Journal of Engineering Science and Technology
Vol. 16, No. 4 (2021) 2876 - 2887

MUSIC CLASSIFICATION USING ASSOCIATION RULE AND K NEAREST NEIGHBOR

KALYANI C. WAGHMARE*, BALWANT A. SONKAMBLE

Pune Institute of Computer Technology, No. 27, Near, Trimurti Chowk, Bharati Vidyapeeth Campus, Dhankawadi, Pune, Maharashtra 411043, India
* Corresponding author: kcwaghmare@pict.edu

Abstract

Music is a melodious combination of different notes. The repeating patterns in it are usually used to classify it. The music experts can do this task easily but for naïve users, it is difficult or next to impossible. This demands automatic identification of patterns from an audio sample of the music. The automatic identification of repeating patterns is a complex task that includes signal processing as well as machine learning concepts. In this paper, Indian Classical Music is considered to classify based on its Ragas. The most frequent patterns are automatically identified using Association Rule Mining from machine learning by applying of pitch values calculated using the Autocorrelation method in Signal Processing. The concept of Term Frequency Inverse Document Frequency in Information Retrieval is applied to find the Pattern Frequency and Inverse Sample Frequency (PFISF) of each pattern. The K-Nearest Neighbour and Support Vector Machine classifiers are used with PFISF features and analysed performance using Accuracy and F1-score. The little increase in Accuracy of nearly 1-2% and more increase in F1-score nearly 5% is observed for the proposed method than the Pitch Class Distribution method. The increase in F1-score indicates that True Positive values are increased.

Keywords: Association rule mining, Indian classical music, Pattern frequency inverse sample frequency (PFISF), Supervised learning algorithms, Term frequency inverse document frequency.

1. Introduction

Music is a melodious combination of vocal and instrumental sounds. When people hear any music, the catchy and repeating phrases from it carve in their mind, and whenever we want to find any distinction or similarity, we use that knowledge. In Indian Classical Music (ICM), trained people usually identify catchy phrases from music. They try to pinpoint characteristics of Raga from it and then conclude which Raga it belongs to. In ICM knowing Raga of any sample is very helpful as Raga has a very rich background. It is a collection of melodic phrases of different notes or Swaras in 3 different octaves as lower, middle, and higher. The combination of different notes and the way it is sung makes Raga unique and effective. Every Raga evokes different emotions from it and induces these feelings in listeners [1, 2]. This gave birth to music therapy using Raga as supplementary medicine. The indexing of music by Raga name will help in such music therapy applications for efficient searching.

In ICM, freedom is given to the artist for improvising the performance by systematically following the rules of Raga. Due to this, the performances of the same Raga by two different signer or even the same singer may vary from time to time. The melodic phrases in input can be obtained by using pitch values [3]. The pitch values are used by the researcher's different ways such as Pitch Class Distributions (PCD), constructing statistical models to identify the Raga of a given input [4].

Chordia and Rae [5], Koduri et al. [6], and Chordia and Senturk [7] created variants of PCD based on bin weighting and note stabilization. In this approach, the loss of temporal properties in melodic phrases is observed, which is important for Raga Recognition. To overcome the loss of temporal properties authors increased the number of bins in the pitch distribution but did not prove to be much useful. The authors tried with different combinations of distribution functions and distance measures [6, 7].

To overcome the limitations of pitch distribution by modelling pitch sequence information, the use of the n-gram model or Markov Model is proposed by researchers. Krishna et al. [8] presented, pitch contour is extracted, for manually marked motifs. Hidden Markov Model (HMM) is defined for each motif. The authors observed that the motifs are identified correctly if the phrases are long enough. The authors observed that a significant understanding of rhythm and acoustics is necessary for extracting melodic motifs.

Kumar et al. [9] indicate, the results are improved by combining the concept of the n-gram model and Pitch Distribution. Kullback Leibler (KL) distance measure is used for comparing two Pitch distribution. A 4-gram histogram of notes is created. A multiclass Support Vector Machine (SVM) classifier is used by combining 2 linear kernels. The overall accuracy is improved than previous approaches.

