



SALERO

D5.3.2 Report on Context Based Audio Feature Extraction

SALERO Deliverable 5.3.2



Report on Context Based Audio Feature Extraction

SALERO Deliverable D5.3.2

SALERO identifier: SALERO_D5.3.2_ContextBasedAudioFeatureExtraction
-v10.doc

Deliverable number: D 5.3.2

Author(s) and company: G. Holmberg (UPF), P. Cano (UPF), J. Jose (UG),
Charlie Cullen (DIT)

Work package / task: WP5

Document status: Final

Confidentiality: Public

Version	Date	Reason of change
1	2007-10-09	Document created (UPF)
2	2007-10-14	Filled Introduction, Related work, and experiments' sections
3	2007-10-25	Added several contributions from the partners
4	2007-10-31	Formatting altogether
5	2007-11-17	Added latest changes from partners
6	2007-27-11	Internal Reviewed by URL
7	30/11/2007	Proof-read & final formatting

The work presented in this document was partially supported by the European Community under the Information Society Technologies (IST) priority of the 6th framework programme for R&D.

This document does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of its content.

This document contains material, which is the copyright of certain SALERO consortium parties, and may not be reproduced or copied without permission. All SALERO consortium parties have agreed to full publication of this document. The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the SALERO consortium as a whole, nor a certain party of the SALERO consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk, and does not accept any liability for loss or damage suffered by any person using this information.

Table of Contents

Table of Contents	iv
1 Executive Summary	1
2 Introduction	2
2.1 Purpose of this document	2
2.2 Related Documents.....	2
3 Context vs. Content based Audio Retrieval	3
3.1 Introduction.....	3
3.1.1 <i>Semantic descriptors</i>	4
3.1.2 <i>Manual annotation</i>	4
3.2 Related work	4
3.2.1 <i>Limitations of pure Content-based analysis</i>	4
3.2.2 <i>Limitations when bridging the Semantic Gap via bottom-up Machine Learning</i>	5
3.2.3 <i>Moods and emotions as a paradigmatic example</i>	7
4 The Notion of Context	8
4.1 User Context	8
4.2 Information Retrieval (Query) Context	9
5 Context Based Audio Retrieval	11
5.1 Temporal Context.....	11
5.1.1 <i>Audio Analysis</i>	11
5.1.2 <i>Text to speech analysis</i>	12
5.1.3 <i>Speech Analysis</i>	13
5.2 Context Aware Similarity	17
5.2.1 <i>Dimensionality Reduction with RBF Neural Network</i>	17
5.2.2 <i>Query Sensitive Music Descriptor Generation Scheme</i>	18
6 Experimental Results	20
6.1 Text-to-speech analysis and synthesis	20
6.1.1 <i>Unit-selection module</i>	20
6.1.2 <i>Prosodic modeling</i>	21
6.1.3 <i>Discriminating expressive speech styles by voice quality parameterization</i>	22
6.2 Tag propagation based on audio similarity	24
6.2.1 <i>Experiments</i>	24
6.2.2 <i>Style labels</i>	25
6.2.3 <i>Mood labels</i>	27
6.2.4 <i>Conclusions</i>	29
6.3 Query Sensitive Music Descriptor Generation	29

6.3.1	<i>Test Configuration</i>	29
6.3.2	<i>Experimental Results</i>	30
	Query Effectiveness	30
	Query and Training Efficiency	32
6.3.3	<i>The query accuracy and training trade-off</i>	33
6.3.4	<i>Summary of Experiments</i>	35
7	Conclusions	36
8	References	37
9	Glossary	40

1 Executive Summary

This document describes investigations into using audio and its context to improve the performance of multimedia, and in particular, audio annotation and retrieval systems.

A background review of the role of the audio in retrieval is provided, along with the motivation for using audio information in retrieval, and the importance of the context (versus the content) in this task. Techniques for using audio in context are described, and experiments are presented to show how audio can aid in contextual retrieval.

2 Introduction

2.1 Purpose of this document

Work Package 5 addresses Objective O1: “To research and develop practical methods of context-based information retrieval that simplify the location and retrieval of characters, sounds, images, movements or behaviours from very large datasets and media storage systems.” In the first 18 months, the WP5 conducts studies and developments aimed at setting the scene, and develop content analysis techniques suitable for the underlying media objects and associated context-based retrieval techniques.

This document addresses the need to study and evaluate the use of audio within context based retrieval.

2.2 Related Documents

Before reading this document it is recommended to be familiar with the following documents:

- D5.2.1 Multimedia analysis and representation techniques and Development Research Roadmap
- D5.3.1 Study on Retrieval Improvements using Context Based Audio
- D5.5.1 Development of Prototype Context-based Retrieval System and User Interface

3 Context vs. Content based Audio Retrieval

3.1 Introduction

Traditional IR offers the ability to search and browse large amounts of text documents and presenting results in a “ranked-by-relevance” interface. For the case of multimedia there are two main approaches: The first is to generate textual indices manually, semi automatically or automatically and then use traditional IR. The other approach is to use content-based retrieval, where the query is non textual and a similarity measure is used for searching and retrieval.

Sound effect management systems rely on classical text descriptors to interact with their audio collections. Librarians tag the sounds with textual description and file them under categories. Users can then search for sounds matching keywords as well as navigating through category trees. Audio filing and logging is a labour-intensive error-prone task. Moreover, languages are imprecise, informal and words have several meanings as well as several words for each meaning. Finally, sounds are multi modal, multicultural and multifaceted and there is not an agreement in how to describe them.

Despite the difficulties inherent in creating SFX metadata, there are need to catalogue assets so as to reuse afterward. Media assets have value. As Flank and Brinkman point out, there are many situations where reusing media content is, not only not economically appealing—think of the cost of sending a team to record Emperor penguins in their natural habitat—but sometimes audio cannot be re-recorded—like natural catastrophes or historical events (Flank and Brinkman, 2002). Complete digital media management solutions include media archiving and cataloguing, digital right management and collaborative creative environments. This section focuses on the knowledge management aspects of sound effect descriptions with the purpose of making metadata easily searchable, less expensive to create and reusable to support possible new users—including computers—and applications.

MPEG-7 offers a framework for the description of multimedia documents (Manjunath et al., 2002; Consortium, 2001, 2002). The description tools for describing a single multimedia document consider semantic, structure and content management descriptions. MPEG-7 content semantic description tools describe the actions, objects and context of a scene. In sound effects, this correlates to the physical production of the sound in the real world, “1275 cc Mini Cooper Door Closes” , or the context, “Australian Office Atmos Chatter Telephones”. MPEG-7 content structure tools concentrate on the spatial, temporal and media source structure of multimedia content. Indeed, important descriptors are those that describe the perceptual qualities independently of the source and how they are structured on a mix. Content management tools are organized in three areas: Media information—which describes storage format, media quality and so on, e.g.: “PCM Wav 44100Hz stereo”—, Creation information—which describes the sound generation process, e.g.: who and how created the sound—and finally usage information—which describes the copyrights, availability of the content and so on (Manjunath et al., 2002).

One of the most time-demanding and error-prone task when building a library of sound effects is the correct labelling and placement of a sound within a category. The information retrieval model commonly used in commercial search engines is based on keyword indexing. Librarians add descriptions for the audio. The systems match the descriptions against the users’ query to retrieve the audio. Sounds are difficult to describe with words. Moreover, the librarian must add the text thinking on the different ways a user may eventually look for the sound, e.g.: “dinosaur, monster, growl, roar” and at the same time with the maximum detail. The vagueness of the query specification, normally one or two words, together with the ambiguity and informality of natural languages affects the quality of the search: Some relevant sounds are not retrieved and some irrelevant ones are presented to the user. Sound effect management systems also allow browsing for sounds in manually generated categories. It is difficult to manage large category structures. Big corpses may be labelled by different librarians that follow somewhat different conventions and may not remember under which category sounds should be placed (e.g: Camera:clicks or clicks:camera). Several ways of describing a sound include: source centered description, perceptual, post-production specific and creation description.

3.1.1 *Semantic descriptors*

Semantic descriptors usually refer to the source of the sound, that is, what has physically produced the sound, e.g: "car approaching". They also refer to the context, e.g: "Pub atmos". The importance of source-tagging for sound designers is questioned by L.Mott (1990). Mott explains that the sound engineer should concentrate on the sound independently on what actually produced it because in many occasions the natural sounds do not fulfill the expectations and must be replaced with sounds of distinct origin, e.g: "arrow swishes" or "gun bangs". There are, however, cases where having the true sound can add quality to a production (e.g: Using the real atmosphere of a Marrakesh market tea house). Besides, describing the source of a sound is sometimes easier than describing the sound itself. It is difficult to describe the "moo of a cow" without mentioning "moo or cow" but just perceptual attributes..

3.1.2 *Manual annotation*

Manual annotation of multimedia data is an arduous task, and very time consuming. Automatic annotation methods, normally fine-tuned to reduced domains such as musical instruments or limited to sound effects taxonomies, are not mature enough to label with great detail any possible sound. Yet, in the music domain the annotation becomes more complex due to the time domain frame. The purpose of making sound effects and music easily accessible implies a condition of describing music in such a way that machine learning can understand it [Pachet 2005]. Specifically, these two steps must be followed: to build music descriptions which can be easily maintained, and to exploit these descriptions to build efficient music access systems that help users find music in large collections. There are a lot of ways to describe music content, but we can basically classify the descriptors in three groups: editorial meta-data, cultural meta-data, and acoustic meta-data.

As a paradigmatic example, the Music Genome Project is a big effort to "capture the essence of music at the fundamental level" by using over 400 attributes to describe songs. To achieve this, more than 40 musicologists have been annotating thousands of files since 2000. Based on this knowledge, a well-known system named Pandora¹ creates playlists by exploiting these human-based annotations. It is clear that helping these musicologists can reduce both time and cost of the annotation task.

3.2 Related work

Nowadays, content-based retrieval systems cannot classify, identify and retrieve as well as humans can. This is a common problem in the multimedia field, like in image or video annotation. But in the latter fields many attempts have been made ([Jeon 2003], [Wenyin 2001]). Semantic audio annotation, however, has not been as studied as image or video annotation, except the work by Whitman [Whitman 2005] or Barrington et al. ([Barrington 2007], [Turnbull 2007]). They have made significant advances in semantic annotation of songs for music information retrieval (MIR) using MFCC's to describe music content and HMM's trained on timbre and rhythm for computing similarity between songs. Their idea was basically based on other work that represented image semantic annotation as a supervised multi-classification problem [Carneiro 2005].

3.2.1 *Limitations of pure Content-based analysis*

The single most important reason why automatic Content Analysis is yet today (after more than a decade of research) not fully commercially deployed in any larger scale is the fact that most algorithms perform more or less well only under certain specific circumstances. A task that is quite trivial for a human, as to determine if a song is happy or sad or if it is danceable or not, can be far from easy for a machine to analyze with acceptable degree of accuracy. But at the same time, a task which can be difficult for a human to do with a high level of accuracy, as determining BPM (beats per minute) in music, can today be done with extraordinary precision by a machine [Gouyon 2006].

The need of going from lab-conditions to real-world scenarios has been a big challenge for the research community in content analysis over the past years, especially in the sense of getting robust and scalable algorithms with high enough accuracy over large databases. It is a generally known issue that although promising results may be obtained over a database of some hundreds of files, a more or less linear decrease in accuracy and increase in computational costs can be expected when the data set

¹ <http://www.pandora.com>

gets in the size of millions of files. For example, general audio classification methods normally concentrate on small domains, such as musical instrument classification or very simplified audio taxonomies. Classification state of the art is far from classifying with detail any possible sound, but work reasonably on domain specific tasks.

Furthermore, in automatic classification tasks, researchers normally assume the existence or define a well defined hierarchical classification scheme of a few categories (less than a hundred at the leaves of the tree for isolated samples, e.g: musical instruments and less than ten for classification of stream of audio). In real-world conditions, there is on the contrary no general acceptance of a music genre taxonomy or a fixed classification scheme for all possible sounds.

To tackle this problem, a general sound annotator & classifier was developed by the Music Technology Group in the AUDIOCLAS EU project. To find a reasonable solution to the ontology definition problem, WordNet, a semantic network that organizes real world knowledge in a large scale – over 100,000 concepts – was used [Cano2005]. In order to overcome the need of a huge number of classifiers to distinguish many different sound classes, [Cano2005-2] used a nearest-neighbour classifier with a database of isolated sounds unambiguously linked to WordNet concepts. The results are sufficient for a semi-automatic annotation framework where the system proposes concepts with certain likelihood (e.g.: this could be a “cat purring” or a “car tickling”). Besides improvements on audio features and complicating the classifier, it became clear that in order to achieve higher precision the use of context or hints from other multimedia annotators are needed to remove the ambiguity. For further discussion on classification of general sound, we refer to [Cano2005-2].

Large-scale evaluation is therefore one of the most important tasks in to establish which feature types work well for which audiovisual search tasks. Being adaptable to noisy conditions & high compression rates of new media formats (such as mobile content, podcasts or video-blogging), is another.

Much of the recent research on content analysis and annotation concentrate on material from news archives, digital audio libraries and TV programs/movie. In these cases audio content is typically professionally created, edited and partially annotated during the production phase. The content management and annotation for mobile phone created personal audiovisual material is a less researched area and it poses new challenges for content description tools and algorithms, but it also brings new promising elements such as user & context information.

Large-scale information retrieval of structured, semi-structured, textual and audio-visual data on the web and in industrial enterprises is pushing the boundaries of what it is possible to achieve with current Information Retrieval and Database technology.

The most used information retrieval schemes such as Vector Space Model [Carmel2003, Mass2002, Kakade2005] and Language Model [Croft2003] have also been extended to incorporate structured information, in order to return results with increasing accuracy and precision. A test data collection for evaluating XML retrieval, INEX [INEX2004], has been initialized and has attracted increasing interest and participation from various research groups. Still, most of these search techniques and systems are still in their beginning phase. The need for more complex metadata descriptions for audio-visual content, geospatial data and web-services (MPEG-7, FGDC, Dublic Core) has only exacerbated the problem of large scale indexing across databases.

