Unsupervised Techniques for Audio Content Analysis and Summarization

A First Year Report
Submitted to the School of Computer Engineering
of the Nanyang Technological University

by

Wang Lei

for the Confirmation for Admission
to the Degree of Doctor of Philosophy

May 3, 2008

Abstract

This thesis explores unsupervised algorithms for broadcast audio analysis and summarization. For this work, the audio content analysis and summarization task will be to extract semantic structures from audio databases and organize them into a hierarchical presentation such as a table of contents for applications such as audio retrieval and browsing.

Previous works on audio content analysis and summarization focus mainly on single audio content type, e.g. broadcast news only segments, music and songs structure analysis, commercials detection. As these works operate only on specific audio type, their applications are limited. This motivates us to propose a framework that can examine general audio broadcast database, e.g. radio broadcast or news channel broadcast.

In this thesis, our main focus is to investigate unsupervised techniques to detect repeating patterns in large audio database and explore audio classification methods to assign repeating patterns into semantic classes with priori knowledge. Within each semantic class, existing segmentation techniques could be applied to further classify broadcast audio program into smaller semantic units such as individual news stories so that a table-of-content like audio content structure could be obtained. One novelty of our approach will be to use Acoustic Segment Model to simultaneously process speech and music segments without the need to first discriminate between them [1]. To find repeating patterns, we explore two methods: a) token strings representation of the audio segments and classical string repetition discovery techniques, and b) vector based suffix tree approach.

To verify our research, we will examine our approach on public domain audio database such as the TRECVID and Channel News Asia. We will also apply large vocabulary continuous speech recognizer to generate basic transcription. These text transcriptions will be used along with our proposed approach to find repeating patterns to index news stories and label detected commercials.

Acknowledgments

During the past two years, it was my great honors to work with the groups of distinguished mentors and outstanding fellows in Nanyang Technological University (NTU) as well as Institute for Infocomm Research (I^{2}R), Singapore. Without their advice and encouragement this thesis would not be possible.

Foremost, I would like to express my gratitude to my supervisor, Dr. Chng Eng Siong (NTU), and co-supervisor, Dr. Li Haizhou (NTU, I^{2}R) for their excellent guidance and great support throughout my candidature. I would like to thank them for all their rewarding discussions, corporations, and kind suggestions to my study and life. Their extensive knowledge and sharp insights immensely influenced my research and their lasting encouragements help me to become a capable and confident researcher.

I would like to thank the fellows in our team in NTU for their generous help and corporation. I want to especially express my appreciation to Mr. Xiao Xiong, my senior, for his patience in teaching me the fundamentals of speech recognition as well as his encouragement in adapting myself to the research work. Moreover, I want also thank Mr. Eugene Koh Chin Wei, my teammate, for the useful discussion. My gratitude also goes to other fellows: Mr. Nguyen Trung Hieu, Mr. Omid Dehzangi and Mr. Ehsan Younessian for their friendship and support.

I also would like to thank the colleagues in Speech and Dialogue Processing Lab of I^{2}R for their generous help. I want show my appreciation to Dr. Ma Bin for his discussion on my research work and Dr. Ma Tin Lay Nwe as well as Mr. Wan Kong Wah for sharing me the large amount of database. I also want to thank other current and former colleagues, Ms. Tong Rong, Dr. Kinnuen Tomi Henrik, Mr. Denny Iskandar, Dr. Maddage Namunu Chinthaka, Dr. Sim Khe Chai, Ms. See Swee Lan, Dr. Sun Hanwu, etc for their help and corporation.

In addition, I thank the technical staffs in the Emerging Research Lab at NTU for providing me all the facilities, and thank the technical staffs in Parallel & Distributed Computing Centre (PDCC), BioInformatics Research Centre (BIRC), Centre for Computational Intelligence (C^2i), Database Technology Lab for their kind help.

Last but not least, I want to thank my parents in China, for their constant love and encouragement.

Contents

List of Figures

List of Abbreviation

ANN	Artificial Neural Network
ASM	Acoustic Segment Model
ASR	Automatic Speech Recognition
BER	Band Energy Ratio
BW	Bandwidth
DTW	Dynamic Time Wrapping
EM	Expectation Maximization
FFT	Fast Fourier Transform
GMM	Gaussian Mixture Model
HMM	Hidden Markov Model
HTK	Hidden Markov Model Toolkit
k-NN	k-Nearest Neighbor
KWS	Keywords Spotting
LDA	Linear Discriminant Analysis
LVCSR	Large-Vocabulary Continuous-Speech Recognizer
MAP	Mean Average Precision
MFCC	Mel-Frequency Cepstral Coefficient
NIST	National Institute of Standards and Technology
PCA	Principal Component Analysis
RT	Rich Transcription
SAX	Symbolic Aggregate approXimation
SDR	Spoken Documents Retrieval
SRILM	SRI Language Modeling Toolkit
STE	Short-Term Energy
STFT	Short-Time Fourier Transform
SVM	Support Vector Machine
TREC	Text REtrieval Conference
TRECVID	TREC Video Retrieval Evaluation
VSM	Vector Space Model
VQ	Vector Quantization
ZCR	Zero Crossing Rate

Chapter 1

Introduction

1.1 Problem statement

The amount of multimedia and audio data available in the Internet has increased exponentially in the past decades [2–5], e.g. databases such as broadcast news archives [6], radio recordings [4], music and songs collections [5], TV program archives [7], lecture and presentation recordings [8], meeting room recordings [9], podcasts [10], personal archives [11], etc have become more assessable to the public. However, most of these data have limited label information, or worse yet, have no label information. Hence, it is not easy for users to locate desired clips in such databases as compared to searching for desired text using search engines such as Google [12]. To overcome this limitation, much work have gone into researching for methods that can automatically analyze and summarize multimedia content for search and retrieval purposes [13]. This has also given rise to several established platforms for researchers to exchange and compare their ideas in this area. For example, the TREC Video Retrieval Evaluation (TRECVID) [7] competition and NIST Rich Transcription (RT) evaluation series [9] are well known competitions for content-based digital video retrieval technologies, and automatic speech recognition technologies respectively. These competitions have attracted growing numbers of researchers to investigate three major thrust of related research [13, 14], namely: auditory [15, 16], visual [17, 18] and textual information [19, 20] to analyze multimedia data.

Past research on multimedia data had primarily focus on video and text analysis [13]. However, due to the rapid growth of unstructured audio database [4], the demand for audio content analysis has also increased. In this thesis, we will focus on audio data analysis technology to automatically summarize audio content. By audio summarization, we mean the task to extract semantic structures from audio database and organize them into a hierarchical presentation such as a table of contents for browsing and retrieval purpose. Specifically, we will explore unsupervised algorithms for the required task. The target database to be used for our experiments will be publicly available audio databases such as radio recordings, songs or TV program archives.

1.2 Motivations and research objectives

Existing audio content analysis research primarily focus on single audio content/application type such as transcribing news only segments [3], performing music and songs structure analysis [5], detecting commercials in TV broadcast program [21], transcribing lecture presentations [8], etc. As these research operate only on specific audio types, their applications are limited to the selected audio types and cannot be applied generally. Hence in such works, there is a need to first classify and remove non-desired audio segments as part of the application. This limitation motivates us to propose a framework which can examine more general audio broadcast database such as radio or news broadcast consisting of news reporting, music, commercials, etc simultaneously.

In our proposed approach, we will simultaneously process speech and music segments without the need to first discriminate between them. Our approach incorporates hidden Markov model (HMM) based music models with speech phoneme models in a large-vocabulary continuous-speech recognizer (LVCSR) to decode any given audio segments. Using vector space model [22] techniques, the HMM's output statistics are cast into vectors which can be used to index the audio database. These vectors can also be used as inputs to unsupervised pattern discovery techniques to find repeating audio segments. We will explore two unsupervised repeating pattern discovery methods, specifically a) sub-string repetition search, b) vector based suffix tree repeating pattern search.

Towards the audio content summarization goal, we will investigate audio classification methods to label segments into semantic classes and unsupervised methods to detect repeating patterns in large audio database. Within each semantic class and repeating audio segment, classification and segmentation techniques would be further applied to tag the audio segments into smaller semantic units such as individual news stories, and commercials to generate a table of contents.

To verify our research, we will develop applications to summarize publicly available audio database such as the TRECVID and Channel News Asia. Large vocabulary continuous speech recognizer will first be applied to generate basic transcription. The generated text transcriptions coupled with our proposed features and unsupervised pattern discovery approach will be used towards our audio content summarization task.

1.3 Organization of the report

This report is organized as follows:

Chapter 2 reviews current applications and techniques for audio content analysis and summarization - we review published works of music structure analysis, audio indexing and retrieval, broadcast TV program summarization, audio segmentation and classification methods. We will also survey existing techniques to perform unsupervised repeating pattern discovery.

Chapter 3 introduces the acoustic segment model and vector space model. Specifically, we will present the training process of data-driven music acoustic models.

Chapter 4 introduces the application of the music acoustic models. An audio database indexing and retrieval framework is proposed.

We conclude in Chapter 5, and the future research goal will be presented.

Chapter 2

Audio Content Analysis and Summarization Literature Review

The audio content analysis and summarization task is a broad research topic that has generated much research literature in the past two decades [23]. This chapter reviews recent literature on techniques, implementation, and solutions for the audio content analysis and summarization task. We note that many existing works have focused primarily on single audio content type to limit the complexity and scale of the problem.

The chapter is organized as follows: section 2.1 describes applications developed towards specific audio content class, section 2.2 examines existing solutions and section 2.3 discusses a few unsupervised techniques to detect repeating patterns for both symbolic sequence and time series.

2.1 Current applications of audio content analysis and summarization

Currently, many large audio databases such as music recordings [5], podcasts [10], broadcast news [6], dialogues [24] and conversations [25], meetings/conferences [26], lectures [8, 27] and presentations [28], etc are available to users in the internet. However, as most of these data have limited tag information, interested users cannot easily search for desired audio segments. Hence, the ability to automatically analyze and tag these data has received much interest as the amount of multimedia data available in the internet has increased tremendously.

In this section, we will review applications and techniques developed to automatically analyze and summarize music and broadcast TV recordings. We note that many existing research and techniques have mainly focused on single audio content type such as music-only, speech-only analysis to simplify research issues.

2.1.1 Music structure analysis and summarization

Music/songs data are among the most popular audio content. Much of these data have been produced commercially with additional information online such as title, genre, performer, lyrics, history, and so on [29] to consumers. Although these additional data allow interested users to search for desired songs based on above search criteria, its use is limited when user seeks to query songs based on the music itself, e.g. query by humming [30, 31], or query by similar melody characteristic [32]. For such queries, musically salient features of desired music pieces should be used to query the music databases instead. Such a requirement motivates the research of music structure analysis.

The music structure analysis research [5] is a field of audio analysis that seeks to locate structures such as chorus, verses, bridge, vocal runs, in a music piece. As human listeners usually recognize music structure through the perception of repetition and other relationships within music, segments which occurred frequently such as chorus and verses are key segments to be identified. Therefore, an important component of music structure analysis research is to first detect repetitions [33].

2.1.2 Audio indexing and retrieval

As the huge amount of multimedia data available on the Internet are unstructured, it is not easy to locate desired segments. This prompts researchers to propose many applications to index large audio database, e.g. spoken documents retrieval (SDR) [19, 34–36], keywords spotting (KWS) [37], and query-by-example [21, 38] systems.

Most of the SDR systems aim to retrieve the most relevant spoken documents given a text query from a large speech corpus [19]. For example, the benchmarking platform TREC Spoken Document Retrieval Track [34] focuses on audio retrieval of broadcast news clips. To retrieve spoken documents, a standard approach which most systems [19, 35, 36] used is to first transcribe audio followed by building an index using the

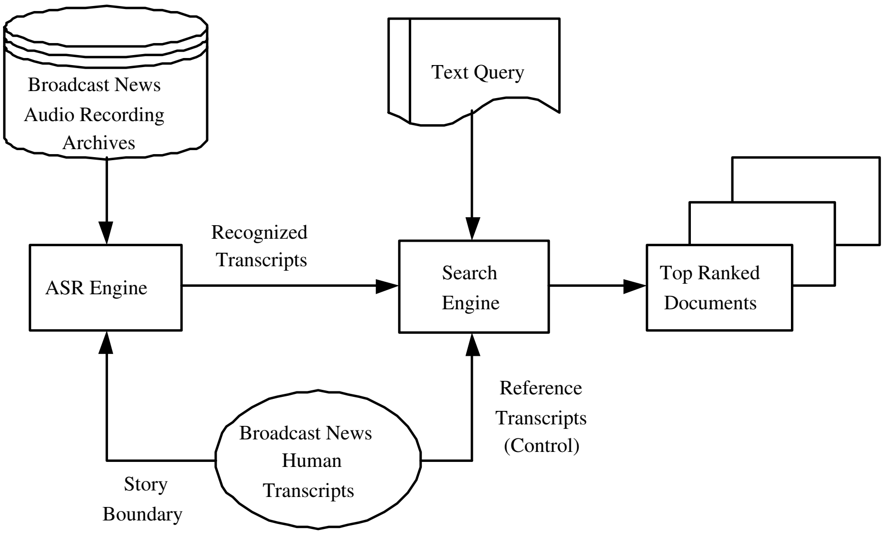


Figure 2.1: SDR system recommended by TREC Spoken Document Retrieval Track 1997.

transcription, as illustrated in Figure 2.1. However, these systems face two major issues: i) out-of-vocabulary queries and ii) high word error rate. To address these problems, the vocabulary-independent approach to speech indexing has been proposed. In this way, speech utterances are first transcribed into subwords [39] or acoustic lattice [40, 41] and then index and search.

