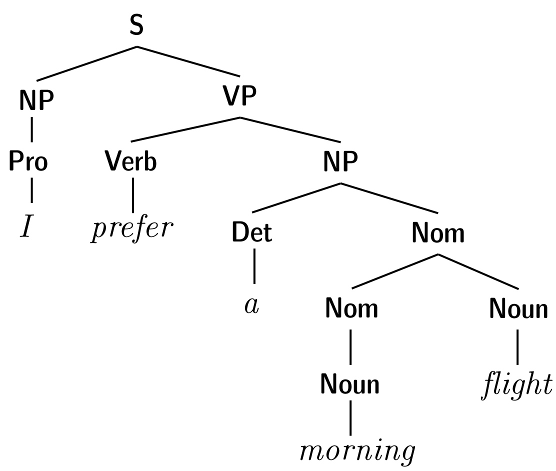


Figure 3.11: Parse tree example of a context-free grammar derivation.

CFGs are very useful in order to parse a natural language sentence and understand its underlying grammar structure: the integration of a CFG in a search engine allows users to perform semantic queries.

Chapter 4

Implementation of the System

The system presented in this thesis is the evolution of *Janas* [9, 1], a music search engine based on textual semantic queries. *Janas* uses a semantic model that defines two types of music descriptors, emotion descriptors (ED) and non-emotion descriptors (NED). This model suffers from various drawbacks: there is not a unique model for ED and NED, the mapping in the V-A plane of ED terms defines semantic relations even between terms belonging to different semantic planes, and finally it does not relate different NED among them.

As a review, in this chapter we give a brief description of the overall system, including the semantic model implemented in *Janas*. We also present a semantic model based on Latent Semantic Indexing (LSI) [2]. Finally, we illustrate our semantic model, that is named *Contextual-related Semantic Model*.

The overall structure of the system is represented in figure 4.1. It is composed by five main modules:

- the *Semantic Description Model*, that specifies how to interpret and represent the semantic description of music pieces
- the *Music Content Annotator*, that assigns appropriate annotations to new music items
- the *Query Model*, that formally represents the user's requests to the system
- the *Retrieval Model*, that identifies the music items that best match the user's request

- the *Graphical User Interface*, that allows users to interact with the system

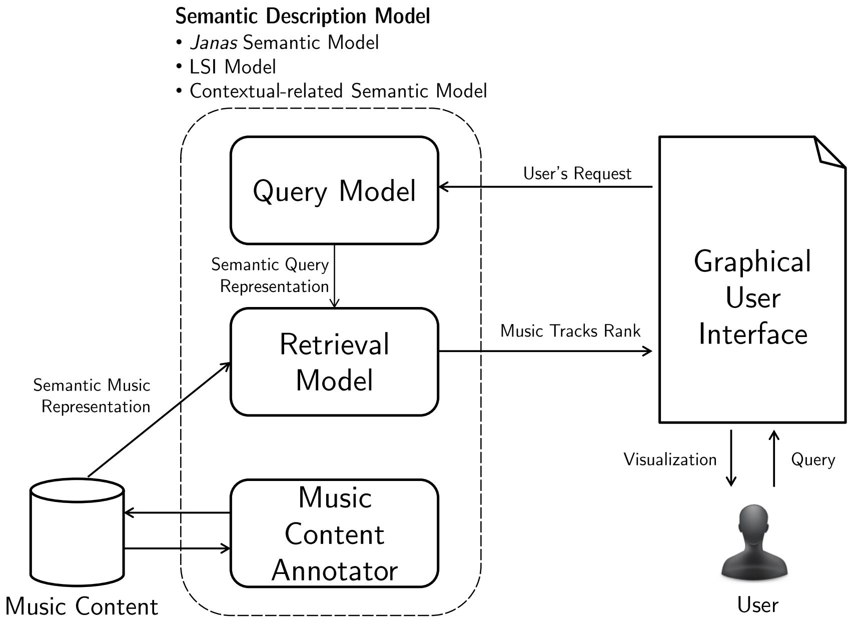


Figure 4.1: Architecture of the System

The Music Content Annotator, the Query Model and the Retrieval Model use the Semantic Description Model in order to semantically represent concepts and music content.

In the following sections we illustrate these basic components and we compare their functioning when the different semantic description models are used.

4.1 Semantic Description Model

The semantic description model is the core of a search engine based on semantic queries, and many of the modules in the system depend on this component. It specifies the relations between terms belonging to the vocabulary and how to use them in order to semantically describe a music piece. The usage of a specific semantic description model strongly influences the final retrieval performances, as shown in the result section in Chapter 5.

We compare our *Contextual-related Semantic Model* with the Janas Semantic Model and the LSI Model. In the following sections we discuss the implementation of these three models.

4.1.1 Janas Semantic Model

Janas assumes that music content and music-related concepts can be represented with both affective and non-affective semantic spaces. For this reason it defines two types of descriptors for music: emotional descriptors (ED) and non-emotional descriptors (NED).

4.1.1.1 Emotional Descriptors

According to [8], emotions can be mapped in the two-dimensional Valence-Arousal (V-A) plane. Its dimensions are: *a) Valence*, that describes the positiveness/negativeness of an emotion *b) Arousal*, that specifies the intensity of an emotion

In the next paragraphs we show how emotional descriptors are used in order to represent concepts and music content (i.e. songs) in *Janas*.

Concept Modeling In order to obtain a mapping of emotion concepts into the V-A plane, the authors exploited the ANEW dataset [10]. It contains a set of emotional-related terms manually tagged by human annotators in the V-A space. Each term is described with a value of mean and standard deviation on both Valence and Arousal dimensions. Since ANEW contains also generic terms, the authors decided to intersect the dataset with Wordnet-affect, a lexical database of emotion terms [45]. In *Janas*, each emotional concept c_{ED} is represented by a specific term $t_{i_{ED}}$ in the dataset $\mathcal{V}_J = \{happy, sad, angry, ...\}$ and it is modeled with a normal distribution:

$$c_{ED} \sim \mathcal{N}(\boldsymbol{\mu}_{VA}(t_{i_{ED}}), \boldsymbol{\Sigma}_{VA}(t_{i_{ED}})), \tag{4.1}$$

where:

- $\boldsymbol{\mu}_{VA}(t_{i_{ED}}) = [\mu_V(t_{i_{ED}}), \mu_A(t_{i_{ED}})]^T$ is the mean of the distribution of the term $t_{i_{ED}}$ in the V-A plane
- $\boldsymbol{\Sigma}_{VA}(t_{i_{ED}}) = diag(\boldsymbol{\sigma}_{VA}(t_{i_{ED}})) = diag([\sigma_V(t_{i_{ED}}), \sigma_A(t_{i_{ED}})]^T)$ is the covariance matrix of the distribution of the term $t_{i_{ED}}$ in the plane

Music Content Modeling Music content are annotated with emotion terms into the V-A plane in the same way as emotion concepts. In *Janas* the authors considered the dataset *MsLite* [63], that consists in a set of songs manually annotated in a 9-point scale both in Valence and Arousal dimensions.

In *Janas*, the authors computed mean and standard deviations of the annotations for each song d_j in the dataset and then they modeled it as a normal distribution in the V-A plane:

$$d_{j_{ED}} \sim \mathcal{N}(\boldsymbol{\mu}_{VA}(d_j), \boldsymbol{\Sigma}_{VA}(d_j)), \tag{4.2}$$

where:

- $\boldsymbol{\mu}_{VA}(d_j) = [\mu_V(d_j), \mu_A(d_j)]^T$ is the mean annotated value of Valence and Arousal related to the song d_j
- $\boldsymbol{\Sigma}_{VA}(d_j) = diag(\boldsymbol{\sigma}_{VA}(d_j))$ is the covariance matrix of the distribution of the song d_j in the plane
- $\boldsymbol{\sigma}_{VA}(d_j) = [\sigma_V(d_j), \sigma_A(d_j)]^T$ is the annotations' standard deviation of Valence and Arousal related to the song d_j

4.1.1.2 Non-Emotional Descriptors

Since a music piece cannot be described exhaustively by using only emotion terms, the authors included non-emotional facets in order to enrich the description. They defined a set of bipolar high-level descriptors related to structural, kineasthetic and judgement features of a music piece, and a mid-level descriptor related to the tempo of a track. We examine only a subset of the NED descriptors defined in *Janas*, in order to consider only descriptors shared among all the considered semantic models (table 4.1).

Non-emotional high-level descriptors	
soft - hard	(soft) 1 - 9 (hard)
static - dynamic	(static) 1 - 9 (dynamic)
flowing - stuttering	(flowing) 1 - 9 (stuttering)
roughness	(not rough) 1 - 9 (rough)
Non-emotional mid-level descriptors	
tempo (BPM)	30-250

Table 4.1: List of non-emotional descriptors used in Janas

In the following paragraphs we describe how non-emotional descriptors are represented in *Janas*.

Concept Modeling In *Janas*, a non-emotional concept is represented by the combination of multiple bipolar descriptors. Each non-emotional bipolar high-level descriptor $(t_l - t_r)$ is linearly modeled in a separate one-dimensional space as $t_{i_{NED}}(t_l - t_r)$. The authors modeled the mid-level descriptor of tempo by means of the mapping with beats per minutes described in table 4.2. It uses the conventional Italian vocabulary for tempo of classical music. The tempo descriptor $t_{i_{NED}}(tempo)$ is finally modeled as a mixed distribution between a normal and a uniform distribution.

Tempo Markings	BPM
Adagio	66-76
Andante	76-108
Moderato	108-120
Allegro	120-168
Presto	168-200

Table 4.2: Tempo markings and correspondent ranges of BPM

A concept c_{NED} is finally formalized as:

$$c_{NED} = \{t_{i_{NED}}(t_l - t_r) \cup t_{i_{NED}}(tempo)\}. \tag{4.3}$$

Music Content Modeling The authors of *Janas* set up a survey in order to annotate the music tracks in the dataset with non-emotional descriptors.

Every song in the dataset d_j is modeled as a normal distribution for each high-level bipolar descriptor $(t_l - t_r)$:

$$d_{j_{NED}}(t_l - t_r) \sim \mathcal{N}(\mu_j(t_l - t_r), \sigma_j(t_l - t_r)), \tag{4.4}$$

where $\mu_j(t_l - t_r)$ and $\sigma_j(t_l - t_r)$ are respectively the mean and the standard deviation values of the bipolar descriptor annotated in the survey

In order to annotate the tempo of songs, the authors used a tempo estimator [64] and modeled the descriptor as a normal distribution with *a*) mean $\mu_j(tempo)$, that corresponds to the computed tempo and *b*) standard deviation $\sigma_j(tempo)$, estimated as a fraction of the mean.

The complete non-emotional semantic model for the song d_j is defined as the set:

$$d_{j_{NED}} = \{d_{j_{NED}}(t_l - t_r) \cup d_{j_{NED}}(tempo)\}. \tag{4.5}$$

On the whole, each music piece d_j in the dataset is modeled by a set that contains both emotional and non-emotional representations:

$$d_j = \{d_{j_{ED}} \cup d_{j_{NED}}\}. \tag{4.6}$$

4.1.2 Latent Semantic Indexing Model

We implemented *Latent Semantic Indexing* (LSI) [2], one of the most used techniques for multimedia retrieval in commercial applications. LSI is a dimensional method that exploits co-occurrences of annotations in songs in order to infer semantic relations between them. This approach differs from the one defined in *Janas* because it initially assumes that all the emotional and non-emotional concepts are independent, then it estimates their relations.

In the next paragraphs we explain how concepts and music items are represented in the LSI model.

Concept Modeling We defined a vocabulary $\mathcal{V}_{LSI}$ of 40 suitable terms t_i for describing both emotional and non-emotional facets of music, as shown in table 4.3. In this model a concept is represented by one or more terms $t_i \in \mathcal{V}_{LSI}$ and it is formalized with a vector $\mathbf{c} \in \mathbb{R}^{40}$, for which each element w_i is directly related to the term in $t_i \in \mathcal{V}_{LSI}$:

$$\mathbf{c} = [w_0, ..., w_i, ..., w_{40}]^T, \tag{4.7}$$

with $w_i \in [0, 1]$. The value of the element w_i is proportional to the weight of the term $t_i \in \mathcal{V}_{LSI}$ in the concept.

Vocabulary $\mathcal{V}_{LSI}$			
Aggressive	Dark	Hard	Serious
Angry	Depressed	Harsh	Slow
Annoyed	Dynamic	Heavy	Smooth
Anxious	Exciting	Joyful	Soft
Boring	Fast	Light	Static
Bright	Flowing	Nervous	Stuttering
Calm	Frustrated	Quiet	Sweet
Carefree	Fun	Relaxed	Tender
Cheerful	Funny	Rough	Tense
Clean	Happy	Sad	Warm

Table 4.3: List of terms in the vocabulary $\mathcal{V}_{LSI}$

Music Content Modeling Music content in the dataset are generally annotated with multiple terms belonging to the vocabulary $\mathcal{V}_{LSI}$. Each song

d_j is initially described by a vector $\mathbf{d}_j \in \mathbb{R}^{40}$ that has non zero elements in correspondence of the selected annotation term t_i:

$$\mathbf{d}_j = [w_{j,0} = 0, ..., w_{j,i} \geq 0, ..., w_{n,40} = 0]^T. \tag{4.8}$$

In general the weight $w_{j,i}$ associated to the relation between the song d_j and the term t_i could be binary or continuous in the range $[0, 1]$, allowing the term to express the degree of descriptiveness of a music piece.

In order to capture the semantic relations between terms in the vocabulary, we build a term-song matrix $\mathbf{A}$ by using the vectors defined in the previous step:

$$\mathbf{A} = [w_{j,i}] = [\mathbf{d}_1, ..., \mathbf{d}_J], \tag{4.9}$$

where J is the total number of music content in the dataset. LSI computes a truncated version of the SVD of $\mathbf{A}$, keeping only the k largest singular values:

$$\mathbf{A_k} = \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{V}_k^T. \tag{4.10}$$

This low-rank approximation of the term-song matrix exploits the co-occurrences of the tracks annotations and merges terms with similar semantic along the same dimension. As a consequence of this, similar songs and similar concepts will be near in the reduced space. Each song $\mathbf{d}_j$ can be represented in the reduced space as:

$$\tilde{\mathbf{d}}_j = (\mathbf{U}_k \mathbf{\Sigma}_k^{-1})^T \mathbf{d}_j \tag{4.11}$$

In the same way, the representation of the concept C is computed as:

$$\tilde{\mathbf{c}} = \mathbf{\Sigma}_k^{-1} \mathbf{V}_k^T \mathbf{c} \tag{4.12}$$

In the implementation of the LSI model we experimentally decided to approximate the rank of the matrix $\mathbf{A}$ to $k = 20$.

4.1.3 Contextual-related Semantic Model

The approaches defined in the previous paragraphs raise substantive issues for the implementation of a music search engine based on semantic queries. *Janas* does not define a unique model for ED and NED. In particular, the semantic model considers the mapping of the terms in the V-A plane provided in ANEW for describing emotional concepts: this dataset is very rich and contains more than one thousand English terms, but at the same time the words were annotated without a specific focus in music. Furthermore, pairs of terms in the V-A plane have a dimensional relation even if they do not belong to the same semantic context. On the other hand, the semantic model for non-emotional descriptors do not consider the semantic relations among different NED. The Latent Semantic Indexing model is instead too simplistic: it just considers the co-occurrences between annotations among songs, thus it is strongly biased by the way in which songs were annotate.