The pitch-based method's first requirement is the accurate identification of tonic. Tonic identification is itself one research topic [7]. So, few researchers implemented Raga identification using Mel Frequency Cepstrum Coefficients (MFCC). Audio files are divided into overlapping frames and identified MFCC features for every frame [10]. It is observed that only MFCC features are not sufficient for Raga Identification So to improve the result, the MFCC features are combined with Chromagram which is the distribution of energies in 12 semitones [11].

In the manual Raga identification process, experts try to identify repeating and catchy phrases in musical excerpts. These phrases reveal the characteristics of Raga which are useful for Raga Recognition. Gulati et al. [12] performed automatic identification of melodic phrases based on the Time-Delayed Melody Surface (TDMS) which is considering the concept of delay coordinates from Time series analysis. The performance of the K-nearest Neighbour (KNN) classifier is analysed with different distance measures.

Gulati et al. [13] considered patterns in 2s segments and created a vocabulary of the pattern. Term Frequency Inverse Document Frequency (TFIDF) concept from text classification is implemented with different classifiers in machine learning.

Gulati et al. [14] modified the Locality Sensitive Hashing (LSH) concept used by Ryynnen and Klapuri [15] for Query by humming. LSH is a technique that can match similar time series into the same bucket. LSH is close to how humans identify Ragas. The LSH is used to measure the similarity of ragas and allocated a raga based on a ranking of the similarity measure. Gulati et al. [12, 13]mentioned that the performance of algorithms doing well for distinct ragas, but unable to capture the delicate melodic tones required to differentiate similar ragas. Authors claimed that the method proposed in this paper is scalable.

Ranjani et al. [16] showed that prescriptive notations contain Raga attributes and sufficient to identify a Raga of Carnatic music. The work is restricted to the notations of 7 notes and suppresses the finer note position information. A dictionary-based approach is used to captures the statistics of repetitive note patterns within a Raga notation.

In all the above reference authors extracted features applied various data mining algorithms such as KNN, Decision Tree, SVM, Naïve Bayes on it. These Distinct approaches heavily rely on feature extraction mechanisms. Anand [17], Ross et al. [18], Chowdhuri [19], and Madhusudhan and Chowdhary [20] tried to develop a Neural Network for Raga identification. The KNN algorithm works in real-time, i.e., it does not have a training phase. Due to the availability of the training phase Decision Tree, SVM, Naïve Bayes is computing faster in the testing phase than KNN but as dataset changes training will be required for them. In this paper, our focus is on Pattern identification, so we considered a simple KNN classifier, which is easy to understand and implement. The results of PCD and pattern-based features are compared using KNN.

The overall summary of the existing classification algorithm based on Raga is the loss of temporal characteristics occurs in the pitch distribution method which is necessary for Raga. The existing phrase-based methods are implemented for limited input and manually annotated data. Very few people worked on the automatic identification of pattern for Raga Identification.

The paper is organized as follows: Section II briefs about the proposed Pattern Frequency Inverse Sample Frequency (PFISF) algorithm. Section III gives details of experimental results and the analysis. Section IV Conclusion.

2. Proposed Method

The Ragas is the central notion of Indian Classical Music. Trained people usually identify catchy phrases from music and try to identify, which Raga it belongs to. The melodic repeating phrases are cues for listeners to identify the Raga of the music. As the identification of catchy phrases in music is a challenging task, very few researchers worked on identifying melodic phrases automatically and used in

Raga Recognition. Identifying catchy phrases is very difficult for untrained people even trained persons cannot recognize it 100% correctly.