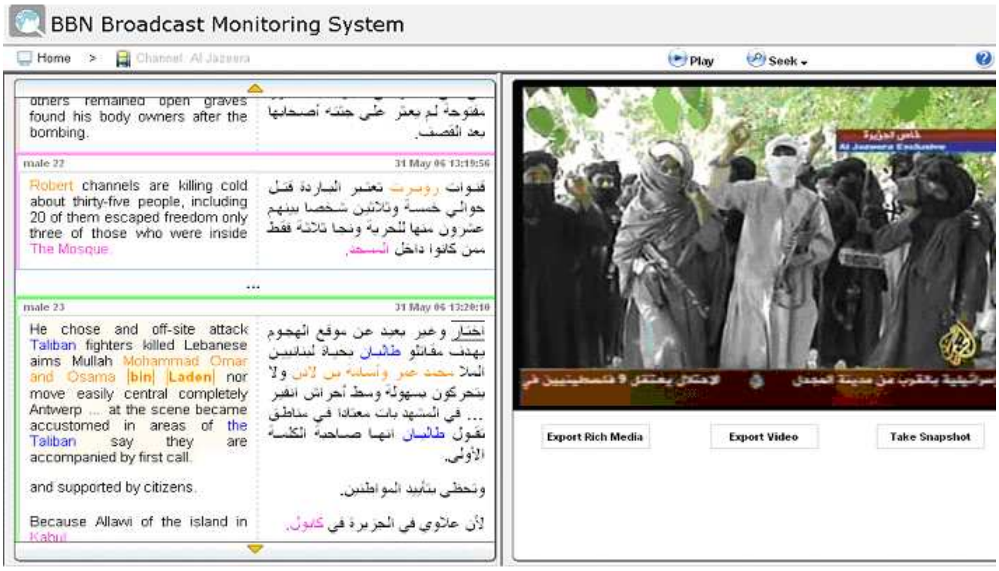
Another application on speech indexing and retrieval is KWS. KWS systems are used for detection of selected words in speech utterances [37]. Actually, KWS is quite similar to SDR as both of them aim to locate queries in speech corpus. Some SDR systems have been extended for KWS task, such as [40].

To deal with generic audio segments, query-by-example applications have been developed to locate similar audio queries. To retrieve similar audio segments, Herre *et al.* [42] extracted fingerprints from audio signal and searched for the queries by matching the fingerprints. However, the approach degrades when there is slight distortion in the audio signals. In contrast, Velivelli *et al.* [38] modeled the predefined audio query using hidden Markov model (HMM) and use the models as queries. However, this approach is not robust since the HMM estimation is poor for short query segment. In another approach, classification methods are popularly involved in query-by-example applications [43], and the similar segments that located in the same semantic class as the query are retrieved.

2.1.3 Speech analysis and summarization

Broadcast TV and radio programs are of major interest to the public. Hence applications to automatically analyze and summarize these data are highly sought. The basic approach will be to first generate a basic transcription of the spoken audio using a LVCSR and then used the transcribed text for indexing purposes or to further apply text summarization techniques to generate summarized information [3, 6, 16, 24, 44, 45].

Speech summarization techniques have been applied to different speech data sources including spoken news [6, 16, 45, 46], presentations [28], dialogues [24] and conversations [25], lectures [27], meetings [47], voicemail messages [48] and so on. For example, one representative system on spoken news summarization is the Rough'n'Ready system developed by BBN Technologies [6]. The system incorporated a wide range of speech technologies to index speech data, create a structural summarization and provide tools for browsing. Figure 2.2 shows the BBN Broadcast Monitoring System which is a variation of the Rough'n'Ready system.



others remained open graves found his body owners after the bombing.	مفتوحة لم يعثر على جثته أصحابها بعد القصف.
male 22	31 May 06 13:19:56
Robert channels are killing cold about thirty-five people, including 20 of them escaped freedom only three of those who were inside The Mosque.	قنوات روبرت تعتبر الباردة قتل حوالي خمسة وثلاثين شخصا بينهم عشرون منها للحرية ونجا ثلاثة فقط ممن كانوا داخل المسجد.
...	
male 23	31 May 06 13:20:10
He chose and off-site attack Taliban fighters killed Lebanese aims Mullah Mohammad Omar and Osama [bin] [Laden] nor move easily central completely Antwerp ... at the scene became accustomed in areas of the Taliban say they are accompanied by first call.	اختار وعبر بعيد عن موقع الهجوم يهدف مقاتلو طالبان بحياة لبنانيين الملا محمد عمر وأسامة بن لادن ولا يتحركون بسهولة وسط أحراش انفير ... في المشهد بات معتادا في مناطق تقول طالبان انها صاحبة الكلمة الأولى.
and supported by citizens.	وتحظى بتأييد المواطنين.
Because Allawi of the island in Kabul	لأن علاوي في الجزيرة في كابول.

Figure 2.2: BBN Broadcast Monitoring System. The application can show transcription in both the original language and English with the synchronized video. (From BBN website)

Another example of speech summarization technology is the research undertaken by the Japanese national project "Spontaneous Speech: Corpus and Processing Technology" in 1999 [49]. They [16] proposed a probabilistic method to extract relatively important words from transcribed speech and use these words to generate summarized text sentences. In their recent work [3], a two-stage summarization framework consisting of important sentence extraction and word-based sentence compaction was used to produce text summaries. Their approach also directly produces audio summaries by identifying important sentences, words and between-filler units from original utterances. Also using utterance extraction techniques, Zechner [24] proposed an automatic summarization system called DIASUMM to summarize open-domain spoken dialogues. The system use dialogue transcription as input and identify important utterances by detecting and removing speech disfluencies, determining sentence boundaries and linking cross-speaker information units.

Besides sentence extraction techniques, other methods based on transcription also have been explored. Valenza *et al.* [44] combined acoustic confidence measures [50] and inverse frequency values to evaluate individual words in the news transcription so that higher ranked words could be included in the summaries. Jin and Hauptmann [45] introduced several supervised strategies to generate titles for broadcast news with the aid of pre-selected news titles in the training corpus.

In the previously mentioned works, the LVCSR was applied to generate basic transcription. However, the LVCSR recognition performance on broadcast data is poor. This leads to much work to improve LVCSR performance by incorporating prosodic features into transcription [51–53]. The prosodic features such as intonational variation in pitch, energy, pause and speaking rates are found to signal the relative importance of speech segments [54]. An early application of prosody on speech summarization was for voice-mail messages [48]. The researchers explored the use of prosodic and lexical features to discriminate the important segments/words of the transcription. In the spoken news summarization applications, prosodic features such as pitch, power and pause were employed to locate stressed words [51]. For talk shows [53], prosodic features such as pitch, phoneme duration, sentence length and power were used together with linguistic information to evaluate the importance of each sentence in the transcription.

2.1.4 Sport events detection

Broadcast sports program is also an important class of multimedia data. Much research have been done on sport event detection using audio and video information. This section discusses existing works which exploit the use of audio information for sports event detection.

Most sports video consumers are usually more interested in the highlights of a sports video than the bulk of scenes, e.g., for a soccer game, the highlights are the goal scoring scenes, foul scenes, penalty shots, etc. Hence, automatic highlights extraction techniques [55–58] have been examined and proposed. To detect these scenes, audio content are sometimes exploited. For example, in [55], the announcers' excited speech and ball-bat impact sound were used to detect possible highlight candidates. In [56], audio signals were classified into semantic classes such as applause, cheering, music, speech and speech with music and used as features for subsequent classifiers to identify highlight segments. In another approach, the ball hits have been detected to indicate the highlights in table tennis games [57]. Moreover, high energy segments were found to closely follow the occurrences of goals in basketball games [58].

2.1.5 Movie and situation comedies content analysis

In contrast to sports program, movie and sitcom programs cannot be easily broken down to a series of events as many portions of the program may be equally important and the event borders may not be obvious [59]. For a review of the existing works on these data, see [60].

The audio classification approach has been commonly included in some frameworks to segment and analyze movie and sitcom [60–65]. In [60] and [62], audio features such as short-term energy and pitch were extracted to measure a movie's tempo so that the movie could be segmented according to the changes of tempo. To assign segments into semantic classes, HMM was used to identify emotional events such as laughter and horror [61, 63]. In other research, different audio features including short-time energy [64],band energy ratio, zero-crossing rate, frequency centroid, bandwidth and MFCCs were combined to classify gunfires, explosions, car braking and other audio effects in action movies [65].

2.2 Current approaches to audio analysis and content summarization

To perform the audio content analysis and summarization task, the basic operations are summarized in Figure 2.3. These operations are: feature extraction, audio classification and segmentation techniques, repetition detection techniques, and rule-based techniques. In this section, we will discuss the generic modules: feature extraction, audio classification and segmentation, and repetition detection.

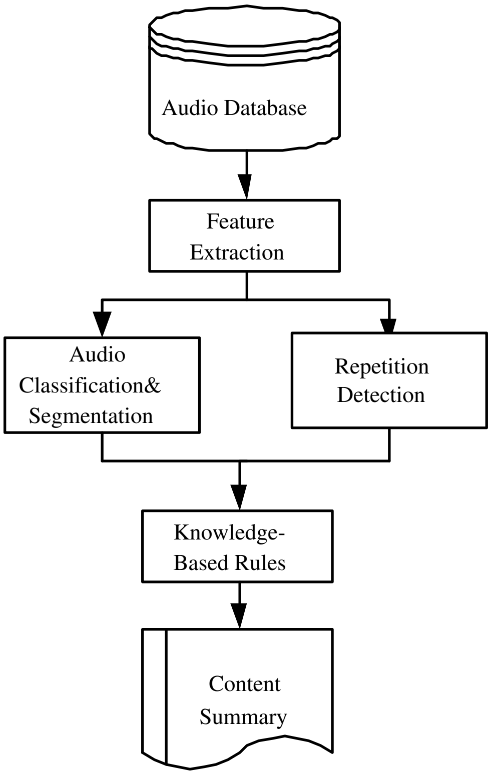


Figure 2.3: Current approaches for audio content analysis and summarization

2.2.1 Audio feature extraction

Many audio features have been proposed to characterize audio signals for different purposes. In some literatures [14], audio features have been classified into two main categories: a) physical features extracted directly from the audio waves, and b) perceptual features that related to audio perception of human beings. Some of the most commonly used features are described below [14, 66]:

Short-term Energy (STE): STE is also referred as volume or loudness and it is the total spectral power of an audio signal. It is widely used in voice activity detection where the distinction between speech and silence is required.

Band Energy Ratio (BER): BER models the distribution of spectral power in different sub-bands of frequency domain. It is obtained by normalizing sub-band energies by the frame STE.

Zero Crossing Rate (ZCR): ZCR is computed as the total number of times that the waveform crosses the zero axis. It is useful to discriminate speech from non-speech components such as music.

Brightness: It is computed as the centroid of the short-time Fourier magnitude spectra measured in log frequency. It is sometimes referred to frequency centroid (FC).

Bandwidth (BW): BW represents the power-weighed standard deviation of the spectrum in a frame. Brightness and bandwidth are usually combined to describe statistical characteristics of the spectrum in a frame.

Pitch: It is defined as the sensation of frequencies of a sound. It also refers to the fundamental frequency (F0) of an audio segment. F0 is a physical term of audio waveform, however, pitch is a perceptual term and it is the perceived F0. Normally voiced speech and music have well-defined pitch.

Mel-Frequency Ceptral Coefficient(MFCC): MFCC [67, 68] is the most popular features for LVCSR. The feature captures the short time segment statistics based on models of human auditory perceptive system with respect to different frequency range.

Chroma Feature: Chroma feature [69] is a powerful representation of music audio. Chroma features captures the salient features of music by projecting the spectrum into 12 bins of semitones (or chroma) of the musical octave.

