SONG RETRIEVAL SYSTEM USING HIDDEN MARKOV MODELS

AKSHAY CHANDRASHEKARAN
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
akshayc@cmu.edu

ANOOP RAMAKRISHNA
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
anoopr@andrew.cmu.edu

ABHISHEK JAIN
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
ajain2@andrew.cmu.edu

GE YANG
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
younger@cmu.edu

NIDHI KOHLI
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
nkohli@andrew.cmu.edu

R SWAMINATHAN
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
rswamin1@andrew.cmu.edu

TEJAS CHOPRA
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
tchopra@andrew.cmu.edu

YITING XIE
Department of ECE
Carnegie Mellon University
Pittsburgh, PA 15213
yitingx@andrew.cmu.edu

Abstract

This project proposes a novel method to retrieve the song corresponding to a recording of a snippet of the same. The method used converts the song into a string of music phonemes, which are MFCC Feature Vectors using Hidden Markov Models. The sequence of these phones are then stored. A similar procedure is performed upon the snippets. The snippet sequence and song sequence are then compared using a sliding window method with the Levenshtein edit distance as the score metric. This effectively reduces the problem to a string search problem. The usage of HMMs adds robustness against noise. The application for these projects include user based content search, song retrieval systems like Shazam® etc.

1 Introduction

Music identification involves determining the identity of the song by matching a partial recording given by a user. These systems can be used by content distribution networks like Google and Youtube to identify copyrighted audio within their systems. The task of identification is difficult due to lack of clarity of the snippet as it may have many components of noise in it. Secondly, it also requires that the system be efficient in terms of memory and time which is usually a difficult task given the

sheer size and number of songs in a typical database.

The previous approaches to retrieving songs and matching music were based on a hash-search algorithm which required an exact match between the snippet and the song. Hence, in the presence of noise, the system would yield incorrect results.

Another method for song retrieval involves decoding the Mel-fequency Cepstral Component(MFCC) [1] features over the audio stream into a sequence of audio events wherein the matching is driven by Hidden Markov Models(HMM)[2]. The system, however looks for atomic sound sequences of a particular length so that the complexity can be reduced.[3] A more recent paper by E. Weinstein et al [4] use HMMs for song representation and then reduce the space complexity by using WFSTs.

The method used by us follows steps similar to [4]. However, it diverges from the procedure in the final stages for Database generation and song retrieval. As a baseline, initially the task was performed using a general spatial correlation, correlation of the Fourier transforms of the songs and the snippets and the Short-time Fourier transforms of the same. Though the methods are simple to understand and implement, they are highly inefficient in terms of physical space requirement and time complexity. Hidden Markov Models, whose observations are Gaussians or mixture of Gaussians, are an effective tool to statistically model the dynamic causal data and hence, it is used in the proposed song retrieval system.

2 Music identification

Unlike speech, which has a set of well defined phonemes, music is essentially a non-stationary stochastic process and has time and space varying joint PDFs. This is because music, being polyphonic, is composed of several random variables like amplitude, frequency, phase, etc. Non stationary processes have varying mean and variances, so we break the songs into several quasi-stationary segments. These songs are to be represented by a set of music phonemes.

2.1 Acoustic Modeling

The acoustic modeling involves learning a joint set of music phonemes and finding a set of phonemes that best represent each song. First, the Mel-Frequency Cepstral Coefficient(MFCCs) Features for each song is computed. We use 100 ms windows with a hop size of 25 ms over the feature stream. The first 12 coefficients, their energy, and their first and second derivatives are found, thus resulting in a 39 dimensional feature vector.