In this section, a different method is designed and developed to recognize Raga of Music by automatically identifying repeating phrases and considering its PFISF. Data Mining is a multidisciplinary field. It performs the task of identifying the unknown and interesting patterns. Association rule learning method from Data Mining finds relationships among the variables. This is also known as Market Basket analysis [21]. The Association Rule Mining is successfully implemented in the intrusion detection system [22], heart disease prediction [23], and Recommender system [24]. In Indian Musical data, we have 12 Swaras in three different octaves. The concept of the Association Rule Learning concept is used to find the relations among these Swaras and identified a frequent set of Swaras. TFIDF from Information retrieval is implemented to get the feature vector for calculating the similarity of different samples. The TFIDF method implemented in research paper classification [25], document clustering [26]. The PFISF value is calculated to find the similarity among samples. The block diagram of the proposed Phrase-based Raga Identification process is shown in Fig. 1.

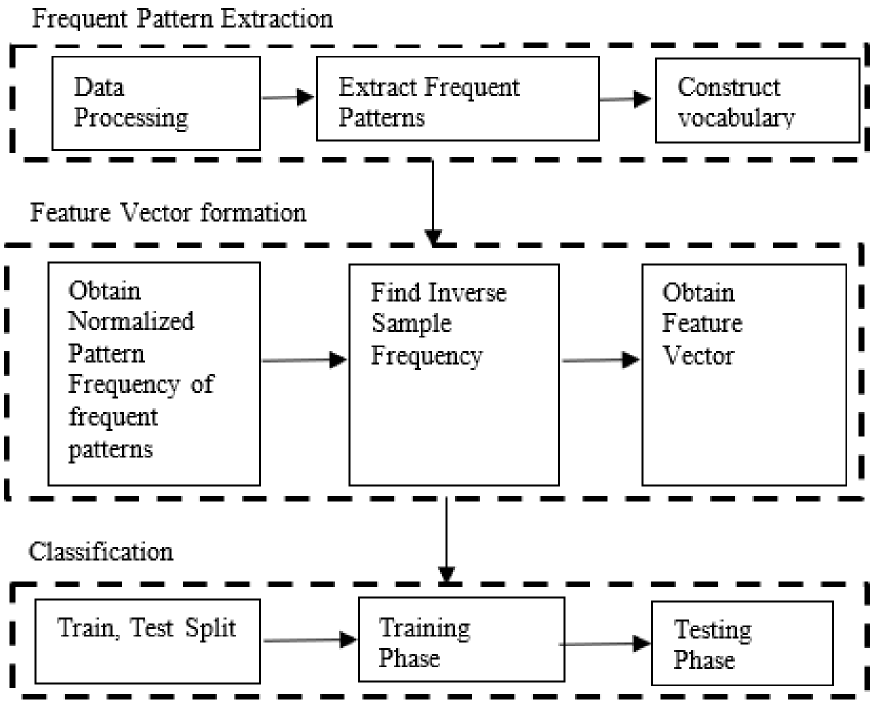

Fig. 1. Block diagram for Phrase-based Raga identification.

This process is divided into 3 modules: Frequent Pattern Extraction, Feature Extraction, and Classification. In Data Processing, block Pitch values to 220 Hz as tonic pitch is extracted using autocorrelation. Pitch values distribution is done with 12 bins per octave. The first octave is from 110 Hz to 220 Hz, Second, 220 Hz to 440 Hz, and 440 Hz to 880 Hz. In the block extract frequent pattern, first, patterns with 2 notes (bigram) in 'L_2' is calculated by checking two contiguous notes. Then identified frequent '*K*' size pattern i.e., 'L_k' by performing self-join of 'L_{k-1}' having k-2 items common. Frequent patterns are extracted for all samples in the dataset and constructed vocabulary, like dictionary of words by taking unique patterns from all the samples. The following steps are followed to extract frequent patterns and constructing vocabulary.

- Step 1

  Calculate Normalized Pitch values (NAP) of every sample to tonic 220 Hz using autocorrelation [27].

- Step 2

  Construct bins of notes in three octaves, Eq. (1).

$$FBIN = 220 \times 2^{i/\beta} \tag{1}$$

  where β is number of bins / octaves.