2.2.2 Audio classification and segmentation

The audio classification and segmentation techniques refers to research to classify and segment audio streams into different sound classes such as speech, music genre, background noise, etc. The techniques proposed in this research have been widely employed in systems such as multimedia indexing and retrieval systems [15, 70, 71], multimedia content analysis frameworks [65, 66, 72] and applications such as speaker diarization [9].

The following sections introduce different techniques of audio segmentation and classification for three tasks: traditional speech/music discrimination, generic audio segments classification and music genres classification.

2.2.2.1 Traditional speech/music discrimination

Speech and music discrimination is a traditional audio classification problem motivated by the need to identify speech only or music only segments for subsequent processing. For example, in LVCSR application of broadcast news, the music segments must first be removed [73]. There are many existing works to discriminate speech from music and most of them explored robust audio features to train pattern classifiers in a supervised manner, e.g. classifiers such as GMM [73], artificial neural network (ANN) [74, 75] and nearest neighbor (NN) classifier [76] have been widely employed.

Besides the above mentioned works on speech/music discrimination, new approaches have been proposed classify audio segments into a wider range of semantic classes. Such works will be discussed in the following section.

2.2.2.2 Generic audio segments classification

As an extension of speech/music discrimination techniques, many systems have been proposed to classify components of audio recordings into broader categories: speech, music, speech with music background, laughter, silence and environmental sounds [72, 77]. E.g., in [77], the audio signals were from meeting room recordings while [72] examined TV broadcast recordings. In speaker diarization task [66], the identification of speech segment is is further refined to segment speech by different speakers.

Musical instrument sounds and sound effects classification have been also been studied in [15, 71, 78]. E.g., Wold *et al.* [15] first studied content-based classification of a wide

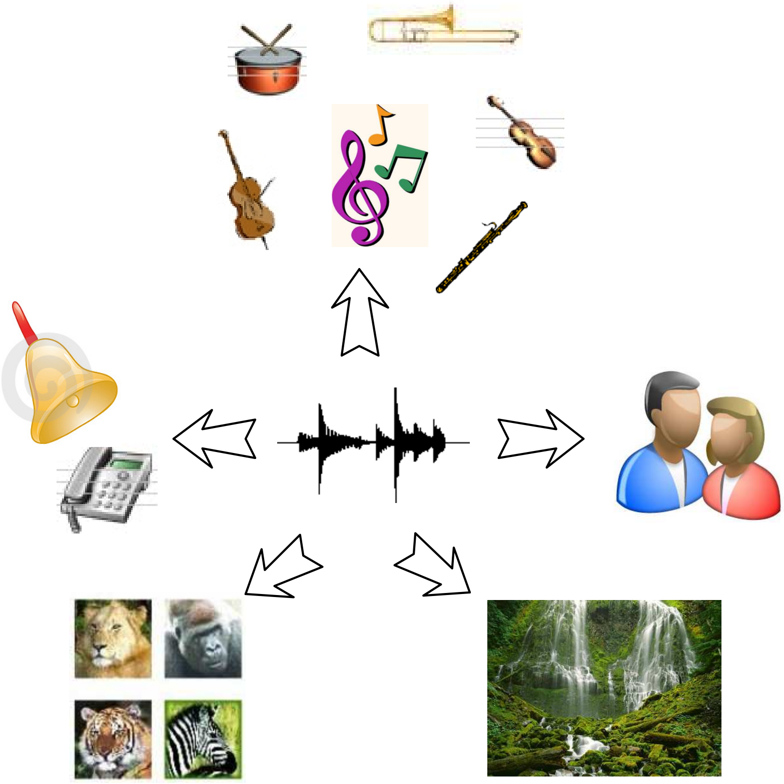


Figure 2.4: Content-based classification of sounds in [15]. They categorized a wide variety of sounds into five major classes: animals, machines, musical instruments, speech and nature. Some major classes can be further divided according their contents.

range of sounds that shown in Figure 2.4. More recently, Cai *et al.* [71] reported a flexible framework to detect applause, car-racing, cheer, car-crash, explosion and other key audio effects in movies and entertainment TV shows. They further extended their work by including inference methods to extract high semantic level context information. Another approach on automatic context classification [78] built k-Nearest Neighbor (k-NN) based and HMM based semantic models for contexts such as street environment using simplistic low-dimensional feature vectors.

2.2.2.3 Music genres classification

A music genre is a term that describes the process of dividing popular music into categories. Although musical genres do not have a strict definition, at least pieces of music in the same category should share similar characteristics such as rhythmic structure, instruments, etc [79]. Therefore, automatic music genres classification task refers to classification of music and songs according to their styles such as pop, jazz, rock, techno, soul, orchestra, etc. The music genre classification is not strict, and is often controversial - Wikipedia alone has listed hundreds of music genres [80].

Music genres classification is a pattern classification problem [81]. Its realization basically involves two steps: feature extraction and multi-class classification. Early work reported the use of three types of features: timbral texture features, rhythmic content features and pitch content features [79]. Timbral textual features are calculated from short-time frames based on short-time Fourier Transform (STFT). It includes physical features such as ZCR, low energy, and brightness as well as cepstral features such as MFCC. In addition, Daubechies wavelet coefficient histogram [82], octave-based spectral contrast [83] and octave-based modulation spectral contrast [84] have also been investigated. Different to the timbral textual features which commonly used for general audio signal classification, rhythmic content features are mainly extracted for music beat selection [85] based on the wavelet transform. Pitch content features [86] characterize the presence of musical tones so that they can describe the melody of the music.

To classify the features, multi-class classifiers such as the hidden Markov model (HMM) [79], k-Nearest Neighbor (k-NN) [79], Gaussian Mixture Model (GMM) [83], linear discriminant analysis (LDA) [84], have been studied. In contrast to the above flat classification of musical genres, hierarchical taxonomy [82] has also been used to identify the relationships among musical genres as well.

Instead of directly using the features extracted from music signal for classification, an intermediate step that first transcribes music to symbols was proposed in [87]. Chen *et al.* decoded music into sequence of predefined symbols using HMM and applied text categorization method to generate n-grams features followed by multi-class classification. This work leads to a framework which can analysis both speech and music audio simultaneously. We will discuss more details of such framework in Chapter 3.

2.2.3 Repetition detection

Another approach for audio content analysis and summarization is to detect the repeating patterns in the audio database. The repeating patterns refer to the recurrent segments with similar statistics and semantics in the audio segments. Examples of repeating patterns include the commercials, theme-music, music/song pieces, etc in audio broadcast program [4, 21] as well as the chorus or and verse sections in a piece of music.

To detect repeating patterns in audio broadcast program, [21] and [4] proposed to construct templates for different audio events and match the audio corpus to these templates in a brute-force manner. However, such methods are not suitable for large database as not only it is computationally expensive but also it requires a manual generation of audio templates. To efficiently and automatically discovery repeating patterns, unsupervised techniques have been widely examined and will be discussed in the next session.

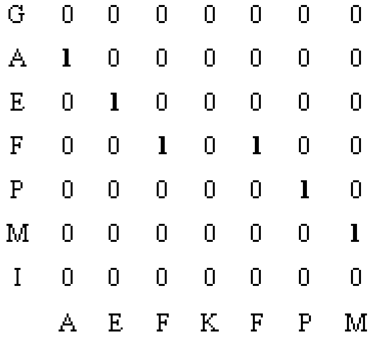
2.3 Unsupervised techniques for repeating pattern discovery

2.3.1 Techniques for symbolic sequence

Repeating pattern discovery of symbolic sequence has been used in many different applications such as motif finding in genomic sequences [88, 89] and themes extraction in music [90]. Researchers have studied different data representation techniques to efficiently detect repeating or similar patterns in data source. From the late 1990s, research on symbolic sequence has resulted in the development of numerous methods to discover repeating patterns or motifs. For example, Bailey and Elkan [88] extended Expectation Maximization (EM) algorithm to produce the Multiple EM for Motif Elicitation tool. This tool is able to discovery new motifs in a set of genomic sequences without priory knowledge. More methods to discovery repeating patterns can be found in [89] and [91]. Among these methods, we will only discuss three approaches: suffix tree [92–95], dot-matrices [90, 96], and incremental search [97, 98].

One approach to find repeating pattern is to represent the token sequence as a suffix tree structure so that the occurrences of each subsequence can be retrieved from each non-leave nodes [92]. Linear time algorithms such as [93] have been proposed to construct suffix tree efficiently. To further improve the efficiency of suffix tree to detect repeating substring patterns, He [94] proposed to first find all possible candidates for repeats and then prune the false alarms. However, suffix tree only works effectively for exact match sequence. For symbolic sequence interspersed by unwanted characters, an inexact-suffix tree structure which allows motifs discovery has been studied in [95].

Another approach to discovery repeating patterns in symbolic sequence is to construct dot-matrices [96]. The dot-matrices were initially proposed to represent pairwise alignments of genomic sequences. Each element in the dot matrix indicates whether the corresponding two symbols are identical or not, as illustrated in Figure 2.5. The dot-matrices have been further extended in [90] to produce correlative matrix which used to find the nontrivial repeating patterns from music MIDI sequence. However, such approach is in a brute-force manner so that it is not suitable for long symbolic sequence.



$$\begin{array}{cccccccc}
G & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
A & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
E & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
F & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
P & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
M & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 & A & E & F & K & F & P & M
\end{array}$$

Figure 2.5: A dot matrix for pairwise symbolic sequences "GAEFPML" and "AEFKFPM". The element "1" represents the corresponding symbols from the two sequences are the same; In contrast, "0" represents they are different.

In contrast, the incremental search approach aims to detect repeating patterns in long symbolic sequence. Such algorithms first generate short candidate strings and search them in the long sequence. The candidates that never rarely occur are removed, and the remaining candidates will be appended with new symbols to form new candidates strings for the next iteration [97]. To avoid costly repetition of searching, an efficient algorithm that examines fewer intermediate repeating patterns has been proposed in [98]. This algorithm scans the long sequence once to generates short potential repeating substrings followed by connecting them into a directed graph. Maximum-length repeating pattern searching is carried out by examining the edges of the graph.

2.3.2 Techniques for vector-based sequence

In contrast to symbolic sequence, vector-based sequence discovery has also been applied. Examples include multidimensional time series such as fetal ECG data [99, 100] as well as sequence of feature vectors extracted from audio segment [101]. In addition, the numerical time series [102] is also included here as it can be considered as 1 dimensional vector sequence.

2.3.2.1 Clustering techniques

Unlike symbolic sequence, it is impossible to detect repeating patterns based on exact matching for a vector-based sequence. Instead, researchers attempted to convert vector-based sequence into symbolic sequence so that the techniques in section 2.3.1 could be employed. The common approach to solve this problem is using data clustering or vector quantization (VQ) [68, 103], specially unsupervised clustering is first performed and each target vector is represented by its class identity. Existing clustering methods have been reviewed in many literatures [104, 105] and different taxonometric presentations of the methods are available. In this thesis, our taxonomy of clustering methods follows [105] and it is shown in figure 2.6.

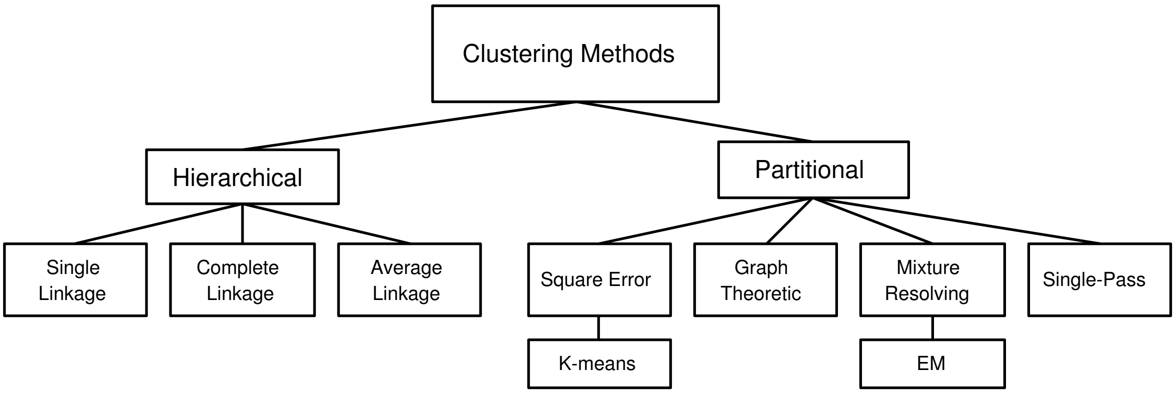


Figure 2.6: A taxonomy of clustering methods [105].