3.2.2 Limitations when bridging the Semantic Gap via bottom-up Machine Learning

While the bottom-up extraction of features and patterns from audio continues to be a very active research area, it is also clear that there are strict limits as to the kinds of music descriptions that can be directly extracted from the audio signal. When it comes to intuitive, human-centred and contextual characterizations such as “peaceful” or “aggressive music” or highly personal categorizations such as “music I like to listen to while working”, there is little hope of analytically defining audio features that unequivocally and universally define these concepts. Yet such concepts play a central role in the way people organize and interact with and “use” their music.

That is where automatic learning comes in. The only way one can hope to build a machine that can associate such high-level concepts with music items is by having the machine learn the correct associations between low-level audio features and high-level concepts, from examples of music items that have been labelled with the appropriate concepts.

Inductive learning as the automatic construction of classifiers from pre-classified training examples has a long tradition in several subfields of computer science. The field of statistical pattern classification

[Duda2001, Hastie2001] has developed a multitude of methods for deriving classifiers from examples, where a “classifier” can be regarded as a black box that takes as input a new object to be classified (described via a set of features) and outputs a prediction regarding the most likely class the object belongs to. Classifiers are automatically constructed via learning algorithms that take as input a set of example objects labelled with the correct class, and construct a classifier from these that is (more or less) consistent with the given training examples, but also makes predictions on new, unseen objects; that is, the classifier is a generalization of the training examples. Training examples would be music items (e.g., songs) characterized by a list of audio features and labelled with the appropriate high-level concept (e.g., “this is a piece I like to listen to while working”), and the task of the learning algorithm is to produce a classifier that can predict the appropriate high-level concept for new songs (again represented by their audio features).

Common training and classification algorithms in statistical pattern classification [Duda2001] include nearest neighbour classifiers (k-NN), Gaussian Mixture Models, neural networks (mostly multi-layer feed-forward perceptrons), and support vector machines [Christianini2000].

The field of Machine Learning is particularly concerned with algorithms that induce classifiers that are interpretable, i.e., that explicitly describe the criteria that are associated with or define a given class. Typical examples of machine learning algorithms that are also used in music classification are decision trees [Quinlan1986] and rule learning algorithms.

Decision trees [Quinlan1986] are probably the most popular class of classification models in machine learning, and they are widely used also in Music Information Retrieval. In [West2004], for instance, decision tree learning algorithms have been used to build a model of the distribution of frame values.

Because of its known merits, k-NN classification is widely used. Sometimes, the feature values - possibly after feature selection - of each piece are regarded as a vector, and the distance used for k-NN classifier is the Euclidean distance between individual pieces (e.g. [Costa2004, Gouyon2004]) or to representative reference vectors (e.g. [Hellmuth2004, Kastner2004]).

Support Vector Machines (SVMs) are also applied to music classification: e.g. [Xu2003] use them for genre classification, and [Li2003] train several SVMs to recognize mood labels, where each SVM decides if one specific label is present in the music.

Gaussian Mixture Models (GMMs) are useful for estimating the distribution of feature values. They can be used as a classifier by modeling each class as a GMM; an instance is then classified by calculating, for each class (GMM), the likelihood that the instance was produced by the respective GMM, and predicting the class with the maximum likelihood. In [Liu2003], mood detection in classical music is done based on this approach. GMM classifiers have also been used in [Bureed2003, Tzanetakis2002] for genre classification.

Neural Networks have also been applied to music classification: [Costa2004] use a multilayer perceptron to determine the class of a piece given its feature vector. [Hellmuth2004] use a more elaborate approach by training a separate neural network for each class, and an additional one that combines the outputs of these networks.

Learned classifiers must be evaluated empirically, in order to assess the kind of prediction accuracy that may be expected on new, unseen cases. This is essentially done by testing the classifier on new (labelled) examples which have not been used in any way in learning, and recording the rate of prediction errors made by the classifier. There is a multitude of procedures for doing this, and a lot of scientific literature on advantages and shortcomings of the various methods. The basic idea is to set aside a part of the available examples for testing (the “test set”), then inducing the classifier from the remaining data (the “training set”), and then testing the classifier on the test set. A systematic method most commonly used is known as n-fold cross-validation, where the available data set is randomly split into n subsets (“folds”), and the above procedure is carried out n times, each time using one of the n folds for testing, and the remaining n - 1 folds for training. The error (or conversely, accuracy) rates reported in most learning papers are based on experiments of this type. A central issue that deserves some discussion is the training data required for learning. Attractive as the machine learning approach may be, it does require (large) collections of representative labelled training examples, e.g., music recordings with the correct categorization attached. Manually labelling music examples is a very laborious and time-consuming process, especially when it involves listening to the pieces before deciding on the category. Additionally, there is the copyright issue. Ideally, the research community

would like to be able to share common training corpora. If a researcher wants to test her own features in classification experiment, she needs access to the actual audio files.

There are some efforts currently being undertaken in the Music Information Retrieval community to compile large repositories of labelled music that can be made available to all interested researchers without copyright problems. Noteworthy examples of this are Masataka Goto's RWC Music Database (<http://staff.aist.go.jp/m.goto/RWC-MDB>), the IMIRSEL (International Music Information Retrieval System Evaluation Laboratory) project at the University of Illinois at Urbana-Champaign (<http://www.music-ir.org/evaluation>), and the FreeSound Initiative (<http://freesound.iua.upf.edu>) of the Music Technology Group.

All these methods presented in this section deal with the content itself. They do not exploit the contextual information surrounding the objects. Thus, given their limitations, section 4 introduces the notion of context, taking into account: the user as well as the multimedia object. Before introducing the notion of context, though, we briefly present the motivation of music and context, in the context of the user's moods and emotions.

3.2.3 Moods and emotions as a paradigmatic example

A paradigmatic example to bridge the semantic gap are the features related with (music) moods. In this sense, there is some work already done in the area of context based audio retrieval, to infer moods from the audio features, plus the associated context of the song.

People listen to music mostly to change their emotional state like demonstrated in [Juslin 2004] by Juslin and Laukka. Consequently, we tend to think that an automatic system able to build contextual playlists by mood would be appreciated by the users.

When dealing with such a subjective question, we are faced with several issues. One of them is the representation paradigm. Which taxonomy should we use? Which set of tags is relevant? One option is to use basic emotions. There is a huge debate on what should be considered as basic emotions. Many researchers in different fields like in Psychology, Musicology or Cognitive Science have proposed their sets. The choice of basic emotions may differ according to the context. But as advised by Juslin et al. in [Juslin 2001], in order to make our experiment and to build a ground truth that achieve the best agreement between people, we should consider few categories.

But detecting mood in music is a very challenging task. In general, in-depth research on music and emotions are quite recent [Juslin 2001]. In the Music Information Retrieval field, only a few works are dealing with this problem using exclusively audio content ([Shi 2006] [Feng 2003], [Lu 2006] and [Li 2003]) and most of them are using machine learning techniques, training a classifier with some selected features. Although some results are promising like in [Lu 2006], there is no standard or very defined ideas about the categories to use and the features that are working well.

4 The Notion of Context

The notion of context is important in work package 5, and a model of user context has already been outlined in D3.1.2, and D5.5.1. As outlined in these previous documents, two main kinds of context can be considered in the SALERO project:

- The context in which the user works (i.e. the context of queries, interaction, etc.)
- The context in which the information artefacts (Intelligent Multimedia Objects) exist

User context is outlined in section 4.1, following on and refining the models described in reports D3.1.2 and D5.5.1. Section 4.2 covers context from the point of view of the multimedia objects embedded in the media itself.

4.1 User Context

From D3.1.2, we defined a rough contextual hierarchy:

- Organisational context: the roles and/or description of the overall organisation in which one or more users are engaged.
- Individual user context: the overall role or roles of a particular user with an organisation
- Work Task context: a particular task in which a user may be engaged in their work. Such task contexts may typically be seen as domain-dependent, and relates the user's role in the organisation. For example, “project management”, “technical support”, “mentoring” or “training”
- Information seeking context: information seeking or searching can be viewed, as here, as a sub-task, existing within a larger work task context. The information seeking context may be considered as the searching task in which the user is engaged in, in order to resolve their work task.

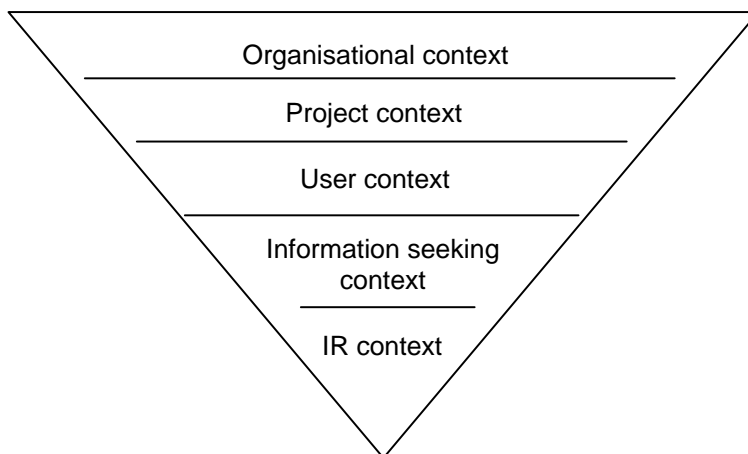


Figure 1: Latest model of user context, following on from D5.5.1 and D3.1.2.

This model was refined in report D5.5.1 to also include a project context, which corresponds to the project or projects which a user may be involved within (such as a media production). The project context may be considered as subservient to the organizational context, or be first class (e.g. a single animation project may involve multiple organisations). Based on these refinements, we can alter the model of context from D3.1.2, illustrated by Figure 1.

These changes aim to better match the user requirements, and better reflect real world complexities. For example, we consider “project” (or “production”) more important than organisation, since many organisations may contribute to a single project, and a single user may play a number of different roles within multiple projects.

4.2 Information Retrieval (Query) Context

The information retrieval, or “query” context is the most short-term of the contexts in , and is designed to capture the short term needs of a user carrying out a specific query. A retrieval context encapsulates the individual actions of the user with the user, their trail of actions when carrying out a single query. The information seeking context captures the trail of different queries which the user can carry out to satisfy their information need.

For example, the user may create a new information seeking context, called “insect like cartoon characters”, which corresponds to a need, that of finding existing cartoon characters which look like insects. Within this information seeking context the user may execute a number of different queries, such as “ants”, “Jiminy cricket”, etc. Each of these separate queries, and their associated query results, exists within a separate “information retrieval” or “query context”. Within each of these individual query contexts, a user may then interact with the system, generating a path of user actions, such as viewing result items, viewing metadata for the results, etc.

Working from the opposite direction, from individual user actions we get:

- a single user action, such as typing a query or highlighting a shot, is contextualised by the retrieval context in which that action has occurred
- a single retrieval (or query) context is a single step in a larger information seeking context
- an information seeking context is part of the larger user context in which it is part

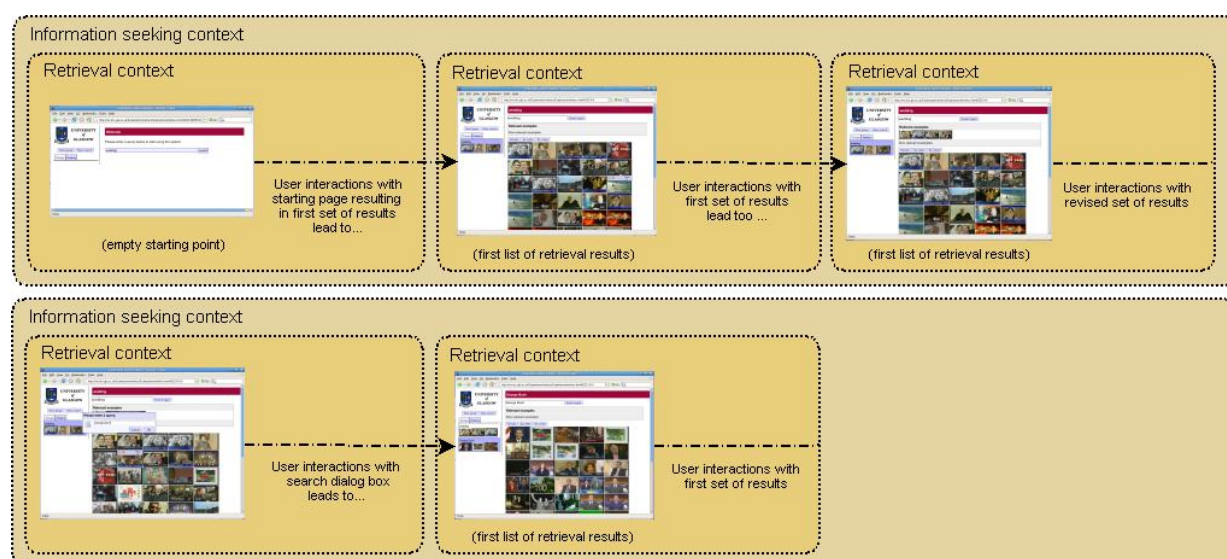


Figure 2: Illustration of retrieval (query) contexts within a larger information seeking context.

Figure 2 illustrates the relationship between retrieval (query) contexts and the larger information seeking contexts in which they are part. Of particular importance in the query retrieval context is the list of user actions which can be tracked by a search interface. This list will obviously change as the actions possible with an interface changes. The actions currently used in the existing prototype described in D5.5.1 are:

- Video shot marked relevant to the user’s information need
- Video shot marked not relevant to the user’s information need
- Video shot is added to a “group”, a user created grouping of shots
- Text query altered
- Mouse over a keyframe on the results list
- Play a video shot

Some of these actions result in a change to the state of the application (for example, marking a shot as relevant will add that shot to the list of relevant shots), while other are tracked simply for their value in understanding or implicitly deriving relevance information. For example, the action of a user clicking a keyframe to start viewing the video clip can be taken as an implicit indication that the user finds that shot relevant to their need. This is valuable in a full system, since this action can be contextualised within an information seeking context, allowing the system to utilise that evidence in other retrievals (including other retrievals by other users).

