Global Journal of Computational Intelligence Research.
ISSN 2249-0000 Volume 1, Number 1 (2011), pp. 57-70

http://www.ripublication.com/gjcir.htm

Feature Extraction and Classification of Ragas in South Indian Classical Music

[1]Surendra Shetty and [2]K.K. Achary

[1]Department of Computer Applications,
NMAM Institute of Technology, Nitte, Karnataka, India
E-mail: hsshetty4u@yahoo.com
[2]Dept of Statistics, Mangalore University,
Mangalagangothri, Karnataka, India
E-mal: kka@mangaloreunivarsity.ac.in

Abstract

Audio Data Mining deals with mining of data in the form of signals such as speech, music, radio etc. Audio data mining is a technique which includes converting raw data in the form of audio signals into a meaningful text/numerical data using digital signal processing techniques and then extracting useful patterns from it for knowledge discovery. The type of data considered for audio mining in our paper are *ragas* in South Indian Classical Music System. The combination of several notes woven into a composition in a way which is pleasing to the ear is called a *Raga* or Melody in music. In this paper, the automatic classification of musical signals into different musical *ragas* is explored. More specifically, the feature set for representing pitch content is proposed, which is represented as, Note distribution count matrix. Using the proposed feature set, Artificial Neural Network (ANN) classification system is developed and implemented on South Indian Classical Music

Keywords: Audio Data Mining, Raga Identification, Note Transcription, Fundamental Frequency, Pitch Detection, South Indian Classical Music, ANN, Classifier.

Introduction

Audio mining research includes many ways of applying machine learning, speech processing, and language processing algorithms [1]. Audio recognition is a classic example of things that the human brain does well, but digital computers do poorly.

Digital computers can store and recall vast amount of data, perform mathematical calculation at blazing speed and perform repetitive tasks without becoming bored or inefficient. Computer performs very poorly when faced with raw sensory data. Teaching the computer to understand audio is a major task. Digital signal processing generally approaches the problem of audio recognition in two steps, viz; feature extraction and feature matching. It helps in the areas of prediction, search, explanation, learning, and language understanding. A new class of learning systems can be created that can infer knowledge and trends automatically from data, analyze and report application performance, and adapt and improve over time with minimal or zero human involvement. Effective techniques for mining speech, audio, and dialog data can impact numerous business and government applications.

The technology for monitoring conversational audio to discover patterns, capture useful trends, and generate alarms is essential for intelligence and law enforcement organizations as well as for enhancing call center operation. It is useful for a digital object identifier analyzing, monitoring, and tracking customer preferences and interactions to better establish customized sales and technical support strategies. It is also an essential tool in media content management for searching through large volumes of audio warehouses to find information, documents, and news. As personal computing power increases, so do both the demand for and the feasibility of automatic music analysis systems. Soon content discovery and indexing applications will require the ability to automatically analyze, classify and index musical databases, according to perceptual characteristics such as genre or mood. For Western music, a variety of studies have been undertaken to define what emotions music elicits and to understand the musical factors that underlie them. In the field of classification many attempts are made by extracting the spectral features from the audio signals. Many different types of classification strategies have been used like k-nearest neighbor schemes (K-NN) [2], Multivariate single gaussian models [2], Gaussian mixture models [3], Self organizing maps [4], Neural networks, Hidden Markov models , Support vector machines [5] and supervised hierarchical implementations of the aforementioned classifiers. It has been observed that in most of the cases, varying the classifier methodology used did not affect the classification accuracy much. However, feature set used for representing a pattern plays vital role in case of classification, so selection of feature set is very important and varying the feature set used for classification may affect the classification accuracy.

But, these classification systems have considered Western Music as the input data. In this paper, classification system is designed for Indian classical music, which classifies songs into different *ragas*. *Ragas* are sometimes defined as melody types. The *raga* system is a method of organizing tunes based on certain natural principles. Tunes in the same raga use the same (nominal) *swaras* in various combinations and with practice, the listener can pick up the resemblance. Indian classical music is defined by two basic elements - it must follow a *Raga* (classical mode), and a specific rhythm, the *Taal*. In any Indian classical composition, the music is based on a drone, i.e. a continual pitch that sounds throughout the concert, which is atonic [6]. *Raga* identification is a process of listening to a piece of music, synthesizing it into sequence of notes and analyzing the sequence of notes for identifying the raga it follows. Note Transcription is the first step in raga identification.