The clustering methods can be divided into two broad categories at the top level: hierarchical and partitional methods. The hierarchical agglomerative clustering algorithm initially treats each pattern vector as an individual cluster and then merge the most similar pair of clusters agglomeratively. Eventually, a hierarchical structure is obtained to show the relationship among all the pattern vectors. To measure the distance between two clusters, linkage is used to select single points that can represent the clusters. Three linkage types are commonly used: 1) single linkage defines the distance between two clusters as the minimum distance of all possible pairs of pattern vectors in the two clusters. 2) Complete linkage is opposite to the previous one and it uses the maximum distance. 3) Average linkage calculates the mean distance.

In another category, partitional algorithms aim to obtain a single partition of the data instead of constructing a hierarchical structure. There are several strategies to partition data [105]. First, squared error criterion is the most frequently used because of its simplicity in implementation. One typical example using square error is k-means clustering which iteratively reassigns the data to clusters based on the distance between data and clusters until convergence. Secondly, graph theoretic clustering presents the relationship among all the data so that the partitions will be clearly archived [106]. The well-known graph-theoretic divisive clustering algorithm obtains the partitions by removing the long edges from the minimal spanning tree constructed for data. Thirdly, mixture resolving methods assume the data are drawn from different distributions so that their goal is to determine the parameters of the models, e.g. the EM algorithm [104] has been applied to parameter estimation. Lastly, single-pass methods [107] aim at reducing computation effort for large size of data and partitioning data in linear time. The single-pass algorithm is based on a first-come-first-serve discipline and assigns data into the currently nearest cluster. Such method can only produce an approximative partition.

Besides the above classic clustering methods, other approaches have also been proposed to symbolize the vector-based sequence. Lin *et al.* [102] proposed Symbolic Aggregate approXimation (SAX) to represent any numerical time series into to lower resolution sequence of symbols. Tanaka *et. al* [100] further extended SAX algorithm by employing principal component analysis (PCA) to process multi-dimensional time series data.

2.3.2.2 HMM based techniques

In contrast to first symbolizing vector-based sequence, HMM-based techniques that can directly discovery repeating patterns have been proposed. The hidden Markov model (HMM) [103, 108] is a statistical modeling technique that characterizes the spectral properties of the frames of an audio signal. Although HMM is usually trained in supervised manner, recently unsupervised variations of HMM have been investigated [109–112]. For

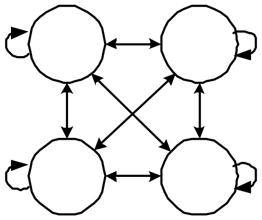


Figure 2.7: Structure of a 4 state ergodic HMM used in [109]. Each state is modeled by a Gaussian distribution.

example, the ergodic HMM [109], illustrated in Figure 2.7, was used in the music structure analysis task to group and label similar fixed-length segments using MFCC features. Heuristics were then applied to choose the key phrase from among the frames with the most frequent labels. Aucouturier and Sandler [110] further explored ergodic HMM using spectral envelope features. In their work, each state of the HMM was assumed to represent a musical part so that music piece could be segmented. In their works, each feature vector captures only 20-30msec statistics of the audio data and hence will not model music data well. To improve modeling, Abdallah *et al.* [111] used longer duration cepstral feature vectors to train the HMM instead. In their implementation, the decoded Markov state sequence was modeled using histograms and the histograms are clustered to find repeating patterns. In yet another variation of HMM, the hierarchical HMM was explored to discovery the recurrent patterns in video achieves [112], and the researchers proposed unsupervised adaptation algorithm to automatically model newly occurred events using both auditory and visual features.

Although HMM-based techniques are capable of capturing the temporal statistics of the data sequence, complex methods such as EM algorithm [104] are required to estimate their parameters. Instead of using such complex techniques, straightforward techniques such as self-similarity matrix have been proposed.

2.3.2.3 Self-similarity matrix

The self-similarity matrix [113–117] has been widely studied for repetition detection in audio segments, especially for music structure analysis. Although such brute-force technique is time consuming, it is feasible to analyse music piece as such data lasts for only several minutes.

A self similarity matrix is constructed based on measures such as correlation between the feature vectors of an audio segment. This idea was first proposed by Foote [113] in which he constructed a brute force frame-to-frame similarity matrix to visualize similarities between segments of the music. The repeating musical patterns are then found using image processing techniques by processing the sub-diagonal lines in the similarity matrix. Foote [118] also described a method to detect the music segment boundaries by analyzing local similarity of adjacent musical signals. Foote's work was however based on MFCC features which although is robust for speech recognition applications, is not suitable to analyze music data [119]. Other researcher working on similarity matrix to detect repeating patterns have proposed better music features. Bartsch and Wakefield [69, 114] introduced the chroma features which can represent the cyclic attribute of pitch perception for music notes. Their experiments have shown that the chroma features were able to represent redundant structures such as chorus and verse within a given song. Their work was extended by Goto [115] to detect and differentiate the chorus section from other repeating patterns in music audio signals. Other extensions of chroma feature include Zhang and Samadani's work [116] which used similarity measure based on blocks of chroma features to detect longer repeated melody. In another approach, Lu *et al.* [117] applied constant Q transform on music piece to extract features and used image processing techniques prior to repeating pattern detection process to enhance the stripes in the features self-similarity matrix.

Inspired by the previous work of repetition detection, we believe that these techniques will benefit audio analysis and summarization task. However, before exploring unsupervised repetition detection methods, robust features should be generated for general audio signal such as broadcast TV program. In the next chapter, we propose a novel feature motivated by the Acoustic Segment Model (ASM) and implemented using LVCSR and Vector Space Model (VSM).

Chapter 3

Audio Content Characterization based on Audio Segment Models

General audio broadcast program usually contains audio segments that are speech only, music only, speech and music mixed segments, and special effects sounds segments. However, most existing applications can only operate on a mono-audio type, e.g. applications such as news summarization and news retrieval [6, 16], music genre classifications [87, 115], audio effects classifications [71], etc. The basic approach adopted by these researchers is to first segment the audio signals into different classes [72, 73, 77], and then subsequently apply specific methods to further classify/analyse the signals as discussed in Chapter 2.

In our study, we aim to remove the need to classify between speech and music segments in our audio analysis system. Towards this goal, we propose novel features which can represent them simultaneously to detect repeating patterns.

The basic system used to analyze recorded broadcast news speech is the LVCSR system. The typical LVCSR system models speech sounds using the HMM [68], and each HMM models a phoneme [68, 103] of the language. The training of these HMMs is carried out in a supervised manner and is therefore language and task dependent. As such, the speech data needs to be manually transcribed.

To analyze music, we will require an equivalent HMM system for music data. Although the same HMM models and decoder structure of the LVCSR can be used, However, the procedure to train the phoneme acoustic model cannot be easily applied as there is no simple or effective way to generate music transcription. To remove the need to define acoustic models and manual transcription for music, we propose a data-driven acoustic modeling technique, the Acoustic Segment Model (ASM) [120], to train our music HMM.

In the following sections, a brief introduction of speech acoustic models and music acoustic models is given. In addition, we will examine two different decoding output of the decoder and the Vector Space Model (VSM) to generate new features from the statistics of the decoding output.

3.1 Acoustic modeling for speech and music

3.1.1 Speech phoneme models

The basic acoustic symbols for a speech decoder can be words [121], subwords [122] or phonemes [68, 103]. Depending on the size of the speech recognition task, from small vocabulary, medium to large vocabulary, the acoustic models used by the decoder may be different.

For the LVCSR task, existing state of the art recognition system use the HMM to model phonemes. The HMM is used to model the statistical behavior of the audio features for the phoneme and has the ability to compensate for some of the variability of speech production [68].

In our systems, the LVCSR module uses 39 HMM models to model the 39 phonemes of the English language with an additional silence model. To generate the acoustic models, we have used the Wall Street Journal database [123, 124] and the HTK [125] to train the HMM. Each HMM consists of three states and each state has 16 Gaussian mixtures. The features used are the MFCC features as well as their first and second derivatives.

3.1.2 Acoustic Segment Model (ASM) for music

The procedure to train the HMM requires transcribed data. As it is not straightforward to transcribe music, a data-driven approach to create music acoustic models is used. The data-driven approach we have applied is the Acoustic Segment Model (ASM) approach. The ASM was first proposed to characterize fundamental speech sounds for speech recognition in [120] and subsequently extended by Li *et al.* [22] to train universal phoneme models for the language identification task.

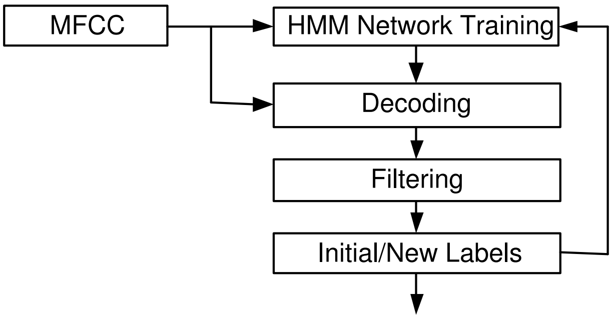


Figure 3.1: Iterative HMM training procedure of music ASMs.

The ASM training process, illustrated in Figure 3.1, is applied to generate 40 HMMs for music data. We have decided on 40 music models so that the final system will have the same number of speech and music HMM models. The ASM procedure is as follow:

Step 1) Segment the music signals of the training corpus into equal length segments.

Step 2) Cluster the segments into 40 clusters using k-means clustering; Label the segments in the entire corpus with the found cluster identity. The initialization step is illustrated in Figure 3.2.

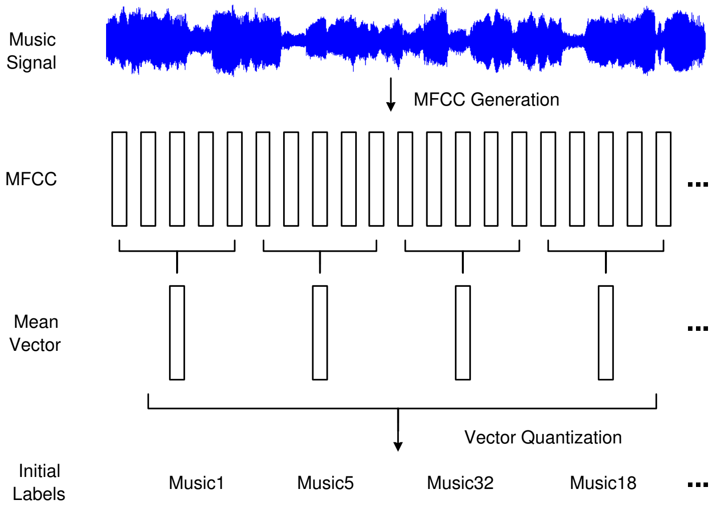


Figure 3.2: Initialization of the training of music ASMs.

Step 3) Create 40 initial HMM models to represent cluster identities found in Step 2 and adapt them using the labeled data.

Step 4) The trained 40 HMMs are used to decode the training corpus. The decoded tokens with duration less than a pre-defined threshold are removed. In our experiment, segments with duration of less than 100msec are removed.

Step 5) The HMMs are re-adapted using the features from the new token labels found after Step 4.

Step 6) Repeat steps 4 and 5 until convergence.

The above approach iteratively generates better HMM models to represent the music data. Each HMM model can be interpreted as a model to represent a particular sound class just like k-means clustering. The ability to capture temporal information of sound class, however makes HMM significantly more robust than k-means clustering. For our implementation, the music training corpus consists of 50 pieces of pure music played by five different instruments and 30 pieces of songs in three styles. All the music pieces are converted from MP3 format and down-sampled to 11 kHz to be consistent with the speech training data. At the end of the training procedure, 40 music HMM models are obtained.

With the trained 40 phoneme models and 40 music ASM models, we are now able to convert an audio signal into text-like transcripts. The HMM decoder can generate results in different formats [125], e.g. top-1 acoustic sequence, top-n acoustic sequences, and acoustic lattice. Our system considers two of the output formats, the top-1 acoustic sequence and the acoustic lattice. We will extract statistics from these two decoder output to generate new features using the Vector Space Model (VSM) approach. The VSM approach is discussed in the next section.

3.2 Vector Space Modeling (VSM)

The Vector Space Model (VSM) [126] was first introduced to represent text documents as vectors of terms in the text retrieval task. The elements of the vectors are counts of occurrence of respective words in a given text. Hence the vector captures the frequency of occurring word in the given text. This idea has been widely applied and extended in different areas such as text retrieval [20], language identification [22], music retrieval [127], music genres classification [87], etc.

In the language identification task [22], a VSM-based approach was proposed to extract the n-gram statistics of the phonemes for a given speech segment to generate a *bag-of-sounds* vector. Inspired by this method, we propose to use the decoded acoustic sequence or lattice to generate an n-gram vector to index the audio segment. In other words, we use the VSM to transform the decoder's output into a vector containing the segment's feature n-gram statistics.