5 Context Based Audio Retrieval

This chapter outlines the main approaches taken in the SALERO project in tackling audio based retrieval in context. The chapter is split into a number of main sections, outlining the techniques and methodology used by the different partners in approaching audio based retrieval, that is: audio analysis, text to speech analysis, and speech analysis.

5.1 Temporal Context

5.1.1 Audio Analysis

Perceptual descriptors describe the temporal perceptual qualities independently of the source that actually created the sound. Classical research on auditory perception has studied the world of sounds within a multidimensional space with dimensions such as pitch, loudness, duration, timbral brightness, and so on. Since they refer to the properties of sound, sometimes there is a mapping between sound descriptions to perceptual measurable features of the sound. Another possibility to describe sounds is the use of onomatopoeia, words that imitate sounds and are extensively used in comics— e.g: “roar”, “mmm”, “ring”.

To describe the audio it is very usual to decompose the audio signal with spectral as well as temporal descriptors. Now, we present some of the most important low level features to describe the audio signal in the temporal context:

Spectral Flatness is the ratio between the geometrical mean and the arithmetical mean of the spectrum magnitude. *Spectral Centroid* is a concept adapted from psychoacoustics and music cognition. It measures the average frequency, weighted by amplitude, of a spectrum. The (individual) centroid of a spectral frame is defined as the average frequency weighted by amplitudes, divided by the sum of the amplitudes. *Strong Peak* intends to reveal whether the spectrum presents a very pronounced peak.

Spectral Kurtosis is the spectrum 4th order central moment and measures whether the data are peaked or flat relative to a normal (Gaussian) distribution.

Zero-Crossing Rate (ZCR), is defined as the number of time-domain zero-crossings within a defined region of signal, divided by the number of samples of that region.

Spectrum Zero-Crossing Rate (SCR) gives an idea of the spectral density of peaks by computing ZCR at a frame level over the spectrum whose mean has previously been subtracted.

Skewness is the 3rd order central moment, it gives indication about the shape of the spectrum in the sense that asymmetrical spectra tend to have large Skewness values where Y is the mean, s is the standard deviation, and N is the number of data points.

Bark-band energy are the energies after dividing the spectrum into the 24 Bark bands, corresponding to the first 24 critical bands of hearing (Zwicker and Fastl, 1990). The published Bark band edges are given in Hertz as [0, 100, 200, 300, 400, 510, 630, 770, 920, 1080, 1270, 1480, 1720, 2000, 2320, 2700, 3150, 3700, 4400, 5300, 6400, 7700, 9500, 12000, 15500]. The published band centers in Hertz are [50, 150, 250, 350, 450, 570, 700, 840, 1000, 1170, 1370, 1600, 1850, 2150, 2500, 2900, 3400, 4000, 4800, 5800, 7000, 8500, 10500, 13500]. These bands are perception-related and have been chosen to enable systematic, instead of database-dependent, division of the spectrum. In order to cope with some low-frequency information, the two lowest bands have been split into two halves (Herrera et al., 2002).

Mel-Frequency Cepstrum Coefficients (MFCCs) are widely used in speech recognition applications. They have been proved useful in music applications as well (Logan, 2000). They are calculated as follows:

1. Divide signal into frames.
2. For each frame, obtain the amplitude spectrum.
3. Take the logarithm.

4. Convert to Mel spectrum.
5. Take the discrete cosine transform (DCT).

Step 4 calculates the log amplitude spectrum on the so-called Mel scale. The Mel transformation is based on human perception experiments. Step 5 takes the DCT of the Mel spectra. For speech, this approximates principal components analysis (PCA) which decorrelates the components of the feature vectors. Logan (2000) proved that this decorrelation applies to music signals as well. As they can be used as a compact representation of the spectral envelope, their variance was also recorded in order to keep some time-varying information. 13 MFCCs are computed frame by frame, and their means and variances are used as descriptors.

5.1.2 Text to speech analysis

A concatenative Text-to-Speech (TTS) system needs to access to the information stored in the speech corpus in order to generate the synthetic speech. This information consists of all tagging files corresponding to the audio files containing the prosodic data according to the tagging framework described in D6.1.1 and D6.1.2, which is currently being extended, for instance, by including voice quality parameters. Moreover, in spite of the defined tagging scheme is mainly oriented to TTS applications, it can be also used for prosodic modeling and voice quality analysis and modeling.

In addition, the input text to be synthesized can be tagged to guide the synthesis process. This information, included in a tagged file (e.g. SMIL and SSML) (see D6.4.1), may be used to determine, for example, the speaker name/voice/identity or the speech sub corpus (e.g. introduction, forecast, farewell in the weatherman scenario) and, thus, the speech corpus to be used within corpus-based TTS synthesis strategies, besides defining the desired prosodic adjustments of the TTS system. At synthesis time, the unit-selection module of the corpus-based TTS system selects the most appropriate set of units from the speech corpus and retrieves them to build the synthetic message. In the current implementation of the URL TTS systems the basic units stand for diphones and triphones of the target language.

The TTS system makes use of information regarding voice parameters in the time domain. On one hand, there is the prosodic information at unit level like: energy, duration and pitch (see D6.4.1 for more details). On the other hand, there are other parameters related to voice quality, such as jitter, shimmer, HNR (presented in D6.1.1 and D6.1.2), etc.

In terms of the retrieval process of the acoustic units, the unit-selection module of the TTS system retrieves the units that best match the requirements provided by the linguistic module. To that effect, different dynamic programming algorithms are being studied (e.g. A* or Viterbi [Formiga&Alfías 2006], see section 6.1.1.) with different types of normalization and selection cost functions. The main goal is to obtain a sequence of units that minimize the amount of digital signal processing (DSP) modification, and, thus, improving the output speech quality in corpus-based TTS framework. The DSP module adapts the retrieved signals to the desired prosodic parameters (energy, duration and pitch at unit level). Generally speaking, it is agreed the more DSP is applied to the signal, the less natural sounding speech is obtained [Black&Taylor 1997].

If the TTS system does not include any prosodic prediction module defining the target prosody of the units to be selected from the corpus, the selection module is focused on obtaining the sequence of units which present the longest path of consecutive units from the original speech recording (named as “text component” in the weather forecast scenario). This strategy is very useful on cases of limited or restricted domain where the recorded speech contains sufficient samples of the desired prosody patterns, thus, making unnecessary to predict its prosodic pattern from the input text solely.

In most cases, the prosody which determines the expressivity of a TTS system is based on the prediction of the following parameters: pitch (i.e. the perception of the fundamental frequency), energy and duration. The prosodic modeling of the TTS system is performed by using a set of linguistic descriptors (e.g. stress and syllables) extracted from text. There are basically three main approaches to extract prosodic patterns from the input text based on: *i*) rule-based approach. The rules for predicting the prosody are defined by experts and are obtained from the corpus analysis. These rules decide the most suitable prosody pattern for guiding the speech generation. This approach needs a pre-processed corpus, usually with ToBI [Silverman et al. 1992] labels, which can be estimated using various machine learning systems. These labels are used to transcribe intonation patterns and other aspects of the prosody of English utterances (it is based on intonational accentual groups), *ii*) Machine Learning (ML)

techniques, such as Case Based Reasoning (CBR) [Iriondo et al. 2006] or Hidden Markov Models (HMMs) [Tokuda et al. 2002]. The former is based on a memory of cases (i.e. attribute-value pairs and prosody parameters) that are used to retrieve similar cases to resolve new cases, i.e. to assign the prosody pattern to a new input sentence according to the most similar one learned by the system. The latter is based on a statistical processing of the prosodic parameters [Gonzalvo et al. 2007] and the use of a decision tree-based clustering. Both techniques, with their particular advantages and disadvantages (see the reported experiments in section 6.1.2), try to define a prosodic pattern capable to allow producing highly expressive synthetic speech (i.e. a similar prosody close to the natural speech) and high naturalness (i.e. related to obtaining a smooth prosody contour) to guarantee that the speech units used to synthesis (e.g. diphones) are not modified excessively. Voice quality parameters are studied so as to know if they may be useful to discriminate among expressive speech styles, and thus, knowing if this information could be used by TTS system in order to improve the naturalness of the synthesized speech. The capability of voice quality parameters to discriminate among different expressive speech styles has been analyzed in several experiments described in section 6.1.3 [Monzo et al, 2007]. To that effect, the data distribution of these parameters, directly measured from the acoustic speech signal, is used to train a Linear Discriminant Analysis (LDA) classifier. As a result, the most relevant voice quality (VoQ) patterns for discriminating expressive speech styles are obtained for Spanish speech corpus with five expressive speaking styles: neutral, happy, sad, sensual and aggressive, considering diphone and triphone as basic speech units.

5.1.3 Speech Analysis

Speech analysis in the Salero project is performed in relation to the logical rhythmic tagging framework (D6.1.1 and D6.1.2), to provide a combination of acoustic, linguistic and emotional dimension information for a speech audio clip. The tagging framework is currently implemented within the LinguaTag speech analysis application (D6.1.2). The inclusion of manual linguistic and emotional dimension analysis within the application allows the user to define various important aspects of a speech signal within a single common file format, which can then be used in processes such as lip-synching animation and content storage and retrieval (see Figure 3):

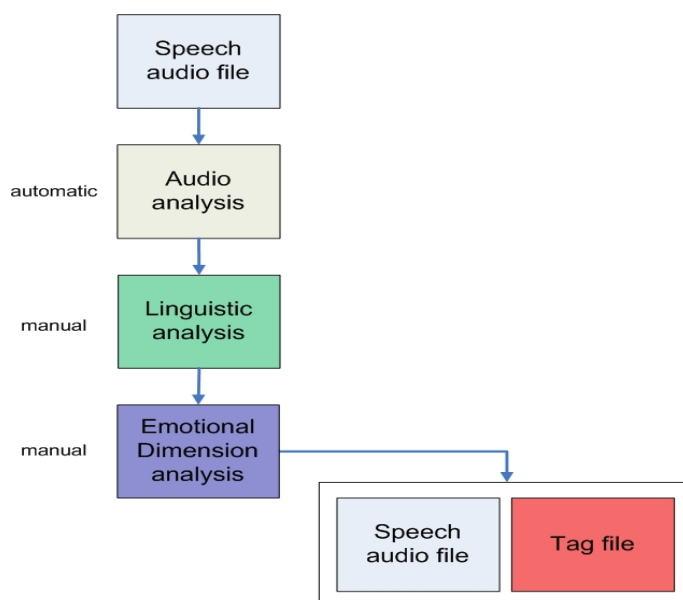


Figure 3: Workflow diagram of the LinguaTag application

LinguaTag uses the Praat analysis engine to obtain acoustic information about vowel events in a speech signal, which are then displayed in the application GUI for acoustic, linguistic and emotional analysis prior to output of this information in SMIL file format. The isolation of vowel events in a speech signal is a common approach in speech analysis. Although a syllable may be formed around non-vocalic events, most speech patterns involve the alternation of vowels and consonants. The definition of the 'pseudo-syllable' is based on the observation that the CV structure is the most common structure, and thus leads to the use of a vowel onset detection algorithm to determine the occurrence of each

vowel (and hence each CV) in a speech event. Prosody in speech can be considered in many different ways, from purely linguistic analysis which places little focus on the acoustic elements of speech to a supra-segmental approach including pitch, loudness and speech rate. The acoustic attributes of pitch, intensity and duration were chosen as fundamental features of a speech event, which are agreed as being common to all 3 representational models of speech. Using this method, a vowel event can thus be graded in terms of threshold values relative to these three parameters (see Figure 4):

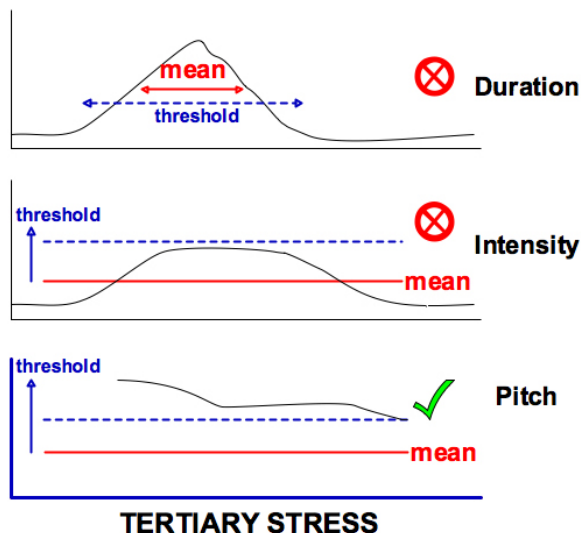


Figure 4: Rating of vowel stress levels

In the LinguaTag application, if a particular vowel crosses a threshold value defined by the user, it is promoted to a higher level of stress. By determining the overall combination of threshold values for an event, it is then defined as either a primary, secondary or tertiary stress (Figure 5):