Fig 1.1 shows the general technique for audio data mining process. The first step obviously deals with collection of data to be mined. Here, the set of collected data are audio signals. After collection of data it is preprocessed where the raw signals are converted into a meaningful data and then the relevant features are extracted. In the next step, the set of features of different patterns is used as a training set for any type of classifier, where we get a classified model as a result.

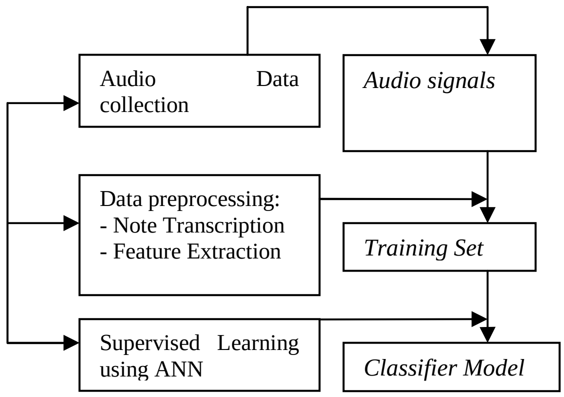

Figure 1.1: Audio Data Mining Technique.

The organization of the contents of our paper is as follows. Section 1 gives brief introduction to the classifier system, its context and applications. Section 2 describes the work done related to the analysis of western music and Indian classical music. In section 3, we discuss about the aspects of the design and implementation of the system. Section 4, presents the experimental studies of the system and finally in section 5, conclusion based on our study is given.

Related Works

Some of the related works in case of western music signal classification are mentioned in the above discussions. Here we explore the research work related to Indian classical music. Pandey [6] developed a system named 'Tansen' to automatically identify ragas *Yaman* and *Bhupali* using a Hidden Markov model. He demonstrated a success rate of 77% with a test data of thirty-one samples in a two-target test. But the methodology was not well documented. An additional stage that searched for specific pitch sequences improved performance to 87%. In an exploratory step, Chordia [7] classified one hundred thirty segments of sixty seconds each, from thirteen *ragas*. The feature vector was the Harmonic Pitch Class Profile (HPCP) for each segment. Perfect results were obtained using a K-NN classifier with 60/40% train/test split. This was further developed in later stages to improve the efficiency with different types of feature sets. Some of the classification was performed using Support Vector Machines, Maximum aposteriori (MAP) rule using a multivariate normal likelihood model (MVN), and random forests. The features were directly calculated from the input audio signals.

Note Transcription of music is defined to be the act of listening to a piece of music and of writing down the musical notation for the sounds that constitute the piece. In other words, this means converting an acoustic signal into a symbolic representation, which comprises musical events and their parameters. Very few attempts are made towards identification and analysis of Note transcription system in South Indian classical music [8][9]. Krishnaswamy [9], has described a method on how pitch tracking is useful for Note Transcription of South Indian Classical Music. He presented the results of applying pitch trackers to samples of South Indian classical (*Carnatic*) music. He investigated the various musical notes used and their intonation and tried different pitch tracking methods and observed their performance in *Carnatic* music analysis. Rajeshwari and Geetha, [8] have proposed a system for *swara*(Note) identification for South Indian Classical Music. It deals with the identification of the *swaras*(notes) in a given composition. The frequencies associated with the segments are identified and the exact tagging of note is performed to find the seven *swara* combinations in the given music signal.

Technical Work Preparation

In our system we have attempted to classify the *raga* of a given song in which it is composed. *Raga* in Indian classical Music is a very complex structure. The relationship between a *raga* and mood (feel) of the song called as *rasa* is strictly defined. A group of *ragas* may convey the same mood in a song but with slight modification. By extracting the underlying *raga* of a song, we achieve extracting melody patterns of song. We have used audio mining approach to get the *raga* patterns from the song, where the raw audio data signals are collected and processed to convert raw data into a meaningful text/numerical values and then useful patterns are extracted from the set of data. In the following part, we explain the two major steps involved in developing the system. The first step is Data Pre-Processing which involves Note Transcription & Feature Extraction. The next step is Classification.