- Step 3

  Construct bigram of every sample to get frequent pattern of size 2.

$$B = \sum_{i=1}^{N} \sum_{j=1}^{d} E\left(nap_i, FBIN_j\right) \tag{2}$$

  where $E(a,b) = \begin{Bmatrix} 1 & if\ (nap_i == FBIN_i\ AND\ nap_{i+1} == FBIN_j) \\ 0 & otherwise \end{Bmatrix}$ (3)

- Step 4

  Identify frequent '*K*' itemset from bigram.

  Let $L_2 = B_{ij}$ $\forall_{\substack{i=1..N \\ j=1..d}} B_{ij} > 0$

  Let C_k be the frequent '*K*' itemset.

$Ck = \{ \{(b_1=1) \wedge (b_2=1) \wedge \ldots \wedge (b_k=1)\}\}$ where $k \geq 3$

$$C_k = L_{k-1} \bowtie L_{k-1} \tag{4}$$

- Step 5

  Construct vocabulary of all patterns.

$$V = \bigcup_{r=1}^{N} C_k \tag{5}$$

  where C_k is frequent '*k*' itemset.

The vocabulary of frequent pattern is given as input to the next Feature Vector Formation module. In the block obtain Normalized Pattern Frequency of frequent patterns, initially calculate the frequency count of each pattern in every sample. To normalize the frequency count of a pattern for a sample, the frequency count of a sample is divided by the maximum frequency count of that pattern. Equation (6) shows calculations of Normalized Pattern Frequency.

$$NPF_{i,j} = \frac{PF_{i,j}}{max(PF_{i,1..N})} \tag{6}$$

where:

$PF_{i,j}$ is frequency pattern of i^{th} Pattern in j^{th} sample.

N is a few samples in the dataset.

So, the pattern, having a maximum frequency in a sample will be equal to 1 and for other samples, it will be less than 1. In Indian Classical Raga, the pattern having the highest frequency does not imply, it is catchy phrases or distinguishing phrases of Raga. It may be a common phrase used in almost Ragas. So, in the next block of Find Inverse Sample Frequency, we are calculating Inverse Sample Frequency as Eq. (7),

$$ISF = \ log_8 \left(\frac{N}{SF_i}\right) \tag{7}$$

N number of samples in dataset and

$$SF_i = \sum E(C_k, S_i) \tag{8}$$

where,

$$E(a,b) = \left\{ \begin{matrix} 1 & if\ a \exists\ b \\ 0 & otherwise \end{matrix} \right\} \tag{9}$$

Si is a sample, *Ck* is frequent Pattern. The log is taken by considering base ‘8’ as we want to divide the dataset into the groups of 8 Ragas.

In obtain feature vector block, normalized pattern frequency is multiplied with inverse sample frequency and obtaining feature vector, *F* [*N* × *M*]. Where *N* is a total number of samples in dataset, *M* number of Frequent Patterns in vocabulary.

The feature vector, *F* is given as input to the classification module. Input splitting is done in a 70 : 30 ratios. In this paper, KNN and SVM algorithms are implemented.

K-Nearest neighbour (KNN) classifier

Lamba and Kumar [28] surveyed KNN and its variants. KNN is easy to understand and implement. It does not require any parameter from given data to evaluate so it is defined as a non-parametric method. It compares every tuple in a dataset with a trained model and not with global data, so it is considered as instance-based learning. The main problem in KNN is the selection of '*K*' value. Usually, researchers run the KNN algorithm several times with preferably odd values of *K* and choose the *K* that reduces the number of errors. The well-known 'elbow' method is usually applied to decide the '*K*' value. The complexity of the KNN algorithm is *O(NK + ND)* where *N* is a few testing tuples, *D* is the number of training tuples and *K* is the number of neighbours. The training data stored in ‘*X*’ having *N* tuples with ‘*d*’ dimensions.