We developed an innovative music-centric semantic model in order to overcome these issues, named *Contextual-related semantic model.* Since we assume that terms could have different meanings depending on the context we are considering, we defined three main contexts for describing music facets:

1. **Perceived Emotion**, that concerns the concepts able to describe the mood of a song
2. **Timbre Description**, that refers to the terms used to describe the sound characteristics of music
3. **Dynamicity**, that is related to the dynamic characteristics of a music piece

We considered the vocabulary $\mathcal{V}_{LSI}$ defined in the previous section and we assigned each of its 40 terms to these three contexts through a survey, described in 5.2. The obtained clusters are shown in table 4.4 and they form the *Context Vocabulary* $\mathcal{V}_{CTX}$.

In order to quantify the relatedness between pairs of terms belonging to the same context $\psi \in \Psi = \{1 \Rightarrow \textit{Perceived Emotion}\ ,\ 2 \Rightarrow \textit{Timbre Description},\ 3 \Rightarrow \textit{Dynamicity}\}$, we collected through a second survey (described in 5.2) annotations about their semantic similarity between pair of terms. Given two terms $t_i, t_j \in \psi$, their semantic similarity s_{ij}^{ψ} is modeled as the mean similarity value assigned with the survey to the pair (t_i, t_j), and it ranges between 0 (when they have opposite meaning) to 1 (when they have the same meaning):

$$s_{ij}^{\psi} = \frac{1}{N_{ij}^{\psi}} \sum_{n=1}^{N_{ij}^{\psi}} a(n)_{ij}^{\psi}, \tag{4.13}$$

where:

- N_{ij}^{ψ} is the number of annotations for the pair (t_i, t_j), with $t_i, t_j \in \psi$
- $\{a(1)_{ij}^{\psi}, ..., a(N_{ij}^{\psi})_{ij}^{\psi}\}$ is the set of gathered annotations for the pair (t_i, t_j)

These results are used for creating a vector space model. Given a context ψ, the *semantic similarity matrix* $\mathbf{S}^{\psi}$ between terms is defined as:

$$\mathbf{S}^{\psi} = [s_{ij}^{\psi}], \tag{4.14}$$

Context Vocabulary $\mathcal{V}_{CTX}$	
Perceived Emotion	Timbre Description
Aggressive	Bright
Angry	Clean
Annoyed	Dark
Anxious	Hard
Boring	Harsh
Calm	Heavy
Carefree	Rough
Cheerful	Smooth
Dark	Soft
Depressed	Warm
Exciting	
Frustrated	Dynamicity
Fun	Calm
Funny	Dynamic
Happy	Fast
Joyful	Flowing
Light	Quiet
Nervous	Relaxed
Quiet	Slow
Relaxed	Static
Sad	Stuttering
Serious	
Sweet	
Tender	
Tense	

Table 4.4: List of terms for each context cluster, obtained with the survey

where s_{ij}^{ψ} is the semantic similarity of the pair (t_i, t_j) in the context ψ. We assume that the semantic similarity is symmetric, thus s_{ij}^{ψ} is equal to s_{ji}^{ψ}.

Concept Modeling In this model a concept is represented as the combination of one or more terms included in the context vocabulary $\mathcal{V}_{CTX}$. The terms that describe a concept could belong to more than one context ψ. For example, a concept that corresponds only to a term $t_i \in \psi$ is modeled as a vector in the context ψ:

$$\mathbf{c}^{\psi} = \begin{cases} [w_0 = 0, ..., w_i \geq 0, ..., w_N = 0]^T & \text{if } t_i \in \psi \\ [w_0 = 0, ..., w_N = 0]^T & \text{if } t_i \notin \psi, \end{cases} \tag{4.15}$$

This notation allows to express a concept by using multiple terms t_i in more than one context ψ, by assigning to them different weights w_i^{ψ} in the range $[0, 1]$. Values of w_i^{ψ} lesser than 0.5 represent semantic dissimilarity with the term and values greater that 0.5 represent semantic similarity. In general the concept is represented by three vectors, each related to a different context in $\mathcal{V}_{CTX}$:

$$C = \begin{cases} \mathbf{c}^1 = [w_0^1, ..., w_i^1, ..., w_{N_1}^1]^T & \psi = 1 \Rightarrow \text{Perceived Emotion} \\ \mathbf{c}^2 = [w_0^2, ..., w_i^2, ..., w_{N_2}^2]^T & \psi = 2 \Rightarrow \text{Timbre Description} \\ \mathbf{c}^3 = [w_0^3, ..., w_i^3, ..., w_{N_3}^3]^T & \psi = 3 \Rightarrow \text{Dynamicity.} \end{cases} \tag{4.16}$$

In order to map the concepts to the *Contextual-related semantic model*, we multiply each context vector $\mathbf{c}^{\psi}$ with the corresponding semantic similarity matrix $\mathbf{S}^{\psi}$:

$$\tilde{\mathbf{c}}^{\psi} = \mathbf{S}^{\psi} \mathbf{c}_i^{\psi}. \tag{4.17}$$

The result of this operation captures the semantic similarity of the non zero elements of $\mathbf{c}^{\psi}$ with other terms in $\mathcal{V}_{CTX}$.

The concept C is finally described by the following set of vectors:

$$\tilde{C} = \{\tilde{\mathbf{c}}^1, \tilde{\mathbf{c}}^2, \tilde{\mathbf{c}}^3\}. \tag{4.18}$$

Music Content Modeling Music content in the dataset is annotated with one or more terms that belong to one or more contexts in $\mathcal{V}_{CTX}$.

Each song d_j in the dataset is initially represented by a set of three vectors $D_j = \{\mathbf{d}_j^1, \mathbf{d}_j^2, \mathbf{d}_j^3\}$, each of them related to a specific context. The weights of each vector assume non zero values in correspondence with the annotation terms elements:

$$D_j = \begin{cases} \mathbf{d}_j^1 = [w_0^1, ..., w_j^1, ..., w_{N_1}^1]^T & \psi = 1 \Rightarrow \text{Perceived Emotion} \\ \mathbf{d}_j^2 = [w_0^2, ..., w_j^2, ..., w_{N_2}^2]^T & \psi = 2 \Rightarrow \text{Timbre Description} \\ \mathbf{d}_j^3 = [w_0^3, ..., w_j^3, ..., w_{N_3}^3]^T & \psi = 3 \Rightarrow \text{Dynamicity.} \end{cases} \tag{4.19}$$

The song is mapped to the *Contextual-related semantic model* by multiplying each context vector $\mathbf{d_j}^{\psi}$ with the related semantic similarity matrix $\mathbf{S}^{\psi}$:

$$\tilde{\mathbf{d}}_j^{\psi} = \mathbf{S}^{\psi}\mathbf{d}_j^{\psi}. \tag{4.20}$$

This mapping enriches the song annotation by assigning a weight $w_i^{\psi} \geq 0.5$ to the terms $t_i \in \psi$ that are semantically correlated to the terms in the annotation.

Eventually, the music content is represented by a set of vectors:

$$\tilde{D}_j = \{\tilde{\mathbf{d}}_j^1, \tilde{\mathbf{d}}_j^2, \tilde{\mathbf{d}}_j^3\}. \tag{4.21}$$

In table 4.5 we summarize the notation for the representation of concepts and music content for each model.

Concept Modeling	
Janas Semantic Model	c_{ED}, c_{NED}
LSI Model	$\tilde{\mathbf{c}}$
Contextual-related Semantic Model	$\tilde{C} = \{\tilde{\mathbf{c}}^1, \tilde{\mathbf{c}}^2, \tilde{\mathbf{c}}^3\}$
Music Content Modeling	
Janas Semantic Model	$d_{j_{ED}}$, $d_{j_{NED}}$
LSI Model	$\tilde{\mathbf{d}}_j$
Contextual-related Semantic Model	$\tilde{D}_j = \{\tilde{\mathbf{d}}_j^1, \tilde{\mathbf{d}}_j^2, \tilde{\mathbf{d}}_j^3\}$

Table 4.5: Modeling notation for both concept and music content

4.2 Music Content Annotation

Music content in the system must be adequately annotated by using the descriptors defined by each semantic model. Manual annotation is a very expensive process because it needs expert human annotators willing to label a great number of music items. Therefore, a machine learning tool that partly automatize this process is preferred. We implemented a system for automatically annotating each song in the dataset. The annotation system is based on supervised machine learning techniques that uses a training dataset. The training set consists in a subset of 240 music excerpts and it has been obtained with a proper mapping between *Janas* semantic space and our model. The annotation procedure is described in Chapter 5. The basic structure for music content annotation is represented in figure 4.3 and in the following we describe the mapping between *Janas* annotations in the *Contextual-related semantic model*.

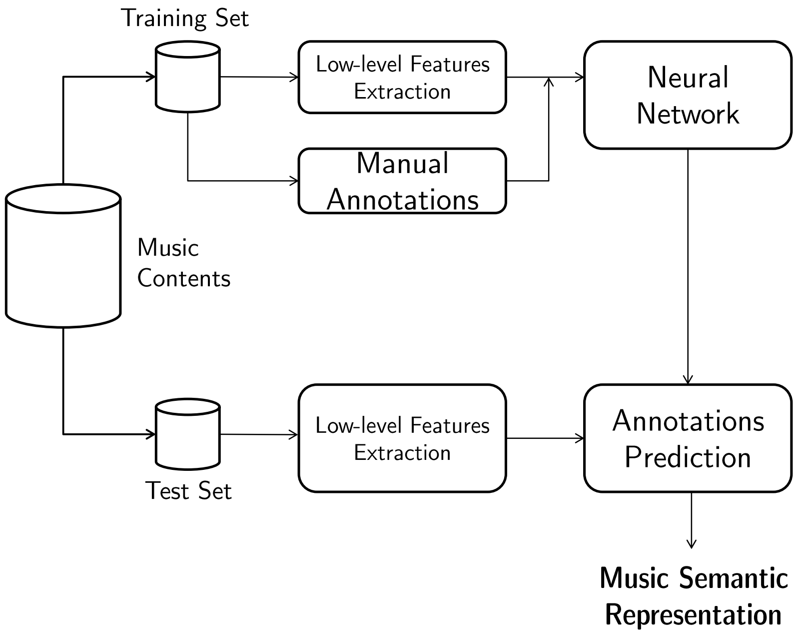


Figure 4.2: Music content annotation architecture

Mapping We experimentally define two metrics in order to map the music content description from *Janas* to the *Contextual-related semantic model*.

The first metric MAP_{ED} defines a mapping with emotional annotations belonging to the V-A plane. Given an affective term t_i belonging to the vocabulary $\mathcal{V}_{CTX}$, we consider its position and its distribution in the V-A plane by following ANEW specifics. Then we compute the similarity between the term t_j and the annotated track d_j in the V-A plane by following the metric:

$$Sim_{ED}(d_j, t_i) = D_{KL}(\mathcal{N}_{d_j} \| \mathcal{N}_{t_i}) \cdot (1 - \|d_{j_{VA}} - t_{i_{VA}}\|_1) \cdot sgn(CS(d_j, t_i)), \tag{4.22}$$

where:

- $D_{KL}(\cdot)$ represents the Kullback-Leibler divergence, intended as a measure of the difference between two multivariate normal distributions [65]
- $\mathcal{N}_{d_j}, \mathcal{N}_{t_i}$ are the multivariate normal distributions on the plane V-A of the song d_j and the term t_i
- $\|d_{j_{VA}} - t_{i_{VA}}\|_1$ is the norm-1 distance between the position of the song d_j and the term t_i in the V-A plane
- $CS(d_j, t_i)$ is the cosine similarity between the song d_j and the term t_i in the V-A plane
- $sgn(\cdot)$ is the sign function

The track d_j is then annotated with the affective terms t_i for which the similarity metric exceed a certain threshold ξ_{ED}. Since some terms in $\mathcal{V}_{CTX}$ assumed controversial values in the ANEW mapping with the V-A plane, we manually filtered ambiguous annotations. For example the term *smooth* is annotated in the V-A emotion plane with the point $(0.395; -0.0025)$, but in the musical context this term represents only a timbric characteristic and does not assume any affective meaning.

Each track is characterized by a set of annotations related to emotion terms:

$$MAP_{ED} : \{t_j \in (\mathcal{V}_{CTX} \cap ANEW) | Sim_{ED}(d_j, t_i) > \xi_{ED}\}. \tag{4.23}$$

The second metric MAP_{NED} defines a mapping with the non-emotional bipolar descriptors $(t_l - t_r)$used in *Janas*. The similarity of the song d_j with the left term t_l and the right term t_r of the descriptor are respectively computed as:

$$Sim^l_{NED}(d_j, t_i) = (1 - t_l), \tag{4.24}$$

$$Sim^r_{NED}(d_j, t_i) = t_r. \tag{4.25}$$

Each track is finally characterized by a set of annotations related to non-emotional descriptor by following the rule:

$$MAP_{NED} : \{w_l | Sim^l_{NED}(d_j, t_i) \geq \xi_{NED}\} \cup \{w_r | Sim^r_{NED}(d_j, t_i) \geq \xi_{NED}\}, \tag{4.26}$$

where ξ_{NED} is the threshold that the similarity metric should exceed. Overall, each music content d_j in the dataset is annotated with the set of terms defined by:

$$MAP = MAP_{ED} \cup MAP_{NED}. \tag{4.27}$$

They assume a weight equals to their respective similarity metric Sim. For example, the track "*Les Rythmes Digitales - Take A Little Time*" has been mapped with the annotations showed in table 4.6.

Annotation	Similarity Weight
Dynamic	0.781
Exciting	0.437
Happy	0.477
Light	0.474
Smooth	0.470

Table 4.6: Mapped annotations from Janas semantic model for the track "Les Rythmes Digitales - Take A Little Time"

Automatic Annotation In order to test the scalability of our method, we annotated a subset of 140 tracks through a neural network that uses the remaining music content in *Janas* as training set $\bar{\tau}$. The output of the regression consists in 140 annotated tracks, creating a complete dataset of 380 songs. Each new song in the dataset is modeled as:

$$\hat{D}_j = \{\hat{\mathbf{d}}_j^1, \hat{\mathbf{d}}_j^2, \hat{\mathbf{d}}_j^3\} \text{ with } \hat{\mathbf{d}}_j^\psi = f^\psi(\bar{\mathbf{x}}_j) \; \forall \psi \in \Psi, \tag{4.28}$$

where:

- $f^\psi(\cdot)$ is the estimation function defined by the neural network model for the ψ-th context
- $\bar{\mathbf{x}}_j$ is the feature vector of the track d_j, with $j \in \bar{\tau}$

In figure 4.3 we illustrate the structure of the music content annotation.

4.3 Query Model

The system is based on free-text queries, such as "*I want a happy song*" or "*Please, retrieve an aggressive track*". In order to perform a research on a certain user request, we define two main modules:

1. a *Natural Language Parser*, that understands a user request by parsing the query and by extracting relevant terms for the research
2. a *Semantic Query Modeling Tool*, that represents the relevant terms in the semantic models defined in the previous paragraph

The architecture of the query model is represented in figure 4.4. In the following sections we describe the implementation of these two modules.