Data Preprocessing

Music is a melody composed by combination of notes. A note is a basic unit of music. In Indian Classical music there are seven basic notes namely, s*a re ga ma pa da ni.* These seven notes are further extended to twelve notes namely, *sa re1 re2 ga1 ga2 ma1 ma2 pa da1 da2 ni1 ni2.* This group of twelve notes is called as an octave. There are three such octaves in Indian Classical Music called as Lower (*Mandra*), Middle (*Madhya*) and Higher (*taara*) octave. [10]

Note-Transcription of music is defined to be the act of listening to a piece of music and of writing down the musical notation for the sounds that constitute the piece. In other words, this means transforming an acoustic signal into a symbolic representation, which comprises musical events and their parameters. Feature Extraction is the process of extracting relevant feature from the set of notes which helps identify the uniqueness of a *raga*.

Note Transcription

A note is built of a set of frequencies consisting of fundamental frequency (f_0) and harmonic frequencies ($2f_0$, $3f_0$, $4f_0$…..). Pitch is the subjective perception of frequency that the human ear detects which is nothing but the fundamental frequency together with its harmonics. The Note transcription process involves identifying the fundamental frequencies of all the notes in a song and mapping it to corresponding notes. The input is taken in the form of wave-file, a raw signal. Fig 3.1 shows wave form for a signal of a song. Here, the first step involves breaking down the wave file into a set of segments (parts) where each segment corresponds to one note. Segmentation is done by dividing an audio clip of a song into a frame size of 50ms and 75% of overlap between the frames. Auto correlation Pitch calculation method is used to get the fundamental frequency of each frame. The adjacent frames which have similar fundamental frequencies are grouped into one segment.

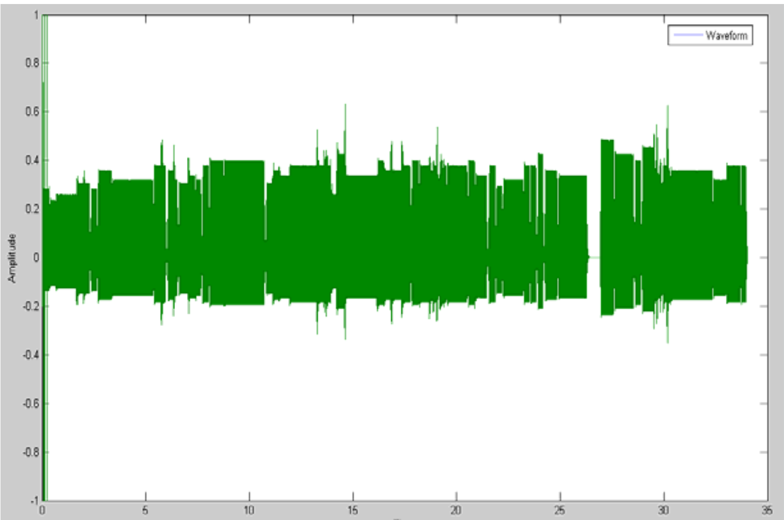

Figure 3.1: Wave-form of a signal.

Fig 3.2 shows a graph drawn for a list of fundamental frequencies for a monophonic audio clip of 34seconds duration. The X-axis indicates the frame numbers and Y-axis indicates the values for the fundamental frequency. The values remain same for the set of frames belonging to the same note. The fundamental frequency of the note called as 'sa' defines the scale of the song. The fundamental frequency used to play/sing any other note is related to this scale with a fixed ratio.

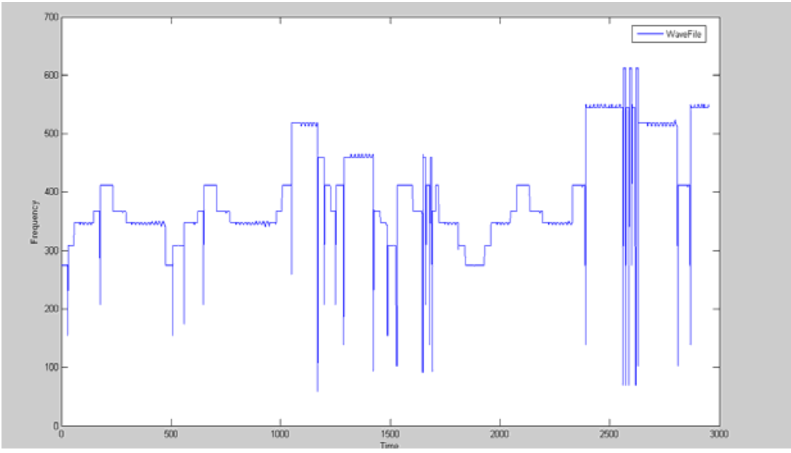

Figure 3.2: Fundamental Frequency List graph for a Monophonic song of 34sec.