3.2.1 Top-1 acoustic sequence

The top-1 decoder's result is a sequence of phonemes and music HMM models with timing boundary to represent the given audio segment based on maximum likelihood decoding criterion [108] using the Viterbi algorithm. The top-1 sequence can be used to derive an n-gram vector by counting the frequencies of each n-gram terms. For example, if we wish to find the 1-gram (unigram) vector, we simply count the occurrences of the acoustic/music models for the given audio segment. In our experiments, the number of HMM models is 80, i.e. 40 music, 39 speech plus 1 silence model. Hence the vector generated is of length 80, i.e., the elements of the vector is a count or a normalized count of the model occurrence in the given audio segment. Such vectors have been referred as "hard-counting" of the decoding output in some literatures [127]. Alternatively, we could have generated the bi-gram VSM, i.e., we can generate a vector of 81×81 (one additional dummy model is introduced to represent the starting/ending point of the top-1 sequence), where each term in the vector registers the frequency of occurrence of the bi-gram term.

3.2.2 Acoustic lattice

One other possible output from the decoder is the acoustic lattice. The acoustic lattice is an intermediate representation of the decoding process [125]. It is defined as a connected loop-free directed graph with each node representing a time frame and each arc representing a token hypothesis with a likelihood score. At every time frame during decoding, the potential acoustic sequences are updated and the higher ranked tokens ending at that

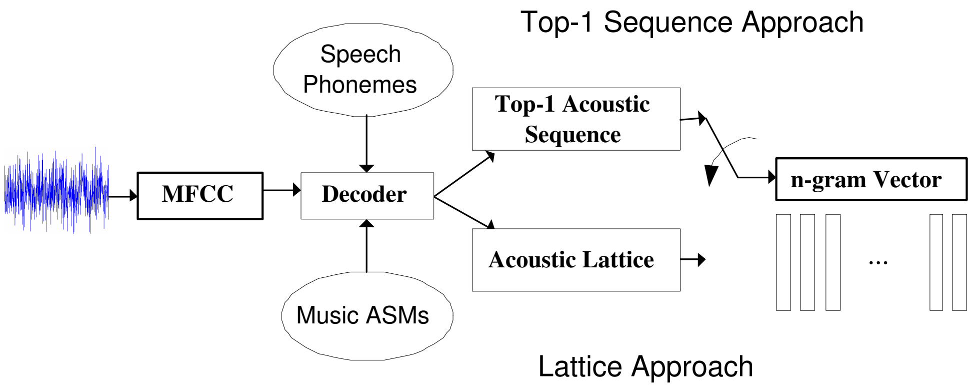


Figure 3.3: Two approaches of audio decoding.

frame are stored. Therefore, the lattice stores multiple token hypotheses for every point throughout the audio segment [128].

Acoustic lattice have been widely used in many speech processing areas such as speech indexing and retrieval [40, 128], language recognition [129], etc. There are several reasons to use lattice: 1)Acoustic lattice can provide more accurate information than top-1 decoding sequence. 2)Similar to phonemes and sub-words [39], it also does not suffer from out-of-vocabulary problem.

The acoustic lattice generated by the decoder can also be used to generate n-gram vector. Currently, the SRILM Toolkit [130] is used for this purpose. The expected count of one n-gram term is computed using the lattice posterior probabilities of all the arcs corresponding to this term. Details of generating n-gram vector from lattice can be found in [129]. All the vectors are normalized to unit length. Other researchers have sometimes referred to this process as "soft-counting" of the decoding output [127].

Fig 3.3 illustrate the two possible ways to generate a VSM from the decoder's output.

3.3 Conclusion

This chapter describes a novel audio feature which can process speech and music simultaneously. We use ASM to train the music acoustic models, and incorporated them with the speech phoneme models to decode generic audio signal into both Top-1 acoustic sequence and acoustic lattice. In addition, VSM is introduced to extract statistics from the two types of decoding outputs. In the next chapter, we will study the performance of features generated from these two ways for an audio indexing and retrieval application.

Chapter 4

Audio Database Indexing and Retrieval

In this chapter, we will explore the usefulness of our proposed ASM trained HMM music models and new audio features (Chapter 3) for an audio indexing and retrieval application. Our application would accept queries by examples of the desired audio segment, and output the N most similar segments from the archived database. In our studies, the measurement of similarity is with respect to the generated features for indexing, and the database considered would be TV broadcast recordings such as Channel News Asia.

The following sections present details and experimental results of our application and approach to the audio indexing and retrieval task.

4.1 Proposed audio indexing and retrieval framework

Our proposed audio indexing and retrieval application will focus on advertisement queries. We have target advertisement queries as these data usually contain both speech and music components which allow us to compare the effectiveness of our proposed features and approach to operate on speech and music data simultaneously.

4.1.1 Indexing the audio archives

Fig. 4.1 illustrates our system's structure. The system consists of the feature extraction module followed by HMM decoding to generate higher level features to form the index features. The index features are created at every s-seconds from an audio segment of

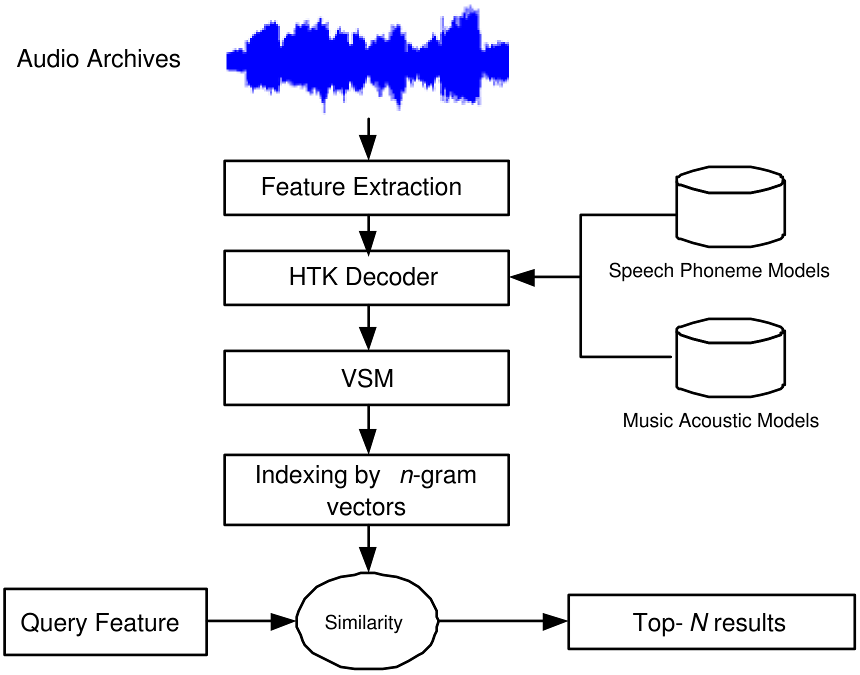


Figure 4.1: Overview of audio indexing and retrieval system.

t-seconds duration. Retrieval is simply a brute force comparison to find the closest N indexed features to the query segment's feature. The following paragraphs describe each module in greater details.

The feature used by the decoder is the MFCC features. The feature generation module extracts the MFCC features as well as their delta and double-delta features to form a feature vector of 39 coefficients. The duration of each frame's is 25 millisecond frames with a hop size of 10 millisecond. In our experiments, we have used the HTK [125] system to generate the features with the audio sampling frequency sampled at 11KHz.

The generated features are then fed into a HMM decoder consisting of both speech and music HMM models. The music HMM models are trained using the techniques described in Chapter 3. By having both speech and music models in the decoder simultaneously, the speech and music segments can be processed together without the need to discriminate between them. We note that there will be processing errors for audio segments that contain both speech and music, however, as the main goal is to represent the speech segment into a token sequence, and not to generate a transcription, the accuracy is not as crucial.

We consider two types of output from the decoder for subsequent processing: specifically, we consider the top-1 acoustic sequence or the acoustic lattice generated for each audio input. The top-1 sequence or the acoustic lattice representation cannot be directly used for indexing as it is of variable length/size, i.e., they cannot be compared to the query feature due to it being of different length. This motivates us to use VSM to capture the decoder's output in a fixed length vector format. In our implementation, we converted the decoder's output to n-gram statistics [130]. The n-gram for the audio segment is then used as the indexing term. The details of converting the top-1 and acoustic lattice n-gram statistics into vector representation are discussed in section 3.2.

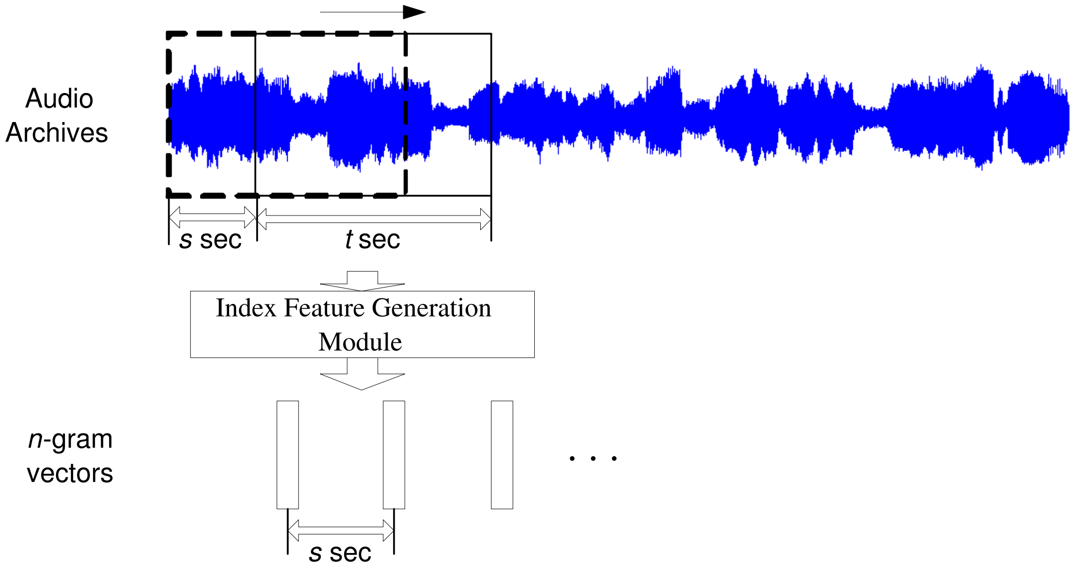


Figure 4.2: Indexing of audio archives. The n-gram vectors can be generated from either top-1 acoustic sequence or acoustic lattice.

Given the generated n-gram statistics, we created an index vector for every t-second window with s-second shift for the entire audio archive as illustrated in 4.2.

4.1.2 Audio segments retrieval

Given an audio query segment, the feature generation method as discussed in the previous section 4.1.1 is applied to generate the query features. To find similar audio segments to the query, our implementation simply compares the similarity of the query features against the entire archive's indexed vectors to retrieve the top-N closest match. Figure 4.3 illustrates the retrieval process.

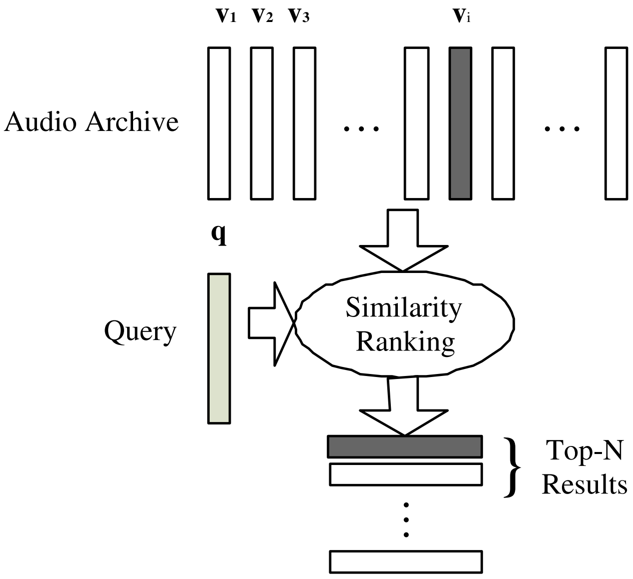


Figure 4.3: Audio segment retrieval process. A brute force approach is adopted to find the N-nearest match.

The given query segment should have a duration longer than t-seconds, where t-second is the duration window used to generate the feature vectors for the index. For the query segment, the feature extraction module extracts the n-gram feature vector $\mathbf{q}$. If the query duration is sufficiently long, more query vectors $\mathbf{q}$ can be generated. The query vectors $\mathbf{q}$ are compared with all the audio archive feature vectors $\mathbf{v}$ to find the closest N match using the Pearson correlation coefficient measure [131]. The Pearson correlation coefficient between vector $\mathbf{q}$ and $\mathbf{v_i}$, where i denotes the i^{th} index vector, is given by the following equation,

$$r_i = \frac{\sum_{k=1}^{n}(q_k - \overline{q})(v_{ik} - \overline{v_i})}{\sqrt{\sum_{k=1}^{n}(q_k - \overline{q})^2 \sum_{k=1}^{n}(v_{ik} - \overline{v_i})^2}}. \tag{4.1}$$

In this way, the segments of the archive are ranked in terms of similarity to the query. As our purpose is primarily to measure accuracy, we did not consider more efficient methods of retrieval such as inverted index [132] for this module of work.

