Published in AJSE, Vol:18, Issue: 02
Received on 23rd June 2019
Revised on 11th August 2019
Accepted on 11th August 2019

AIUB JOURNAL OF SCIENCE AND ENGINEERING
ISSN: 1608 – 3679 (print) 2520 – 4890 (Online)

Genre of Bangla Music: A Machine Classification Learning Approach

Abhijit Bhowmik, AZM Ehtesham Chowdhury

***Abstract*—The necessity for designing autonomous indexing tools to establish expressive and efficient means of describing musical media content is well recognized. Music genre classification systems are significant to manage and use music databases. This research paper proposes an enhanced method to automatically classify music into different genre using a machine learning approach and presents the insight and results of the application of the proposed scheme to the classification of a large set of The Bangla music content, a South-East Asian language rich with a variety of music genres developed over many centuries. Building upon musical feature extraction and decision-making techniques, we propose new features and procedures to achieve enhanced accuracy. We demonstrate the efficacy of the proposed method by extracting features from a dataset of hundreds of The Bangla music pieces and testing the automatic classification decisions. This is the first development of an automated classification technique applied specifically to the Bangla music to the best of our knowledge, while the superior accuracy of the method makes it universally applicable.**

Keywords—audio classification; music information retrieval; musical genre classification; musical instrument recognition;support vector machine; temporal feature integration

I. Introduction

Bangla, also referred to as “Bengali”, is the 6th largest language in the world in terms of number of native speakers [1][2]. According to Ethnologue [3] and other sources, about 210 million people use Bangla as their mother tongue in Bangladesh and India’s state of West Bengal in addition to a large worldwide diaspora.

The Bangla language has a rich literary heritage [4] and the Bangla music has been an integral part of how this language and the culture evolved over many centuries [5]. However, it is crucial to approximate the number of Bangla songs, there are many millions that are widely enjoyed by the native speakers. Like the rest of the world, digital transformation is fast taking roots in this part of the world, with digital archiving and distribution over Internet becoming popular, especially with the new generation. Most of the Bangla songs are now available on the internet platform and as well as via mobile applications such as Amazon.in, Gaan.com, GP Music, Yonder Music, Jio SAAVN, Gaana, Google Play Music, Banglalink VIBE etc [6].

The leading categories of the Bangla music are “Rabindrasangeet”, “Nazrulgeeti”, Folk, Baul, “Adhunik”, and Band music, among others [7]. The origin of “Rabindrasangeet” is from the works of Nobel winner poet, play writer and novelist, Rabindranath Tagore. “Rabindrasangeet” itself is largely classified into several sub-genres which are knotted by a mutual refrain of love and philosophy [8]. Bangla folk music has been being contributed by numerous novelists, poets. Folk is now one of the most significant and influencing musically in lives of Bengalis on both sides of the border (West Bengal-Bangladesh). LalonFokir, Hason Raja, Ramesh Shill, Abbas Uddin, etc. [7] had played the role to internationalize the Bangla Folk songs. Folk songs are characterized by simple words and musical structure. “Adhunik” literally means “modern songs”. Though the “Adhunik” or “modern” songs have become old, these continue to be called by the same name. Remarkably, these songs have grown faster than any other genres. It can be accommodated anything that fails to fit elsewhere as it is a miscellaneous category. So, although the nomenclature itself might not be as insightful, the genre itself is still well-defined [5]. A Bangla band uses western principles of music performs uniquely or mainly in the Bangla language and its uses. Bangla bands became popular with young people in the 1970s, both in India and Bangladesh, and became entrenched in modern Bengali culture.

Due to the recent widespread use of digital music, the demand for classifying music files more efficiently is increasing. Although there is a rich body of research work on music file classification, so far there has not been an effort to focus on automated classification of The Bangla music. In this paper, we introduce a new machine learning approach to automate the Bangla music genre classification and a corresponding software implementation. We demonstrate improved efficiencies in automatic analysis and classification of the leading genres with our proposed algorithms applied on a large music database.

Rest of the research paper is organized as follows, Section II presents the related works with a concise description of the previous research works, Section III contains a short description of the classification Proposed Methodology, Section IV discusses the results of experiments that were

Abhijit Bhowmik
American International University-Bangladesh
E-mail: abhijit@aiub.edu

AZM Ehtesham Chowdhury
American International University-Bangladesh
E-mail: ehtesham@aiub.edu

carried out for classifying and Conclusion is included in Section V.