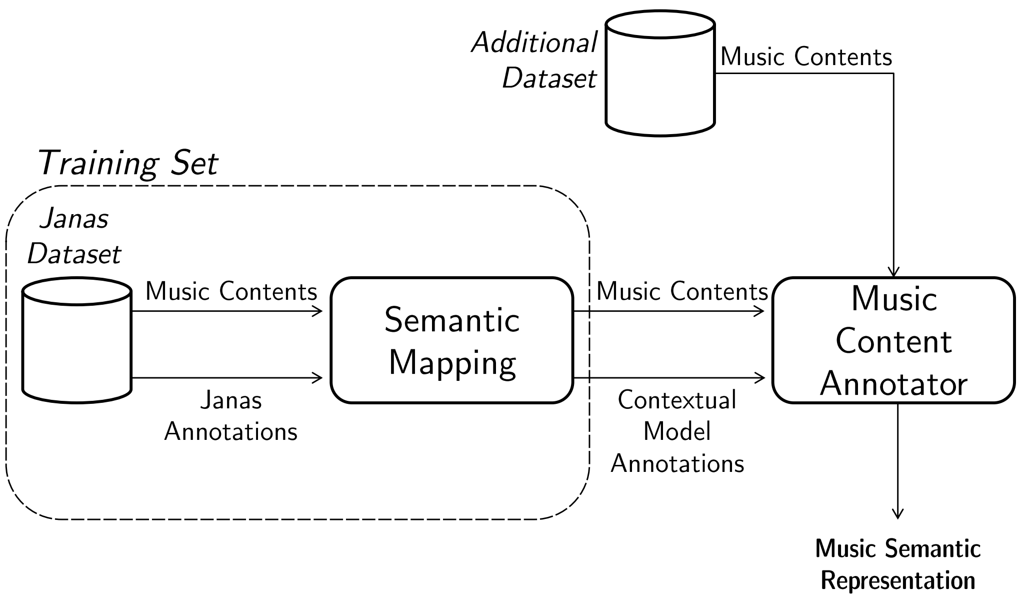


Figure 4.3: Music content annotation

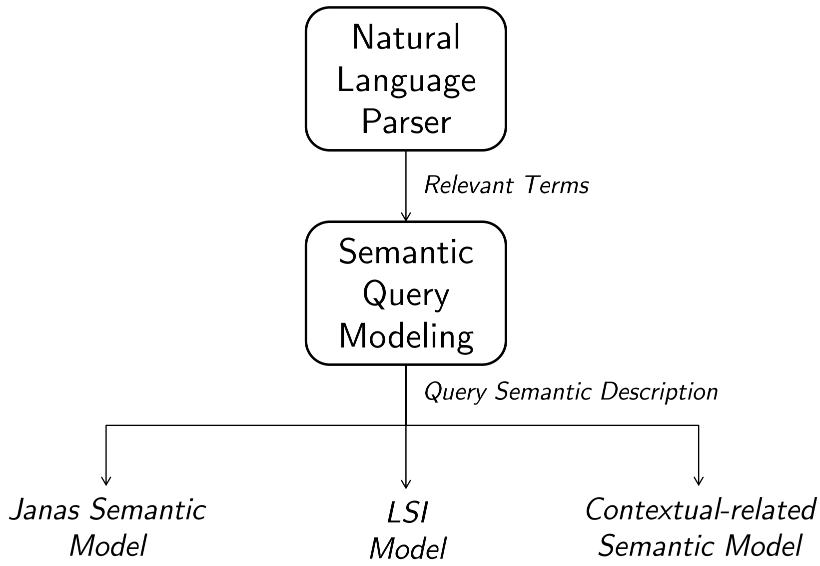


Figure 4.4: Query model structure

4.3.1 Natural Language Parser

In order to parse the query we employed the Natural Language Toolkit (NLTK) [66], a Python platform for natural language processing that is based on WordNet and provides various tools for part-of-speech tagging and sentence parsing.

The module parses the query for discovering the grammar role of each term in the request by using a context-free grammar. Only relevant parts-of-speech (adjectives in the dictionary) are considered.

4.3.2 Semantic Query Modeling

Once the parser extracts relevant terms t_i from the user's request, the system models them as a formal query according to the selected semantic model. The query may be referred to one or more terms in the vocabulary, furthermore each term could be characterized by a qualifier that defines the intensity expressed by that word. In the next paragraphs we describe the usage of qualifiers in the queries and then we analyze how queries are formalized in each of our considered semantic models.

Qualifiers When people describe concepts or objects by using adjectives, they usually add qualifiers in order to specify the intensity of the description. For example, a music piece can be defined as *very* sad, *partly* sad, *moderately* sad, *not* sad *at all*, etc. In a text-based search system it is important to consider qualifiers in the description paradigm. In order to deal with this type of description, the natural language parser retrieves qualifiers by analyzing siblings and children of a word in the parse tree and then the query model assign to each relevant term a weight proportional to the qualifier.

We defined a set of 23 weights $\mathcal{Q}$ for common qualifiers (table 4.7) by following the rating scale defined in [5]. The relevant terms t_i in the user request are directly associated to their corresponding qualifier $\rho_i \in \mathcal{Q}$ with the tuple r_i :

$$r_i = (t_i, \rho_i) \text{ with } i = 1, ..., R \ , \tag{4.29}$$

where R is the total number of relevant terms in the request.

Qualifiers' Set $\mathcal{Q}$			
Qualifier	Weight	Qualifier	Weight
a little	0.56	moderately	0.73
average	0.72	not	0.04
completely	1	not at all	0.0
considerably	0.9	partly	0.65
extremely	1	quite	0.59
fairly	0.75	quite a bit	0.91
fully	1	rather	0.78
hardly	0.5	slightly	0.56
highly	0.97	somewhat	0.7
in-between	0.72	very	0.92
mainly	0.85	very much	0.98
medium	0.72		

Table 4.7: Qualifiers and mean value weights from [5], re-mapped to our model

In general each relevant term t_i in the query is associated to the value of te qualifier $\rho_i \in \mathcal{Q}$. When a term is not directly associated to a qualifier in the user's request, we consider a weight $\rho_i = 0.9$, in order to differentiate it from the case of extremely positive qualifiers.

4.3.2.1 Janas Semantic Query

In order to model the query, *Janas* initially build two separate representations, one for the non-emotional and one for the emotional descriptors:

- if a non-emotional term is present in the list of relevant terms, it is added to the set $\mathcal{Z}_{NED}$ along with its qualifier
- if an emotional term (or a synonym of the term) that appears in ANEW is also present in the list of relevant terms, it is added to the set $\mathcal{Z}_{ED}$ along with its qualifier

Afterwards, the query q is modeled as a set of normal distributions, one for each non-emotional descriptor in $\mathcal{Z}_{NED}$, plus a multivariate distribution in the V-A plan if any emotional term is present in $\mathcal{Z}_{ED}$:

$$q = \begin{cases} q_{ED} \sim \mathcal{N}(\mu_{ED}, \Sigma_{ED}), \\ q_{NED} \sim (\mathcal{N}(\mu_{NED_1}), \sigma^2_{NED_1}), ..., \mathcal{N}(\mu_{NED_{10}}, \sigma^2_{NED_{10}})). \end{cases} \tag{4.30}$$

We refer the reader to [9] for further implementation details of *Janas*.

4.3.2.2 Latent Semantic Indexing Query

In the Latent Semantic Indexing model we initially build a query as a vector $\mathbf{q} \in \mathbb{R}^{40}$ in which each relevant term t_i extracted by the natural language parser assumes a weight w_i equal to its qualifier $\rho_i \in \mathcal{Q}$:

$$\mathbf{q} = [w_1, ..., w_i = \rho_i, ..., w_{40}]^T. \tag{4.31}$$

For example, the semantic query "*I want an angry and very sad song*" that contains the terms $t_i = angry$ and $t_j = sad$ associated to the qualifier $\rho_j(very)$ will be represented by the vector $\mathbf{q}$ in which the weights w_i and w_j assume the values of the qualifiers $q_i, q_j \in \mathcal{Q}$ associated to terms. As we already mentioned in the previous section, we decided to assign a weight 0.9 to the terms for which there is no associated qualifier. Our example is represented as:

$$\mathbf{q} = [w_0 = 0, ..., w_i = \rho_i, ..., w_j = \rho_j, ..., w_{40} = 0]^T. \tag{4.32}$$

Finally the query is mapped in the reduced semantic space defined by LSI as:

$$\tilde{\mathbf{q}}_k = (\mathbf{U}_k \mathbf{\Sigma}_k^{-1})^T \mathbf{q}, \tag{4.33}$$

where:

- k is the number of dimensions considered in the LSI
- $\mathbf{U}_k$ and $\mathbf{\Sigma}_k$ are the rank-k approximated version of the matrices obtained by applying the SVD on the term-song matrix

4.3.2.3 Contextual-related Semantic Query

In the *Contextual-related semantic model* each relevant term in the user's request is represented by a set of 3 *context query vectors* $Q = \{\mathbf{q}^1, \mathbf{q}^2, \mathbf{q}^3\}$, one for each valid context $\psi \in \Psi = \{1 \Rightarrow$ *Perceived Emotion* , $2 \Rightarrow$ *Timbre Description*, $3 \Rightarrow$ *Dynamicity*$\}$. Since some terms in the context vocabulary $\mathcal{V}_{CTX}$ belongs to more than one context, we developed a procedure that assigns to each term a probability of being part of the context ψ. We run a survey where the testers were asked to assign a given term to one or more contexts in Ψ and then we estimated the probability of the term to belong to a context as:

$$p(t_i|\psi) = \frac{N(t_i, \psi)}{N(t_i)}, \tag{4.34}$$

where:

- $N(t_i, \psi)$ is the number of time that the testers associated the term t_i to the context $\psi \in \Psi$
- $N(t_i)$ is the total number of times that the term t_i has been annotated

On the base of this probability, the system is able to model the probabilistic query related to the term t_i as:

$$Q_i = \{\mathbf{q}_i^1, \mathbf{q}_i^2, \mathbf{q}_i^3\} = \begin{cases} p(t_i|\psi) \cdot [w_0 = 0, ..., w_i = \rho_i, ..., w_{N_\psi} = 0]^T & \text{if } t_i \in \psi \\ \mathbf{0}_\psi & \text{if } t_i \notin \psi, \end{cases} \tag{4.35}$$

where:

- ρ_i is the weight of the qualifier associated to the term t_i
- N_ψ is the number of terms in the context ψ
- $\mathbf{0}_\psi$ is a zero vector of size N_ψ

When the user's request contains T terms, each of them associated to a qualifiers, the overall query is computed as the sum of the probabilistic query of each term:

$$Q = \{\mathbf{q}^1, \mathbf{q}^2, \mathbf{q}^3\} = \sum_{i=1}^{T} Q_i = \{\sum_{i=1}^{N_1} \mathbf{q}_i^1, \sum_{i=1}^{N_2} \mathbf{q}_i^2, \sum_{i=1}^{N_3} \mathbf{q}_i^3\}. \tag{4.36}$$

Once the query Q is built, we project it into the *Contextual-related semantic model* by multiplying each context query vector $\mathbf{q}^\psi$ with its related semantic similarity matrices $\mathbf{S}^\psi$:

$$\tilde{\mathbf{q}}^\psi = \alpha \cdot \mathbf{q}^\psi + (1 - \alpha) \cdot \mathbf{S}^\psi \mathbf{q}^\psi, \tag{4.37}$$

where α is a tuning parameter that assigns a higher weight to the terms specified in the user query with respect to their semantically related terms. We experimentally set up the parameter $\alpha = 0.25$. The final query is modeled as:

$$\tilde{Q} = \{\tilde{\mathbf{q}}^1, \tilde{\mathbf{q}}^2, \tilde{\mathbf{q}}^3\}. \tag{4.38}$$

Specific Case: Semantic Dissimilarity Qualifiers In the case in which the user's request contains a qualifier that expresses a negative correlation (e.g. *I want a song not happy at all*), we need to appropriately model the query.

In a vector space such as the one defined by the *Contextual-related semantic model*, we can easily measure the similarity between vectors by using common metrics like the cosine similarity, but computing their dissimilarity it is not straightforward. However, the *Contextual-related model* provides us a simple method for reversing the problem: given that this model assigns a value from 0 (opposite meaning) to 1 (same meaning) to each pair of adjectives t_i, t_j, we can compute a *semantic dissimilarity matrix* in each context ψ:

$$\mathbf{B}^\psi = \mathbf{J}^\psi - \mathbf{S}^\psi, \tag{4.39}$$

where:

- $\mathbf{S}^\psi$ is the similarity matrix associated to the context ψ, as defined in section 4.1.3
- $\mathbf{J}^\psi$ is the all-one matrix, with the same size of $\mathbf{S}^\psi$

The element b_{ij} of the matrix $\mathbf{B}^\psi$ assumes a value of 1 when the terms t_i and t_j are completely dissimilar (opposite meaning) and a value of 0 when they have the same semantic (same meaning).

A context query $\mathbf{q}^\psi$ that contains both similarity and dissimilarity qualifiers is split into two parts:

$$\mathbf{q}^\psi = \mathbf{q}^\psi_{SIM} + \mathbf{q}^\psi_{DISS}, \tag{4.40}$$

where $\mathbf{q}^\psi_{SIM}$ is the similarity-related query and $\mathbf{q}^\psi_{DISS}$ is the dissimilarity-related query.

The the similarity-related query $\mathbf{q}^\psi_{SIM}$ is projected in the *Contextual-related semantic model* as:

$$\tilde{\mathbf{q}}^\psi_{SIM} = \mathbf{S}^\psi \mathbf{q}^\psi_{SIM}. \tag{4.41}$$

The the dissimilarity-related query $\mathbf{q}^{\psi}_{DISS}$ (in our case the terms associated to the qualifiers *not* and *not at all*) is projected as following:

$$\tilde{\mathbf{q}}^{\psi}_{DISS} = \mathbf{B}^{\psi}\mathbf{q}^{\psi}_{DISS}. \tag{4.42}$$

The final query is then modeled as:

$$\tilde{Q} = \{\tilde{\mathbf{q}}^{1}_{SIM} + \tilde{\mathbf{q}}^{1}_{DISS},\ \tilde{\mathbf{q}}^{2}_{SIM} + \tilde{\mathbf{q}}^{2}_{DISS},\ \tilde{\mathbf{q}}^{3}_{SIM} + \tilde{\mathbf{q}}^{3}_{DISS}\}. \tag{4.43}$$

4.4 Retrieval Model

Once the query has been adequately mapped in the selected model, it is compared with the semantic representation of music content in the dataset. The comparison allows to determine which music tracks mainly reflect the characteristics required by the user. Most similar tracks are sorted in a rank that constitutes the output of the retrieval system. The structure of the retrieval model is shown in figure 4.5.

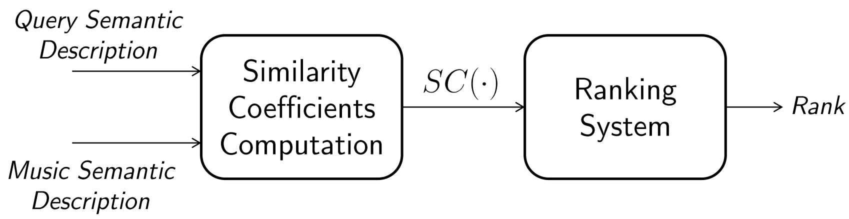


Figure 4.5: Retrieval model structure

In the following paragraphs we describe the ways in which retrieving process is performed in the three semantic models.

4.4.1 Janas Retrieval Model

In *Janas* the authors defined a probabilistic approach that compares each relevant term t in the user's request with the a-priori probability of a song in the dataset, both for emotional and non-emotional descriptors.

The similarity coefficient $SC(\cdot)$ between a query q and a song d_j can be expressed as:

$$SC(q, d_j)_J = \left(\prod_{\forall t \in Z_{ED} \cup Z_{NED}} P(q_t|d_j) \cdot P(d_j) \right)^{\frac{1}{|Z_{ED} \cup Z_{NED}|}}, \tag{4.44}$$

where:

- Z_{ED} and Z_{NED} represent the sets of relevant terms in the query that are associated respectively to emotional and non-emotional descriptors
- $|Z_{ED} \cup Z_{NED}|$ is the total number of terms in the query
- $P(q_t|d_j)$ is the conditional probability of the term t in the query q to be associated to the song d_j
- $P(d_j)$ is the prior probability of the song d_j

Both $P(q_t|d_j)$ and $P(d_j)$ are computed by using a Bayesian decision model [67]. Further computational details are presented in [9].