4.2 Experiments on TRECVID Data

Experiments using the video database extracted from TRECVID 2003 and 2004 [133] were carried out. TRECVID data were recorded from the ABC and CNN network news during the year 1998. The audio information from 28 hour of video clips were extracted. We manually label all the commercials presented in these 28 hour database and select 100 unique commercials as our query commercials. In this database, there are 242 additional instance of the 100 unique commercials.

We conducted several experiments to determine the effect of query duration to retrieval performance. The duration of the queries (t) were chosen to be 3, 5 and 10 seconds with hop size s set to 0.5 second.

We also compared the performances of the system based on top-1 sequence and lattice based approaches to generate the index vectors. We studied both unigram ($n = 1$) and bigram ($n = 2$) vectors as index features. Since there are a total of 80 HMM models, the unigram vector has 80 dimensions. The bigram vector has 6561 (81x81) dimensions as we consider the starting point and ending point of the sequence as one extra token.

To evaluate the retrieved results, retrieved boundary of the segment within ±1 second of the actual boundary is considered as correct answer. In our experiments, we retain the top-10 results for evaluation as no queries in our database has more than 10 instants of the query.

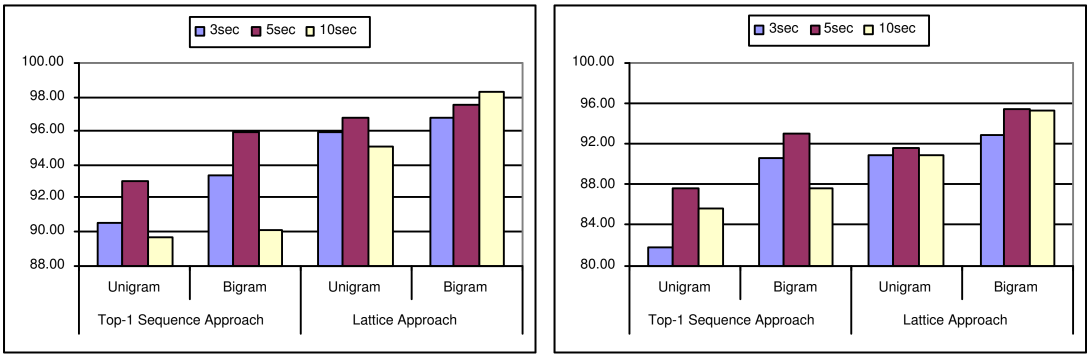
The recall and Mean Average Precision (MAP) are used to evaluate the system's performance and is defined by the following equation:

$$\text{Recall} = \frac{Number\ of\ relevant\ results\ retrieved}{Total\ number\ of\ actual\ relevant\ results}, \tag{4.2}$$

$$\text{MAP} = \frac{1}{M}\sum_{m=1}^{M} AP_m, \tag{4.3}$$

$$AP_m = \frac{1}{K}\sum_{i=1}^{K} \frac{i}{r_i}, \tag{4.4}$$

where M is the total number of queries, K is the number of relevant results retrieved for query m, and r_i is the rank of relevant result i.



(a) Recall (b) Mean Average Precision

Figure 4.4: Performance comparisons between top-1 sequence approach and lattice approach to generate index vectors.

The recall and MAP of our experimental results are illustrated in Figure 4.4(a) and 4.4(b) respectively. The results showed that the lattice approach outperforms the top-1 sequence approach. This can be explained by the fact that the lattice representation captures more statistics than just the top-1 decoder output and hence is more robust.

As both approaches utilize the temporal information of the signal, the window size t is an essential parameter to the system. Figure 4.4(a) shows the highest recall 98.35% is achieved by the 10 second window based on the lattice bigram vectors. Its corresponding MAP is also above 95% which shows that our proposed method is robust.

4.3 Conclusion

This chapter presents our experiments to examine the use of joint music/speech HMM models to decode audio data. Two types of decoding outputs are further processed by VSM to generate n-gram vectors to index the audio data. In the same way, n-gram vector can also be generated for audio query segment. In such cases, the query segment can be easily compared to the audio archives by measuring the similarities between the n-gram vectors in brute-force manner. Therefore, the audio segments which most similar to the query can be retrieved.

The experiments have been carried out using TRECVID data and the experimental results show the lattice approach outperforms the top-1 sequence approach. This is because the acoustic lattice reserves more information than top-1 sequence approach during the HMM decoding process. The high MAP scores also show that our ASM based models are robust and effective to transcribe broadcast audio data containing both music and speech.

Chapter 5

Conclusions and Future Works

5.1 Conclusions

This thesis aimed to explore unsupervised techniques for the audio content analysis and summarization task. We focus on the problem to extract the semantic structures from audio database so that we can organize the content into a hierarchical presentation such as a table for browsing and retrieval purpose.

Towards the above goal, we present a novel approach to operate on audio segment that contains both speech and music without first discriminating between them. The approach uses a decoder which contains both the English phonetic models and music ASM based HMM models together to convert the given audio segment into a top-1 phoneme/music model sequence or acoustic lattice. The decoded sequence is then further processed using the vector space modeling approach to generate the n-gram statistics. The new features have been applied to a query-by-example indexing and retrieval task to study its performance. The experiments show that the proposed features and framework are effective for the indexing and retrieval of broadcast audio program.

In the current work, we have only applied our new features/approach to query-by-example indexing and retrieval task. We believe that the presented approach can be used as a framework for the automatic audio content summarization task. In future work, we will explore efficient unsupervised repeating detection methods to analysis the audio content and to generate the content's structure. The following sections present the future works.

5.2 Future Works

Our future work will target technologies/applications to summarize the audio content of TV broadcast programs such as news channel [7] by generating a table of contents. From the observation of such programs, researchers have noticed that the non-news items such as music/songs [4], commercials and theme-music [21] occur repeatedly. Inspired by these observations, some existing work have proposed to detect the repeating audio segments by matching audio input stream to audio templates in brute-force manner [4, 21]. Such approaches however have severe limitations on large audio collections: 1) Audio templates have to be predefined or automatically generated; 2) The searching process has to start over for new coming templates; 3) Brute-force searching is time consuming. This motivates us is to examine more efficient unsupervised algorithms that can analyze large audio collections.

From the literature review, we know that repetition detection algorithms have been used to analyze symbolic sequences such as genomic data [88] and music MIDI scores [90, 98] as well as to examine vector-based sequences in the applications such as music structure analysis [69, 90, 98, 113, 115–117, 119]. However, the former category of works require to convert the input data into symbolic representation while the latter category usually operates on audio segments such as music pieces which have short duration, typically a few minutes. Thus, we believe that novel unsupervised techniques which can efficiently discover repetitions in long vector-based sequences may be able to help in our goal to find repeating patterns in large audio database.

Towards this research, we will first explore new algorithms and apply them to music structure analysis task as the existing works on music structure analysis have provided a benchmark to evaluate our proposed algorithms. After that, we will apply the new algorithms to general broadcast audio database. The plan for future works is as following:

(i) **Music structure analysis**

The first stage of our research work is to explore unsupervised techniques to automatically detect the repeating patterns in music. There are two possible approaches to achieve the goal:

(a) We can first convert music data into symbolic representation such as MIDI scores or another token-based sequences and then efficient algorithms such as suffix tree can be applied to discover the repeating patterns. However, the token-based sequences may suffer from noise and other unexpected factors in the data source so that the exact matching algorithms will fail in this case. Therefore, the inexact matching methods such as inexact-suffix tree [95] will be explored.

(b) Alternatively, we may directly manipulate on the vector-based feature sequences of music data. The existing methods have used frame-level feature vectors to construct self-similarity matrix [69, 113, 115–117, 119] and analyze music structure by detecting the stripes in the matrix. However, such approach requires $O(n^2)$ for both time and space consuming so that it cannot be applied to large audio database. Thus, this motivates us to explore a novel technique which can detect the repetitions based on feature vectors in linear time.

(ii) **General audio content summarization**
The second stage of this research work is to apply the above mentioned techniques to general broadcast audio programs using our proposed feature vectors. By first detecting the repeating segments, a rough content structure of the audio database can be inferred using prior knowledge. Different analysis methods can be applied to different audio portions. For examples, a LVCSR engine can be used to transcribe the speech portion in order to get a brief news description. In addition, predefined commercial templates can be used to match and label the commercial segments. Figure 5.1 presents the proposed application of our research work.

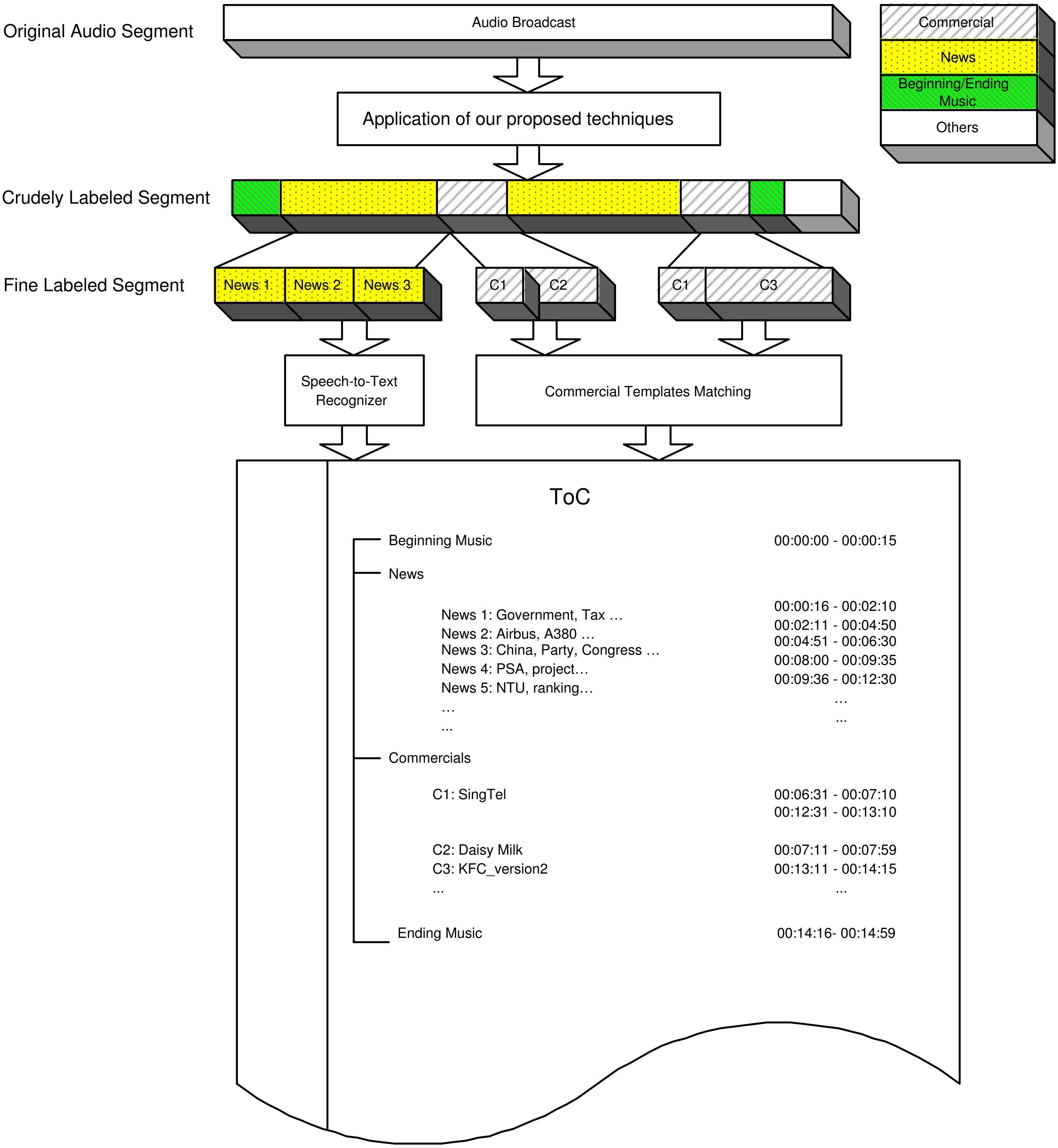


Figure 5.1: A overview of the proposed application of audio content summarization.

Publication

Published

(i) **Lei Wang**, Haizhou Li and Eng Siong Chng, “A vector-based approach to broadcast audio database indexing and retrieval”, in *proceedings of IEEE International Conference on Multimedia and Expo 2007 (ICME ’07)*, Beijing, China, July 2-5, 2007, pp. 512-515.