II. Related Works

From past two decades, several researches have been being conducted to classify the music genre based on the researchers' native linguistic features and patterns. Researchers proposed several machine learning methodologies for the automatic music genre classification. A group of researchers classified the music genres considering the audio signals based on the polynomial kernel functions and machine learning algorithms [9]. Multiple feature vectors were considered in another approach where a pattern recognition ensemble and reported an accuracy of 65%. Another group of researchers [10] used a Support Vector Machines by means of designing an automatic music genre classifier where they used a dataset consisting of 500 Italian songs spanning 5 genres, the authors achieved 82% accuracy. David Pye used the content-based classification and retrieve a MPEG layer III digital music archive, based on Gaussian Mixture Modeling and Tree-based Vector Quantization [11]. The database consisted of 175 songs, including blues, classical, easy listening, dance (techno) and opera genres. The author reported an accuracy of 96%. An automatic genre classifier was proposed based on the separation of musical tracks based on Non-Negative Matrix Factorization [12]. Another researcher described an algorithm which was acquiesced to the MIREX 2007 contest where task was to classify the Audio Genre [13]. Based on a dataset of 400 songs including classical, hiphop, dance, jazz, blues, pop and rock music, this work reports an accuracy of 72%. The paper [14] presented a technique that extracted Spectral roll-off, Bandwidth, Loudness and Spectral Flux features by splitting all the signals into 21.3 milliseconds frames. Using a dataset of 2266 music extracted spanning 29 genres, they achieved an accuracy of 87%. In paper [15] author used a multi-resolution algorithm for the classification of music audio signals. The dataset consisted 1515 audio contents with 10 genres, yielding an accuracy of 87%. In 2016, a study argued to incorporate semantic study of audio signals to automatic and intelligent process of the signal and augmented interaction to find the category in an effective way [16]. Stober et al. performed a non-conventional survey [17] regarding the evaluation of genre identifications. Aggregate features were considered on AdaBoost to classify the genres by Bergstra et al. [18]. The AdaBoost applied the classification in an iterative approach where it combined the weighted votes of weak learners. Ntalampiras introduced data non-conventional selection method using Gaussian mixture model to develop the Universal Modeling (UM) [19] where the dataset characteristics adapted itself to achieve a greater recognition rate. Another group of researchers proposed a support vector machines (SVMs) based method for content-based retrieval and audio classification, achieving 87% accuracy on a dataset consisting of 409 pieces of songs [20] while Rafael et al. [21] considered data augmentation using Convolutional Neural Network which provided a significant amount of improvement in the accuracy of music genre classification. AK Koyla et. al proposed a theoretical approach to recognize music genre applying single layer Feedforward NN considering the musical features [22]. A study from Asfaqur Rahman provided the classification of only four basic genres of Bangladeshi music. However, the classifier did not incorporate the features of other genres [23]. The existing research works did not consider the features of the Bangla language.

In this paper, we propose a novel machine learning approach to automatically classify the Bangla music classification and a corresponding software application implementation. This research paper demonstrates the improved accuracies in automatic analysis and classification of the versatile and leading Bangla genres with our proposed algorithms applied on a large music database.

III. Proposed Model

This section describes the flow of our proposed approach. First, a number of features are selected for the implementation with the following definitions. A series of frames that depicts the spectral data as a two-dimensional image using 2D method of moments. The description of the spectrograph includes the changes over a short time frame [24]. The Root Mean Square (RMS) defines a measure which is the power of a signal over a window. To construct a rhythmic regular histogram the Beat Histogram correlates the RMS for all bins. This is used as a base feature for determining the best tempo match. A histogram depicts the relative strength of signal's diverse periodicities rhythmically [25]. A beat creates interference between two sounds having different frequencies which has variations periodically in volume whose rate is the variance between the two frequencies. Beat Sum sums all bins in the beat histogram. This measure has the significance of regular beats in a signal [26]. Noisiness is estimated using the spectral smoothness, Compactness. The components of a window's magnitude spectrum of fast-Fourier transform (FFT) with that of its neighboring windows [27] are compared to calculate the Noisiness. Fraction of Low Energy Frames is depicted as the fraction of previous windows where RMS is less than the mean RMS. This is a significant indication of the variability of the amplitude of windows. The fraction of the final 100 windows has an RMS which is less than the mean RMS of the final 100 windows. This indicate the amount of signal section is fairly relative to the other part of signal section [28]. Magnitude spectrum is a degree of the strength of diverse frequency components which is derived from the FFT. Method of Moments has the first five statistical moments of the spectrograph: the area, mean, density of power spectrum, skewness of spectrum, and spectral kurtosis. These features depict the form of the spectrograph of a window in 5 orders [28]. Q Liu et al. [29] proposed a formula to calculate the Mel-Frequency Cepstral Coefficients. The calculations are implemented in our work inspired from the Orange Cow voice recognition project [30] which beneficial to describe a spectrum window.