The relationship in terms of ratios between the scale of the song and the frequencies of different notes in an octave is shown below.

Table 3.1: The relationship between frequency of notes and scale of the song.

Note	Ratio	Note	Ratio	Note	Ratio	Note	Ratio
Sa	1	ga1	32/27	ma2	729/512	da2	27/16
re1	256/243	ga2	81/64	pa	3/2	ni1	16/9
re2	9/8	ma1	4/3	da1	128/81	ni2	243/128

The three octaves have same set of twelve notes and the relationship of frequencies between the notes is as shown above. The frequencies of notes belonging to different octaves are also related. The use of three octaves in a song constitutes the possibility of using thirty six notes. The problem in Indian music is that the pitch or scale of the song is highly variable, which is dependent on the singer and the relationship between the fundamental frequency and the note are not fixed; it depends on the scale of the song chosen. The recent works done in this area [6, 7] have chosen the fixed scale. The mapping of fundamental frequency into a note is done using the above ratios and assuming that the scale is known. These notes are represented as numbers from 1 to 36, where numbers 1-12 are Lower octave notes, 13-24 are Middle octave notes and 25-36 are Higher octave notes. So a script containing sequence of notes played is generated for a given song.

Feature Extraction

After converting the raw signal into a set of meaningful data, the next step is to extract features from the data. *Raga* is an attractive combination of notes, engaging tonal context. What distinguish each *raga* are the characteristic phrases (of notes), sequence of notes, and the treatment given to each note in terms of its timing rendition, prominence or ornamentation. *Raga* identification is a process of listening to a piece of music, synthesizing it into sequence of notes and analyzing the sequence of notes for identifying the *raga* it follows.

A raga is defined by the choice of notes (from the twelve notes), ascending and descending sequences (*arohana & avarohana),* the nature of inflexion on different notes *(gamaka/meend),* Characteristic phrases *(swara sanchara /chalan/ pakad).* The type of the features extracted here are, the choice of notes and the Arohana-Avarohana pattern. These features help to identify the *raga*. A *raga* is constructed of five to seven consistent *swaras* (melodic steps). Each *raga* has an ascending (*arohana*) and descending (*avarohana*) form. These forms may be different i.e. the *arohana* and *avarohana* may contain different *swaras*; a*rohana* is the ascending sequence of notes which the *raga* follows. Any ascending sequence in improvised portions of the *raga* follows the pattern defined in *arohana* strictly. Similarly, a*varohana* is the corresponding descending sequence. Table 3.2 shows the *arohana/ avarohana* and Note combination of some of the *ragas*. In our system the features which are extracted from the set of data corresponds to the Note Combination and arohana/avarohana features of *raga*. These characteristics help unique identification of a raga and hence these are used to draw the features from the given set of data.

Table 3.2: Arohana-avaroahana pattern in ragas.

	NOTE combination	AROHANA / AVAROHANA
1 Todi	S r1 g1 m1 p d1 n1	S r1 g1 m1 p d1 n1 s'/ s' n1 d1 p m1 g1 r1 s
2 Dhanyasi	s r1 g1 m1 p d1 n1	S g1 m1 p n1 s' /s' n1 d1 p m1 g1 r1 s
3 Varali	s r1 g1 m2 p d1 n2	s r1 g1 m2 p d1 n2 s'/s' n2 d1 p m2 g1 r1 s
4 Mayamalavagaula	s r1 g2 m1 p d1 n2	s r1 g2 m1 p d1 n2 s'/s' n2 d1 p m1 g2 r1 s
5 Saveri	s r1 g2 m1 p d1 n2	s r1 m1 p d1 s'/s' n2 d1 p m1 g2 r1 s
6 chakravak	s r1 g2 m1 p d2 n1	s r1 g2 m1 p d2 n1 s'/s' n1 d2 p m1 g2 r1 s
7 Gaula	s r1 g2 m1 p n2	S r1 m1 p n2 s'/s' n2 p m1 g2 r1 s
8 Kamavardhini	s r1 g2 m2 p d1 n2	s r1 g2 m2 p d1 n2 s'/s' n2 d1 p m2 g2 r1 s
9. Saurashtra	s r1 g2 m1 p d2 n1 n2	s r1 g2 m1 p d2 n2 s'/s' n1 d2 n1 d2 p m1 g2 r1 s
10. Abheri	s r2 g1 m1 p d2 n1	s g1 m1 p n1 s'/s' n2 d2 p m1 g1 r2 s
11. Anandabhairavi	s r2 g1 m1 p d2 n1	S g1 r2 g1 m1 p d2 p s'/s' n1 d2 p m1 g1 r2 s
12. Bhairavi	s r2 g1 m1 p d1 d2 n1	s r2 g1 m1 p d2 n1/s' n1 d1 p m1 g1 r2 s