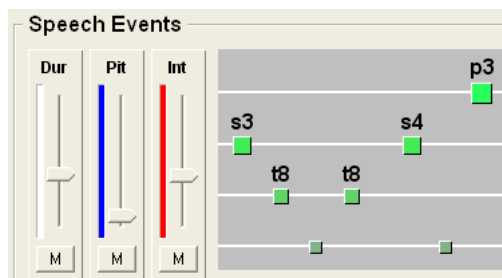


Figure 3

Figure 5: User specification of vowel threshold levels in LinguaTag

Prioritizing vowel events in this manner helps determining elements of prominence in a speech signal. The analysis of such prominent events aims to provide means of focus when seeking to determine the acoustic correlates of emotional speech. There are many types of information that can be extracted from a speech signal, and this framework seeks to define a means of considering this information relative to its overall salience within the clip. Each vowel event in a speech clip is also rated for stress based on its duration, intensity and pitch. Voice quality attributes such as jitter, shimmer, HNR and Hammarberg Index are also obtained for each vowel event, which can then be analyzed in conjunction with duration, intensity and pitch information. All analysis information is displayed within the application, allowing the user to manually check for specific voice quality values (Figure 6):

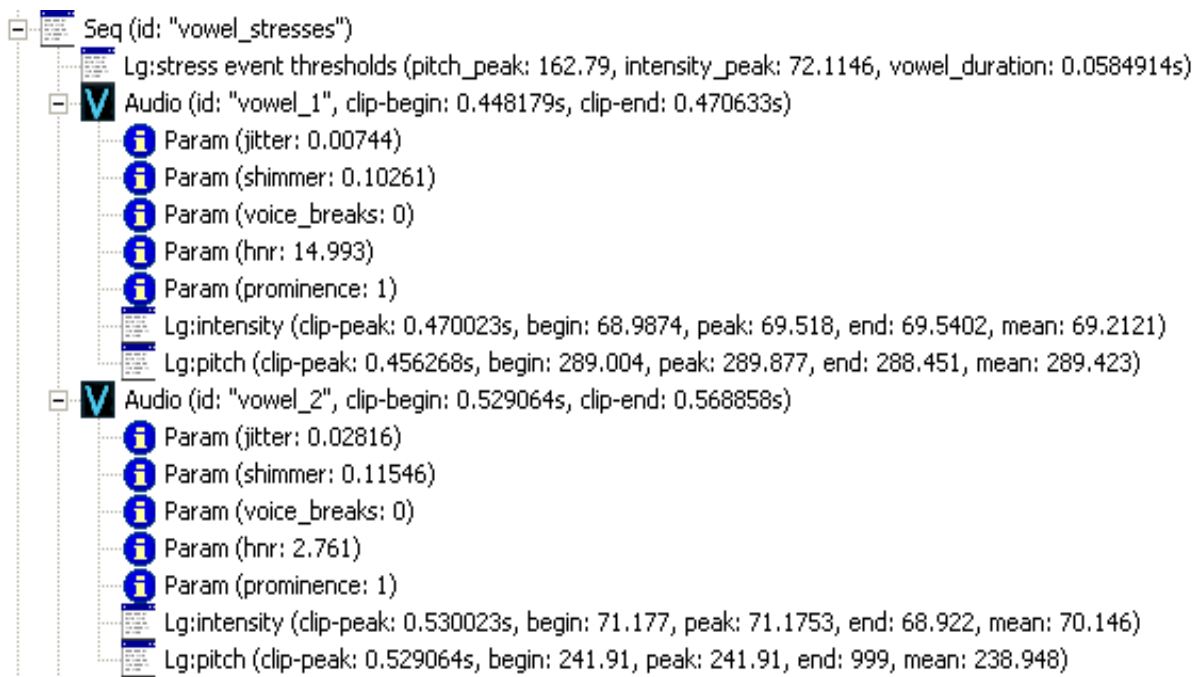


Figure 6: LinguaTag output file display screen

The LinguaTag application allows the speech audio file to be edited to demark areas of linguistic interest (Figure 7):

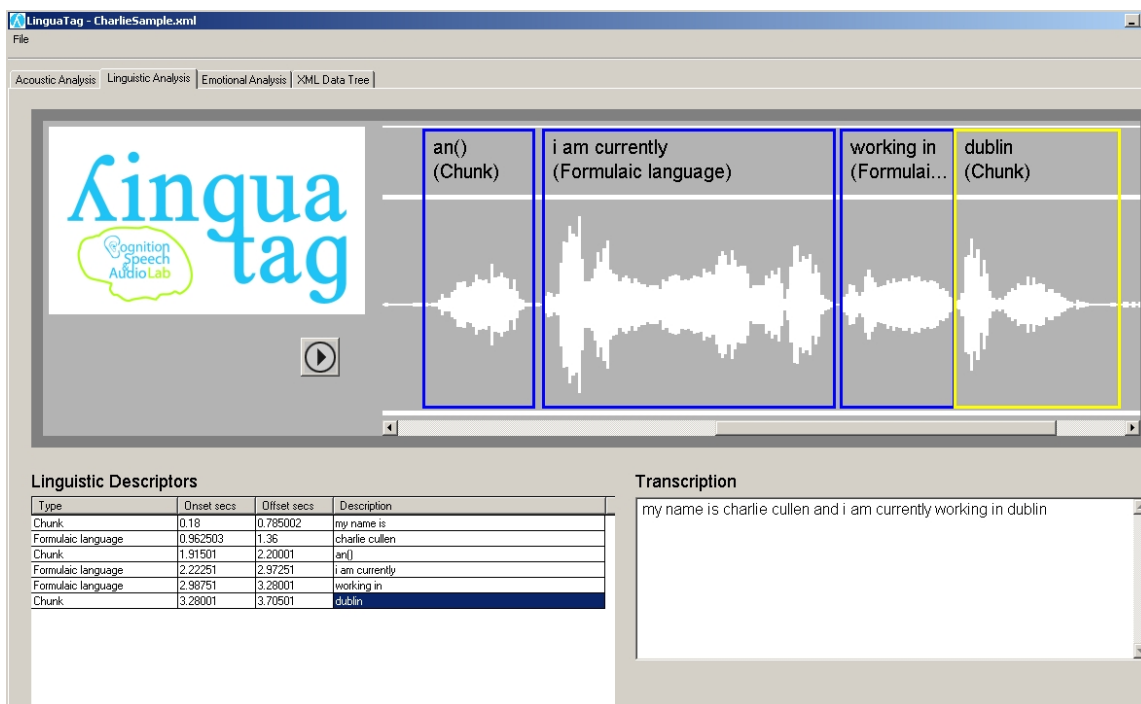


Figure 7: Linguistic analysis screen

Linguistic features such as formulaic language and chunks can be annotated in the current version, and it is intended to include provision for many other salient features of linguistic relevance such as speed of delivery, power relationship and prominence in later versions. These features will allow the LinguaTag application to be used for more effective and multidisciplinary analysis of a speech signal in a single pass.

In this research, circumplex emotional modelling is used to rate speech assets on unit scales of activation and evaluation (Figure 8):

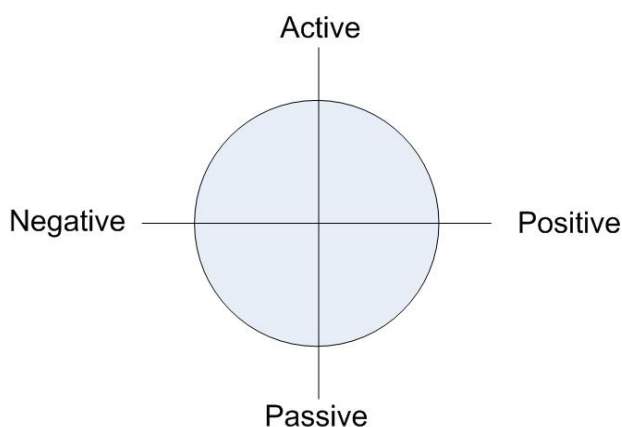


Figure 8: Circumplex emotional model denoting dimensions of activation and evaluation, adapted from Scherer

This dimensional rating allows the speech corpus to be defined in terms of its emotional content (using statistical listening tests) prior to acoustic analysis using LinguaTag. Determination of the acoustic correlates of emotional speech is an open research question, with no definitive results being available at the present time . Correlations between fundamental frequency and emotional dimensions have been observed , but again further work is needed. LinguaTag allows prominent events to be analysed for a variety of parameters, which may prove to be acoustic correlates of emotional speech. In this process, an asset obtained using experimental mood induction procedures is first manually rated in terms of its emotional dimensions (Figure 9):

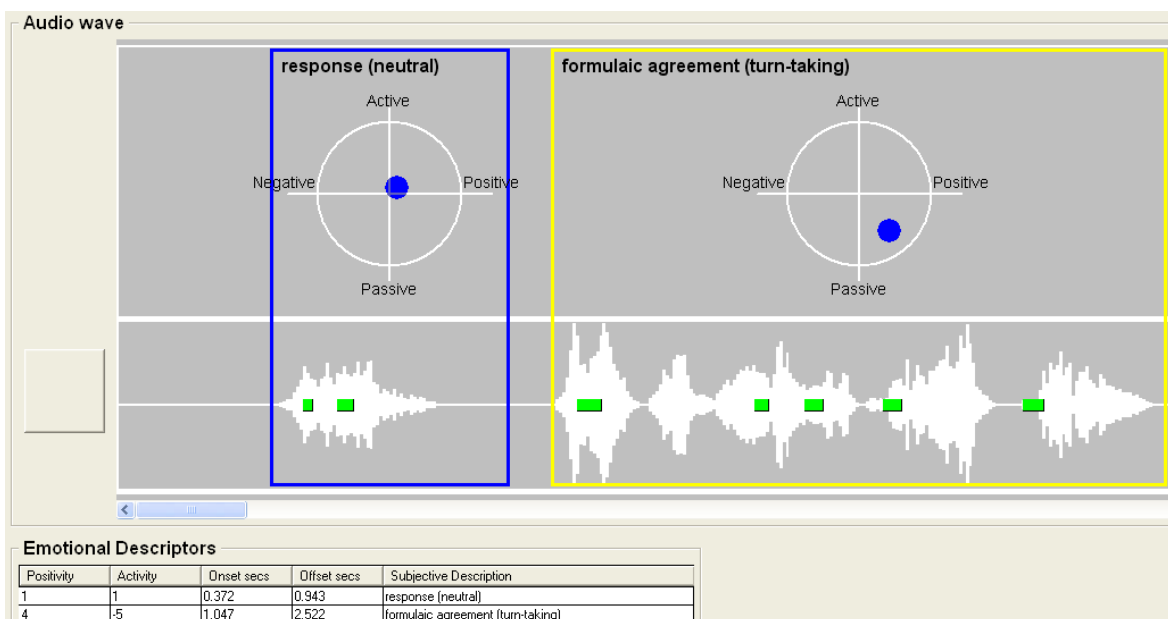


Figure 9: Emotional dimension rating in LinguaTag

This user rating forms part of a statistical evaluation of the perceived emotional dimensions in a clip, whereby the collation of user group results is used to define overall values of activation and evaluation. In this manner, a statistically robust approach to the definition of emotional dimensions is made by consensus, rather than individual ratings or small expert groups. Once rated, the output metadata relating to the acoustic parameters in each vowel event can then be queried for analysis. The querying of groups of assets within the emotional speech corpus will then be used to investigate the presence of acoustic correlates within emotionally rated speech clips.

5.2 Context Aware Similarity

As one of the most important multimedia data created by human beings, music has permeated into every corner of our daily life. Recently, empowered by advances in networking, data compression and physical storage, modern information systems deal with ever-increasing amounts of music data and are becoming more and more important. Since content-based music retrieval is a means to access large multimedia databases for various domain applications, it gains considerable momentum and has become an attractive research topic. Many techniques have been developed to support automatic classification or retrieval of music.

The most fundamental component in modern content-based music information retrieval systems is the scheme to construct content descriptors for music objects. There is a relatively long history to extract low-level features from music objects. Little effort has been focused on constructing a music descriptor for different query types. In fact, the existing approaches are not "smart". By "smart", we mean that they are not query sensitive in the sense that, the features used for genre based similarity search are the same as the ones used for instrument based similarity search. In addition, it has been proved that using physical features from the acoustic signal to effectively represent various high-level semantic similarity notions (such as genre) is an extremely difficult task. The main obstacles are as follows. First, as a rich semantic media, there exists a wide variety of acoustic characteristic within a music signal (e.g. timbre texture, harmony, rhythm structure, melody and pitch). Therefore, it is impossible to obtain effective music content representation for the retrieval purpose by only considering a single kind of acoustic feature. On the other side, the high dimensionality of composite feature², can easily lead to inefficient query processing based on the existing multidimensional indexing structure. Second, a large semantic gap exists between high-level concepts and low-level physical representation. This problem is associated with the human music similarity notion measurement and perception process of human beings. Recent studies in psychology support the belief that human beings perceive music by combining diverse acoustics information via a "non-linear" way. This indicates assuming that each type of acoustic features contributes equally in music recognition is not supported in human music cognition system. Linear concatenation of different low-level multimedia features can not provide "semantic concise" and compact representation. Third, human beings have a unique and amazing capability to identify, classify and understand music. It calls for understanding and integrating those factors or relative high level information on developing effective techniques for music information retrieval (in particular, query processing).

Unfortunately, relatively little attention has been paid on this issue. From above, we can find the key criteria for the multimedia descriptor obtained must be

- Comprehensive – represent raw object concisely and containing semantic information about raw data for effective query processing and classification,
- Compact – it must be small, and;
- Efficient – require less computational resource to generate.

In below sections, we present a fast and robust descriptor generation method for music data. It applies multi-layer structure consisting of two major component layers including pre-processing module and hybrid neural network for nonlinear dimensionality reduction. Distinguished from traditional approaches, our approach can easily fuse various acoustic characteristics and human classification information (high level) into the small feature vector to enhance retrieval process via learning. Experimental results demonstrate various kinds of superiority.