Spectral Centroid is the mass of the power spectrum. Spectral Centroid Variability is calculated by the standard deviation of the spectral centroid for latest 100 windows. Spectral Flux is the spectral correlation among consecutive windows which is used as a suggestion of the rate of change of the spectrum among windows. It is measure that is calculated the change in the magnitude spectrum from frame to frame of the quantity of spectral change in a signal. Spectral rolloff is another frequency where all 85% of the energy in the spectrum is underneath of this point, indicating the skewness of the frequencies which present in a window. From the highest bin in the beat histogram, the strongest beat in a signal is determined which is compared to other potential beats. By counting occurrences of domain signal that crosses zero within a given window Zero Crossing is calculated. Strongest Frequency FFT Maximum, Spectral Centroid, or Zero Crossings is the strongest frequency component of a signal, represents in Hz, depicted by finding the FFT bin with the highest power, the spectral centroid, or the number of zero-crossings, respectively [25].

In addition to the features described above, which have been described and utilized in previously reported methods (see references), we defined and used a new feature which we term as the Continuity feature. It represents the change of a single feature over the all the windows, determined from the a, b, and c coefficients of the following second-degree polynomial equation via curve fitting technique: y = ax2 + bx + c. Next, we define the four classifiers that we used in our experiment. Support vector machines (SVMs) are a collection of supervised learning approaches that are used for supervised learning, classification and regression. SVM constructs a splitting hyperplane in that space by considering viewing input data as two sets of vectors in an n-dimensional space where it maximizes the brim between the two datasets. Two parallel hyperplanes are constructed to calculate the margin, one on each side of the separating hyperplane. Each plane "pushes up against" the two data sets. The hyperplane that has the largest distance to the neighboring data points of both classes is achieved spontaneously. Here, in general the larger the margin the lower the generality of error of the classifier. The second classifier we used is J48 (C4.5 Pruned / Unpruned Decision Tree). Using C4.5 algorithm, a decision tree used to generate. The decision generated by C4.5 can be used to classify and reason where C4.5 is often denoted to as a statistical classifier [31] [32]. Next, a naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem (from Bayesian statistics) with strong (naive) independence assumptions. In simple terms, a naive Bayes classifier undertakes that the presence (or absence) of a particular feature of a class which can be unrelated to the presence (or absence) of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 4" in diameter. Even though these features depend on the existence of the other features, a naive Bayes classifier considers all of these properties to independently contribute to the probability that this fruit is an apple [32]. Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, naive Bayes model can be used without considering the Bayesian probability or Bayesian methods [32]. Despite providing apparently over-simplified assumptions and the naive design and, naive Bayes classifiers shows better in many complex real-world situations.

An advantage of the naive Bayes classifier is that it requires a small quantity of training data to estimate the means, deviations, variances of the variables which is crucial for classification. The variances of the variables for each class is determined and not the entire covariance matrix as independent variables are assumed. This research paper shows that a software package using Java is developed to implement and verify the obtained results with the proposed method, including Weka decision making and machine learning module.