The musical script consisting of a sequence of notes where each note is represented as numbers is analyzed to generate Note Distribution Count matrix of order 12 × 12. The matrix is indexed by the Note numbers from N_1-N_{12}, both row and column wise. The notes belonging to three octaves are merged into one, which means that the note number 1(Lower Octave 'Sa') is same as note number 13(Middle Octave 'Sa') and note number 25(Higher Octave 'Sa') and all these three are merged into note number 1. The same notes belonging to different octaves are considered as single note. Each element, A_{ij}, in the Note Distribution matrix corresponds to the count of occurrence of pair of notes Ni and Nj where i,j= 1…..12. Thus, we write,

$$Aij=(COUNT (NiNj) * 100) / C \quad (1)$$

Where, 'C' is the total number of notes in the script. Value of A_{ij} denotes the contribution of pair (N_i,N_j), in composing the song and this value will be high for the pair in a*rohana/ avarohana* of the *raga*. A_{ij} will be equal to zero if the pair (N_iN_j) or any one of N_i or N_j does not exist in the song. All these features depend on the characteristics of the *raga* of a song. The set of 12 features are extracted as the average of the sum of all the values in each row and column as shown in (2).

$$Fi= (\sum_{i=1}^{12} Aij + \sum_{i=1}^{12} Aji)/2 \quad (2)$$

The set of features collected can be given as input to a number of classifiers. Selection of classifier does not make much difference since the features selected are very important in classification scheme. In our system, we have used Artificial Neural Network as the classifier system for *raga* identification.

Classification

An Artificial Neural Network model is created for classification purpose. The feature vector $(F_1,F_2,\ldots.F_{12})$ extracted from an input pattern is used for training and testing the system. In ANN, training is a process of strengthening the weights of the interconnections between the neurons by using known input/output pattern and learning algorithm, where input is the set of features and output is a target label. Each *raga* to be classified is assigned a unique target label for classification. The unique target labels are the numbers from 1 to 21, as 21 ragas are considered for classification. Testing is a process of extracting the output using a known input and the weights of the interconnection.

Experimental Studies and Results

The system is designed for mining *ragas* of musical compositions from South Indian classical music. There are around 2000 *ragas* in Indian classical music [10] and new *ragas* are continuously being invented and added to this collection. But, only around 50 *ragas* among these are commonly used. We have collected a group of songs belonging to 21 different *ragas* based on the availability of monophonic songs. Few songs and short audio clips were collected from the website and few were recorded with the help of musicians. Both vocal and instrumental songs are chosen. It makes no difference because fundamental frequency of a note heard, remains the same either in vocal or instrumental form. The difference in these two comes with harmonics of a note. We collected 100 songs making sure that at least 5 songs belong to the same *raga*. 75% of these collections were used for training purpose and the rest 25% for testing purpose. In the test set we have considered two different composition of the same raga. The ANN is designed as multi-layer network, given as 12×12×1; the structure is shown in the Fig 4.1 below

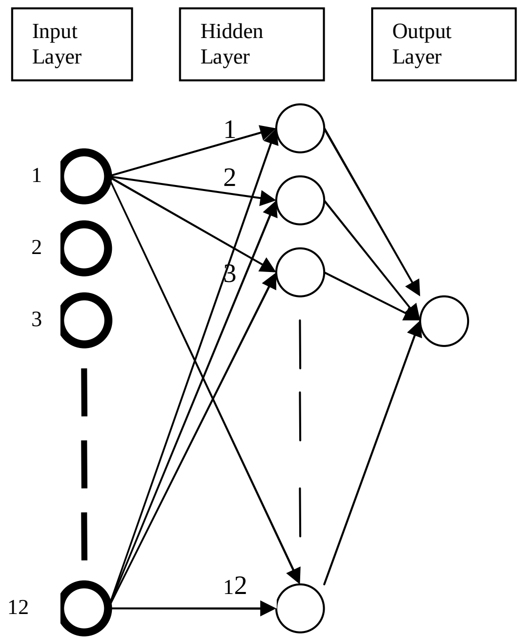

Figure 4.1: ANN Structure.