Once all the similarity coefficients are computed, they are sorted in a reversed-order list and a rank of songs similar to the query $\{d_1, ..., d_N\}$ is built.

4.4.2 Latent Semantic Indexing Retrieval Model

In the Latent Semantic Indexing model, the retrieving is performed by computing the similarity between the query vector and all the vectors related to music pieces that were previously mapped in the reduced space defined by LSI. We decided to use the cosine similarity as similarity metric in order to compute the similarity coefficients.

Given the query $\tilde{\mathbf{q}}_k \in \mathbb{R}^k$ and the vector representation of the music content $\tilde{\mathbf{d}}_j \in \mathbb{R}^k$ in the k-reduced space defined by LSI, the similarity is defined as:

$$SC(\tilde{\mathbf{q}}_k, \tilde{\mathbf{d}}_j)_{LSI} = \frac{\tilde{\mathbf{q}}_k^T \tilde{\mathbf{d}}_j}{\|\tilde{\mathbf{q}}_k\| \|\tilde{\mathbf{d}}_j\|}. \tag{4.45}$$

When the query and the music content have a similar representation, the similarity coefficient will have a higher value.

Once the similarity among the query and the whole music dataset has been computed, the system produces a list of tracks $\{d_1, ..., d_N\}$ ranked in decreasing order, with the most relevant tracks at the top of the list.

4.4.3 Contextual-related Semantic Retrieval Model

In order to retrieve the best matching music pieces in the *Contextual-related semantic model*, the system computes the similarity between the query $\tilde{Q} = \{\tilde{\mathbf{q}}^1, \tilde{\mathbf{q}}^2, \tilde{\mathbf{q}}^3\}$ and the tracks representation $\tilde{D}_j = \{\tilde{\mathbf{d}}_j^1, \tilde{\mathbf{d}}_j^2, \tilde{\mathbf{d}}_j^3\}$ in every context defined by the model. Similarly to the LSI query model, the system uses cosine similarity between query and tracks vector as similarity metric.

Given the query $\tilde{Q}$ and the representation of the music content $\tilde{D}_j$, the similarity coefficient is defined as the arithmetic mean of the similarity in every context:

$$SC(\tilde{Q}, \tilde{D}_j)_{CTX} = \frac{1}{3}\sum_{\psi=1}^{3} SC(\tilde{Q}, \tilde{D}_j)_{\psi} = \frac{1}{3}\sum_{\psi=1}^{3} \frac{(\tilde{\mathbf{q}}^{\psi})^T \tilde{\mathbf{d}}_j^{\psi}}{\|\tilde{\mathbf{q}}^{\psi}\| \|\tilde{\mathbf{d}}_j^{\psi}\|}. \tag{4.46}$$

The similarity coefficients are then sorted in descending order, producing a rank of music content $\{d_1, ..., d_N\}$ similar to the query.

4.5 Graphical User Interface

We developed a graphical interface in Python that acquires a semantic query from the user and visualizes the results of the ranking module.

The main window contains a search bar, similar to the one used by web search engines, as shown in 4.6.

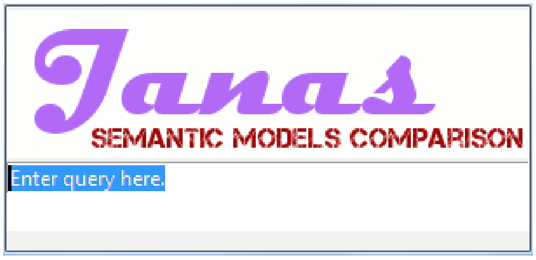


Figure 4.6: Music search interface

When the user submits a request, the system computes the ranking list of relevant songs. A threshold is imposed in order to show only interesting results (we experimentally chose 0.5 for LSI and the *Contextual-related semantic model*).

Since the goal of the thesis is to compare the performances of the semantic models, we display simultaneously the results of the three models (figure 4.7). Each music piece of the ranking list is displayed. showing information about the title, the artist and the album and allowing to play, stop and pause the tracks.

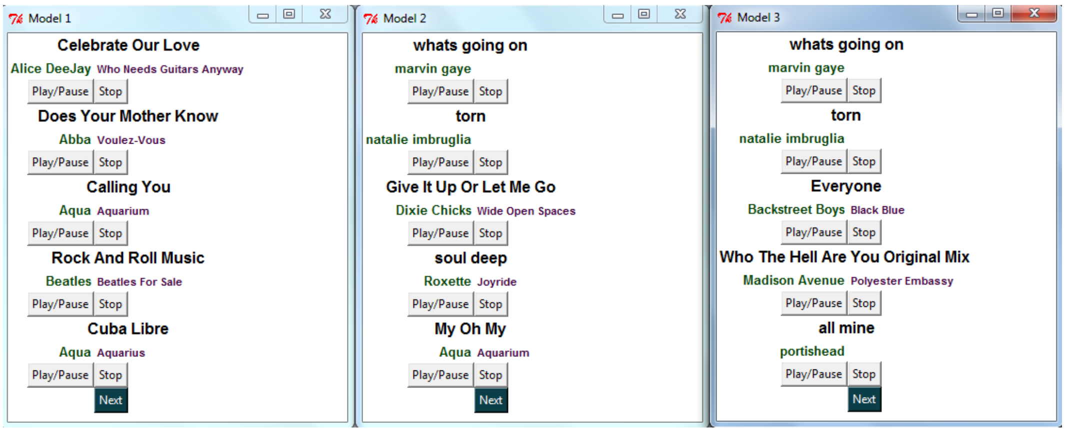


Figure 4.7: Result windows for the three models

Chapter 5

Experimental Results

In this chapter we analyze the performances of the *Contextual-related Semantic Model* with respect to the original *Janas* model and the LSI model. We collected the results through a questionnaire, where the testers were asked to evaluate the individual performances of each semantic model. In order to develop our model we initially carried out an online survey for collecting data about the contextual pertinence and the semantic similarity among the terms defined in section 4.1.3. At the same time, we examined the scalability of our model by annotating a subset of the original music collection through a machine learning process based on neural networks.

In the next section we analyze the annotation process, the results collected through the survey and the evaluation of the system performances obtained with the questionnaire.

5.1 Dataset Annotation

The original dataset used in *Janas* was composed by 380 songs excerpts annotated with both emotional and non-emotional descriptors. In order to build an initial dataset, we mapped the music content description of a subset of 240 excerpts from *Janas* to the *Contextual-related semantic model* representation, as described in 4.2.

In order to test the scalability of the system we automatically annotated the remaining 140 songs in the *Contextual-related semantic model* using the content-based approach described in 3.1.2. For the annotation process, representative 15 seconds for each song has been considered. The excerpts were sampled at 44.1 KHz and they have a bit rate of 320kbps. In appendix A we list the dataset of songs used by the sytem. We used the 240 initial excerpts to train a neural network (see 3.1.2). We extracted a set of 18 audio features for each music excerpt in the dataset, as described in 3.3: MFCC (for a total of 13 components), Spectral Centroid, Zero Crossing Rate, Spectral Skewness, Spectral Flatness and Spectral Entropy. We considered audio frames of 50 milliseconds in order to compute the selected features. The features were extracted using the *MIRToolbox* [3], a *Matlab*[1] toolbox dedicated to the extraction of musical features from audio files. The features were normalized in order to have zero mean and unitary standard deviation (Z-Score). The neural network consisted in:

- one input layer X with $P = 18$ neurons, where each neuron corresponds to a certain audio feature
- one hidden layer Z with $M = 50$ neurons, where the number of neurons was experimentally set up
- one output layer Y with $K = 40$ neurons, where each neuron corresponds to a certain term defined by the *Contextual-related Semantic Model*

We used a Sigmoid function for both the activation function $\sigma(\cdot)$ and the transformation $g_k(\cdot)$. The back-propagation algorithm was limited to a maximum of 800 iterations and the learning rate γ_r was defined as an adaptive parameter that updates by following L-BFGS specifics [68]. In order to prevent the model from overfitting, we added a weight decay term to the error function with a tuning parameter $\lambda = 10^{-4}$.

5.2 Music Semantic Survey

We designed and implemented an online survey called *Music Semantic Survey*, in order to collect data about semantic properties of the terms defined by the *Contextual-related Semantic Model.* The survey was in English and it was available online from January 15th to February 16th, 2014. The web technologies used in order to implement it are HTML, PHP, JavaScript and CSS. The survey was divided in two parts.

In the first part of the survey we asked people to assign the 40 terms to the contexts defined by our model. A subset of randomly chosen terms was proposed to each testers, who selected the contexts in which the terms assume a meaning (figure 5.1). A total of 135 people took the survey. At the end of the first part, on average each term was evaluated 68 times.

[1] MathWorks Matlab, `http://www.mathworks.com/products/matlab/`

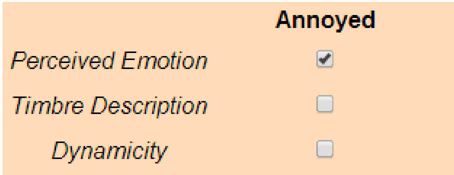


Figure 5.1: Layout of the first part of the survey

In order to assign each term t_i to a certain context $\psi \in \Psi = \{1 \Rightarrow$ *Perceived Emotion* $, 2 \Rightarrow$ *Timbre Description*$, 3 \Rightarrow$ *Dynamicity*$\}$, we first computed the following ratio:

$$r(t_i, \psi) = \frac{N_{t_i}^{\psi}}{N_{t_i}}, \tag{5.1}$$

where $N_{t_i}^{\psi}$ is the number of times that the testers assigned the term t_i to the context ψ and N_{t_i} is the total number of annotations for the term t_i. We assigned the term t_i to the context ψ when the ratio exceeds a threshold ξ:

$$r(t_i, \psi) > \xi. \tag{5.2}$$

We experimentally set up the threshold $\xi = 0.7$. In table 5.1 we show the set of terms obtained for each contexts.

In the second part of the survey, 170 people were asked to quantify the semantic similarity between pairs of terms that they assigned to the same context. A list of pairs of terms was proposed to each tester, that annotated the similarity by setting a slider, as shown in figure 5.2. In the second part of the survey we collected at least three semantic similarity annotations for each pair of terms.

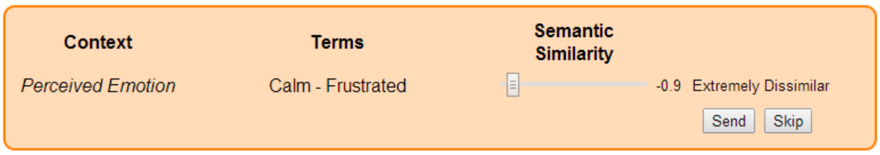


Figure 5.2: Layout of the second part of the survey

Given a pair of terms $(t_i, t_j) \in \psi$, the n-th tester annotated their semantic relation with a value $a(n)^{\psi} \in [-1, 1]$, where -1 means complete semantic dissimilarity, 0 means semantic neutrality and 1 means complete semantic similarity. We modeled their semantic similarity in the context ψ as the

Perceived Emotion	Timbre Description
Aggressive	Bright
Angry	Clean
Annoyed	Dark
Anxious	Hard
Boring	Harsh
Calm	Heavy
Carefree	Rough
Cheerful	Smooth
Dark	Soft
Depressed	Warm
Exciting	
Frustrated	**Dynamicity**
Fun	Calm
Funny	Dynamic
Happy	Fast
Joyful	Flowing
Light	Quiet
Nervous	Relaxed
Quiet	Slow
Relaxed	Static
Sad	Stuttering
Serious	
Sweet	
Tender	
Tense	

Table 5.1: List of terms for each context, obtained through the survey

mean of the annotations:

$$s_{ij}^{\psi} = \frac{1}{N_{ij}^{\psi}} \sum_{n=1}^{N_{ij}^{\psi}} a(n)_{ij}^{\psi}, \tag{5.3}$$

where:

- N_{ij}^{ψ} is the number of annotations for the pair (t_i, t_j), with $t_i, t_j \in \psi$
- $\{a(1)_{ij}^{\psi}, ..., a(N_{ij}^{\psi})_{ij}^{\psi}\}$ is the set of gathered annotations for the pair (t_i, t_j)

The results have been normalized in the range $[0, 1]$ in order to use them in a vector space model. In appendix B we show the results of the survey.

We observed similar annotation results among English mother tongue testers and the other participants for both parts of the survey, thus we believe that the results are not biased by the language knowledge of the testers.

5.3 Model Evaluation

In order to evaluate the performances of the analyzed semantic models, we proposed a test to 30 subjects. The complete text of the test is provided in appendix C. During the test, subjects were left alone. Each subject made one only test. They were asked to answer to a questionnaire with three evaluation sections:

- Predefined Query Evaluation
- Models Comparison
- Overall System Evaluation

In order to analyze possible deviations in the results we also collected information about how frequently the testers listen to music. 50% of the subjects have been classified as *beginners*, since they declared to listen to music less than three hours a day, 27% of subjects have been classified as *experts*, since they affirmed to listen to music more than 3 hours a day, and 23% have been classified as *professionals*, since they claimed to work in a field related to music. In figure 5.3 we show this distribution. In the next paragraphs we discuss the obtained results for each section of the questionnaire.

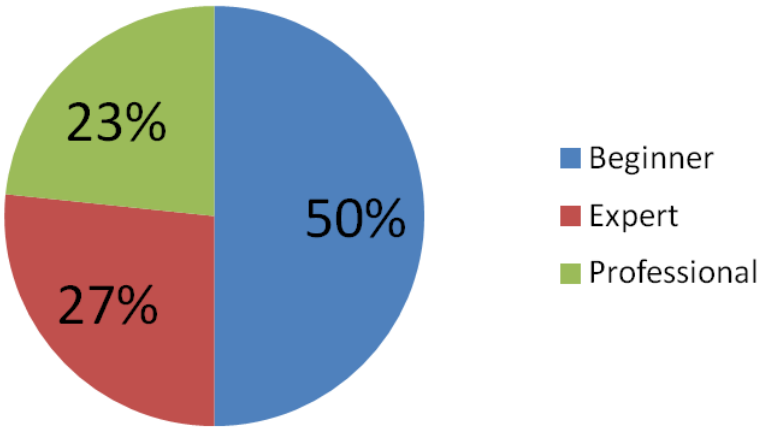


Figure 5.3: Music listening profiles in the test population

5.3.1 Predefined Query Evaluation

Nine predefined queries have been proposed to the subjects. They were asked to evaluate the playlist generated by the three semantic model for each query with a rate between 1 and 9, where 5 indicates a neutral mark, rates higher than 5 indicate a positive evaluation and rates lower than 5 indicate a negative evaluation. The subjects were unaware about which model produced the playlist that they were evaluating. In the following, we discuss the subject evaluation for each predefined query. For each model we show mean, standard deviation and mode of the rates.

I want a highly relaxed and depressed song The evaluation of the query "*I want a highly relaxed and depressed song*" is shown in table 5.2. Our model obtained a mode rate of 8, while the LSI model and the *Janas* models respectively obtained a mode rate of 7 and 6. Nevertheless, the LSI model obtained the best mean rating, but it has been subjected to a higher standard deviaton. We did not notice any substantial difference in the evaluation among the subject categories.

I would like to listen to a moderately angry track The evaluation of the query "*I would like to listen to a moderately angry track*" is shown in table 5.3. Our model and the LSI model obtained the same mode rate of 7. The LSI model achieved a mean rate of 6.87, that is slightly higher than

	Mean	**Std**	**Mode**
Janas Semantic Model	5.83	1.51	6
Context-related Semantic Model	6.77	1.14	**8**
LSI Model	**6.9**	1.47	7

Table 5.2: Evaluation of the first question for the three semantic models

the mean rate obtained by our model. The original *Janas* model performed very bad and obtained an average evaluation of 3.4.