2.2 Model Initialization

2.2.1 Segmentation

To find the initial set of segmentations, the audio is segmented by sliding a window over the feature stream and fitting a diagonal covariance Gaussian to each window. Then, the KL Divergence is calculated between adjacent windows. The KL Divergence can be calculated as:

$$\mathrm{KL}_{sym}(G1,G2) = \mathrm{tr}\left(\Sigma_2\Sigma_1^{-1}\right) + \mathrm{tr}\left(\Sigma_1\Sigma_2^{-1}\right) + (\mu_2-\mu_1)^{\top}\left(\Sigma_1^{-1}+\Sigma_2^{-1}\right)(\mu_2-\mu_1) - 2m$$

If the value of KL divergence between adjacent windows is above some empirically determined threshold, the song is segmented at those points. This is done over all the songs to generate a set of segments as shown in figure 1.

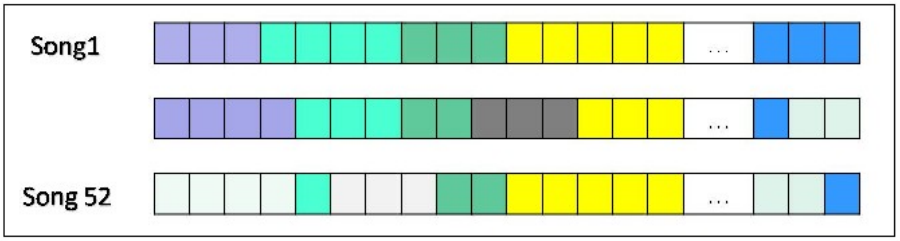

Fig.1: Segmentation in songs

2.2.2 Clustering

The means of all the segments are then taken and a K means clustering is performed on them. The number of clusters is predetermined, and is taken as a power of 2. Corresponding number of HMMs are trained using these segments. These HMMs are not ergodic. Each state of the HMM has only self transition or can transit to only the next adjacent state. The observations of each state are independent Gaussians with diagonal covariance.

2.2.3 Phoneme Generation

Next, all the segments are decoded using the HMMs and are assigned to that HMM for which they give the maximum log likelihood, as shown in figure 2. HMMs are then trained using the reassigned segments for clusters

Thus, this method results in the generation of quasi-stationary music phonemes which can be used to represent this database.

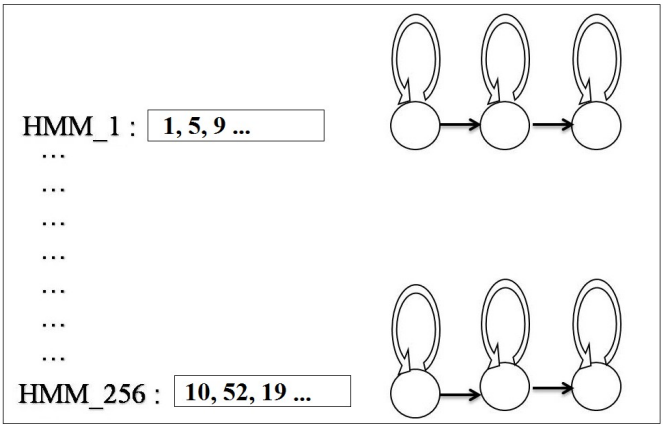

Fig.2: Creation of HMMs after clustering

2.2.4 State Flow Model

At this point, we have all the segments assigned to different HMMs. Our next step is to combine these different HMMs into a single large HMM called the State Flow HMM which has parallel branches, with each branch representing the HMM of one cluster and with their starts and ends being tied together via non-emitting states, as shown in figure 3.