We have used built-in function available in Matlab Signal Processing Toolbox for reading the audio clip in wave-file format. The input songs collected were first converted into wave-file format using the available software. The music file of 1 ms duration contains 44 samples since music signals are sampled at the rate of 44.1 KHz. Normally, the duration of the song will be at least 2 minutes, that means we need to process at least 50 lakhs samples to get all the required information. The set of samples are divided into number of frames and each frame is processed individually. In our study, the music signal processing includes mainly three factors - deciding the frame size, applying a pitch calculation method and combining the frames into segment based on the pitch value. We have tried with different frame sizes ranging from 128 to 4096; the interval values are taken as powers of 2 to ease the pitch calculation computation. We found that the frame size should be at least 512 samples to identify the fundamental frequency of a note and below which the fundamental frequency of all the small silence periods is also calculated. The maximum limit of the frame size is derived using the relation between tempo and the number of samples in a note. Tempo of a song is defined as the number of beats per minute (bpm). A beat is duration of one pulse in a song. The fast track will have higher tempo and vice versa. In music, the duration of one full note is same as the duration of one beat. The tempo of the song ranges from 80 bpm to 140bpm. The Table 4.1 shows the values for the number of samples in full, half, quarter notes for different tempo of a song. If a quarter note is played in a song with 140 bpm, then duration of the note is 4800 samples. So, if we consider a frame size greater than 4800 then there is a possibility that some of the frames will contain more than one adjacent frequency.

Table 4.1: Relationship of tempo of a song and number of samples in single note.

Tempo (bpm)	Full Note	Half Note	Quarter Note
80	33075	16537	8300
90	29500	14750	7400
100	26460	13230	6615
110	24000	12000	6000
120	22060	11030	5500
130	20400	10200	5100
140	18900	9450	4800

By these discussions, we derive that the frame size can be in the range of 512 to 4096. The overlapping must be at least 25% to maintain the continuity of values in signal. The lower frame size will have the advantage of computing all the fundamental frequency values in a detailed manner. The disadvantages are that the speed of computation will be slow as there will be many frames and there will be more possibility of calculating the frequency of the part of the signal other than note like noise, silence etc. As we reduce the frame size, the above mentioned factors will affect more. As we increase the frame size the above mentioned factors will have less effect, but the disadvantage of larger frame size is that, sometimes we may miss the exact location of the edge between two notes. If we want to calculate the exact duration of a note in a signal, the large frame size will give less accurate result. So the frame size chosen must be a balance between lower and the higher limit. In our paper, the frame size is taken as 2048 samples with an overlapping of 75%. A valid fundamental frequency will span across at least 5-6 frames; other values can be neglected.

There are mainly two methods for calculating fundamental frequency of a signal; one is the cepstrum method (frequency domain) and another is an autocorrelation method (time domain). We have tried both the methods and found that both can calculate the pitch accurately. Both the techniques yield almost same result but we selected autocorrelation method which is more suitable for periodic signal as it computes the pitch directly from the waveform. This technique is computationally easy because the original signal will be in time domain so pitch will be directly calculated by correlating the signal by itself with different time lags.

Table 4.2: Note Distribution Count Matrix for a song of 'Dhanyasi' raga.

	s	**r1**	**r2**	**g1**	**g2**	**m1**	**m2**	**p**	**d1**	**d2**	**n1**	**n2**
S	0	0	2.38	0	0	1.58	0	0	0	0	0	0
r1	0	0	0	0	0	0	0	0	0	0	0	0
R2	1.58	0	0	0	0	0	0	0	0	0	0	0
g1	0	0	0	0	0	0	0	0	0	0	0	0
g2	0	0	0	0	0	5.55	0	0	0	0	0	0
m1	0.79	0	0	0	5.55	0	0	7.14	0	0.79	0	0
m2	0	0	0	0	0	0	0	0	0	0	0	0
p	0.79	0	0	0	0	6.34	0	0	0	8.73	0	0
d1	0	0	0	0	0	0	0	0	0	0	0	0
d2	0	0	0	0	0	0.79	0	7.93	0	0	7.93	0
n1	0.79	0	0	0	0	0	0	0.79	0	7.93	0	0
n2	0	0	0	0	0	0	0	0	0	0	0	0