	Mean	**Std**	**Mode**
Janas Semantic Model	3.4	1.90	3
Context-related Semantic Model	6.73	1.39	**7**
LSI Model	**6.87**	1.50	**7**

Table 5.3: Evaluation of the second question for the three semantic models

I want a happy and rather exciting music piece The evaluation of the query "*I want a happy and rather exciting music piece*" is shown in table 5.4. Our *Contextual-related semantic model* outperformed the other two models. In particular, it obtained a mean rate of 7.07 and a mode rate of 8. The mean of rates assigned by professionals is 7.43, that is slightly better than the general mean. The LSI obtained positive results, with a mean rate of 6.6 and a mode rate of 6. The 57% of the subjects assigned to the *Janas* model a negative evaluation. On the overall it obtained a neutral mode rate and a negative mean rate of 4.3.

	Mean	**Std**	**Mode**
Janas Semantic Model	4.3	1.93	5
Context-related Semantic Model	**7.07**	1.23	**8**
LSI Model	6.6	1.35	6

Table 5.4: Evaluation of the third question for the three semantic models

Give me a tender and considerably bright song The evaluation of the query "*Give me a tender and considerably bright song*" is shown in table 5.5. In this question, our model achieved the best retrieval performances, obtaining a mode rate equal to 8 and a mean rate of 7.1. The performances of the LSI model were positive, with a mode rate equal to 7 and a mean rate of 6.37. The subjects assigned on the overall a netrual evaluation to the *Janas* model .

	Mean	Std	Mode
Janas Semantic Model	5.03	1.3	5
Context-related Semantic Model	**7.1**	1.18	**8**
LSI Model	6.37	1.1	7

Table 5.5: Evaluation of the fourth question for the three semantic models

Retrieve a little relaxed, somewhat bright and static song The evaluation of the query "*Retrieve a little relaxed, somewhat bright and static song*" is shown in table 5.6. The LSI and the *Janas* model obtained a mode rate equal to 8, while our model achieved only a mode rate equal to 6. Nevertheless, professionals rated our model with a mode value of 8 and the LSI model with a mode value of 7. The mean rates of the three models are very similar.

	Mean	Std	Mode
Janas Semantic Model	**6.87**	1.38	8
Context-related Semantic Model	6.57	1.79	6
LSI Model	6.63	1.65	**8**

Table 5.6: Evaluation of the fifth question for the three semantic models

I would like to listen to a dynamic and quite a bit carefree track The evaluation of the query "*I would like to listen to a dynamic and quite a bit carefree track*" is shown in table 5.7. Our model clearly outperformed the other ones. It obtained an average rate of 7.53 and a mode rate equal to 8. It is interesting to notice that none of the testers evaluated the performances of our model with negative rates for this question. The LSI model obtained a mean rate equal to 6.2 and a mode rate of 6, while the *Janas* model obtained a mean rate of 5.83 and a mode rate of 5.

	Mean	Std	Mode
Janas Semantic Model	5.83	1.62	5
Context-related Semantic Model	**7.53**	1.11	**8**
LSI Model	6.2	1.27	6

Table 5.7: Evaluation of the sixth question for the three semantic models

Please give me a hard, slightly aggressive and fast song The evaluation of the query "*Please give me a hard, slightly aggressive and fast song*" is shown in table 5.8. Both our model and the *Janas* model obtained a mode rate equal to 5, while the LSI model achieved a mode rate of 6. The *Contextual-related semantic model* obtained the best mean rate, equal to 5.87.

	Mean	**Std**	**Mode**
Janas Semantic Model	5.33	1.32	5
Context-related Semantic Model	**5.87**	1.50	5
LSI Model	5.83	1.29	**6**

Table 5.8: Evaluation of the seventh question for the three semantic models

Give me a little frustrated and partly calm song The evaluation of the query "*Give me a little frustrated and partly calm song*" is shown in table 5.9. Our model achieved the best performances for this query, obtaining a mean rate of 7.07 and a mode rate of 9. In particular, 23.3% of the subjects evaluated it with the maximum rate. LSI model obtained a mean rate equal to 5.83 and a mode rate of 6. *Janas* model performed badly to this query. In fact, 63.3% of the testers assigned a negative rate to the retrieval performances obtained with this model. On the overall, *Janas* model obtained an average rate of 4.27 and a mode rate equal to 4.

	Mean	**Std**	**Mode**
Janas Semantic Model	4.27	1.34	4
Context-related Semantic Model	**7.07**	1.62	**9**
LSI Model	5.83	1.23	6

Table 5.9: Evaluation of the eighth question for the three semantic models

Give me a mainly dark, quite flowing and partly nervous track The evaluation of the query "*Give me a mainly dark, quite flowing and partly nervous track*" is shown in table 5.10. This complex query is particularly interesting because it includes high-level description of emotional, rhythmical and timbral aspects. We noticed that our model is the only one that obtained positive mean and mode rates. In particular, 96.67% of the subjects positively evaluated the performances of the *Contextual-related semantic model*, while only 36.67% and 53.33% respectively assigned positive rates to the *Janas* and LSI models.

	Mean	Std	Mode
Janas Semantic Model	4.17	1.66	3
Context-related Semantic Model	**6.53**	1.43	**6**
LSI Model	4.7	1.42	5

Table 5.10: Evaluation of the ninth question for the three semantic models

5.3.2 Models Comparison

The subjects were asked to try some free-text queries. Finally, they had to evaluate the overall performances of each of the three models, taking into consideration the retrieving performances obtained with the predefined queries and with free-text queries. The results are presented in table 5.11.

	Mean	Std	Mode
Janas Semantic Model	5.03	1.3	5
Context-related Semantic Model	**7.1**	1.18	**8**
LSI Model	6.37	1.1	7

Table 5.11: Overall evaluation for the three semantic models

On the overall, the subjects evaluated our model with highest rates. The mode of the ratings for our model is 8, agreed by 36.66% of the testers. On the overall, the LSI model obtained positive results, while the original *Janas* semantic model has been evaluated on average with neutral rates. It is interesting to notice that professional subjects preferred our model. In fact they evaluated our model with an average rate of 7.43, while at the same time they assigned an average rate of 6.14 to the LSI model and 5.43 to the original *Janas* semantic model.

5.3.3 Overall System Evaluation

In the last section of the questionnaire the subjects were asked to give an overall evaluation of the system. The results are reported in table 5.12. The subjects evaluated positively the overall system, its usefulness and the possibility to use it in real life, assigning a mode rate equal to 7 to all the questions.

5.3.4 Result Analysis

The obtained results show that our *Contextual-related semantic model* clearly outperformed the original semantic model proposed in [9], that combined ED

	Mean	Std	Mode
Do you think this system is useful?	7.07	1.46	7
Would you ever use this kind of system?	6.59	1.69	7
Taking into account the results, the idea of semantic research and the implementation, the functionalities, usefulness and potentials, how do you evaluate the system in general?	7.34	1.4	7

Table 5.12: Overall evaluation of the system

and NED descriptors. The LSI model obtained positive evaluations for some predefined queries, but in general the testers preferred the retrieving performances of our model. By defining three different musical contexts, our *Contextual-related semantic model* is particularly useful when the user query contains multiple terms belonging to different contexts. The other models instead, are not able to distinguish the contexts and they define semantic relations between terms even if they belong to different contexts.

At the end of the questionnaire, the subjects were asked to express some considerations. Some testers referred that they found the dataset of the system too small, producing similar results to different queries. Other subjects suggested to add a genre specification, in order to retrieve only songs with a similar high-level description that belong to the same genre.

Chapter 6

Conclusions and Future Developments

In this chapter we review the work presented in the thesis and we provide a list of possible applications and future developments for this study.

6.1 Conclusions

In this thesis we proposed a new approach for high-level description of music content. The purpose of the work is to define a music-related semantic model that represents music description in a dimensional space composed by three different contexts. The contexts are: *perceived emotion*, *timbre description* and *dynamicity*. Contrary to the most popular dimensional semantic models that define descriptor as points in a space, our approach is focused on the semantic relation between pairs of descriptors belonging to the same context. The semantic relations between descriptors, as well as their contexts membership, have been manually annotated through an online survey.

Our work belongs to the Music Information Retrieval research field. It aims at building an effective music search application that allows users to retrieve music content by semantic description. In order to evaluate the performances of our model, we integrated it in a music search engine based on textual queries [9]. The retrieving results of our system have been compared to the results obtained with other two music description approaches, the approach originally used in [9] (*Janas*) and a co-occurrence approach based on Latent Semantic Indexing [2], a popular model for music retrieval. We conducted an experiment in order to collect ratings of these different models. Testers evaluated our model as the best one, followed by LSI and *Janas* approaches. In particular, our model outperformed the other ones when complex queries containing multiple terms in different contexts are requested by the user. In fact, our model is the only one able to map music descriptors on multiple semantic spaces defined by the contexts. Overall, the testers appreciated the idea of the system.

We believe that our model can be easily integrated in commercial retrieval systems that make use of much bigger music collections.

6.2 Future Developments

In this section we present several future developments and applications that could derive from this work.

6.2.1 Contextual-related Semantic Model Refinement

In this work we defined three music contexts: *Perceived Emotion*, *Timbre Description* and *Dynamicity.* Each context contains a set of specific terms, for which we estimated their semantic relations through a survey. In a future implementation, this innovative approach could be expanded with the definition of new contexts for music description, such as the *Genre* and the *Player Performance.*

Moreover, several terms in different music contexts may be semantically related. For example, the affective term *Aggressive* and the dynamic term *Fast* have a semantic relation even if they do not share the same context. A next development of the model may introduce a formal relation weight between terms belonging to different contexts.

As a proof of concept, our implementation contained a total of 40 popular terms for music description. A further improvement of the system consists in the introduction of new terms in all the contexts.

6.2.2 Dataset Expansion, Semantic Web Integration and Social Tagging

In our work we used a dataset that includes 380 song excerpts of 15 seconds each. Expanding the size of the dataset may produce more accurate results for the user. Furthermore, a future development may consist in adding song meta information by using a music-based Semantic Web service, such as *MusicBrainz*[1]. This approach could leads to semantically enriched queries, like: "*I want to listen to a happy 1962 jazzy track by John Coltrane recorded at Van Gelder Studio*". Since our *Contextual-related semantic model* deals

[1]MusicBrainz, `http://musicbrainz.org/`

with the semantic similarity of music descriptor, it can be also used in order to enrich the song annotations gathered from a social tagging platform, such as Last.fm.

6.2.3 User Personalization

High-level description of music carries a great semantic significance, but at the same time it is highly subjective. A possible improvement consists in building a description model suited to users, where semantic relations between terms in different contexts are personalized. With this approach we obtain a personalized model, biased by the user semantic interpretation of terms and songs. Nevertheless, a manual personalization process implies a high cognitive load for the users. Thus, an automatic process that infers user's semantic perception should be preferred.

6.2.4 Speech-driven Query

Emerging technologies aims at facilitating the interaction between computers and human. New applications like *Google Glass*[2] allow users to express query by speech for retrieving multimedia information. Future development of our work may consist in the integration of the system with a speech-driven query module.

6.2.5 Online Music Streaming

Our system is scalable, since it is based on a content-based approach. Therefore, it could be easily integrated as a plugin into an online music streaming service, such as *Spotify*, in order to provide an alternative music browsing experience.

6.2.6 Time-varying Description

Our system dealt with song excerpts of 15 second each, that have been annotated with one or more high-level descriptors, representing the overall music content. In order to capture the song evolution over time, it could be interesting to study emotion-related and non emotion-related descriptors in a time-varying fashion. This could allow users to interrogate the system with queries like: "*Give me a song that is happy for at least 30 second and then it is anxious for 15 second*".

[2] Google Glass Project, `http://www.google.com/glass/`

Appendices

Appendix A

List of Songs

In the following part we list the songs in the dataset of the system. The songs that have been automatically annotated are indicated in bold.

Artist	Album	Title
1	1	I Can t Believe
1	1	Sweet
10cc		**For you and i**
3 Doors Down	The Better Life	Be Like That
3 Doors Down	The Better Life	Duck And Run
Aaron neville		**Tell it like it is**
Abba	Arrival	Tiger
Abba	Voulez-Vous	The King Has Lost His Crown
Abba	Voulez-Vous	Does Your Mother Know
Abc		**Poison arrow**
AC/DC	Back In Black	What Do You Do For Money Honey
AC/DC	Back In Black	Hells Bells
Ac dc		**Dirty deeds done dirt cheap**
Ace of Base	The Sign	Happy Nation
Aerosmith		**Dude looks like a lady**
Aerosmith	Nine Lives	Taste Of India
Aerosmith	Live Bootleg	Back In The Saddle
Aerosmith	A Little South of Sanity - Disk 1	Same Old Song And Dance
Aerosmith	Nine Lives	Pink
Aimee mann		**Wise up**
Air		**Sexy boy**
Al green		**Sha-la-la make me happy**
Alan Jackson	Who I Am	Let s Get Back To Me And You
Alan Jackson	Who I Am	All American Country Boy
Alanis Morissette	MTV Unplugged	Ironic
Alice cooper		**Elected**
Alice DeeJay	Who Needs Guitars Anyway	Celebrate Our Love
Alice in chains		**No excuses**
Alicia keys		**Fallin**
All Saints	All Saints	Never Ever
All Saints	All Saints	Lady Marmalade
Allman brothers band		**Melissa**
Ani difranco		**Crime for crime**
Andrews sisters		**Boogie woogie bugle boy**
Animals		**Im crying**
Antonio carlos jobim		**Wave**
Aphex twin		**Come to daddy**
Aqua	Aquarium	Calling You
Aqua	Aquarius	Cuba Libre
Aqua	Aquarium	My Oh My

Aretha franklin		**Dont play that song**
Art tatum		**Willow weep for me**
Ashford and simpson		**Solid**
Association		**Windy**
A tribe called quest		**Bonita applebum**
Backstreet Boys	Black Blue	Everyone
Backstreet boys		**As long as you love me**
Bad Brains	I Against I	House Of Suffering
Badly drawn boy		**All possibilities**
Band		**King harvest has surely come**
Barenaked ladies		**Its all been done**
Barry white		**Cant get enough of your love babe**
B.b. king		**Sweet little angel**
BBMak	Sooner Or Later	Love On The Outside
Beatles	A Hard Day s Night	If I Fell
Beatles	Magical Mystery Tour	All You Need Is Love
Beatles	Beatles For Sale	Everybody s Trying To Be My Baby
Beatles	A Hard Day s Night	And I Love Her
Beatles	Beatles For Sale	Rock And Roll Music
Beatles		The Long And Winding Road
Beatles		**Strawberry fields forever**
Bee gees		**Stayin alive**
Ben Folds Five	Whatever And Ever Amen	Brick
Ben folds five		**Brick**
Billie holiday		**God bless the child**
Billy Joel	Piano Man	Captain Jack
Billy Joel	The Stranger	Scenes From an Italian Restaurant
Billy joel		**We didnt start the fire**
Black sabbath		**Black sabbath**
Blind Melon	Blind Melon	Holyman
Blink 182	Enema Of The State	Dysentery Gary
Blood Sweat Tears	Blood Sweat Tears	Spinning Wheel
Bloodhound Gang	One Fierce Beer Coaster	Shut Up
Blue oyster cult		**Burnin for you**
Blur		**Country house**
Bob Dylan	Live at Budokan Disc 1	Ballad of a thin man
Bobby womack		**Womans gotta have it**
Bon Jovi	New Jersey	Living In Sin
Bon Jovi	Slippery When Wet	Livin On a Prayer
Bonnie tyler		**Total eclipse of the heart**
Boston	Boston	Foreplay Long Time
Boston		**More than a feeling**
Brad sucks		**Overreacting**
Breeders		**Cannonball**
Bruce springsteen		**Badlands**
Bruce Springsteen	Live 1975-1985 disc 3	The Promised Land
Bryan Adams	On A Day Like Today	Inside Out
Bryan Adams	On A Day Like Today	Where Angels Fear To Tread
Bryan Adams	So Far So Good	Cuts Like A Knife
Bryan adams		**Cuts like a knife**
Buddy holly		**Peggy sue**
Buena vista social club		**El cuarto de tula**
Buggles		**Video killed the radio star**
Busta Rhymes	Extinction Level Event - The Final World Front	Just Give It To Me Raw
Busta Rhymes	Anarchy	Here We Go Again
Byrds		**Wasnt born to follow**
Cab calloway		**Minnie the moocher**
Cake		**Perhaps**
Cake	Fashion Nugget	She ll Come Back To Me
Cardigans		**Lovefool**
Carly simon		**Youre so vain**
Charles mingus		**Mood indigo**
Cheap Trick	Silver	Day Tripper
Cheap Trick	Silver - Disc 1	World s Greatest Lover
Chet baker		**These foolish things**