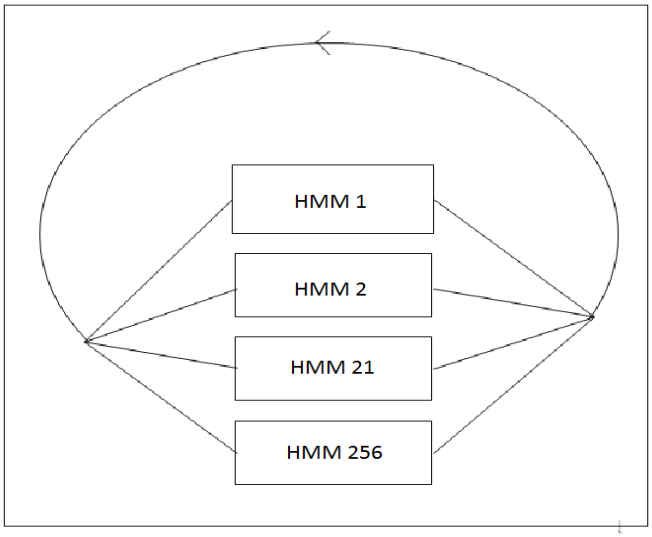

Fig.3: State Flow Model HMM

2.2.5 Song Representation

After generating the state flow model, the MFCC feature matrix of each song is passed through the state flow model and its path i.e. the HMMs it goes through from the start to the end is found. Thus, an uncompressed song in WAV file format, which is usually an array nearly a million bytes long, has been represented by a string of symbols.

2.2.6. Song Retrieval

At this point we have managed to represent each song as a string of alpha-numeric characters, and we can save this to memory, and this is in fact a one time job, to develop the data base. Now when we receive a snippet from a user, we simply run the snippet through the State Flow HMM and retrieve a transcription for that as well. We now simply calculate an edit distance between the snippet transcription and that of all the songs in the database, the song with the lowest edit distance is classified as the best match.

3 Results

	Spatial Correlation	STFT Correlation	HMM based Retrieval
Noiseless Snippets	0.72	0.83	0.92
Noisy Snippets	0.51	0.55	0.60

Table 1: Comparison of accuracy of Various methods for noiseless and noisy signals

The baseline case involved applying correlation between the snippet and the songs. Although the results were not very bad, it was a computationally expensive method and consumed a lot of memory space and time. This was further refined to an extent by the short-time Fourier transform method. Slightly better results were obtained using the STFT based correlation. However, the complexity was very large. Our algorithm showed an improvement over both the methods in terms of efficiency, speed, memory space occupied and the accuracy. The results have been statistically determined and tabulated as shown in Table 1.

4 Conclusion

This method was able to successfully match noiseless snippets to songs with extremely high accuracy. This approach did not rely on an exact match between the snippet and the song. We have music phone sequences of variable length depending on the clustering and segmentation exercise we performed for each song. As a part of the project we have been successful at effectively transcribing the songs into music units. Usage of HMMs adds robustness to the matching procedure. The string search method incorporated also takes into consideration changes in the expected value of the snippet string due to noise. However, further improvement can be achieved in case of noiseless signals. Though this method's training takes up a longer time, this is offset by the speed of online song retrieval.

5 Future Work

Though this algorithm gives a very high accuracy for noiseless snippets, its performance in the presence of noise is however, lacking. Though the algorithm has a better temporal efficiency compared to other methods, it can still be improved further. A further possible step is to implement other more efficient and accurate string search algorithms.

6 Acknowledgment

We would like to sincerely thank Professor Bhiksha Raj, LTI, Carnegie Mellon University for his critical suggestions and help. His constant motivation was the driving factor which enabled us to complete the project.

7 References

[1]. E. Weinstein and P. Moreno, Music identification with weighted finite-state transducers, International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Honolulu, Hawaii, 2007.

[2] A tutorial on hidden markov models and selected applications in speech recognition (1989), Lawrence R. Rabiner, Proceedings of the IEEE

[3] E. Batlle, J. Masip, and E. Guaus, “Automatic song identification in noisy broadcast audio,” in IASTED International Conference on Signal and Image Processing, Kauai, Hawaii, 2002.

[4] Mehryar Mohri, Pedro Moreno, and Eugene Weinstein. Efficient and Robust Music Identification with Weighted Finite-State Transducers, IEEE Transactions on Audio, Speech, and Language Processing, 2009

[5] "Binary codes capable of correcting deletions, insertions, and reversals". Soviet Physics Doklady, Vladmir Levenshtein, 1966