The sequence of notes for a given signal are processed to generate the Note Distribution Count Matrix(NDCM) using a method described in the section 3. We have followed the *Melakarta Raga Classification System*[10], the system which clearly defines the classification of set of ragas into 72 number of *melakartas*. One *melakarta* is defined by the combination of seven/six/five notes used out of the twelve basic notes of Indian classical music. The *ragas* belonging to one *Melakarta* will have same combination of notes, but different *arohana/avarohana* patterns. Different *ragas* are mainly distinguished based on the *arohana* / *avarohana* pattern of notes. So, we have attempted to extract features based on this concept and obtain the Note Distribution Count Matrix (NDCM). The table 4.2 shows the Note Distribution Count Matrix for the *raga* Dhanyasi. As this *raga* does not use the notes 'r2, g2, m2, d2 and n2', the rows and columns corresponding to these notes are filled with values zero. These factors are helpful to distinguish between the *ragas* which use different set of notes. But, there is a possibility that two *ragas* use same set of notes of an octave. In such cases, the values representing the count in the matrix become useful. Table 4.3 shows the matrix for the raga 'Khamaj'.

Table 4.3: Note Distribution Count Matrix for a song of 'Khamaj' Raga.

	s	**r1**	**r2**	**g1**	**g2**	**m1**	**m2**	**p**	**d1**	**d2**	**n1**	**n2**
s	0	3.68	0	1.38	0	0	0	0	0	0	0	0
r1	5.99	0	0	0	0	0	0	0	0	0	0	0
r2	0	0	0	0	0	0	0	0	0	0	0	0
g1	0	2.76	0	0	0	5.06	0	0	0	0	0	0
g2	0	0	0	0	0	0	0	0	0	0	0	0
m1	0	0	0	4.14	0	0	0	8.75	0.46	0	0	0

m2	0	0	0	0	0	0	0	0	0	0	0	0
p	0	0	0	0.92	0	8.29	0	0	2.76	0	7.83	0
d1	0	0	0	0	0	0	0	7.37	0	0	0	0
d2	0	0	0	0	0	0	0	0	0	0	0	0
n1	0	0	0	0	0	0	0	2.30	4.60	0	0	0
n2	0	0	0	0	0	0	0	0	0	0	0	0

Table 4.4 shows the set of features extracted for some of the audio songs. The names of the *ragas* are mentioned and also the target labeling used for training the network is mentioned in the table. In these tables, we can clearly distinguish between different *ragas* primarily due to the number of notes used and secondarily due to the contribution of each pair in a *raga*. In the training part, these 12 features of each pattern and the target labels are used to construct a weight matrix and this weight matrix is used to get the output in testing process. We are getting 98% results for the current training and testing sets

Table 4.4: Features sets based on test results for songs of 10 ragas with two songs for each raga. F1 to F12 indicates feature variables, and T indicates the value of the target labeling used for a particular raga.

RAGA	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	T
arabh	.111	0	.169	0	.069	.138	0	.103	0	.269	0	.003	0.8
arabhi	.120	0	.184	0	.072	.108	0	.124	0	.092	0	.020	0.8
Dhanyasi	.055	.062	0	.071	0	.133	0	.191	.070	0	.073	0	0.4
Dhanyasi	.083	.077	0	.155	0	.143	0	.149	.056	0	.058	0	0.4
Hansadwani	.111	0	.186	0	.150	0	0	.140	0	0	0	.078	0.3
Hansadwani	.120	0	.222	0	.140	0	0	.925	0	0	0	.050	0.3
kalyani	.073	0	.113	0	.113	0	.136	.181	0	.136	0	.073	0.2
kalyani	.049	0	.079	0	.101	0	.138	.168	0	.150	0	.069	0.2
Kamavardini	.041	.065	0	0	.083	0	.059	.130	.166	0	0	.083	0.5
Kamavardini	.040	.056	0	0	.102	0	.132	.122	.102	0	0	.035	0.5
kamboji	.086	0	.103	0	.119	.111	0	.138	0	.123	.059	0	0.9
Kamboji	.095	0	.095	0	.113	.095	0	.128	0	.098	.053	0	0.9
kanada	.081	0	.137	.102	0	.172	0	.106	0	.108	.042	0	0.7
kanada	.121	0	.128	.057	0	.100	0	.500	0	.078	.092	0	0.7
mohana	.072	0	.125	0	.166	0	0	.166	0	.093	0	0	0.1
mohana	.937	0	.231	0	.212	0	0	.081	0	.056	0	0	0.1
Shankara	.074	0	.109	0	.102	.118	0	.126	0	.118	0	.081	1.0
Shankara	.049	0	.133	0	.165	.168	0	.120	0	.730	0	.269	1.0
Todi	.066	.121	0	.135	0	.101	0	.027	.124	0	.082	0	0.6
Todi	.069	.101	0	.105	0	.111	0	.108	.139	0	.077	0	0.6