Chicago	Chicago X	Gently I ll Wake You
Christina Aguilera	Christina Aguilera	So Emotional
Christina Aguilera	Christina Aguilera	I Turn To You
Chumbawamba	Tubthumper	Amnesia
Chumbawamba		**Tubthumping**
Cilla black		**Alfie**
Clash		**Lost in the supermarket**
Coldplay		**Clocks**
Collective Soul	Hints Allegations and Things Left Unsaid	Breathe
Collective Soul	Collective Soul	Gel
Collective Soul	Hints Allegations and Things Left Unsaid	Wasting Time
Counting Crows	This Desert Life	Hanginaround
Counting Crows	Across A Wire - Live In NYC From The Ten Spot CD 2	Raining In Baltimore
Counting Crows	August and Everything After	Perfect Blue Buildings
Counting Crows	Across A Wire - Live In NYC From The Ten Spot CD 2	Round Here
Craig David	Born To Do It	Last Night
Cream		**Tales of brave ulysses**
Creedence clearwater revival		**Travelin band**
Creedence Clearwater Revival	Pendulum	It s Just A Thought
Creedence Clearwater Revival	Cosmo s Factory	Before You Accuse Me
Crosby stills and nash		**Guinnevere**
Cyndi lauper		**Money changes everything**
Cypress Hill	IV	Dead Men Tell No Tales
Cypress Hill	Live at the Fillmore	Riot Starter
D'Angelo	Voodoo	Chicken Grease
Dave Matthews Band	Live at Red Rocks 8 15 95 Disc 1	Best Of What s Around
Dave Matthews Band	R.E.M.ember Two Things	The Song That Jane Likes
De la soul		**Eye know**
Dead kennedys		**Chemical warfare**
Def Leppard	Adrenalize	I Wanna Touch U
Deftones	White Pony	Rx Queen
Depeche mode		**World in my eyes**
Depeche Mode	People Are People	People Are People
Devo		**Girl u want**
Dido		**Here with me**
Dionne warwick		**Walk on by**
Dire straits		**Money for nothing**
Disturbed	The Sickness	Down With The Sickness
Disturbed	The Sickness	Voices
Dixie Chicks	Wide Open Spaces	Never Say Die
Dixie Chicks	Wide Open Spaces	Give It Up Or Let Me Go
DMX	Flesh Of My Flesh Blood Of My Blood	Bring Your Whole Crew
Don McLean	Favorites And Rarities - Disc 1	American Pie
Donovan		**Catch the wind**
Dr. Dre	00	Forgot About Dre ft Eminem
Duran Duran	Arena	Hungry Like The Wolf
Elvis Presley	Elvis Christmas Album	I Believe
Eminem		**My fault**
Enya	Watermark	Orinoco Flow
Erasure		**Chains of love**
Erasure	Chorus	Joan
Eric Clapton	Crossroads 2 Disc 4	Kind hearted woman
Eric Clapton	Crossroads 2 Disc 2	Layla
Eric Clapton	Unplugged	Tears in Heaven
Eric Clapton	Unplugged	Old Love
Eric clapton		**Wonderful tonight**
Eurythmics		**Sweet dreams**

Evanescence		**My immortal**
Everclear	So Much For The Afterglow	I Will Buy You A New Life
Everclear	Sparkle And Fade	Pale Green Stars
Everlast	Whitey Ford Sings the Blues	Hot To Death
Everlast	Eat At Whitey s	I Can t Move
Everlast	Whitey Ford Sings the Blues	Years
Everything but the Girl	Amplified Heart	Rollercoaster
Everything but the Girl	Amplified Heart	Missing
Faith no more		**Epic**
Fatboy Slim	You ve Come a Long Way Baby	Kalifornia
Finger Eleven	The Greyest Of Blue Skies	Suffocate
Finger Eleven	The Greyest Of Blue Skies	Famous
Fleetwood Mac	The Dance	Dreams
Fleetwood mac		**Say you love me**
Flying burrito brothers		**Break my mind**
Foo fighters		**Big me**
Foreigner	Agent Provocateur	I Want To Know What Love Is
Franz ferdinand		**Come on home**
Garbage	Garbage	Only Happy When It Rains
Garth Brooks	Ropin The Wind The Limited Series	Which One Of Them
Garth Brooks	The Chase	Learning To Live Again
Gary Wright	The Dream Weaver	Made To Love You
Genesis	From Genesis To Revelation Disky version	In The Wilderness
Genesis	Live - The Way We Walk - Volume One - The Shorts	Jesus He Knows Me
Genesis		**Cuckoo cocoon**
George harrison		**All things must pass**
Green Day	Dookie	Burnout
Huey Lewis and the News	Fore	I Never Walk Alone
Ja Rule	Venni Vetti Vecci	World s Most Dangerous feat Nemesis
James brown		**Give it up or turnit a loose**
Jamiroquai		**Little l**
Janet Jackson	Rhythm Nation 1814	Someday Is Tonight
Jeff buckley		**Last goodbye**
Jennifer Paige	Jennifer Paige	Always You
Jennifer Paige	Jennifer Paige	Between You and Me
Jerry lee lewis		**Great balls of fire**
Jessica Andrews	Who Am I	Who Am I
Jimi Hendrix Experience	Are You Experienced	The Wind Cries Mary
Jimi hendrix		**Highway chile**
Joe Cocker	Joe Cocker Live	When The Night Comes
John cale		**Pablo picasso**
John coltrane		**Giant steps**
John Denver	An Evening With John Denver - Disc 2	Take Me Home Country Roads
John lee hooker		**Boom boom**
Joy division		**Love will tear us apart**
Junior murvin		**Police and thieves**
King crimson		**Thela hun ginjeet**
Keith Sweat	Keith Sweat	Chocolate Girl
Kenny Loggins	Outside from the Redwoods	Now And Then
Kraftwerk		**Spacelab**
Kris kristofferson		**The best of all possible worlds**
La Bouche	Sweet Dreams	Fallin In Love
Lara Fabian	Lara Fabian	I am Who I am
Lauryn Hill	The Miseducation of Lauryn Hill	Final Hour
Led Zeppelin	In Through The Out Door	Carouselambra
Led Zeppelin	Led Zeppelin I	You Shook Me
Led zeppelin		**Immigrant song**
Leonard cohen		**Suzanne**
Les Rythmes Digitales	Darkdancer	Take a Little Time
Les Rythmes Digitales	Darkdancer	Sometimes
Lifehouse	No Name Face	Sick Cycle Carousel
Lifehouse	No Name Face	Quasimodo

Live	The Distance To Here	Run to the Water
Live	Throwing Copper	Waitress
LL Cool J	mr smith	I Shot Ya
LL Cool J	G O A T	Imagine That
LL Cool J	G O A T	Back Where I Belong
Lou Bega	A Little Bit Of Mambo	Mambo Mambo
Lou Bega	A Little Bit Of Mambo	The Trumpet Part II
Lou reed		**Walk on the wild side**
Louis armstrong		**Hotter than that**
Lynyrd Skynyrd	Lyve From Steel Town CD 1	Saturday Night Special
Lynyrd skynyrd		**Sweet home alabama**
Madison Avenue	Polyester Embassy	Who The Hell Are You Original Mix
Marilyn Manson	Holy Wood	Coma Black
Marilyn Manson	The Last Tour On Earth	Astonishing Panorama Of the Endtimes
Marvin Gaye	Let s Get It On	Let s Get It On
Marvin gaye		**Whats going on**
Me First and the Gimme Gimmes	Are a Drag	Stepping Out
Metallica		**One**
Michael Jackson	Off The Wall	Rock With You
Michael Jackson	Thriller	Human Nature
Michael Jackson	Off The Wall	Working Day And Night
Michael jackson		**Billie jean**
Miles davis		**Blue in green**
Montell Jordan	Get It On Tonight	let s cuddle up featuring LOCKDOWN
Moby		**Porcelain**
Modest mouse		**What people are made of**
Montell Jordan	This Is How We Do It	Down On My Knees
Morrissey		**Everyday is like sunday**
Mudvayne	L d 50	Prod
Mudvayne	L d 50	Internal Primates Forever
MxPx	On The Cover	No Brain
Mystikal	Let s Get Ready	**Mystikal Fever**
Natalie imbruglia		**Torn**
Neil Diamond	Hot August Night - Disc 1	Sweet Caroline
Neil Diamond	Hot August Night Disk 2	Canta Libre
Neil Diamond	Hot August Night - Disc 1	Shilo
Neil Young	Harvest	Words Between The Lines Of Age
New Radicals	Maybe You ve Been Brainwashed Too	Technicolor Lover
New Radicals	Maybe You ve Been Brainwashed Too	I Don t Wanna Die Anymore
Next	Welcome II Nextasy	Cybersex
Nine Inch Nails	The Fragile Right	The Big Come Down
Nine inch nails		**Head like a hole**
No doubt		**Artificial sweetener**
No doubt		**Simple kind of life**
Norah jones		**Dont know why**
Oasis		**Supersonic**
Olivia Newton-John	Olivia	Summer Nights Grease
Our Lady Peace	Happiness Is Not A Fish That You Can Catch	Blister
Papa Roach	Infest	Broken Home
Paula Abdul	Forever Your Girl	Opposites Attract
Pennywise	Straight Ahead	Might Be a Dream
Pennywise	Straight Ahead	Straight Ahead
Phil Collins	But Seriously	Heat On The Street
Phil Collins	But Seriously	I Wish It Would Rain Down
Phil Collins	Hello I Must Be Going	Thru These Walls
Pink floyd		**Echoes**
Pixies		**Wave of mutilation**
Pj harvey		**Dry**
Placebo	Black Market Music	Passive Aggressive
Portishead		**All mine**
Primus		**Jerry was a race car driver**
Queen	The Game	Save Me
Queen	The Works	**I Go Crazy**

Queen	Live Magic	Is This The World We Created
Queen	The Works	Is This The World We Created
Queen		**We will rock you**
R.E.M.	Dead Letter Office	Burning Hell
R.E.M.	Dead Letter Office	Femme Fatale
Radiohead	OK Computer	No Surprises
Radiohead		**Karma police**
Rage Against the Machine	Renegades	Microphone Fiend
Rancid	and out Come the Wolves	As Wicked
Red hot chili peppers		**Give it away**
Richard Marx	Repeat Offender	Satisfied
Robert johnson		**Sweet home chicago**
Rod Stewart	Vagabond Heart	Rebel Heart
Rod Stewart	Vagabond Heart	If Only
Rod Stewart	Vagabond Heart	Have I Told You Lately
Rolling Stones	Tattoo You	Worried About You
Roxette	Look Sharp	Dance Away
Roxette	Joyride	**soul deep**
Run-D.M.C.	Raising Hell	Hit It Run
Sade	Love Deluxe	Like A Tattoo
Sade	Sade LOVERS ROCK	LOVERS ROCK
Savage Garden	Affirmation	The Animal Song
Scorpions	World Wide Live	Make It Real
Seven Mary Three	American Standard	Anything
Shakira		**The one**
Shania Twain	Come On Over	Honey I m Home
Shania Twain	The Woman In Me	Home Ain t Where His Heart Is Anymore
Sheryl Crow	Live from Central Park	There Goes The Neighborhood
Sisqo	Unleash The Dragon	Unleash The Dragon feat Beanie Sigel
Smiths		**How soon is now**
Sonic youth		**Teen age riot**
Sonny rollins		**Strode rode**
Soul Asylum	Grave Dancers Union	Somebody To Shove
Soundgarden		**Black hole sun**
Spencer davis group		**Gimme some lovin**
Spice girls		**Stop**
Spineshank	Strictly Diesel	Slipper
Spineshank	Strictly Diesel	While My Guitar Gently Weeps
Stan getz		**Corcovado quiet nights of quiet stars**
Steppenwolf		**Born to be wild**
Steve Winwood	Back in the High Life	Split Decision
Stevie Wonder	Songs in the Key of Life Disc 2	Isn t She Lovely
Stevie Wonder	Songs in the Key of Life Disc 2	As
Stevie Wonder	Songs In The Key Of Life Disc 1	Sir Duke
Sting		**Big lie small world**
Stone Temple Pilots	Tiny Music Songs from the Vatican Gift Shop	Adhesive
Stranglers		**Golden brown**
Stroke 9	Nasty Little Thoughts	One Time
Styx	Return To Paradise Disc 2	Fooling Yourself The Angry Young Man
Styx	Return To Paradise Disc 2	Show Me The Way
Styx	The Grand Illusion	Come Sail Away
Talking heads		**And she was**
The Bangles	Different Light	Following
The Bee Gees	Here At Last Bee Gees Live Disc Two	Down The Road
The Cardigans	Gran Turismo	Starter
The Chemical Brothers	Surrender	Out of Control
The Corrs	In Blue	Somebody for someone
The Cranberries	No Need To Argue	Ridiculous Thoughts
The Cranberries	No Need To Argue	Yeat s Grave

The Everly Brothers	The Fabulous Style of	All I Have To Do Is Dream
The Human League	The Very Best of	Heart Like A Wheel
The Police	Live Disc One - Orpheum WBCN Boston Broadcast	Hole In My Life
The Police	Live Disc Two - Atlanta Synchronicity Concert	Walking In Your Footsteps
The Police	Live Disc Two - Atlanta Synchronicity Concert	So Lonely
The Presidents of the United States of America	unknown	Body
The Verve	Urban Hymns	Weeping Willow
Thelonious monk		**Epistrophy**
Tim McGraw	A Place In The Sun	Somebody Must Be Prayin For Me
Tina Turner	Tina Live In Europe CD 1	What s Love Got To Do With It
TLC	FanMail	Don t Pull Out On Me Yet
Toby Keith	How Do You Like Me Now	Do I Know You
Todd rundgren		**Bang the drum all day**
Toni Braxton	Secrets	Come On Over Here
Toni Braxton	Toni Braxton	I Belong to You
Tool	Aenima	Stinkfist
Tool	Aenima	Hooker with a Penis
Tricky		**Christiansands**
U2	All That You Can t Leave Behind	Elevation
Ugly Kid Joe	America s Least Wanted	Cats In The Cradle
Ultravox		**Dancing with tears in my eyes**
Van Halen	98	House of Pain
Wade Hayes	Old Enough To Know Better	Kentucky Bluebird
Weezer		**Buddy holly**
Wes montgomery		**Bumpin**
Westlife	Westlife	I Need You
White stripes		**Hotel yorba**
White Zombie	Supersexy Swingin Sounds	Electric Head Pt Satan in High Heels Mix
Whitney Houston	Whitney Houston	Greatest Love Of All
Wu-Tang Clan	Wu-Tang Forever Disc 2	Dog Shit
Wu-Tang Clan	Enter The Wu-Tang 36 Chambers	WuTang th Chamber Part II
Wu-Tang Clan	u-Tang Forever Disc one	Reunited
Xzibit	Restless	Rimz Tirez feat Defari Goldie Loc Kokane
Xzibit	Restless	D N A DRUGSNALKAHOL feat Snoop Dogg

Appendix B

Semantic Similarity

In the following part we attach the semantic similarity between terms obtained through a survey. The semantic similarity has been defined in the range $[-1, 1]$, where -1 represent opposite semantic (opposite meaning) and 1 represent same semantic (same meaning). When two terms are semantically independent, their semantic similarity is 0.