5.2.1 Dimensionality Reduction with RBF Neural Network

One potential solution to the problem is to apply a neural network to carry out a non-linear dimension reduction. We select Radial Basis Function (RBF) network as neural network component in our system. Since RBF neural network is standard technique, we don't provide description in this report and the interested reader can refer to [Vapnik 2006]. It can provide a general mechanism to conduct complex maps approximately in high dimensional spaces. A neural network can be generally treated as a set of interconnected neurons, which act as the basic computational units for nonlinear mapping. In our

² Feature vector contains information of more than one type of acoustic feature

current implementation, we use unnormalized Gaussians for our basis functions, which are denoted as $\{f_1, f_2, \dots, f_n\}$. The function $f(X)$ can be written as below,

$$f(X) = \sum_{i=1}^r w_i \phi_i(\|X - \mu_i\|, \theta_i)$$

Equation 1

Where X denotes input vector w_i 's and its weight for network. ϕ_i is the basic function and can be written as below,

$$\phi_i(\|X - \mu_i\|, \theta_i) = \exp\{-\|X - \mu_i\| / (\theta_{i1}, \theta_{i2}, \dots, \theta_{in})\}^2$$

Equation 2

Where $\|\cdot\|$ denotes the Euclidean distance function. The procedure of non-linear dimensionality reduction with RBN network is as below. After the network training process is completed, content representation of multimedia objects can be generated by feeding concatenated feature vectors into the network and taking the vectors computed in the hidden units as the lower-dimensional representation. Since the neural network is trained with manual training examples, discriminative information can be fused into these lower-dimension vectors. Therefore, they can be applied for different effective similarities and its size can be configured to be smaller for better efficiency.

The advantage of using a neural network for dimensionality reduction is that it can learn directly from training examples (such as human pre-labelled data) to form a model of the feature data. Its basis is the standard non-linear regression analysis used in a neural network approach. Through training, the distance information of the original data source can be represented as weights between units in successive layers of the neural network. However, the main weakness for such a scheme is a high training cost and complex structure.

Figure 10: Structure of Query Sensitive Dimension Reduction Scheme (RBF neural network). Input is D-dimensional Feature Vector.

5.2.2 Query Sensitive Music Descriptor Generation Scheme

Motivated by above, we have developed a hybrid architecture based on neural network and linear subspace method (LSM) [Vapnik 2006]. This system's structure is illustrated in Figure 11 and it applies a hybrid method that combines two traditional dimensionality reduction methods, LSM and a RBF neural network, into a single architecture. Neural networks can be greatly influenced by the "curse of dimensionality". That implies that the time required for this training method grows sub-linearly with the size of inputs and sometimes it can make training of large network impractical. Thus pre-processing raw data using a linear dimension reducer can yield a great cost advantage not only in efficiency but also in effectiveness. Based on this observation, LDA is used as a "pre-processing" step in our study for linear dimensionality reduction where it provides reduced-dimension feature vectors to train the neural network, and thus speed up the nonlinear dimensionality reduction training time. From above, we can see that since our approach incorporates high level similarity information via training, descriptors produced are both efficient (smaller size) and effective (well-discriminating).

In current setting, we use LDA as linear subspace method to pre-process raw input data for speeding up training process of neural network. LDA is a well method dimension reduction method and its main

advantage over PCA is to discover the feature space with lower dimensional subspace for which the different classes of measurements remain well separated after projection to this subspace. The subspace is spanned by a set of vectors. If the separate classes are sampled from Gaussian distributions, all with identical covariance matrices, then LDA maximizes the mean value of the KL divergences between the different classes. The main advantage of LDA over PCA is that it enjoys better data modelling capacity. In section 6.1, comprehensive experimental results will validate our claims.

Figure 11: Structure of Query Sensitive Dimension Reduction Scheme NN+LSM

Training Data

6 Experimental Results

This chapter presents the results of the evaluations of the different context based audio techniques outlined in Chapters 4 and 5.

6.1 Text-to-speech analysis and synthesis

The context-based audio feature extraction within the text-to-Speech (TTS) system can be understood from two points of view. Firstly, the speech data is parameterized extracting the features, which the TTS system needs to conduct the synthesis process. Therefore, the speech database contains the wav files corresponding to the speech units and their associated acoustic information (duration, energy, pitch, etc.) And secondly, when speech synthesis is being conducted, the information contained in the speech corpus is needed to conduct the unit retrieval process before generating the synthetic speech, and thus, data retrieval is performed. The following experiments are focused on analyzing the performance of *i)* the unit-selection retrieval process in terms of computational cost, *ii)* the prosodic model in terms of speech quality obtained by means of statistical synthesis, *and iii)* the annotation and discrimination of speaking styles capacity through voice quality parameterization.

6.1.1 Unit-selection module

One important issue of the retrieval process within the text-to-speech conversion is the retrieval time (i.e. the computational cost of the unit selection module), which is a critical point so as to obtain a real-time TTS system.

The main difference between A* and Viterbi algorithms is that the former is based on decoding a beam search stack, with a tree-based search model where the partial hypotheses (heuristics) are represented by the different branches of the tree, whereas the latter computes all the distances of the paths connecting the trellis structure (exhaustive search). It is to note, that the Heuristics are a key factor for the success of the A* algorithm. A set of poor heuristics would make the algorithm behave as the Viterbi algorithm in terms of computational cost (and, thus, with an increase of memory cost).

Towards providing a better understanding of the heuristic concept, a toy example is introduced. For instance, consider the case that 12 speech units are to be selected from the speech corpus, and the search state is at the point which 7 units have already been selected but 5 are still to be selected. On that point, the cost associated to that search state is computed as the cost of selecting the 7 units plus an heuristic factor multiplied to the number of search steps to reach the goal (in this example, as 5 units are still to be selected, it is $5 \cdot h'(n)$, where $h'(n)$ stands for the considered heuristic value).

The following experiment compares the execution time of Viterbi and A* algorithms by considering different heuristics for three Catalan phrases covering search units with different number of realizations in the speech corpus

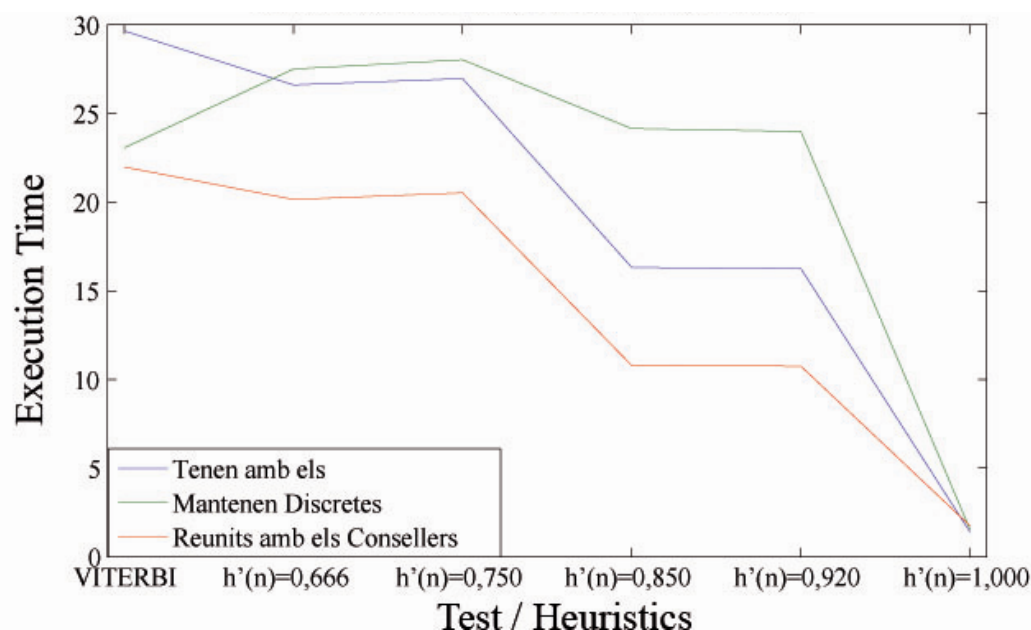


Figure 12: Execution Time depending on the test phrase and the Heuristic value

As it can be observed from figure Z, the A* algorithm has a better behavior when it adopts the most pessimistic heuristic ($h'(n)=1,000$) and is able to decrease time in a considerable manner. In the future, a deeper analysis of the retrieval process results will be conducted in terms of cost function for quantifying the signal processing applied to the signal after being retrieved.

6.1.2 Prosodic modeling

In this work, the prosodic modeling has been performed either by using a machine learning (ML) or a statistical based approach. The prosodic prediction will use the prosody model in order to estimate the correct prosody parameters in basis of the linguistic information extracted from the input text to be synthesized. The prosody prediction performance is important since the final quality and expressiveness of the synthetic speech are closely related to the reliability of the estimation process.

The following experiment was developed to compare the synthetic results of the CBR and HMM-based prosodic models by means of a preference test. Figure 13 depicts the results of this test, using a HMM-TTS synthesis system [Gonzalvo et al. 2007], for two kinds of phrases: interrogatives and exclamatives. Specifically, the two compared approaches are based: only on HMM-based prosodic modeling, and a on mixed CBR+HMM prosody prediction (denoted as Mixed F0 in Figure 13). From the obtained results, it can be concluded that the mixed approach yields the highest preference values (66,67% for INT and 50% for EXC), besides taking into account that the equal option (no preference between the synthetic results) in this case is around 10% and 20%, respectively.

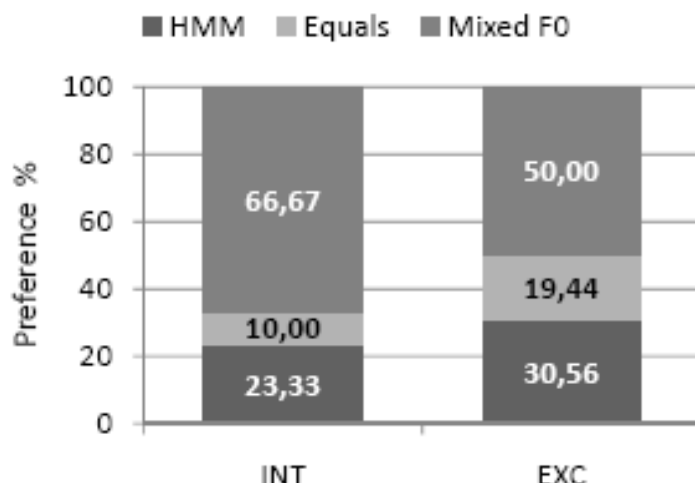


Figure 13: Preference of prosody prediction system for phrase type, INT stands for interrogative and EXC for exclamative sentences.

6.1.3 Discriminating expressive speech styles by voice quality parameterization

These parameters, directly measured from the acoustic speech signal, is used to train a Linear Discriminant Analysis (LDA) classifier . As a result, the most relevant voice quality (VoQ) patterns for discriminating expressive speech styles are obtained for Spanish speech corpus with five expressive speaking styles: neutral, happy, sad, sensual and aggressive, considering diphone and triphone as basic speech units

The VoQ parameters used in this study are directly estimated from the acoustic speech signal; therefore neither extra hardware nor invasive transducers are required [Lugger&Yang, 2006]. According to [Drioli et al. 2003], The selected parameters considered in the different tests were: Jitter, Shimmer, Harmonic-to-Noise Ratio (HNR), Glottal-to-Noise Excitation Ratio (GNE), Spectral Flatness Measure, Hammarberg Index (Hamml) and Drop-off of spectral energy above 1000Hz (Drop_1000) and the relative amount of energy in the high (above 1000Hz) versus the low frequency range of the voice spectrum (Pe_1000).

The aim of the conducted analysis was finding the most representative VoQ parameters that may be useful to discriminate among expressive speech styles. Therefore, the subsequent analysis was conducted by considering descriptive statistics and LDA classification of parameterized speech data. The results obtained from the automatic LDA classifier were validated by means of a t-test analysis on data distributions. The significance level is measured and analyzed among expressive speech styles by means of a pair-wise comparison per VoQ parameter. For each VoQ parameter (input data to the classifier), all expressive speech styles (output classifier classes) are known. By means of a 10-fold cross validation (using a random process for data selection) the training and testing information were obtained. The F1 measure was used to assess the classifier performance as it combines both precision and recall into a single metric and favors a balanced performance of the two metrics (see Figure 14).

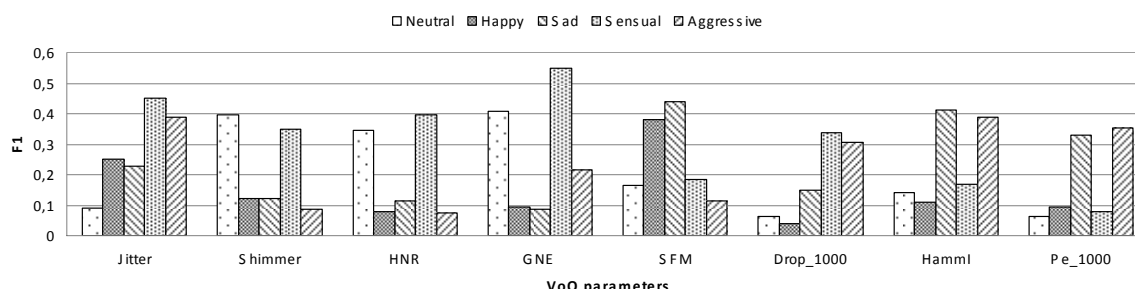


Figure 14: F1 measure of the LDA classifier using VoQ parameters for five expressive speech styles

After conducting this experiment, the best discriminating parameters per expressive style were obtained (see Table 1), and for instance, from this, can be noticed that parameters as jitter are useful for discriminating between sensual and aggressive styles. In addition, the most relevant VoQ parameters related to the discrimination between each speaking style are shown in Table 2.