The class label of 8 Raga stored in ‘Y’ Y = {1 – Asawari, 2 – Bageshree, 3- Bhairavi, 4 - Bhupali, 5- Darbari Kanada, 6- Malkauns, 7- Brindavani Sarang, 8- Yaman}.The steps followed in the KNN algorithm are as follows:

- Step 1

  Calculate Euclidean distance of each data point with other as per Eq. (5).

$$Dist(i,j) = \ \sqrt{\sum_{n=1}^{d} (i_n - j_n)^2} \tag{5}$$

- Step 2: Sort distance of every data-point in ascending order.
- Step3: Select labels of first ‘*K*’ entries from sorted distance *D*.
- Step 4: Return mode of ‘*K*’ labels.
- Step 5: Repeat steps 5, 6 for odd values of *K*.

3. Experimental Results and Discussion

Data considered for experimentation is generated for 8 different Raga. 25 samples sung by each singer of every Raga are recorded in a soundproof room with Tanpura as drone and Tabla for rhythm. The experimentation is done on samples recorded by 6 singers. The file is stored in wav format with sampling frequency 44100 Hz and 16 bps. The frame size is considered as 20 ms with 25% overlapping. The 8 Ragas considered for experimentation are Asavari, Bageshree, Bhairavi, Bhoopali, Darbari Kanada, Malkauns, Vrindavani Sarang, and Yaman. Dataset is a self-generated dataset for research work. The samples of the same Raga from a standard dataset of CompMusic [12, 13] are collected for the experimentation. The samples of Raga from CompMusic are extracted as per the instructions based on Sankalp [29] research.

Initially, pitch values are extracted using autocorrelation. Distribution of these values is done in a PCD of 36 bins. The PCD of the sample stores the frequency count of every bin.

To get the patterns, in music, only a single occurrence of the note is not important, the 1 note is occurring along with which note is important. So now to extract frequent patterns, starting is performed with patterns having 2 notes. The data of the 2-note pattern is considered to find a 3-note pattern. This continues until we can find maximum frequent patterns.

In our dataset, total 1400 samples are presented for every sample we have identified frequent patterns. The vocabulary is constructed from all identified patterns of every sample by removing duplicate patterns. In the dataset of 1400 samples, we are getting 969 unique samples. The frequency count of every pattern in every sample is obtained, and then calculated Normalized Pattern Frequency (NPF) as per Eq. (1). This will create matrix of [969×1400] for our dataset where rows represent pattern number and column represents sample number. Table 1 shows Pattern frequency for the first 10 patterns in 5 samples.

Table 1. Pattern frequency count in first 5 samples for our data.

	S1	S2	S3	S4	S5
P1	0.01416	0.00000	0.00000	0.00000	0.03593
P2	0.01416	0.00000	0.00000	0.00000	0.05072
P3	0.02301	0.00000	0.00000	0.00000	0.00000
P4	0.01593	0.00000	0.00000	0.00000	0.00000
P5	0.09560	0.09280	0.09280	0.00000	0.16062
P6	0.04249	0.07326	0.07326	0.00000	0.00000
P7	0.09383	0.06594	0.06594	0.00000	0.00000
P8	0.08852	0.06105	0.06105	0.03933	0.05072
P9	0.05134	0.04640	0.04640	0.00000	0.00000
P10	0.06816	0.04396	0.04396	0.05589	0.05283

Using Eq. (2), ISF is obtained for every pattern identified in the dataset. The ISF for the first 10 patterns is shown in Fig. 2.

The multiplication of NPF and ISF is performed to get the feature vector PFSIF [1400×969]. Table 2 shows a few entries in pattern frequency inverse sample frequency (PFISF).

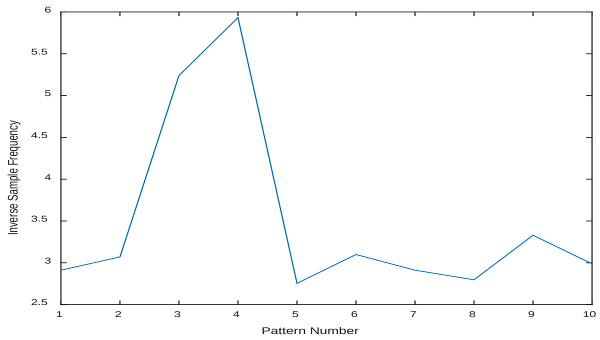

Fig. 2. Inverse Sample Frequency (ISF) for first 10 patterns in our data.