Perceived Emotion

Term 1	**Term 2**	**Semantic Similarity**
Aggressive	Angry	0.3
Aggressive	Annoyed	-0.1
Aggressive	Anxious	0.32
Aggressive	Boring	-0.575
Aggressive	Calm	-0.8
Aggressive	Carefree	-0.267
Aggressive	Cheerful	-0.675
Aggressive	Dark	0.033
Aggressive	Depressed	-0.6
Aggressive	Exciting	0.2
Aggressive	Frustrated	0.375
Aggressive	Fun	-0.6
Aggressive	Funny	-0.575
Aggressive	Happy	-0.35
Aggressive	Joyful	0.075
Aggressive	Light	-0.575
Aggressive	Nervous	0.225
Aggressive	Quiet	-0.967
Aggressive	Relaxed	-0.933
Aggressive	Sad	-0.175
Aggressive	Serious	-0.167
Aggressive	Sweet	-0.85
Aggressive	Tender	-0.9
Aggressive	Tense	0.2
Angry	Annoyed	-0.467
Angry	Anxious	0.15
Angry	Boring	-0.375
Angry	Calm	-0.78
Angry	Carefree	-0.7
Angry	Cheerful	-0.575
Angry	Dark	0.625
Angry	Depressed	0.425
Angry	Exciting	-0.275

Angry	Frustrated	0.575
Angry	Fun	-0.75
Angry	Funny	-0.275
Angry	Happy	-0.88
Angry	Joyful	-0.767
Angry	Light	-0.467
Angry	Nervous	0.475
Angry	Quiet	-0.85
Angry	Relaxed	-0.5
Angry	Sad	0.075
Angry	Serious	0.2
Angry	Sweet	-0.85
Angry	Tender	-0.825
Angry	Tense	0.533
Annoyed	Anxious	-0.15
Annoyed	Boring	0.275
Annoyed	Calm	-0.133
Annoyed	Carefree	-0.8
Annoyed	Cheerful	-0.525
Annoyed	Dark	-0.45
Annoyed	Depressed	0.45
Annoyed	Exciting	-0.733
Annoyed	Frustrated	0.3
Annoyed	Fun	-0.925
Annoyed	Funny	-0.78
Annoyed	Happy	-0.85
Annoyed	Joyful	-0.825
Annoyed	Light	-0.567
Annoyed	Nervous	0.225
Annoyed	Quiet	0.233
Annoyed	Relaxed	-0.967
Annoyed	Sad	0.1
Annoyed	Serious	0.1
Annoyed	Sweet	-0.44
Annoyed	Tender	-0.333
Annoyed	Tense	-0.275
Anxious	Boring	-0.66
Anxious	Calm	-0.5
Anxious	Carefree	-0.65
Anxious	Cheerful	-0.5
Anxious	Dark	0.225
Anxious	Depressed	0.2
Anxious	Exciting	-0.8
Anxious	Frustrated	0.567
Anxious	Fun	-0.675
Anxious	Funny	-0.6
Anxious	Happy	-0.575
Anxious	Joyful	-0.2
Anxious	Light	-0.667
Anxious	Nervous	0.725
Anxious	Quiet	-0.875
Anxious	Relaxed	-0.675
Anxious	Sad	-0.133
Anxious	Serious	-0.067
Anxious	Sweet	-0.34
Anxious	Tender	-0.3
Anxious	Tense	0.775
Boring	Calm	0.167
Boring	Carefree	-0.6
Boring	Cheerful	-0.867
Boring	Dark	0.467
Boring	Depressed	-0.05
Boring	Exciting	-0.5
Boring	Frustrated	0.3
Boring	Fun	-0.925
Boring	Funny	-0.733
Boring	Happy	-0.75
Boring	Joyful	-0.375
Boring	Light	-0.067
Boring	Nervous	-0.1
Boring	Quiet	0.2
Boring	Relaxed	-0.35

Boring	Sad	0.1
Boring	Serious	-0.18
Boring	Sweet	-0.55
Boring	Tender	0.05
Boring	Tense	-0.6
Calm	Carefree	0
Calm	Cheerful	0.125
Calm	Dark	-0.175
Calm	Depressed	0.175
Calm	Exciting	-0.867
Calm	Frustrated	-0.46
Calm	Fun	-0.05
Calm	Funny	0.02
Calm	Happy	0.24
Calm	Joyful	-0.025
Calm	Light	0.325
Calm	Nervous	-1
Calm	Quiet	0.933
Calm	Relaxed	0.8
Calm	Sad	-0.05
Calm	Serious	0.225
Calm	Sweet	0.425
Calm	Tender	0.275
Calm	Tense	-0.925
Carefree	Cheerful	0.375
Carefree	Dark	-0.7
Carefree	Depressed	-0.633
Carefree	Exciting	-0.025
Carefree	Frustrated	-0.825
Carefree	Fun	0.333
Carefree	Funny	0.5
Carefree	Happy	0.767
Carefree	Joyful	0.5
Carefree	Light	0.05
Carefree	Nervous	-0.967
Carefree	Quiet	0.25
Carefree	Relaxed	0.35
Carefree	Sad	-0.6
Carefree	Serious	0.033
Carefree	Sweet	0.52
Carefree	Tender	-0.033
Carefree	Tense	-0.85
Cheerful	Dark	-0.7
Cheerful	Depressed	-0.85
Cheerful	Exciting	0.45
Cheerful	Frustrated	-0.533
Cheerful	Fun	0.52
Cheerful	Funny	0.5
Cheerful	Happy	0.675
Cheerful	Joyful	0.82
Cheerful	Light	0.3
Cheerful	Nervous	-0.4
Cheerful	Quiet	-0.25
Cheerful	Relaxed	0.433
Cheerful	Sad	-0.75
Cheerful	Serious	-0.6
Cheerful	Sweet	0.5
Cheerful	Tender	-0.125
Cheerful	Tense	-0.5
Dark	Depressed	0.68
Dark	Exciting	-0.333
Dark	Frustrated	0.125
Dark	Fun	-0.76
Dark	Funny	-0.375
Dark	Happy	-0.9
Dark	Joyful	-0.725
Dark	Light	-0.74
Dark	Nervous	0.267
Dark	Quiet	0.533
Dark	Relaxed	-0.325
Dark	Sad	0.175
Dark	Serious	0

Dark	Sweet	-0.325
Dark	Tender	-0.433
Dark	Tense	0.25
Depressed	Exciting	-1
Depressed	Frustrated	0.45
Depressed	Fun	-1
Depressed	Funny	-0.775
Depressed	Happy	-0.9
Depressed	Joyful	-0.9
Depressed	Light	-0.825
Depressed	Nervous	0.1
Depressed	Quiet	-0.525
Depressed	Relaxed	-0.05
Depressed	Sad	0.725
Depressed	Serious	-0.15
Depressed	Sweet	-0.375
Depressed	Tender	-0.325
Depressed	Tense	0.167
Exciting	Frustrated	-0.42
Exciting	Fun	0.5
Exciting	Funny	0.4
Exciting	Happy	0.5
Exciting	Joyful	0.333
Exciting	Light	0.033
Exciting	Nervous	0.075
Exciting	Quiet	-0.48
Exciting	Relaxed	-0.5
Exciting	Sad	-0.625
Exciting	Serious	-0.7
Exciting	Sweet	-0.275
Exciting	Tender	-0.4
Exciting	Tense	-0.1
Frustrated	Fun	-0.725
Frustrated	Funny	-0.45
Frustrated	Happy	-0.825
Frustrated	Joyful	-0.925
Frustrated	Light	-0.8
Frustrated	Nervous	0.64
Frustrated	Quiet	-0.725
Frustrated	Relaxed	-0.45
Frustrated	Sad	0.367
Frustrated	Serious	0.067
Frustrated	Sweet	-0.867
Frustrated	Tender	-0.325
Frustrated	Tense	0.475
Fun	Funny	0.267
Fun	Happy	0.625
Fun	Joyful	0.675
Fun	Light	0.183
Fun	Nervous	0.167
Fun	Quiet	0.025
Fun	Relaxed	-0.075
Fun	Sad	-0.85
Fun	Serious	-0.9
Fun	Sweet	0.18
Fun	Tender	-0.133
Fun	Tense	-0.32
Funny	Happy	0.325
Funny	Joyful	0.575
Funny	Light	0.225
Funny	Nervous	-0.525
Funny	Quiet	0.025
Funny	Relaxed	0.05
Funny	Sad	-0.833
Funny	Serious	-0.875
Funny	Sweet	0.45
Funny	Tender	-0.05
Funny	Tense	-0.233
Happy	Joyful	0.975
Happy	Light	0.375
Happy	Nervous	-0.325
Happy	Quiet	0

Happy	Relaxed	0.525
Happy	Sad	-1
Happy	Serious	-0.32
Happy	Sweet	0.48
Happy	Tender	0.3
Happy	Tense	-0.45
Joyful	Light	0.35
Joyful	Nervous	-0.48
Joyful	Quiet	-0.125
Joyful	Relaxed	0.14
Joyful	Sad	-0.925
Joyful	Serious	-0.625
Joyful	Sweet	0.433
Joyful	Tender	0.633
Joyful	Tense	-0.85
Light	Nervous	-0.833
Light	Quiet	0.4
Light	Relaxed	0.3
Light	Sad	-0.55
Light	Serious	-0.625
Light	Sweet	0.7
Light	Tender	0.3
Light	Tense	-0.733
Nervous	Quiet	-0.84
Nervous	Relaxed	-1
Nervous	Sad	-0.14
Nervous	Serious	0.125
Nervous	Sweet	-0.567
Nervous	Tender	-0.625
Nervous	Tense	-0.033
Quiet	Relaxed	0.675
Quiet	Sad	0.033
Quiet	Serious	0.3
Quiet	Sweet	0.48
Quiet	Tender	0.4
Quiet	Tense	-0.52
Relaxed	Sad	0.167
Relaxed	Serious	-0.16
Relaxed	Sweet	0.55
Relaxed	Tender	0.58
Relaxed	Tense	-0.667
Sad	Serious	0.25
Sad	Sweet	-0.275
Sad	Tender	-0.15
Sad	Tense	0.133
Serious	Sweet	-0.35
Serious	Tender	-0.325
Serious	Tense	0.25
Sweet	Tender	0.8
Sweet	Tense	-0.475
Tender	Tense	-0.15

Timbre Description

Term 1	**Term 2**	**Semantic Similarity**
Bright	Clean	0.54
Bright	Dark	-1
Bright	Hard	-0.1
Bright	Harsh	-0.667
Bright	Heavy	-0.867
Bright	Rough	-0.467
Bright	Smooth	0.35
Bright	Soft	0.4
Bright	Warm	-0.025
Clean	Dark	-0.3
Clean	Hard	-0.375
Clean	Harsh	-0.75
Clean	Heavy	-0.5

Clean	Rough	-0.9
Clean	Smooth	0.75
Clean	Soft	0.5
Clean	Warm	0.067
Dark	Hard	0.375
Dark	Harsh	0.233
Dark	Heavy	0.175
Dark	Rough	0.2
Dark	Smooth	-0.325
Dark	Soft	-0.1
Dark	Warm	-0.1
Hard	Harsh	0.4
Hard	Heavy	0.867
Hard	Rough	0.475
Hard	Smooth	-0.675
Hard	Soft	-1
Hard	Warm	-0.567
Harsh	Heavy	0.167
Harsh	Rough	0.725
Harsh	Smooth	-0.925
Harsh	Soft	-0.75
Harsh	Warm	-0.5
Heavy	Rough	0.35
Heavy	Smooth	-0.4
Heavy	Soft	-0.675
Heavy	Warm	-0.35
Rough	Smooth	-0.65
Rough	Soft	-0.7
Rough	Warm	-0.625
Smooth	Soft	0.6
Smooth	Warm	0.575
Soft	Warm	0.433

Dynamicity

Term 1	**Term 2**	**Semantic Similarity**
Calm	Dynamic	-0.65
Calm	Fast	-0.5
Calm	Flowing	0.133
Calm	Quiet	0.833
Calm	Relaxed	0.867
Calm	Slow	0.55
Calm	Static	0.3
Calm	Stuttering	-0.7
Dynamic	Fast	0.2
Dynamic	Flowing	0.3
Dynamic	Quiet	-0.7
Dynamic	Relaxed	-0.58
Dynamic	Slow	-0.675
Dynamic	Static	-1
Dynamic	Stuttering	-0.175
Fast	Flowing	-0.267
Fast	Quiet	-0.75
Fast	Relaxed	-0.933
Fast	Slow	-1
Fast	Static	-0.78
Fast	Stuttering	-0.433
Flowing	Quiet	-0.067
Flowing	Relaxed	0.033
Flowing	Slow	-0.1
Flowing	Static	-0.85
Flowing	Stuttering	-0.775
Quiet	Relaxed	0.3
Quiet	Slow	0.367
Quiet	Static	0.65
Quiet	Stuttering	-0.275
Relaxed	Slow	0.575
Relaxed	Static	0.1

Relaxed	Stuttering	-0.567
Slow	Static	0.467
Slow	Stuttering	-0.375
Static	Stuttering	-0.533

Appendix C

Model Evaluation Test

In the following part we attach the questionnaire used for the evaluation of the system.

Semantic Models Comparison

What kind of listener are you? Please choose only one answer

Beginner (I listen to music less than three hours a day)	
Expert (I listen to music more than three hours a day)	
Professional (I listen to music also for reasons related to my job)	

Predefined queries

Please test these queries and evaluate the quality of results with a mark in a 9 point-scale, where 1 means very bad and 9 is the optimum. Quality is intended as the correspondence of songs results with respect to the query content. 5 indicates a neutral mark.

Query	*1* (1-9)	*2* (1-9)	*3* (1-9)
1) I want a highly relaxed and depressed song			
2) I would like to listen to a moderately angry track			
3) I want a happy and rather exciting music piece			
4) Give me a tender and considerably bright song			
5) Retrieve a little relaxed, somewhat bright and static song			
6) I would like to listen to a dynamic and quite a bit carefree track			
7) Please give me a hard, slightly aggressive and fast song			
8) Give me a little frustrated and partly calm song			
9) Give me a mainly dark, quite flowing and partly nervous track			

Free-text queries

Please try some free-text queries and do evaluate the performances.

Please consider the attached list of currently available adjectives and qualifiers in order to compose the query.

System Evaluation

Please evaluate the overall performances of the three systems:

	1	2	3	4	5	6	7	8	9
Model 1									
Model 2									
Model 3									

General Evaluation

	1	2	3	4	5	6	7	8	9
Do you think this system is useful? (1: not at all - 5 can't really say - 9 : very useful)									
Would you ever use this kind of system? (1: not at all. 5: I don't know. 9: Yes, very often)									
Taking into account the results, the idea of semantic research and the implementation, the functionalities, usefulness and potentials, how do you evaluate the system in general? (1: very bad. 5: neutral. 9: very good)									

Please indicate optional notes

Available Adjectives			
Aggressive	Dark	Hard	Serious
Angry	Depressed	Harsh	Slow
Annoyed	Dynamic	Heavy	Smooth
Anxious	Exciting	Joyful	Soft
Boring	Fast	Light	Static
Bright	Flowing	Nervous	Stuttering
Calm	Frustrated	Quiet	Sweet
Carefree	Fun	Relaxed	Tender
Cheerful	Funny	Rough	Tense
Clean	Happy	Sad	Warm

Available Qualifiers		
a little	highly	quite
average	in-between	quite a bit
completely	mainly	rather
considerably	medium	slightly
extremely	moderately	somewhat
fairly	not	very
fully	not at all	very much
hardly	partly	

Bibliography

[1] M. Buccoli, "A music search engine based on semantic text based queries," Master's thesis, Politecnico di Milano, 2013.