Conclusions and Scope for Future Work

We have successfully designed a classification system for south Indian classical music using audio data mining techniques. The audio mining techniques require the audio signal to be converted into text/numerical values. Hence, the feature extraction process becomes simple. At present we have taken complete song as the input, but this can be changed to read a part of the song and get the required features. The set of features chosen are based on the definition of raga in an Indian Classical Music, so the feature set can be easily evaluated for more number of data sets. The melody pattern in the song mainly depends on the underlying *raga* used to compose the song.. By following such *raga* definition system, our work can be easily extended to thousands of *ragas* defined in the system. There is a clear partition between the data preprocessing step, feature extraction and classification. The present system requires the input song to be of monophonic type, and the scale of the song is known. The polyphonic pitch detection techniques can be applied and there will not be any restriction for the type of the input. By applying pitch detection techniques, the input song can be either monophonic or polyphonic, where we need to replace only the first module of the system i.e data preprocessing. Same feature extraction and classification method can be used as it works successfully. In case, scale is not known, an improved generalized scale detection algorithm can be easily added to the existing system. The accuracy of the system mainly depends on the successful transcription of notes, but as we have considered monophonic song the note transcription is not a problem. For polyphonic music, we need to accurately convert the signals into a set of notes which is a very challenging task

References

[1] Lie Lu and Hong-Jiang Zhang, “Content analysis for audio classification and segmentation.” , IEEE Transactions on Speech and Audio Processing, vol 10, pp:504–516, October 2002.

[2] George Tzanetakis and Perry Cook, “Musical genre classification of audio signals”, IEEE Transactions on Speech and Audio Processing, 2002., Volume 10, Issue 5, Jul 2002 pp: 293 - 302

[3] George Tzanetakis, Georg Essl, and Perry Cook, “Automatic musical genre classification of audio signals”, in Proceedings of ISMIR 2001: The International Conference on Music Information Retrieval and Related Activities, pp 205-210

[4] Pedro J Ponce de Le´on and Jos´e M Iesta, “Feature driven recognition of music styles”, Tech. Rep., Departamento de Lenguajes y Sistemas Inform´aticos, Universidad de Alicante, Ap. 99, E- 03080 Alicante, Spain, 2003, LNCS Vol. 2652,pp 773-781

[5] Changsheng Xu, Namunu C Maddage, Xi Shao, Fang Cao, and Qi Tian, “Musical genre classification using support vector machines”, Proceedings, (ICASSP '03). 2003 IEEE International Conference, vol 5, pp 429-432

[6] Gaurav Pandey, Chaitanya Mishra, and Paul Ipe, “Tansen: A system for automatic raga Identification”, In Proceedings of the 1st Indian International Conference on Artificial Intelligence, pp 1350–1363, 2003.

[7] Parag Chordia, “Automatic rag classification using spectrally derived tone profiles”, In Proceedings of the International Computer Music Conference, 2004, vol 129, pp 83-87

[8] Rajeswari Sridhar, and T.V. Geetha, “Swara Indentification for South Indian Classical Music“, 9th International Conference on Information Technology (ICIT'06), (2006), pp 143-144

[9] Krishnaswamy, A, “Application of pitch tracking to South Indian classical music”, Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on 19-22 Oct. 2003

[10] S. K. Padmanabha Acharya, “PanchamaVeda”, Published by S.K. Padmanabha Acharya, Kasargod, India, 2001