Table 2. PFISF for 5 samples in our data.

	S1	S2	S3	S4	S5
P1	0.04124	0.00000	0.00000	0.00000	0.10461
P2	0.04348	0.00000	0.00000	0.00000	0.15572
P3	0.12057	0.00000	0.00000	0.00000	0.00000
P4	0.09452	0.00000	0.00000	0.00000	0.00000
P5	0.26329	0.25559	0.25559	0.00000	0.44237
P6	0.13167	0.22704	0.22704	0.00000	0.00000
P7	0.27321	0.19200	0.19200	0.00000	0.00000
P8	0.24756	0.17075	0.17075	0.11000	0.14185
P9	0.17094	0.15449	0.15449	0.00000	0.00000
P10	0.20364	0.13134	0.13134	0.16699	0.15786

The Normalized Pattern Frequency of Patterns 1 and 2 are the same but the PFISF value varies for both patterns. The ISF value of Pattern 1 is less than pattern 2 means pattern 1 is more generalize and present in a greater number of samples. So, the PFISF value which is the weight for pattern 1 is less than pattern 2. In this way the pattern which is less occurring in multiple samples is having a greater ISF value and so its weight in that specific sample is high.

The feature vector PCD, PFISF is given to the classification module separately to implement KNN classifiers for identifying the Raga. The literature of KNN suggests the suitable value of K is equal to the square root of the number of samples in the training dataset. Considering this the performance of KNN is observed from K=1 to K= 38 for our dataset and from K=1 to K= 10 for the CompMusic Data set. To finalize the K value the elbow method is applied and finalized the K = 3. Table 3 and 4 show results of accuracy and F1-score, with PCD and PFISF method for our data set, CompMusic Dataset, respectively. The accuracy and F1-score are calculated as per the following Eqs. (11) and (12), respectively [30].

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{11}$$

$$F1 - Score = \frac{2 * TP}{(2 * TP + FP + FN)} \tag{12}$$

Table 3. Performance measures for our data.

Method / Performance measure	Accuracy KNN	Accuracy SVM	F1 Score KNN	F1 Score SVM
Using PCD	88.76%	84.74%	57.86%	39.99%
Using PFISF	89.95%	85.92%	62.96%	44.70%

Table 4. Performance measures for Compmusic data.

Method / Performance measure	Accuracy KNN	Accuracy SVM	F1 Score KNN	F1 Score SVM
Using PCD	83.96%	82.72%	39.26%	40.70%
Using PFISF	85.71%	87.77%	42.86%	53.11%

In accuracy true positive as well as true negatives are considered, so to analyse the performance of multiclass classifiers we evaluated the other performance measure F1-score. F1-Score is a combination of both precision and recall. It makes a balance between precision and recall and that is why it is very useful to choose classifiers for any dataset.

In pitch features, it is necessary to accurately identify tonic and the other notes. The slight error in tonic may affect the note values. In our implementation, PCD method identifies 36 features for 1 sample, while PFISF has 969 features for 1 sample. In these 969 features, many zeros are present as if the pattern is not present in that sample. In the proposed PFISF methods little improvement in accuracy and more improvement in F1-score is observed using traditional KNN. This shows that the pattern-based Raga identification gives a more accurate result than pitch distribution. The manual annotation of a complete dataset is very time consuming and not possible to everyone also pattern annotation may happen wrongly. So automatic identification of patterns is the best solution.

The main hurdle in comparison of results with other existing methods is the dataset used in the existing method and our method are different. Bidkar et al. [31], claimed that 92% accuracy for 1 Raga, and it is almost less than 80% for another Raga. Kumar et al. [9] obtained 83.39 % accuracy with their approach for Compmusic data which was higher than [6]. We have used the same dataset, but samples of different Raga and we observed above 85% accuracy by our approach.