[2] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[3] O. Lartillot, P. Toiviainen, and T. Eerola, "A matlab toolbox for music information retrieval," 2008, pp. 261–268. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-78246-9_31

[4] Y.-H. Yang and H. H. Chen, *Music Emotion Recognition.* CRC Press, 2011.

[5] B. Rohrmann, "Verbal qualifiers for rating scales: Sociolinguistic considerations and psychometric data," *Project Report. University of Melbourne, Australia*, 2003.

[6] J. H. Lee and J. S. Downie, "Survey of music information needs, uses, and seeking behaviours: Preliminary findings," in *In ISMIR Proceedings*, 2004, pp. 441–446.

[7] Ò. Celma, P. Herrera, and X. Serra, "Bridging the semantic gap."

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

[9] M. Buccoli, A. Sarti, M. Zanoni, and S. Tubaro, "A music search engine based on semantic text-based query," in *Multimedia Signal Processing (MMSP), 2013 IEEE 15th International Workshop on*, 2013.

[10] M. M. Bradley and P. J. Lang, "Affective norms for english words (ANEW): Stimuli, instruction manual, and affective ratings," Center

for Research in Psychophysiology, University of Florida, Gainesville, Florida, Tech. Rep., 1999.

[11] M. Lesaffre, L. D. Voogdt, M. Leman, B. D. Baets, H. D. Meyer, and J.-P. Martens, "How potential users of music search and retrieval systems describe the semantic quality of music." *JASIST*, vol. 59, no. 5, pp. 695–707, 2008. [Online]. Available: http://dblp.uni-trier.de/db/journals/jasis/jasis59.html#LesaffreVLBMM08

[12] T. V. Wal, "Folksonomy coinage and definition," 2007. [Online]. Available: http://vanderwal.net/folksonomy.html

[13] J. Surowiecki, *The Wisdom of Crowds.* Anchor, 2005.

[14] C. Marlow, M. Naaman, D. Boyd, and M. Davis, "Ht06, tagging paper, taxonomy, flickr, academic article, to read," in *Proceedings of the Seventeenth Conference on Hypertext and Hypermedia*, ser. HYPERTEXT '06. New York, NY, USA: ACM, 2006, pp. 31–40. [Online]. Available: http://doi.acm.org/10.1145/1149941.1149949

[15] M. Ames and M. Naaman, "Why we tag: Motivations for annotation in mobile and online media," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '07. New York, NY, USA: ACM, 2007, pp. 971–980. [Online]. Available: http://doi.acm.org/10.1145/1240624.1240772

[16] P. Lamere and E. Pampalk, "Social tags and music information retrieval." in *ISMIR*, J. P. Bello, E. Chew, and D. Turnbull, Eds., 2008, p. 24. [Online]. Available: http://dblp.uni-trier.de/db/conf/ismir/ismir2008.html#LamereP08

[17] M. Kuhn, R. Wattenhofer, and S. Welten, "Social audio features for advanced music retrieval interfaces," in *Proceedings of the International Conference on Multimedia*, ser. MM '10. New York, NY, USA: ACM, 2010, pp. 411–420. [Online]. Available: http://doi.acm.org/10.1145/1873951.1874007

[18] Ó. Celma, "Music recommendation and discovery in the long tail," Ph.D. dissertation, University Pompeu Fabra, Barcelona, Spain, 2008. [Online]. Available: http://www.iua.upf.edu/~ocelma/PhD/

[19] D. Turnbull, L. Barrington, D. A. Torres, and G. R. G. Lanckriet, "Semantic annotation and retrieval of music and sound effects." *IEEE Transactions on Audio, Speech & Language Processing*,

vol. 16, no. 2, pp. 467–476, 2008. [Online]. Available: http://dblp.uni-trier.de/db/journals/taslp/taslp16.html#TurnbullBTL08

[20] M. Levy and M. Sandler, "A semantic space for music derived from social tags," in *8th International Conference on Music Information Retrieval (ISMIR 2007)*, 2007. [Online]. Available: http://ismir2007.ismir.net/proceedings/ISMIR2007_p411_levy.pdf

[21] ——, "Learning latent semantic models for music from social tags," *Journal of New Music Research*, vol. 37, no. 2, pp. 137–150, 2008.

[22] C. Wu and Y. Guo, "A semantic relatedness service based on folksonomy," in *Information Science and Digital Content Technology (ICIDT), 2012 8th International Conference on*, vol. 3, June 2012, pp. 506–511.

[23] C. Cattuto, D. Benz, A. Hotho, and G. Stumme, "Semantic analysis of tag similarity measures in collaborative tagging systems," *CoRR*, vol. abs/0805.2045, pp. 39–43, 2008. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr0805.html#abs-0805-2045

[24] P. J. Morrison, "Tagging and searching: Search retrieval effectiveness of folksonomies on the world wide web," *Inf. Process. Manage.*, vol. 44, no. 4, pp. 1562–1579, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1377474

[25] W3C, "W3c semantic web activity," http://www.w3.org/2001/sw/, 2001, [Online; accessed 2014-03-01].

[26] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, May 2001. [Online]. Available: http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21

[27] A. Passant and Y. Raimond, "Combining social music and semantic web for music-related recommender systems," in *Social Data on the Web Workshop of 7th International Semantic Web Conference*, Karlsruhe, Deutschland, Oktober 2008.

[28] Ò. Celma, "Foafing the music: Bridging the semantic gap in music recommendation," in *Proceedings of 5th International Semantic Web Conference*, Athens, GA, USA, 2006, pp. 927–934. [Online]. Available: http://dx.doi.org/10.1007/11926078_67

[29] M. Sordo, F. Gouyon, and L. Sarmento, "A method for obtaining semantic facets of music tags," in *Workshop on Music Recommendation and Discovery, ACM Conference on Recommender Systems*, Barcelona, 2010. [Online]. Available: http://www.inescporto.pt/~fgouyon/docs/SordoGouyonSarmento_WOMRAD2010.pdf

[30] I. Tatli and A. Birturk, "A tag-based hybrid music recommendation system using semantic relations and multi-domain information." in *ICDM Workshops*, M. Spiliopoulou, H. Wang, D. J. Cook, J. Pei, W. W. 0010, O. R. Zaï¿½ane, and X. Wu, Eds. IEEE, 2011, pp. 548–554. [Online]. Available: http://dblp.uni-trier.de/db/conf/icdm/icdmw2011.html#TatliB11

[31] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing," *Int. J. Hum.-Comput. Stud.*, vol. 43, no. 5-6, pp. 907–928, Dec. 1995. [Online]. Available: http://dx.doi.org/10.1006/ijhc.1995.1081

[32] G. A. Miller, "Wordnet: A lexical database for english," *COMMUNICATIONS OF THE ACM*, vol. 38, pp. 39–41, 1995.

[33] H. H. Kim, "A semantically enhanced tag-based music recommendation using emotion ontology." in *ACIIDS (2)*, ser. Lecture Notes in Computer Science, A. Selamat, N. T. Nguyen, and H. Haron, Eds., vol. 7803. Springer, 2013, pp. 119–128. [Online]. Available: http://dblp.uni-trier.de/db/conf/aciids/aciids2013-2.html#Kim13

[34] J. Wang, X. Chen, Y. Hu, and T. Feng, "Predicting high-level music semantics using social tags via ontology-based reasoning," in *Proceedings of the 11th International Society for Music Information Retrieval Conference*, Utrecht, The Netherlands, August 9-13 2010, pp. 405–410, http://ismir2010.ismir.net/proceedings/ismir2010-69.pdf.

[35] S. Rho, S. Song, E. Hwang, and M. Kim, "Comus: Ontological and rule-based reasoning for music recommendation system." in *PAKDD*, ser. Lecture Notes in Computer Science, T. Theeramunkong, B. Kijsirikul, N. Cercone, and T. B. Ho, Eds., vol. 5476. Springer, 2009, pp. 859–866. [Online]. Available: http://dblp.uni-trier.de/db/conf/pakdd/pakdd2009.html#RhoSHK09

[36] L. Chiarandini, M. Zanoni, and A. Sarti, "A system for dynamic playlist generation driven by multimodal control signals and descriptors," in

Multimedia Signal Processing (MMSP), 2011 IEEE 13th International Workshop on. IEEE, 2011, pp. 1–6.

[37] M. Slaney, "Semantic-audio retrieval." in *ICASSP*. IEEE, 2002, pp. 4108–4111. [Online]. Available: http://dblp.uni-trier.de/db/conf/icassp/icassp2002.html#Slaney02

[38] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Towards musical query-by-semantic-description using the cal500 data set," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '07. New York, NY, USA: ACM, 2007, pp. 439–446. [Online]. Available: http://doi.acm.org/10.1145/1277741.1277817

[39] K. O. P. N. Johnson-Laird, "The language of emotions: An analysis of a semantic field," *Cognition and Emotion*, vol. 3:2, pp. 81–123, 1989.

[40] C. C. Moore, A. K. Romney, T. L. Hsia, and C. D. Rusch, "The universality of the semantic structure of emotion terms: Methods for the study of inter- and Intra-Cultural variability," *American Anthropologist*, vol. 101, no. 3, pp. 529–546, 1999. [Online]. Available: http://links.jstor.org/sici?sici=0002-7294\%2528199909\%25292\%253A101\%253A3\%253C529\%253ATUOTSS\%253E2.0.CO\%253B2-G

[41] K. Hevner, "Experimental studies of the elements of expression in music," *The American Journal of Psychology*, vol. 48, pp. 246–268, 1936.

[42] M. Zentner, D. Grandjean, and K. R. Scherer, "Emotions evoked by the sound of music: Characterization, classification, and measurement," *Emotion*, vol. 8,4, pp. 494–521, 2008.

[43] X. Hu, J. S. Downie, C. Laurier, M. Bay, and A. F. Ehmann, "The 2007 mirex audio mood classification task: Lessons learned." in *ISMIR*, J. P. Bello, E. Chew, and D. Turnbull, Eds., 2008, pp. 462–467. [Online]. Available: http://dblp.uni-trier.de/db/conf/ismir/ismir2008.html#HuDLBE08

[44] J. H. . A. Gordon, "The deep lexical semantics of emotions," 2008. [Online]. Available: http://ict.usc.edu/files/publications/EMOT08.PDF

[45] C. Strapparava and A. Valitutti, "WordNet-Affect: An affective extension of WordNet," in *Proceedings of the 4th International*

Conference on Language Resources and Evaluation. ELRA, 2004, pp. 1083–1086. [Online]. Available: MISSING

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

[47] G. L. Collier, "Beyond valence and activity in the emotional connotations of music," *Psychology of Music*, vol. 35, no. 1, pp. 110–131, 2007.

[48] Fontaine, R. J. Johnny, Scherer, R. Klaus, Roesch, B. Etienne, Ellsworth, and C. Phoebe, "The world of emotions is not Two-Dimensional," *Psychological Science*, vol. 18, no. 12, pp. 1050–1057, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1111/j.1467-9280.2007.02024.x

[49] M. I. Mandel and D. P. W. Ellis, "A web-based game for collecting music metadata," in *In 8th International Conference on Music Information Retrieval (ISMIR*, 2007.

[50] D. Turnbull, R. Liu, L. Barrington, and G. Lanckriet, "A game-based approach for collecting semantic annotations of music," in *In 8th International Conference on Music Information Retrieval (ISMIR*, 2007.

[51] E. L. M. Law, L. Von Ahn, R. B. Dannenberg, and M. Crawford, "Tagatune: A game for music and sound annotation," in *International Conference on Music Information Retrieval (ISMIR'07)*, 2007, pp. 361–364.

[52] K. Bischoff, C. S. Firan, R. Paiu, W. Nejdl, C. Laurier, and M. Sordo, "Music mood and theme classification - a hybrid approach." in *ISMIR*, K. Hirata, G. Tzanetakis, and K. Yoshii, Eds. International Society for Music Information Retrieval, 2009, pp. 657–662. [Online]. Available: http://dblp.uni-trier.de/db/conf/ismir/ismir2009.html#BischoffFPNLS09

[53] Y.-C. Lin, Y.-H. Yang, and H. H. Chen, "Exploiting online music tags for music emotion classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 7S, no. 1, pp. 26:1–26:16, Nov. 2011. [Online]. Available: http://doi.acm.org/10.1145/2037676.2037683

[54] J. Yao, B. Cui, G. Cong, and Y. Huang, "Evolutionary taxonomy construction from dynamic tag space." *World Wide Web*, vol. 15, no.

5-6, pp. 581–602, 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/www/www15.html#YaoCCH12

[55] J. Aucouturier and F. Pachet, "Improving timbre similarity : How high's the sky ?" *J. Negative Results in Speech and Audio Sciences*, 2004.

[56] Y. L. e. a. Y.H. Yang, *Advances in Multimedia Information Processing - PCM 2008*. Springer Berlin - Heidelberg, 2008, ch. 8, pp. 70–79.

[57] D. Yang and W. Lee, "Disambiguating music emotion using software agents," 2004, pp. 10–14.

[58] C. Laurier, J. Grivolla, and P. Herrera, "Multimodal music mood classification using audio and lyrics," in *In Proceedings of the 7th International Conference on Machine Learning and Applications (ICMLA' 08). December 2008*, 2008.

[59] X. Hu, J. S. Downie, and A. F. Ehmann, "Lyric text mining in music mood classification." in *ISMIR*, K. Hirata, G. Tzanetakis, and K. Yoshii, Eds. International Society for Music Information Retrieval, 2009, pp. 411–416. [Online]. Available: http://dblp.uni-trier.de/db/conf/ismir/ismir2009.html#HuDE09

[60] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference and prediction*, 2nd ed. Springer, 2009. [Online]. Available: http://www-stat.stanford.edu/~tibs/ElemStatLearn/

[61] H.-G. Kim, N. Moreau, and T. Sikora, *MPEG-7 Audio and Beyond: Audio Content Indexing and Retrieval.* John Wiley & Sons, 2005.

[62] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of english: The penn treebank," *COMPUTATIONAL LINGUISTICS*, vol. 19, no. 2, pp. 313–330, 1993.

[63] Y. Kim, E. Schmidt, and L. Emelle, "Moodswings: A collaborative game for music mood label collection." in *ISMIR*, J. P. Bello, E. Chew, and D. Turnbull, Eds., 2008, pp. 231–236. [Online]. Available: http://dblp.uni-trier.de/db/conf/ismir/ismir2008.html#KimSE08

[64] M. E. Davies and M. D. Plumbley, "Context-dependent beat tracking of musical audio," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 3, pp. 1009–1020, 2007.

[65] J. Duchi, "Derivations for linear algebra and optimization," pp. 13.

[66] S. Bird, "Nltk: the natural language toolkit," in *Proceedings of the COLING/ACL on Interactive presentation sessions*, ser. COLING-ACL '06. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006, pp. 69–72. [Online]. Available: http://dx.doi.org/10.3115/1225403.1225421

[67] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000.

[68] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM J. Sci. Comput.*, vol. 16, no. 5, pp. 1190–1208, Sep. 1995. [Online]. Available: http://dx.doi.org/10.1137/0916069