(ii) Tomi Kinnunen, Chin Wei Eugene Koh, **Lei Wang**, Haizhou Li, and Eng Siong Chng, “Temporal Discrete Cosine Transform: Towards Longer Term Temporal Features for Speaker Verification”, in *Proceedings of 5th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, COLIPS, Q. Huo et al.(Eds.), pp. 547-558, 13-16 December 2006, Singapore.

(iii) K. A. Lee, H. Sun, R. Tong, B. Ma, M. Dong, C. You, D. Zhu, C. W. E. Koh, **L. Wang**, T. Kinnunen, E. S. Chng, and H. Li, “The IIR Submission to CSLP 2006 Speaker Recognition Evaluation”, in *Proceedings of 5th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, LNAI 4274, Q. Huo et al.(Eds.), pp.494-505, 13-16 December 2006, Singapore.

References

[1] L. Wang, H. Li, and E. S. Chng, "A vector-based approach to broadcast audio database indexing and retrieval," in *Proc. ICME '07*, Beijing, China, July 2007, pp. 512–515.

[2] N. Dimitrova, H.-J. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, "Applications of video-content analysis and retrieval," *IEEE Multimedia*, vol. 9, no. 3, pp. 42–55, July-Sept. 2002.

[3] S. Furui, T. Kikuchi, Y. Shinnaka, and C. Hori, "Speech-to-text and speech-to-speech summarization of spontaneous speech," *IEEE Trans. Speech and Audio Processing*, vol. 12, no. 4, pp. 401–408, July 2004.

[4] C. Herley, "ARGOS: Automatically extracting repeating objects from multimedia streams," *IEEE Trans. Multimedia*, vol. 8, no. 1, pp. 115–129, Feb. 2006.

[5] W. Chai, "Semantic segmentation and summarization of music: methods based on tonality and recurrent structure," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 124–132, Mar. 2006.

[6] J. Makhoul, F. Kubala, T. Leek, D. Liu, L. Nguyen, R. Schwartz, and A. Srivastava, "Speech and language technologies for audio indexing and retrieval," *Proc. IEEE*, vol. 88, no. 8, pp. 1338–1353, Aug. 2000.

[7] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation campaigns and TRECVid," in *Proc. ACM MIR '06*, Santa Barbara, California, USA, 2006, pp. 321–330.

[8] A. Park, T. J. Hazen, and J. R. Glass, "Automatic processing of audio lectures for information retrieval: Vocabulary selection and language modeling," in *Proc. ICASSP '05*, Philadelphia, USA, Mar. 2005, vol. 1, pp. 497–500.

[9] J. G. Fiscus, J. Ajot, M. Michel, and J. S. Garofolo, "The Rich Transcription 2006 Spring Meeting Recognition Evaluation," in *Machine Learning for Multimodal Interaction*, pp. 309–322. May 2006.

[10] Z.-Y. Zhou, P. Yu, C. Chelba, and F. Seide, "Towards spoken-document retrieval for the internet: lattice indexing for large-scale web-search architectures," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the ACL*, New York, June 2006, pp. 415–422.

[11] D. P. W. Ellis and K. Lee, "Minimal-impact audio-based personal archives," in *Proc. ACM workshop on Continuous Archival and Retrieval of Personal Experiences '04*, New York, NY, USA, 2004, pp. 39–47.

[12] *http://www.google.com.*

[13] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 2, no. 1, pp. 1–19, 2006.

[14] Y. Wang, Z. Liu, and J.-C. Huang, "Multimedia content analysis -using both audio and visual clues," *IEEE Signal Processing Magazine*, vol. 17, no. 6, pp. 12–36, Nov. 2000.

[15] E. Wold, T. Blum, D. Keislar, and J. Wheaten, "Content-based classification, search, and retrieval of audio," *IEEE Multimedia*, vol. 3, no. 3, pp. 27–36, July-Sept. 1996.

[16] C. Hori and S. Furui, "A new approach to automatic speech summarization," *IEEE Trans. Multimedia*, vol. 5, no. 3, pp. 368–378, Sept. 2003.

[17] C. Colombo, A. Del Bimbo, and P. Pala, "Semantics in visual information retrieval," *IEEE Multimedia*, vol. 6, no. 3, pp. 38–53, July-Sept. 1999.

[18] B. T. Truong and S. Venkatesh, "Video abstraction: A systematic review and classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 3, no. 1, 2007.

[19] M. G. Brown, J. T. Foote, G. J. F. Jones, K. Sparck Jones, and S. J. Young, "Automatic content-based retrieval of broadcast news," in *Proc. ACM Multimedia '95*, San Francisco, California, United States, 1995, pp. 35–43.

[20] I. Mani and M. T. Maybury, *Advances in Automatic Text Summarization*, Cambridge, MA:MIT Press, 1999.

[21] S. E. Johnson and P. C. Woodland, "A method for direct audio search with applications to indexing and retrieval," in *Proc. ICASSP '00*, Istanbul, Turkey, June 2000, pp. 1427–1430.

[22] H. Li, B. Ma, and C.-H. Lee, "A vector space modeling approach to spoken language identification," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 271–284, Jan. 2007.

[23] B.-H. Juang and S. Furui, "Automatic recognition and understanding of spoken language - a first step toward natural human-machine communication," *Proc. IEEE*, vol. 88, no. 8, pp. 1142–1165, Aug. 2000.

[24] K. Zechner, "Automatic summarization of open-domain multiparty dialogues in diverse genres," *Computational Linguistics*, vol. 28, no. 4, pp. 447–485, Dec. 2002.

[25] X. Zhu and G. Penn, "Summarization of spontaneous conversations," in *Proc. Eurospeech '07*, Antwerp, Belgium, Aug. 2007, pp. 1531–1534.

[26] N. Morgan, D. Baron, J. Edwards, D. Ellis, D. Gelbart, A. Janin, T. Pfau, E. Shriberg, and A. Stolcke, "The meeting project at ICSI," in *Proc. HLT '01*, San Diego, 2001, pp. 1–7.

[27] Y. Fujii, N. Kitaoka, and S. Nakagawa, "Automatic extraction of cue phrases for important sentences in lecture speech and automatic lecture speech summarizaiton," in *Proc. Eurospeech '07*, Antwerp, Belgium, Aug. 2007, pp. 2801–2804.

[28] T. Shinozaki, C. Hori, and S. Furui, "Towards automatic transcription of spontaneous presentations," in *Proc. Eurospeech '01*, Aalborg, Denmark, Sept. 2001, pp. 491–494.

[29] B. Pardo, "Finding structure in audio for music information retrieval," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 126–132, May 2006.

[30] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith, "Query by humming: Musical information retrieval in an audio database," in *Proc. ACM MM '95*, San Francisco, California, USA, Nov. 1995, pp. 231–236.

[31] L. Lu, H. You, and H.-J. Zhang, "A new approach to query by humming in music retrieval," in *Proc. ICME '01*, Tokyo, Japan, Aug. 2001, pp. 776–779.

[32] T. De Mulder, J.-P. Martens, S. Pauws, F. Vignoli, M. Lesaffre, M. Leman, B. De Baets, and H. De Meyer, "Factors affecting music retrieval in query-by-melody," *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 728–739, Aug. 2006.

[33] R. B. Dannenberg and N. Hu, "Pattern discovery techniques for music audio," *Journal of New Music Research*, vol. 32, no. 2, pp. 153–163, June 2003.

[34] J. S. Garofolo, E. M. Voorhees, V. M. Stanford, and K. S. Jones, "TREC-6 1997 Spoken Document Retrieval Track Overview and Results," in *Proc. 1997 TREC-6 Conference*, Gaithersburg, MD, Nov. 1997, pp. 83–91.

[35] D. Abberley, S. Renals, and G. Cook, "Retrieval of broadcast news documents with the THISL system," in *Proc. ICASSP '98*, Seattle, WA, May 1998, pp. 3781–3784.

[36] J.-M. V. Thong, P. J. Moreno, B. Logan, B. Fidler, K. Maffey, and M. Moores, "SPEECHBOT: An experimental speech-based search engine for multimedia content in the web," Tech. Rep. CRL 2001/06, Cambridge Research Laboratory, July 2001.

[37] I. Szoke, P. Schwarz, and P. Matejka, "Comparion of keyword spotting approaches for informal continuous speech," in *Proc. Eurospeech '05*, Lisbon, Portugal, Sept. 2005, pp. 633–636.

[38] A. Velivelli, C. Zhai, and T. S. Huang, "Audio segment retrieval using a short duration example query," in *Proc. ICME '04*, Taipei, Taiwan, June 2004, pp. 1603–1606.

[39] B. Logan, P. Moreno, and O. Deshmukh, "Word and sub-word indexing approaches for reducing the effects of OOV queries on spoken audio," in *Proc. Human Language Technology Conference '02*, San Diego, CA, 2002.

[40] P. Yu, K. Chen, C. Ma, and F. Seide, "Vocabulary-independent indexing of spontaneous speech," *IEEE Trans. Speech and Audio Processing*, vol. 13, no. 5, pp. 635–643, Sept. 2005.

[41] M. Saraclar and R. Sproat, "Lattice-based search for spoken utterance retrieval," in *Proc. HLT/NAACL 2004*, Boston, MA, USA, May 2004.

[42] J. Herre, E. Allamanche, and O. Hellmuth, "Robust matching of audio signals using spectral flatness features," in *Proc. IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics '01*, New Paltz, New York, Oct. 2001, pp. 127–130.

[43] Z. Liu and Q. Huang, "Content-based indexing and retrieval-by-example in audio," in *Proc. ICME '00*, New York City, NY, USA, July-Aug. 2000, pp. 877–880.

[44] R. Valenza, T. Robinson, M. Hickey, and R. Tucker, "Summarisation of spoken audio through information extraction," in *Proc. ESCA workshop: Accessing information in spoken audio*, 1999, pp. 111–116.

[45] R. Jin and A. G. Hauptmann, "Title generation for spoken broadcast news using a training corpus," in *Proc. ICSLP '00*, Beijing, China, Oct. 2000.

[46] S. R. Maskey and J. Hirschherg, "Automatic summarization of broadcast news using structural features," in *Proc. Eurospeech '03*, Geneva, Switzerland, Sept. 2003, pp. 1173–1176.

[47] G. Murray and S. Renals, "Towards online speech summarization," in *Proc. Eurospeech '07*, Antwerp, Belgium, Aug. 2007, pp. 2785–2788.

[48] K. Koumpis and S. Renals, "Automatic summarization of voicemail messages using lexical and prosodic features," *ACM Trans. Speech Lang. Process.*, vol. 2, no. 1, pp. 1–24, Feb. 2005.

[49] S. Furui, “Recent progress in spontaneous speech recognition and understanding,” in *Proc. IEEE Workshop on Multimedia Signal Processing ’02*, Dec. 2002, pp. 253–258.

[50] G. Williams and S. Renals, “Confidence measures for hybrid HMM/ANN speech recognition,” in *Proc. Eurospeech ’97*, Rhodes, Greece, Sept. 1997, pp. 1955–1958.

[51] C.-L. Huang, C.-H. Hsieh, and C.-H. Wu, “Spoken document summarization using acoustic, prosodic and semantic information,” in *Proc. ICME ’05*, July 2005.

[52] K. Ohtake, K. Yamamoto, Y. Toma, S. Sado, S. Masuyama, and S. Nakagawa, “Newscast speech summarization via sentence shortening based on prosodic features,” in *Proc. ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003, pp. 167–170.

[53] A. Inoue, T. Mikami, and Y. Yamashita, “Improvement of speech summarization using prosodic information,” in *Proc. Speech Prosody*, Japan, 2004.

[54] J. Hirschberg, “Communication and prosody: Functional aspects of prosody,” *Speech Communication*, vol. 36, no. 1-2, pp. 31–43, Jan. 2002.

[55] Y. Rui, A. Gupta, and A. Acero, “Automatically extracting highlights for TV baseball programs,” in *Proc. ACM MM ’00*, 2000, pp. 105–115.

[56] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T. S. Huang, “Audio events detection based highlights extraction from baseball golf and soccer games in a unified network,” in *Proc. ICASSP ’03*, Hong Kong, China, Apr. 2003, vol. 5, pp. 632–635.

[57] B. Zhang, W. Dou, and L. Chen, “Ball hit detection in table tennis games based on audio analysis,” in *Proc. ICPR ’06*, 2006, vol. 3, pp. 220–223.

[58] S. Nepal, U. Srinivasan, and G. Reynolds, “Automatic detection of ’Goal’ segments in basketball videos,” in *Proc. ACM MM ’01*, Ottawa, Canada, 2001, pp. 261–269.