The Neural Network and its variations are used by [17-20]. The results of Neural Network for Raga Identification are comparatively higher than the results we got using KNN and SVM. The results may improve by applying neural network on PFISF features identified for pattern in our approach and could be considered for future work.

The improvement in the results may happen by modifying parameters in KNN or SVM such as weights to the different features, *K* values, and different distance measures, kernel function, and so on.

4. Conclusions

Music is a collection of melodious patterns. To classify music based on Raga, identifying repeating patterns is a very important task. The automatic identification of repeating patterns in Indian Classical Music is a difficult task for the naïve user. In this paper, an algorithm to identify the frequent patterns by using the Association Rule mining is defined. The PFISF features are extracted for the patterns based on the concept of Term Frequency Inverse Document Frequency from information retrieval. The KNN, SVM classifiers for identifying Raga using PFTSF features are implemented. The method we have implemented is very much mimicking manual Raga Identification. We observed little improvement in accuracy but more improvement in F1-Score value. The improvement F1-score shows that true positive entries are increasing. This proves that repeating patterns of notes are more useful for Raga Identification than considering only the distribution of the notes. In the future, the plan is to build a modified KNN, SVM classifier for Raga identification with an increase in accuracy and efficiency also apply and compare deep learning algorithm performance on our data.

References

1. Valla, J.M.; Alappatt, J.A.; Mathur, A.; and Singh N.C. (2017). Music and emotion-A case for north Indian classical music. *Frontiers in Psychology*, 8(2115), 1-9.
2. Avantika, M.; Vijayakumar, S.H.; Chakrabarti, B.; and Singh, N.C. (2015). Emotional responses to Hindustani raga music: The role of musical structure. *Frontiers in psychology*, 6(513), 1-11.
3. Ganguli, K.K.; and Rao, P. (2019). On the distributional representation of ragas: Experiments with allied Raga pairs. *Transactions of the International Society for Music Information Retrieval*, 1(1), 79-95.
4. Koduri, G.K.; Gulati, S.; Rao, P; and Serra, X..(2012). Raga recognition based on pitch distribution methods. *Journal of New Music Research*, 41(1), 337-350.
5. Chordia, P.; and Rae, A. (2007). Raga recognition using pitch class and pitch class dyad distribution. *8th International Society of Music Information Retrieval Conference*, Vienna, Austria, 431-436.
6. Koduri, G.K.; Gulati, S.; and Rao, P. (2011). A survey of raaga recognition techniques and improvements to the state-of-the-art. *Proceedings of Sound and Music*, Padova, Italy, 1-8.
7. Chordia, P.; and Senturk, S. (2013). Joint recognition of raag and tonic in North Indian music. *Computer Music Journal*, 37(3), 82-98.
8. Krishna, A.S.; Rajkumar, P.V.; Saishankar, K.P.; and John, M. (2011). Identification of carnatic raagas using hidden Markov models. *Proceedings of 2011 IEEE 9th International Symposium on Applied Machine Intelligence and Informatics*, Smolenice, Slovakia, 107-110.
9. Kumar, V.; Pandya, H.; and Jawahar, C.V. (2014). Identifying ragas in Indian music. Proceedings of *IEEE 22nd International Conference on Pattern Recognition*, Stockholm, Sweden, 767-772.
10. Deshmukh, K.M.; and Deore, P.J. (2017). Raga identification using MFCC and chroma features. *International Journal of Advanced Research in Computer Science*, 8(5), 725-729.
11. Dharini, D.; Revathi, A.; and Kalaivani, M. (2018). CD-HMM modeling for raga identification. *Proceedings of the 2nd International Conference on in I-SMAC (IoT in Social, Mobile, Analytics and Cloud)*, Palladam, India, 486-489.
12. Gulati, S.; Serra, J.; Ganguli, K.K.; Senturk, S.; and Serra, S. (2016). Time-Delayed melody surfaces for raga recognition. Proceedings of *17th*

International Society for Music Information Retrieval Conference, New York, USA, 751-757.