VoQ parameter	Expressive style
Jitter	Se, A
Shimmer	N, Se
HNR	N, Se
GNE	N, Se
SFM	H, Sa
Drop_1000	Se, A
Hamml	Sa, A
Pe_1000	Sa, A

Table 1: The highest discrimination of expressive speech styles by VoQ parameterization

Expressive style	VoQ parameter
Neutral	Shimmer, HNR, GNE
Happy	SFM
Sad	SFM, Hamml, Pe_1000
Sensual	Jitter, Shimmer, HNR, GNE, Drop_1000
Aggressive	Jitter, Drop_1000, Hamml, Pe_1000

Table 2: The most relevant relations among expressive speech styles and VoQ parameters

Notice that all parameters are important to discriminate the expressive speech styles. On one hand, the 'Sensual' style, which usually is difficult to be identified by only using prosodic parameters, can be clearly discriminated from VoQ parameters. On the other hand, the 'Happy' style is only discriminated by means of SFM parameter, indicating the low relevance of this expressive style in terms of VoQ parameterization, i.e. it is an expressive style poorly modeled by VoQ parameters

Moreover, these results can be also interpreted in terms of phonation type analysis. For instance, the 'Sensual' style is characterized by a whispering voice (noisy), whereas the 'Neutral' style is characterized by a modal voice, therefore, VoQ parameters such as GNE, related to noise measurement, make possible the discrimination between them.

In order to validate the obtained results, a t-test was conducted. In Table 3 the significance level (p) is shown for each expressive speech style through a pair-wise comparison. Below the threshold value ($p < 0.05$), expressive styles pair-wise comparison is considered significantly different, and thus, making possible the discrimination. Notice that the clearest discrimination results obtained from LDA analysis (see Table 1) are below the significance level (indicated by * in the table), therefore LDA discrimination results among expressive speech styles are statistically validated. However, there is a noticeable exception when comparing 'Happy' and 'Aggressive' styles, where the significance levels for SFM, Drop_1000 and Hamml indicate that these VoQ parameters are not useful for discriminating between these styles.

	N-H	N-Sa	N-Se	N-A	H-Sa	H-Se	H-A	Sa-Se	Sa-A	Se-A
Jitter	0.25	*	*	*	*	*	*	*	*	*
Shimmer	*	*	*	*	*	*	0.31	*	*	*
HNR	*	*	*	*	*	*	0.63	*	*	*
GNE	*	*	*	*	*	*	0.44	*	*	*

	N-H	N-Sa	N-Se	N-A	H-Sa	H-Se	H-A	Sa-Se	Sa-A	Se-A
SFM	*	*	*	*	*	*	0.32	*	*	*
Drop_1000	0.09	*	*	*	*	*	0.08	*	*	*
HammI	*	*	*	*	*	*	0.45	*	*	*
Pe_1000	*	*	0.11	*	*	*	*	*	*	*

Table 3: Significance level value for expressive speech styles pair-wise comparison per VoQ parameter (threshold for significance level is: * $p < 0.05$) (* stands for the significance level threshold)

6.2 Tag propagation based on audio similarity

In this section we present some experiments using tags (user's annotations) to contextually describe the content of the audio files. Based on these tags we can derive relationships with the text and the audio content. Then, using content-based similarity we are able to propose tags to new songs (thus, avoiding the so-called cold start problem).

Finally, we present a metric (based on cosine distance) to get similar songs based on their contextual tags, allowing to improve the pure content based similarity.

6.2.1 Experiments

The main goal of this experiment is twofold: on the one hand we aim at easing the process of annotating music collections, by using content-based similarity distance as a way to propagate contextual tags among songs.

For our purpose, the content--based similarity can be seen as a black box. That is to say, given a seed song, the module returns a list of the i th most similar songs.

This study employs a CB module that considers not only timbral features (e.g. MFCC), but some musical descriptors related to rhythm, tonality, etc.

We present two different experiments. The first one propagates labels that are related with the style of the piece, whereas the second experiment deals with mood labels.

The problem with the Magnatune collection is that there is only one human that annotated the tracks, when normally a ground truth of this nature should be pair--reviewed. Yet, we validated a large amount of the annotated songs by listening to them.

The metrics used to evaluate the styles experiments were initially Precision/Recall and F2-Measure (giving more weight to Recall). In our case, Recall seems to be more informative since our purpose is to know how well the tags can be propagated.

However, neither P nor R take into account the frequencies (i.e. ranking) of the tags obtained from the similar songs. Thus, we used the Spearman's rank correlation coefficient, or Spearman-rho, which is defined as:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

Equation 3: Spearman's rank coefficient

Where d_i represents the distance between each rank of pair of values ---in our case labels in the ground truth and labels in the proposed tags--- % and n the number of all possible pair of values.

To compute the distances we assume that the frequency of manually annotated labels is equal to 1.

6.2.2 Style labels

For the style experiment, we ran different configurations and we computed the average metrics.

Sims.	Constraint	P	R	F_2	ρ
10	None	0.56	0.84	0.72	0.51
	Artist	0.41	0.58	0.51	0.23
	Album	0.50	0.71	0.62	0.34
	Artist & Album	0.43	0.59	0.53	0.19
20	None	0.56	0.82	0.71	0.49
	Artist	0.48	0.61	0.56	0.26
	Album	0.53	0.72	0.64	0.35
	Artist & Album	0.48	0.61	0.56	0.24
30	None	0.60	0.77	0.70	0.45
	Artist	0.50	0.58	0.55	0.28
	Album	0.56	0.67	0.63	0.37
	Artist & Album	0.50	0.59	0.55	0.27

Table 4: Experiments with the 100% annotated collection. The Precision/Recall measure, the F2-measure and the Spearman rho measure are proportional to the number of similar songs. When constraints are present, these measures decrease.

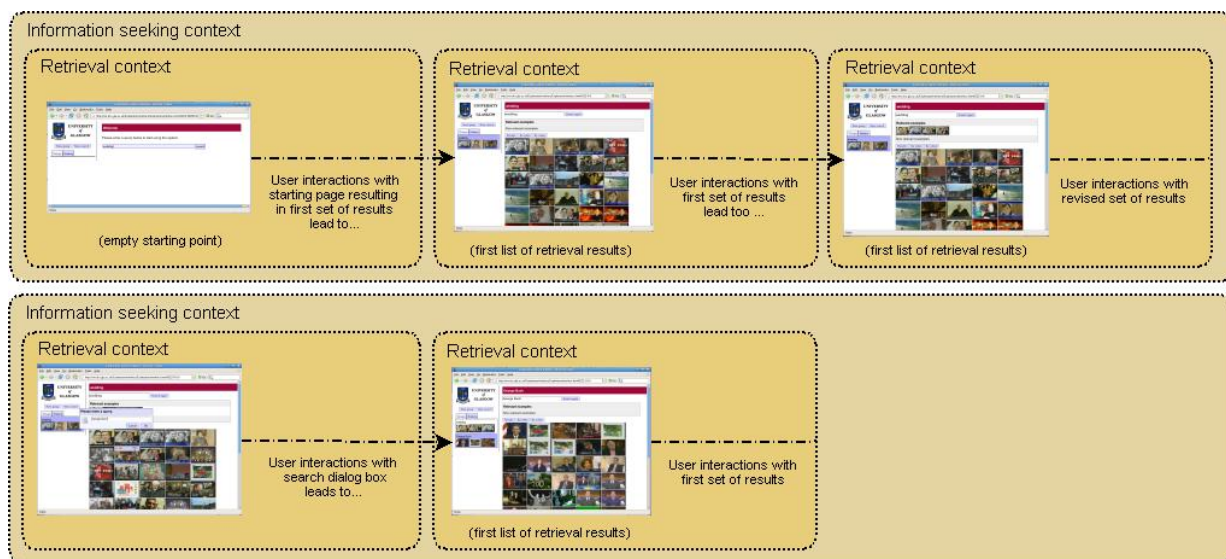


Figure 15: Illustration of retrieval (query) contexts within a larger information seeking context.



Figure 16: Illustration of retrieval (query) contexts within a larger information seeking context.

When filtering by artist or by album we make sure that the most similar songs to a given one are not from the same artist or the same album. That of course decreases the Precision/Recall measure.

We can see from the results, that to achieve more precision and recall when applying a constraint, we need to increase the number of similar songs, which makes sense because we are not taking into account similar songs that are closer to a given one.

Now, Table 5 shows the results of propagating a partially annotated collection. The Spearman-rho coefficient, as well as Precision/Recall and F2-measure, grows when increasing the percentage of songs annotated in the collection. Interestingly enough, the values decrease when increasing the number of neighbors (from 10 to 30) for a given song.

Annotation	Sims.	P	R	F_2	ρ
20%	10	0.32	0.29	0.30	0.24
	20	0.22	0.17	0.19	0.16
	30	0.08	0.05	0.06	0.06
40%	10	0.57	0.59	0.58	0.43
	20	0.56	0.52	0.53	0.41
	30	0.49	0.39	0.42	0.34
50%	10	0.61	0.67	0.64	0.47
	20	0.61	0.61	0.61	0.45
	30	0.57	0.51	0.53	0.41

Table 5: Experiments with the 20%, 40% and 50% annotated collection. The Precision, Recall and F2-measure and the Spearman-rho values grow with a higher percentage of annotated songs, and a smaller number of similar songs.

Finally, we propose another experiment that is to automatically annotate songs in a music collection by means of the propagation process. The results are presented in Table 6.

It is clear that the percentage of songs automatically annotated by CB similarity increases when the number of already annotated songs grows.

But, we can see an interesting exception here, that is the 40% annotated collection performs better (up to 38.68% new propagated labels, with a low Recall 0.4) than the 50% one. This could be due to the random process of splitting the ground truth and the test set from the collection.

Furthermore, we can see how the percentage of songs automatically annotated is inversely proportional to the number of similar songs used by the CB similarity module.

Annot.	Sims.	Propagation with Recall		
		> 0.8	> 0.6	> 0.4
20%	10	17.515%	21.365%	24.977%
	20	8.666%	12.352%	15.453%
	30	2.554%	3.758%	5.145%
40%	10	28.01%	33.46%	38.68%
	20	22.50%	28.92%	34.32%
	30	15.22%	20.82%	26.22%
50%	10	26.77%	31.62%	35.92%
	20	22.66%	28.74%	33.37%
	30	17.48%	23.15%	28.44%

Table 6: Extending annotations of a music collection by means of CB similarity. We observe that the propagation grows with a smaller number of similars and a higher percentage of annotated songs, except for the case of 40% and 50%.

6.2.3 Mood labels

For the moods experiment, the first issue is the choice of the taxonomy. As advised by Juslin et al. in [Juslin 2001], in order to make our experiment and to build a ground truth that achieve the best agreement between people, we should consider few categories. We used a reduced version of the Magnatune online library. This collection offers a set of playlists based on mood (<http://www.magnatune.com/moods/>). We clustered the 150 mood playlists to fit in our few categories paradigm. The adjectives proposed by Juslin: happiness, sadness, anger and fear in [Juslin 2001] have been applied by Feng et al. in [Feng 2003] and proved to give satisfying results. As the collection is mostly focused on popular and classical music, the "fear" adjective has been extended to a larger category called "mysterious". Using Wordnet (<http://wordnet.princeton.edu/>) we have joined the possible playlists together in the following four categories: happy, sad, angry and mysterious. Then, a group of listeners were asked to validate each song mood label.

We obtained a ground truth database of 191 songs with the distribution in mood shown in Table 7. For each song, there is only one mood label. It is not an equal distribution but there is enough data in each category to experiment with the CB similarity.

Mood	Happy	Sad	Angry	Mysterious
Songs	67	61	34	29

Table 7: Mood distribution of the ground truth}

To evaluate the mood results, we used two measures. First we wanted to check if the system was able to guess the correct mood label (there is only one possible label per song). We evaluated the Precision just considering the first result using Precision at 1, also called P@1:

$$P@1 = \begin{cases} 1, & \text{best proposed label} = \text{real label} \\ 0, & \text{otherwise} \end{cases}$$

Equation 4: Precision at 1 (P@1)

We averaged this value over all the examples. This metric helps us to understand if the system can predict the correct mood label. However it does not take into account the relative frequencies. Then

another measure would be needed to evaluate this aspect. We weighted the frequencies of the proposed label and normalized to compute a weighted Precision at 1, that we will call $wP@1$. It is equal to the frequency value of the correct label over the sum of all the proposed label frequencies:

$$wP@1 = \frac{\text{freq. correct label}}{\sum \text{freq. proposed labels}}$$

Equation 5: Weighted Precision at 1, ($wP@1$)

To have an overview of the system performance for each mood, we built a confusion matrix in Table 8. It has been computed using 100% of the collection annotated. Each row gives the predicted mood distribution (considering only the best label) for each mood in the ground truth.

GT/Predicted	Angry	Happy	Mysterious	Sad
Angry	27	7	1	1
Happy	4	55	1	2
Mysterious	8	6	7	5
Sad	4	16	2	35

Table 8: Confusion matrix for the mood experiment with a 100% annotated collection.

Looking at the confusion matrix we observe that a CB similarity approach can propagate relatively well the "happy", "angry", and "sad" labels. However, the "mysterious" label does not give good results. We can explain this by the fact that it might be the most ambiguous concept of these categories. Table 9 presents the average $P@1$ and $wP@1$ values per mood.

	Angry	Happy	Mysterious	Sad	All
$P@1$	0.72	0.89	0.27	0.61	0.62
$wP@1$	0.65	0.62	0.22	0.59	0.52

Table 9: $P@1$ and $wP@1$ values averaged for each mood

It confirms what we have in the confusion matrix, the "happy" category gives the best result. However looking at the values of $wP@1$, we note that if "happy" is the most guessed mood, the system gives more reliability to its results about the label "angry". In our last experiment we wanted to evaluate how well the mood labels can be propagated if we annotate just partially the collection. We computed the $P@1$ for 70%, 50% and 30% of the database and obtain the results written in Table 10. It shows that for 30% of the collection annotated, the system can propagate correctly the tags up to 65% of the collection.

Initial annotation	70%	50%	30%
$P@1$	0.60	0.44	0.5
Correctly annotated after prop.	88%	72%	65%

Table 10: Evaluation of the mood label propagation with the initially percentage of annotated songs.