[59] C. M. Taskiran, Z. Pizlo, A. Amir, D. Ponceleon, and E. J. Delp, "Automated video program summarization using speech transcripts," *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 775–791, Aug. 2006.

[60] Y. Li, S.-H. Lee, C.-H. Yeh, and C.-C. J. Kuo, "Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 79–89, Mar. 2006.

[61] M. Xu, L.-T. Chia, and J. Jin, "Affective content analysis in comedy and horror videos by audio emotional event detection," in *Proc. ICME '05*, Amsterdam, Netherlands, July 2005.

[62] H.-W. Chen, J.-H. Kuo, W.-T. Chu, and J.-L. Wu, "Action movies segmentation and summarization based on tempo analysis," in *Proc. ACM SIGMM international workshop on Multimedia information retrieval '04*, New York, NY, USA, 2004, pp. 251–258.

[63] S. Moncrieff, C. Dorai, and S. Venkatesh, "Affect computing in film through sound energy dynamics," in *Proc. ACM MM '01*, Ottawa, Canada, 2001, pp. 525–527.

[64] Z. Rasheed and M. Shah, "Movie genre classification by exploiting audio-visual features of previews," in *Proc. ICPR '02*, Aug. 2002, vol. 2, pp. 1086–1089.

[65] W.-H. Cheng, W.-T. Chu, and J.-L. Wu, "Semantic context detection based on hierarchical audio models," in *Proc. ACM MIR '03*, Berkeley, California, 2003, pp. 109–115.

[66] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 7, pp. 504–516, Oct. 2002.

[67] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[68] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: A guide to theory, algorithm and system development*, Prentice Hall PTR, 2001.

[69] M. A. Bartsch and G. H. Wakefield, “To catch a chorus: Using chroma-based representations for audio thumbnailing,” in *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics ’01*, Oct. 2001, pp. 15–18.

[70] S. Kiranyaz, A. F. Qureshi, and M. Gabbouj, “A generic audio classification and segmentation approach for multimedia indexing and retrieval,” *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 3, pp. 1062–1081, May 2006.

[71] R. Cai, L. Lu, A. Hanjalic, H.-J. Zhang, and L.-H. Cai, “A flexible framework for key audio effects detection and auditory context inference,” *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 3, pp. 1026–1039, May 2006.

[72] T. Zhang and C.-C. J. Kuo, “Audio content analysis for online audiovisual data segmentation and classification,” *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 4, pp. 441–457, May 2001.

[73] E. Scheirer and M. Slanry, “Construction and evaluation of a robust multifeature speech/music discriminator,” in *Proc. ICASSP ’97*, Munich, Germany, Apr. 1997, pp. 1331–1334.

[74] G. Willianis and D. Ellis, “Speech/music discrimination based on posterior probability features,” in *Proc. Eurospeech ’99*, Budapest, Hungary, Sept. 1999, pp. 687–690.

[75] H. Harb and L. Chen, “Robust speech music discrimination using spectrum’s first order statistics and neural networks,” in *Proc. International Symposium on Signal Processing and Its Applications ’03*, July 2003, vol. 2, pp. 125–128.

[76] K. El-Maleh, M. Klein, G. Petrucci, and P. Kabal, “Speech/music discrimination for multimedia applications,” in *Proc. ICASSP ’00*, Istanbul, Turkey, June 2000, pp. 2445–2448.

[77] D. Kimber and L. Wilcox, "Acoustic segmentation for audio browsers," in *Proc. Interface Conference*, Sydney, Australia, July 1996.

[78] A. J. Eronen, V. T. Peltonen, J. T. Tuomi, A. P. Klapuri, S. Fagerlund, T. Sorsa, G. Lorho, and J. Huopaniemi, "Audio-based context recognition," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 321–329, Jan. 2006.

[79] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, July 2002.

[80] *http://en.wikipedia.org/wiki/List_of_music_genres.*

[81] K. West and S. Cox, "Features and classifiers for the automatic classification of musical audio signals," in *Proc. ISMIR '04*, Barcelona, Spain, Oct. 2004.

[82] T. Li and M. Ogihara, "Music genre classification with taxonomy," in *Proc. ICASSP '05*, Philadelphia, USA, Mar. 2005, vol. 5, pp. 197–200.

[83] D.-N. Jiang, L. Lu, H.-J. Zhang, J.-H. Tao, and L.-H. Cai, "Music type classification by spectral contrast feature," in *Proc. ICME '02*, Aug. 2002, vol. 1, pp. 113–116.

[84] C.-H. Lee, J.-L. Shih, K.-M. Yu, and J.-M. Su, "Automatic music genre classification using modulation spectral contrast feature," in *Proc. ICME '07*, Beijing, China, July 2007, pp. 204–207.

[85] N. C. Maddage, C. Xu, M. S. Kankanhalli, and X. Shao, "Content-based music structure analysis with applications to music semantics understanding," in *Proc. ACM MM '04*, New York, NY, USA, 2004, pp. 112–119.

[86] Y. Zhu and M. S. Kankanhalli, "Precise pitch profile feature extraction from musical audio for key detection," *IEEE Trans. Multimedia*, vol. 8, no. 3, pp. 575–584, June 2006.

[87] K. Chen, S. Gao, Y. Zhu, and Q. Sun, "Music genres classification using text categorization method," in *Proc. IEEE Workshop on Multimedia Signal Processing '06*, Oct. 2006, pp. 221–224.

[88] T. L. Bailey and C. Elkan, "Unsupervised learning of multiple motifs in biopolymers using expectation maximization," *Machine Learning*, vol. 21, no. 1-2, pp. 51–80, Oct. 1995.

[89] A. Brazma, I. Jonassen, I. Eidhammer, and D. Gilbert, "Approaches to the automatic discovery of patterns in biosequences," *Journal of Computational Biology*, vol. 5, no. 2, pp. 279–305, 1998.

[90] J.-L. Hsu, C.-C. Liu, and A. L. P. Chen, "Discovering nontrivial repeating patterns in music data," *IEEE Trans. Multimedia*, vol. 3, no. 3, pp. 311–325, Sept. 2001.

[91] M. M. Gaber, A. Z., and S. Krishnaswamy, "Mining data streams: a review," *ACM SIGMOD Record*, vol. 34, no. 2, pp. 18–26, 2005.

[92] E. M. McCreight, "A space-economical suffix tree construction algorithm," *Journal of the ACM*, vol. 23, no. 2, pp. 262–272, 1976.

[93] E. Ukkonen, "On-line construction of suffix trees," *Algorithmica*, vol. 14, no. 3, pp. 249–260, Sept. 1995.

[94] D. He, "Using suffix tree to discover complex repetitive patterns in DNA sequences," in *Proc. of EMBS '06*, Aug. 2006, pp. 3474–3477.

[95] A. Chattaraja and L. Paridab, "An inexact-suffix-tree-based algorithm for detecting extensible patterns," *Theoretical Computer Science*, vol. 335, no. 1, pp. 3–14, May 2005.

[96] M. Vingron and P. Argos, "Motif recognition and alignment for many sequences by comparison of dot-matrices," *Journal of Molecular Biology*, vol. 218, no. 1, pp. 33–43, Jan. 1991.

[97] M.-F. Sagot, A. Viari, and H. Soldano, "Multiple sequence comparison: A peptide matching approach," in *Proc. 6th Annual Symposium on Combinatorial Pattern Matching '95*, Espoo, Finland, July 1995.

[98] I. Karydis, A. Nanopoulos, and Y. Manolopoulos, "Finding maximum-length repeating patterns in music databases," *Multimedia Tools and Applications*, vol. 32, no. 1, pp. 49–71, Jan. 2007.

[99] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh, "Indexing multidimensional time-series," *The VLDB Journal*, vol. 15, no. 1, pp. 1–20, Jan. 2006.

[100] Y. Tanaka, K. Iwamoto, and K. Uehara, "Discovery of time-series motif from multi-dimensional data based on MDL principle," *Machine Learning*, vol. 58, no. 2-3, pp. 269–300, Mar. 2005.

[101] J. W. Picone, "Signal modeling techniques in speech recognition," *Proc. IEEE*, vol. 81, no. 9, pp. 1215–1247, Sept. 1993.

[102] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *Proc. ACM SIGMOD workshop on Research issues in data mining and knowledge discovery '03*, San Diego, California, 2003, pp. 2–11.

[103] L. Rabiner and B.-H. Juang, *Fundamental of Speech Recognition*, Prentice-Hall International, Inc., 1993.

[104] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons Inc., 2nd edition, 2001.

[105] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, Sept. 1999.

[106] C. T. Zahn, "Graph-theoretical methods for detecting and describing gestalt clusters," *IEEE Transactions on Computers*, vol. C-20, no. 1, pp. 68–86, Jan. 1971.

[107] C. J. V. Rijsbergen, *Information Retrieval*, London: Butterworths, 2nd edition, 1979.

[108] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[109] B. Logan and S. Chu, “Music summarization using key phrases,” in *Proc. ICASSP ’00*, Istanbul, Turkey, June 2000, vol. 2, pp. 749–752.

[110] J.-J. Aucouturier and M. Sandler, “Segmentation of musical signals using hidden Markov models,” in *Proc. the Audio Engineering Society 110th Convention*, Amsterdam, Netherlands, May 2001.

[111] S. Abdallah, K. Noland, M. Sandler, M. Casey, and C. Rhodes, “Theory and evaluation of a bayesian music structure extractor,” in *Proc. the 6th ISMIR Conference*, London, UK, 2005, pp. 420–425.

[112] L. Xie, *Unsupervised pattern discovery for multimedia sequences*, Ph.D. thesis, Columbia University, 2005.

[113] J. Foote, “Visualizing music and audio using self-similarity,” in *Proc. ACM MM ’99*, 1999, pp. 77–80.

[114] M. A. Bartsch and G. H. Wakefield, “Audio thumbnailing of popular music using chroma-based representations,” *IEEE Trans. Multimedia*, vol. 7, no. 1, pp. 96–104, Feb. 2005.

[115] M. Goto, “A chorus section detection method for musical audio signals and its application to a music listening station,” *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1783–1794, Sept. 2006.

[116] T. Zhang and R. Samadani, “Automatic generation of music thumbnails,” in *Proc. ICME ’07*, 2007, pp. 228–231.

[117] L. Lu, M. Wang, and H.-J. Zhang, “Repeating pattern discovery and structure analysis from acoustic music data,” in *Proc. ACM MIR ’04*, New York, NY, USA, 2004, pp. 275–282.

[118] J. Foote, “Automatic audio segmentation using a measure of audio novelty,” in *Proc. ICME ’00*, July-Aug. 2000, pp. 452–455.

[119] Y. Shiu, H. Jeong, and C.-C. J. Kuo, "Similarity matrix processing for music structure analysis," in *Proc. ACM workshop on Audio and music computing multimedia '06*, Santa Barbara, California, USA, 2006, pp. 69–76.

[120] C.-H. Lee, F. K. Soong, and B.-H. Juang, "A segment model based approach to speech recognition," in *Proc. ICASSP '88*, New York, NY, 1988, pp. 501–504.

[121] C.-H. Lee and L. R. Rabiner, "A frame-synchronous network search algorithm for connected word recognition," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 37, no. 11, pp. 1649–1658, Nov. 1989.

[122] R. Singh, B. Raj, and R. M. Stern, "Automatic generation of subword units for speech recognition systems," *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 2, pp. 89–99, Feb. 2002.

[123] J. Garofalo et al., *CSR-I (WSJ0) Complete*, Linguistic Data Consortium, Philadelphia, 1993.

[124] Linguistic Data Consortium, Philadelphia, *CSR-II (WSJ1) Complete*, 1994.

[125] S. Young et al., *The HTK Book (for HTK Version 3.2.1)*, Cambridege University Engineering Department, 2002.

[126] G. Salton, *The SMART Retrieval System*, Prentice-Hall, Inc., 1971.

[127] N. C. Maddage, H. Li, and M. S. Kankanhalli, "Music structure based vector space retrieval," in *Proc. ACM SIGIR '06*, Seattle, Washington, USA, Aug. 2006, pp. 67–74.

[128] D. A. James and S. J. Young, "A fast lattice-based approach to vocabulary independent wordspotting," in *Proc. ICASSP '94*, Adelaide, Australia, Apr. 1994, vol. 1, pp. 377–380.

[129] J.L. Gauvain, A. Messaoudi, and H. Schwenk, "Language recognition using phone lattices," in *Proc. ICSLP '04*, Jeju, South Korea, Oct. 2004.

[130] A. Stolcke, “SRILM - an extensible language modeling toolkit,” in *Proc. ICSLP '02*, Denver, USA, Sept. 2002.

[131] J. Pevsner, *Bioinformatics and Functional Genomics*, John Wiley & Sons, Inc., 2003.

[132] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley Longman, 1999.

[133] *http://www-nlpir.nist.gov/projects/trecvid.*