13. Gulati, S.; Serra, J.; Ishwar, V.; Senturk, S.; and Serra, X. (2016). Phrased based raga recognition using vector space modelling. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Shanghai, China, 66-70.
14. Padmasundari, G.; and Murthy, H.A.; (2017). Raga identification using locality sensitive hashing. *Proceedings of 23rd National Conference on Communications* , Chennai, India, 128-131.
15. Ryynnen, M.; and Klapuri, A. (2008). Query By humming of MIDI and audio using locality sensitive hashing. *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, USA, 2249-2252.
16. Ranjani, H.G.; and Sreenivas, T.V. Sreenivas (2017). Raga identification using repetitive note patterns from prescriptive notations of carnatic music. *arXiv: Audio and Speech Processing*, arXiv, 1, 1-15.
17. Anand, A. (2019). Raga identification using convolutional neural network. *Proceedings of IEEE 2nd International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, Gangtok, India, 1-6.
18. Ross, J.C.; Murthy, R.; Ganguli, K.K.; and Bhattacharyya, P. (2017). Identifying raga similarity in hindustani classical music through distributed representation of raga names. *Proceedings of* 13th *International Symposium on Computer Music Multidisciplinary Research*, Matosinhos, Portugal, 1-12.
19. Chowdhuri, S. (2019). PhonoNet: Multi-stage deep neural networks for raga identification in hindustani classical music. *Proceedings of 2019 International Conference on Multimedia Retrieval*, Ottawa Canada, 197-201.
20. Madhusudhan, S.; and Chowdhary, G. (2019). DeepSRGM - Sequence classification and ranking in Indian classical music with deep learning. *Proceedings of 20th International Society for Music Information Retrieval Conference*, Delft, Netherland, 533-540.
21. Rao, C.S.; Babu, D.R.; Shankar, R.S.; Kumar, V.P.; Rajanikanth, J.; and Sekhar, C.C. (2013). Mining association rules Based on boolean algorithm a study in large databases. *International Journal of Machine Learning and Computing*, 3(4), 347-351.
22. Kong, H.; Jong, C.; and Ryang, U. (2016). Implementation of association rule mining for network intrusion detection. *arXiv: Cryptography and Security*, 2, 1-10.
23. Said, I.U.; Adam, and Garko A.B. (2015). Association rule mining on medical data to predict heart disease. *International Journal of Science, Technology and Management*, 4(8), 26-35.
24. Varsaneh, H.H.; Neysiani, B.S.; Ziafat, H.; and Soltani, N. (2018). Recommendation systems based on association rule mining for a target object by evolutionary algorithms. *Emerging Science Journal*, 2(2), 100-107.
25. Kim, S.W.; and Gil, J.M. (2019). Research paper classification systems based on TF-IDF and LDA schemes. *Human-Centric Computing and Information Sciences*, 9(30), 1-21.
26. Kim, D.; Seo, D.; Cho, S.; and Kang, P. (2019). Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec. *Information Sciences*, 477, 15-29.

27. Waghmare, K.C.; and Sonkamble, B.A. (2020). Timbre with note based features for improving performance of music classification. *Journal of Advanced Science and Technology*, 29(3), 10328-10338.
28. Lamba, A.; and Kumar, D. (2016). Survey on KNN and its variants. *International Journal of Advanced Research in Computer and Communication Engineering,* 5(5), 430-435.
29. Sankalp, G, (2018). Retrieved November 26, 2018, from https://github.com/sankalpg/experiments/tree/master/MotifDiscovery/RagaRecognition/ISMIR_2016.
30. Jiawei, H.; Micheline, K.; and Jian Pei. (2011) . *Data mining: concepts and techniques* (3rd ed.) the Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers.
31. Bidkar, A.A.; Deshpande, R.S.; Dandawate, Y.H. (2018). A novel approach for selection of features for north Indian classical raga recognition of instrumental music. *Proceedings of IEEE International Conference on Advances in Communication and Computing Technology*, Ahmednagar, India, 499-503.