As the CB approach may not consider important aspects that can infer the mood, all these performances should be improved by using dedicated descriptors and approach or meta-data, like information about the title, the style or the lyrics.

6.2.4 Conclusions

The objective of these experiments was to test how the CB similarity can propagate context tags about the audio files. For the styles experiment, we have shown that with a 40% annotated collection, we can reach a 78% (40%+38%) annotated collection with a recall greater than 0.4, only using CB similarity.

In the case of moods, with a 30% annotated collection we can automatically propagate up to 65% (30%+35%). These results are quite encouraging as CB similarity can propagate styles and moods in a surprisingly effective manner. Of course there are some limitations as the example of the "mysterious" label, the concept has to be clearly encoded in the music for the CB propagation to work. For the moods we will try to experiment with a larger database, different taxonomies and more concepts. With our current mood results it may not be possible to generalize but it shows the potential of the technique.

In general, to enhance the performance of such an automatic annotation system we would use a hybrid approach combining CB, user feedback and social networks information.

But as shown by the satisfying results, our propagation system based on CB similarity would already ease a lot the annotation process of huge music collections.

6.3 Query Sensitive Music Descriptor Generation

This section presents experimental results on querying music descriptors based on genre and artist classification. The aim of these experiments is to demonstrate the superiority of our proposed method, to represent music data.

6.3.1 Test Configuration

We are currently testing our scheme on large music dataset for different query classes. We test the scheme using two test collections. The first data set, called Dataset I, contains 3000 music data items covering ten genres with 300 songs per genre. Each item in this collection belongs to exactly one of ten music genre categories: *Classical, Country, Dance, Hip-hop, Jazz, Reggae, Metal, Blues* and *Pop*. This hierarchy is illustrated in Figure 15. The second data set, called Dataset II, contains 3000 music items including song performed by 20 different artists. Ten are female and another ten are male.

All the audio excerpts are converted to 22050 Hz, 16bit, mono audio file. The length of each is 30 sec. With those two datasets, two kinds of queries can be carried out,

- Type I: Given a query example, find the music items having similar genre, and;
- Type II: Given a query example, find the music items performed by the same artist.

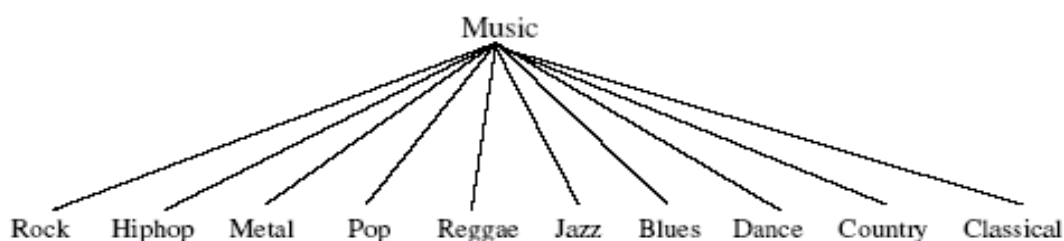


Figure 17: Genre Structure of Music Test Data

With our approach, one music descriptor generator can be built for each kind of query mentioned in above.

We use cross fold validation: 20% of the target data collection is used as training examples and the rest of data is used for query test. Thus, there is no overlapping between test data and training data. In our study, dimensionality of hidden layer for neural network is set to be 10. Thus, size of music descriptor obtained is equal to 10. Multidimensional indexing structure used for fast search is *iDistance* [Jagadish 2005].

In this study, input features for the method we propose include Pitch Histogram, MFCC and STFT. The detailed information about those features can be found in Section 5.1 of this deliverable.

6.3.2 Experimental Results

In this section we evaluate the methods from two aspects:

- Query effectiveness, or how good are the results of the music retrieval? This is evaluated using precision and recall, standard methods in Information Retrieval (e.g. see [van Rijsbergen 1979], Chapter 7 or [Witten 1999], section 4.5). Precision is a measure of the accuracy of the search, defined as the number of relevant and retrieved music clips over the total number of retrieved clips. Recall is a measure of the exhaustiveness of the results, defined as the number of relevant and retrieved clips over the total number of relevant clips in the collection.
- Training and query performance. The method outlined here requires training for each query type (or context), which may be slow depending on the quantity of the training undertaken. Query performance, the speed at which a result for a query can be returned is also important in interactive systems, and will vary depending on the quantity of the training carried out (i.e a smaller feature vector leads to quicker querying speed, but slower training time). These issues are looked at in the later sections.

Query Effectiveness

We use the features listed in Section 5.1.1 as input and the dimensionality of feature vector generated by our method is 15. Table 11 and Table 12 present the retrieval results using the Euclidean distance.

The main premise of our approach is to construct of a semantic vector space for audio data via non-linear map with a hybrid structure. Throughout integration of high level semantic information via learning process, audio descriptors generated significantly improved overall quality of retrieval results. The experiments verify our claim. Table 11 and Table 12 summarize query effectiveness of the different features and their combinations for two different query types. It is shown that MFCC is the worst in terms of MAP and P@10.

Although the technique based on Wavelet achieves better performance than other low level features, improvement is limited. This is because Wavelet only captures low-level physical characteristics of the music signal.

- Furthermore, results also show that any kinds of feature combination via linear concatenation cannot bring performance improvement on query accuracy for both queries. In fact, the experimental results clearly demonstrate that our approach significantly outperforms any other approaches. The main reason is due to integration of human relative information via learning process for final object descriptor generation.

Table 11 shows, compared to the Wavelet based feature, that the NN+LSM method improves the retrieval precision from 0.2297 to 0.3411 and 0.4087 to 0.4807 (in terms of MAP and precision at 10 results). While this is only a preliminary result, it shows the superiority of our method. In the following section, we present some improvements on training and query efficiency.

Metrics	MFCC	STFFT	Pitch	LPC	Wavelet	NN+LSM
MAP	0.0995	0.1123	0.1234	0.1034	0.2297	0.3411
P@10	0.1956	0.2456	0.2543	0.2113	0.4087	0.4807

Metrics	MFCC + STFFT	MFCC + Pitch	MFCC + LPC	MFCC + Wavelet	STFFT + Pitch	STFFT + LPC
MAP	0.0912	0.1032	0.1045	0.1567	0.1127	0.1035
P@10	0.1879	0.1934	0.2042	0.3567	0.1687	0.1542

Metrics	STFFT + Wavelet	Pitch + LPC	Pitch + Wavelet	LPC + Wavelet	MFCC + STFFT + LPC	MFCC+ STFFT + Pitch
MAP	0.1532	0.1033	0.1919	0.1842	0.0897	0.0924
P@10	0.3279	0.2117	0.3728	0.3620	0.1986	0.2018

Metrics	MFCC+ STFFT + Wavelet	STFFT + Pitch + LPC	STFFT + Pitch + Wavelet	Pitch + LPC + Wavelet	MFCC+ STFFT + Pitch + Wavelet
MAP	0.1601	0.1004	0.1156	0.1231	0.1531
P@10	0.3188	0.2321	0.2511	0.2113	0.2123

Table 11: Performance Comparison on Retrieval Effectiveness of Query Type I base on MFCC, STFFT, Pitch, LPC, Wavelet, Various Linear Combinations and NN+LSM. NN+LSM denotes the method developed by the project.

Metrics	MFCC	STFFT	Pitch	LPC	Wavelet	NN+LSM
MAP	0.0895	0.1021	0.1099	0.1145	0.2109	0.3120
P@10	0.1256	0.1855	0.2187	0.2235	0.3700	0.4077

Metrics	MFCC + STFFT	MFCC + Pitch	MFCC + LPC	MFCC + Wavelet	STFFT + Pitch	STFFT + LPC
MAP	0.0895	0.0978	0.1056	0.1251	0.0987	0.1025
P@10	0.1195	0.1971	0.1678	0.2113	0.1271	0.1189

Metrics	STFFT + Wavelet	Pitch + LPC	Pitch + Wavelet	LPC + Wavelet	MFCC + STFFT + LPC	MFCC+ STFFT + Pitch
MAP	0.1456	0.0987	0.1135	0.1034	0.1059	0.1171
P@10	0.1537	0.1372	0.2189	0.1908	0.1277	0.1567

Metrics	MFCC+ STFFT + Wavelet	STFFT + Pitch + LPC	STFFT + Pitch + Wavelet	Pitch + LPC + Wavelet	MFCC+ STFFT + Pitch + Wavelet
MAP	0.1237	0.1078	0.1235	0.1273	0.1145
P@10	0.2199	0.1225	0.1988	0.2545	0.3699

Table 12: Performance Comparison on Retrieval Effectiveness of Query Type II for base on MFCC, STFFT, Pitch, LPC, Wavelet, Various Combination and NN+LSM. NN+LSM denotes our method.

Query and Training Efficiency

In this section, we study efficiency of our proposed method from two aspects, training efficiency and query efficiency.

Our technique (NN+LSM) aims to generate feature vectors tailored to specific types of queries, which will also involve a reduction in the number of features. Due to the “curse of dimensionality”, a large input feature vector can make the learning process for any classifier and existing multidimensional indexing structure very inefficient in terms of training and querying time. Using a small but well-discriminating feature vector generated by our method not only provides superior query accuracy but also saves a large amount of training and query processing time. To further illustrate the performance advantage of using our method, we computed the retrieval time for different methods.

The results in Table 13 indicate that compared to other methods, the speed-up for query processing by our proposed method is significant, with the exception of LPC. Improvement ranges from 15% to 30%. LPC can generate much smaller feature vector (its dimension is 4) and thus leads to best performance in term of query response time.

Query Type	MFCC	STFFT	Pitch	LPC	Wavelet	NN+LSM
I	0.278	0.256	0.237	0.057	0.399	0.121
II	0.276	0.262	0.240	0.056	0.401	0.133

Table 13: Performance Comparison on Query Response Time for Query Type I and II. NN+LSM denotes the method developed by UG (all values in seconds)

On the other hand, efficiency at the training step is very important for any learning based dimensionality reduction methods. The quicker the training, the quicker new types of queries can be indexed from a collection. Table 14 shows that our approach can reduce training cost significantly in comparison to pure neural network. For example, with preprocessing training the pure neural network for dimensionality reduction required 10324 epochs and 10102 epochs for type I and type II query, respectively. In contrast, our proposed approach needed only 4345 epochs and 4767 epochs, nearly 38% and 31% saving. Also, pre-processing by LDA results in less neuron number of input layer and thus overall systematic architecture becomes much simpler.

Query Type	Neural Network (Epoch)	NN+LSM (Epoch)
I	11021	4219
II	10987	4500

Table 14: Performance Comparison on Training Cost for Query Type I and II. NN+LSM denotes our method.

6.3.3 The query accuracy and training trade-off

It is well known that there are a lot of different parameters in a neural network. The configuration of a parameter can influence the final performance of the neural network greatly. One of the major parameters which can be altered is the number of hidden units used in the network. We tested different numbers of hidden users to try to find an optimal choice for the network learning algorithm in the previous experiments.

The number of the hidden units used can affect the network convergence and learning time. Table 11 summaries the learning and query time of neural network with various numbers of hidden units for different datasets. Figure 16 and Figure 19 illustrate the learning curves of the systems containing 10, 20 and 30 neurons in their hidden layer and the result is obtained based on Dataset I and II.

The size of Hidden Layer	Query Accuracy (P@10)		Training Cost (Epoch)	
	Type I	Type II	Dataset I	Dataset II
5	0.4734	0.4019	4521	4590
10	0.4807	0.4077	4219	4500
15	0.4715	0.4072	4098	4001
20	0.4567	0.4063	3921	3902
25	0.4756	0.4056	3609	3821
30	0.4802	0.4199	3521	3709

Table 15: Training Speed and Query Accuracy Comparison based on Different Sizes of Hidden Layer in Neural Network

We can easily observe that the more hidden units the neural network has, the less training cost is required to complete the learning process. For example, based on Figure 18, it takes 4219 epochs and 3521 epochs for system with 10 neurons and 30 neurons to reach converge, respectively. There is a huge difference on training time and main reason for this difference is that that more hidden unit can keep more information and less computational cost will be used to achieve converge. On the other hand, no significant difference on the query accuracy between neural networks containing different number of neurons in hidden layer can be observed.

The system is very robust against change on neuron number in hidden layer. In fact, since the network serves as a dimension reducer, the number of hidden units is restricted to a practical limit. In practice, we need to find a balance the learning cost and performance. Basic principle is that more neuron leads to higher training cost.

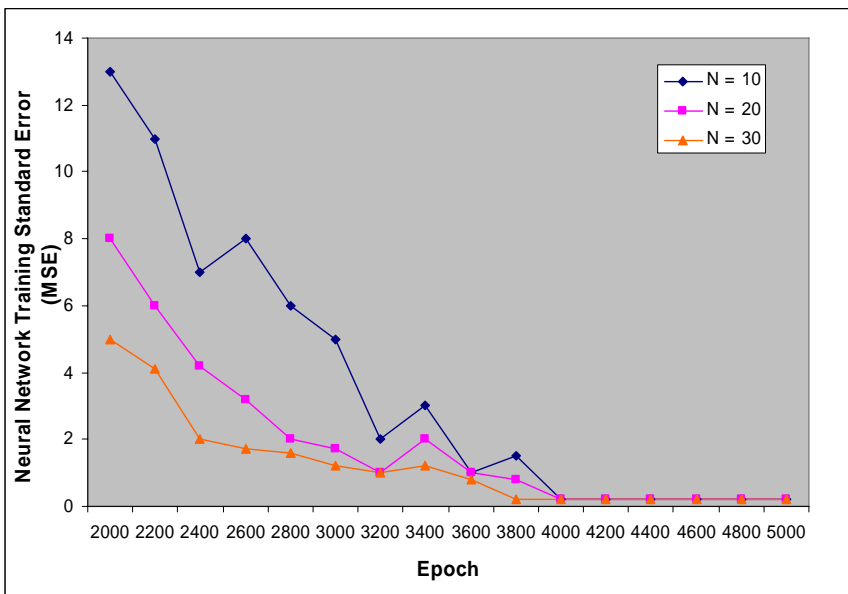


Figure 18: Neural Network Training Standard Error on Dataset I. N equals to neuron number in hidden layer

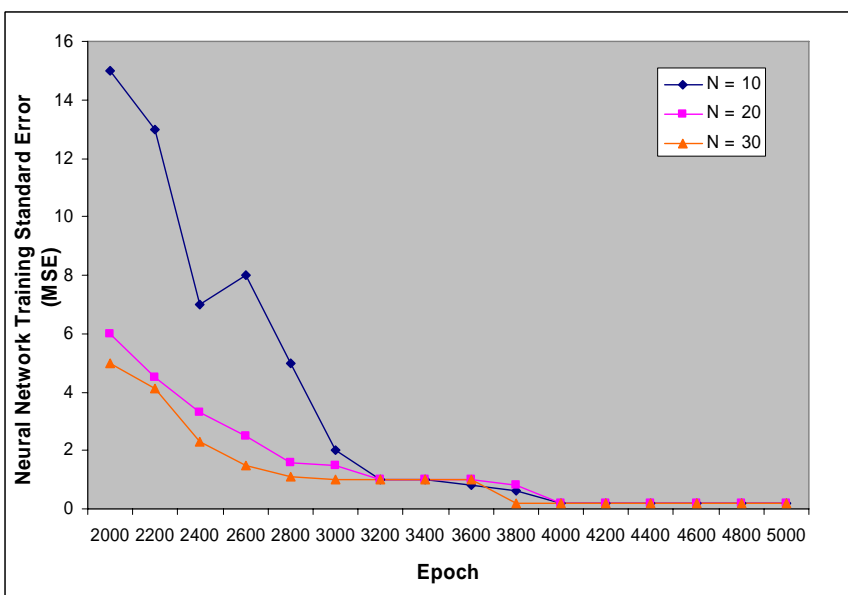


Figure 19: Neural Network Training Standard Error on Dataset II. N equals to the neuron number in hidden layer

6.3.4 Summary of Experiments

The experiments reported above have looked at how the NN+LSM method, which generates feature vectors for music clips which are contextualised by two different query types, has compared with a range of other non-contextual music features (also used in document D5.3.1). The experiments included:

Evaluation of the query effectiveness of the system – how good are the retrieval results? (Tables 11 and 12). This shows the performance of the NN+LSM method is better than using non-context dependent audio features.

The speed of querying using the the new NN+LSM method compared to other feature reduction techniques (table 13), which shows the NN+LSM method is faster at querying an index than the other audio feature types which were considered (with the exception of the LPC feature)

The training cost of the NN+LSM method compared to a standard neural network (table 14), which indicates that the NN+LSM method is significantly faster than using a neural network by itself.

Finally, a comparison of the query accuracy (measured by precision at 10 music clips) versus the number of hidden nodes in the neural network was undertaken (Table 15 and Figures 16 and 17). This shows that a smaller number of hidden nodes can be used, to decrease the training time while not impacting query accuracy significantly.

7 Conclusions

In this deliverable we have presented the state of the art related with context based audio annotation and retrieval. Based on that, we have conducted and described three experiments related with context based audio retrieval in different domains.

The first experiment deals with the audio feature extraction in the text-to-speech problem. The speech data is parameterized extracting the features, which the text-to-speech system needs to conduct the synthesis process. After that, the speech synthesis evaluation is being conducted. The information contained in the speech corpus is needed to conduct the unit retrieval process before generating the synthetic speech, and thus, data retrieval is performed. From the point of view of the experimental results, three different experiments are presented in terms of: *i*) unit-selection (retrieval), *ii*) prosodic modeling, and *iii*) voice quality parameterization to discriminate among expressive speech styles.

The second experiment shows how one can bridge the semantic gap in a concrete field (music moods and emotions). That is, based on the content-based similarity of the audio files, one can derive and automatically annotate new tags to the incoming songs. The sources come then from a social community of users that actively tag and annotate the actual music collection. We think that this approach can ease the cold-start problem, as well as to semi-automatically annotate audio files, taking into account both the content (audio analysis) and the context (social community, and the associated tags).

Finally, the third experiment presents two different ways of classifying artist and genres. The experiment is based on two different types of queries. The first type of query involved finding audio files having similar genre, and; the second one, given a query example, finds the music items performed by the same artist. The results show how our proposed method improves the results from the State of the Art algorithms used for this concrete problem.

8 References

- [Banzhaf1998] Banzhaf, W., Nordin, P., Keller, R.E., Francone, F.D., "Genetic Programming: An Introduction: On the Automatic Evolution of Computer Programs and Its Applications", Morgan Kaufmann, 1998.
- [Barrington et al. 2007] Barrington, L. and Chan, A. and Turnbull, D. and Lanckriet, G. "Audio Information Retrieval Using Semantic Similarity", International Conference on Acoustic, Speech and Signal Processing (ICASSP), Hawaii, 2007.
- [Black&Taylor:1997] A. Black and P. Taylor. "Automatically clustering similar units for unit selection in speech synthesis". In Eurospeech97, volume 2, pages 601--604, Rhodes, Greece, 1997.
- [Cano2005] P. Cano, M. Koppenberger, S. Le Groux, J. Ricard, N. Wack, P. Herrera. "Nearest-Neighbor Automatic Sound Classification with a WordNet Taxonomy" Journal of Intelligent Information Systems Vol.24 .2 99-111, 2005.
- [Cano2005-2] P.Cano, E. Battle, T. Kalker, J. Haitsma. "A Review of Audio Fingerprinting". The Journal of VLSI Signal Processing Vol.41 .3 271 – 284. 2005.
- [Carmel2003] D. Carmel, Y. S. Maarek, M. Mandelbrod, Y. Mass, A. Soffer. "Searching XML Documents via XML Fragments". SIGIR 2003.
- [Carneiro et al. 2005] {carneiro2005fsi} Carneiro, G. and Vasconcelos, N. "Formulating Semantic Image Annotation as a Supervised Learning Problem", Computer Vision and Pattern Recognition, volume 2, IEEE Computer Society Conference, San Diego, 2005.
- [Christianini2000] N. Cristianini and J. Shawe-Taylor. "An Introduction to Support Vector Machines and other Kernel-based Learning Methods". Cambridge University Press, 2000.
- [Costa2004] C. H. L. Costa, J. D. Valle Jr., and A. L. Koerich. "Automatic classification of audio data". In Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics - SMC, Hague, Netherlands, October, 10-13 2004.
- [Croft2003] W. B. Croft. "Language Models for Information Retrieval". Invited Talk. ICDE 2003.
- [Duda2001] R. Duda, P Hart, and D. Stork. "Pattern Classification (2nd Edition)". John Wiley & Sons, New York, 2001.
- [Feng et al. 2003] Feng, Y. and Zhuang, Y. and Pan, Y. "Music Information Retrieval by Detecting Mood via Computational Media Aesthetics", IEEE/WIC International Conference on Web Intelligence, Washington DC, 2003.
- [Formiga&Alías:2006] Lluís Formiga, Francesc Alías; "*Heuristics for implementing the A* algorithm for unit selection TTS synthesis systems*", IV Jornadas en Tecnología del Habla (4JTH06), pp. 219-224, ISBN 84-96214-82-6, november, Zaragoza (Spain). (*in Spanish*)
- [Gonzalvo et al. 2007] Gonzalvo, X., Iriondo, I., Socoró, J.C., Alías, F. and Monzo, C., "HMM-based Spanish speech synthesis using CBR as F0 estimator", Proc. of NoLISP, 2007.
- [Gouyon2004] F. Gouyon, S. Dixon, E. Pampalk, and G. Widmer. "Evaluating rhythmic descriptors for musical genre classification". In Proceedings of the AES 25th International Conference, London, UK, June 17-19 2004
- [Gouyon2006] F. Gouyon, A. Klapuri, S. Dixon, M. Alonso, G. Tzanetakis, C. Uhle, P. Cano. "An experimental comparison of audio tempo induction algorithms" IEEE Transactions on Speech and Audio Processing Vol.14 .5., 2006

- [Hastie2001] T. Hastie, R. Tibshirani, and J. Friedman. "The Elements of Statistical Learning. Springer Verlag", New York, 2001.
- [Hellmuth2004] O. Hellmuth, E. Allamanche, J. Herre, T. Kastner, N. Lefebvre, and R. Wistorf. "Music genre estimation from low level audio features". In Proceedings of the AES 25th International Conference, London, UK, June 17-19 2004.
- [Iriondo et al. 2006] Iriondo, I., Socoró. J.C., Formiga, L., Gonzalvo X., Alías F., Miralles P., "Modeling and estimating of prosody through CBR", JTH 2006 (In Spanish)
- [Jagadish 2005] Jagadish, H., Ooi, B., Tan, Kian-Lee., Yu, C., Zhang, R., "iDistance: An adaptive B+ tree based indexing method for nearest neighbour search", ACM Transactions on Database Systems (TODS), 30(2), pp. 364
- [Kakade2005] V. Kakade, P. Raghavan. "Encoding XML in Vector Spaces". ECIR 2005.
- [Kastner2004] T. Kastner, J. Herre, E. Allamanche, O. Hellmuth, C. Ertel, and M. Schalek. „Automatic optimization of a music similarity metric using similarity pairs". In Proceedings of the AES 25th International Conference, London, UK, June 17-19 2004.
- [INE2004] "Initiative for the Evaluation of XML Retrieval". <http://inex.is.informatik.uni-duisburg.de>, 2004.
- [Jeon et al. 2003] Jeon, J. and Lavrenko, V. and Manmatha, R. "Automatic image annotation and retrieval using cross-media relevance models", 26th ACM SIGIR conference on Research and development in information retrieval pages 119-126, Toronto, Canada, 2003.
- [Juslin 2001] Juslin, P.N. and Sloboda, J.A. Music and Emotion: Theory and Research. Oxford University Press, 2001.
- [Koza1992] Koza, J.R. "Genetic Programming: On the Programming of Computers by Means of Natural Selection", MIT Press, 1992.
- [Li2003] T. Li and M. Ogihara. "Detecting emotion in music". In Proceedings of the International Symposium on Music Information Retrieval (ISMIR'03), Baltimore, MD, USA, October 26-30 2003.
- [Liu2003-2] D. Liu, L. Lu, and H.-J. Zhang. „Automatic mood detection from acoustic music data". In Proceedings of the International Symposium on Music Information Retrieval (ISMIR'03), Baltimore, MD, USA, October 26-30 2003.
- [Lu et al. 2006] Lu, L. Liu, D. Zhang H.J. "Automatic mood detection and tracking of music audio signals", IEEE transactions on audio, speech and language processing, volume 14, pages 5-18, 2006
- [Mass2002] Y. Mass, M. Mandelbrod, E. Amitay, D. Carmel, Y. Maarek, and A. Soffer. "JuruXML -- an XML retrieval system at INEX'02". In Fuhr et al. [8], pages 73—80, 2002
- [Monzo et al 2006] Carlos Monzo, Francesc Alías, Ignasi Iriondo, Xavier Gonzalvo, Santiago Planet; "Discriminating Expressive Speech Styles by Voice Quality Parameterization", 16th International Congress of Phonetic Sciences, pp. 2081-2084, Saarbrücken, Germany
- [Pachet 2005] Pachet, F. "Knowledge Management and Musical Metadata", Encyclopedia of Knowledge Management, 2005.
- [Quinlan1986] J.R. Quinlan. "Induction of decision trees". Machine Learning, 1(1):81-106, 1986.
- [Tokuda et al. 2002] Tokuda, K., Zen, H. and Black, A.W., "An HMM-based speech synthesis system applied to English", Proc. Of IEEE SSW, 2002.

- [Turnbull et al. 2007] Turnbull, D. and Barrington, L. and Torres, D. and Lanckriet, G. "Exploring the Semantic Annotation and Retrieval of Sound", CAL Technical Report CAL-2007-01, San Diego, 2007.
- [Tzanetakis2002] G. Tzanetakis and P. Cook. "Musical genre classification of audio signals". IEEE Transactions on Speech and Audio Processing, 10(5):293-302, 2002.
- [Vapnik 1996] Vapnik, NV., "The Nature of Statistical Learning Theory", Springer, 1996
- [Xu2003] C. Xu, N. C. Maddage, X. Shao, and F. C. Tian. "Musical genre classification using support vector machines". In Proceedings of IEEE ICASSP03, Hong Kong, China, April 6-10 2003.
- [West2004] K. West and S. Cox. "Features and classifiers for the automatic classification of musical audio signals". In Proceedings of the International Symposium on Music Information Retrieval (ISMIR'04), Barcelona, Spain, October, 10-14 2004.
- [Wenyin et al. 2001] Wenyin, L. and Dumais, S. and Sun, Y. and Zhang,
- [Whitman 2005] Whitman, B.A. "Learning the meaning of music", PhD Thesis, Massachusetts Institute of Technology, 2005.

9 Glossary

Partner Acronyms

AM	Activa Multimedia, ES
BLITZ	Blitz Games, UK
CINESITE	Cinesite Europe Ltd., UK
DIT	Dublin Institute of Technology, IE
DTS	Digital Theatre Systems, UK
FBM-UPF	Fundació Universitat Pompeu Fabra, ES
GVG	Grass Valley Germany, DE
JRS	JOANNEUM RESEARCH Forschungsgesellschaft mbH, AT
LFUI	Leopold-Franzenzs Universtät Innsbruck, AT
MTG-UPF	Music Technology Group, UPF, ES
PGP	Pepper's Ghost Productions Ltd., UK
TAIK	Taideteollinen Korkeakoulu, FI
UG	University of Glasgow, UK
URL	Universitat Ramon Llull, ES