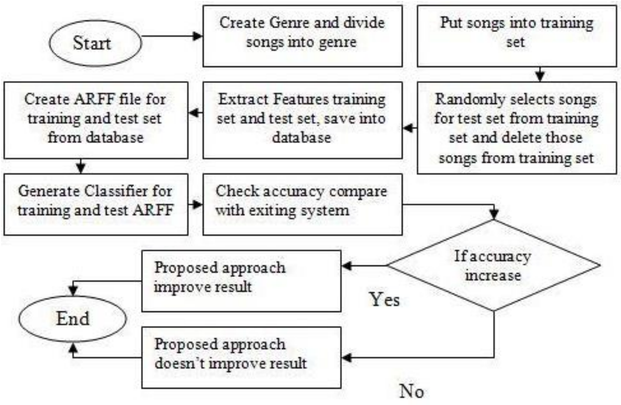
Start
Create Genre and divide songs into genre
Put songs into training set
Create ARFF file for training and test set from database
Extract Features training set and test set, save into database
Randomly selects songs for test set from training set and delete those songs from training set
Generate Classifier for training and test ARFF
Check accuracy compare with exiting system
If accuracy increase
Proposed approach improve result
Yes
End
Proposed approach doesn't improve result
No

Fig. 1. Flow chart of proposed approach.

Fig 1 shows the flow chart of the method. We based our work on a dataset consisting of 1200 Bangla songs, divided into four genres: Folk, "Rabindrasangeet", Band and "Adhunik", encoded in the wav format. The software we developed picks the songs randomly for training and tests from these genres, specifically 250 songs for the training set and 50 for the test set. Next, the program extracts the features from the training and the test sets. These features are extracted by four algorithms: Support Vector Machine, KNN, J48 and Naïve Bayes. As a new feature we have also used the second-degree polynomial function as described above. The extracted features from the training and test sets are stored using MY-SQL database. Next, Attribute-Relation File Format files were created from each song. The ARFF file is ASCII text file that provides a list of instances which share a collection of attributes. ARFF files have two dissimilar sections where the first one depicts the Header information and other one is Data information. The Machine Learning Project of The University of Waikato developed ARFF files to use with the Weka machine learning software [33]. The software then generates decisions from the classifiers derived from the training and test ARFF files and provides the percentage of difference between the two results.

We used the following libraries in our software package to extract the features from the songs. jAudio facilitates the extraction of features from audio files which is a software package. It helps to sharing and developing new features iteratively. To retrieval of various information from music the extracted features can be used. This could be done often using machine learning frameworks such as Weka [33]. Another software application, jMIR is open-source software which had developed using java and run on Java Machine to music information retrieval related researches. It is being used to research on audio music and symbolic formats and extract information regarding cultures from the world wide web and manage music collections. jMIR includes the software features to extract musical features as well as provide the way to applying machine learning methodologies and analyzing metadata. The lo- level features are combined to create the high-level and meaning features. To achieve that, the iterative approach to feature development emphasized in the design of all jMIR components. There are several characteristics of jAudio that allow such iterative feature development. For example, jAudio uses a such modular plugin interface that unable to modify core source code or recompilation when new features are added. It needs to place a newly compiled feature in a plugin folder as well as is referred by reference to it in an XML configuration file, which can refer to remote URLs and local file paths. jAudio can dynamically provide all features with the values of all other extracted features as well as dynamically calculates all feature dependencies inevitably. This allow the feature extraction in an appropriate schedule. jAudio works with audio samples to features as simple arrays which helps researchers work on high level and release to not work directly with Java's low-level issues such as buffering or audio format conversions [34]. LibXtract [35] is another lightweight and portable library for extracting audio feature which provides a relatively comprehensive set of feature extraction primitives. These are designed to be 'cascaded' to create extraction hierarchies. For example, 'variance', 'skewness', 'average deviation' and 'kurtosis', all need the 'mean' of the input vector to be precomputed. The 'mean' will be sent as an argument rather than computing the 'mean' 'inside' each function. The mean is calculated only once, and then sent to any functions based on the user's requirement.

This philosophy of "cascading" characteristics is taken into account throughout the library, for example with features that operate within the spectrum range of the signal vector (eg "Irregularities"), the spectrum magnitude is not calculated "inside" of the corresponding function which is the pointer to the first component in matrix containing the size spectrum sent as an argument. The library became more efficient during computing the large numbers of features. It will be flexible because the extraction functions will be arbitrarily collective.

jSymbolic is a library which extracts high-level musical features from symbolic music representations. It specially extracts features from MIDI file. It has several design characteristics like jAudio which allow feature extensibility. Here feature dependencies are included [34] automatically by automatic provision of all feature values considering each feature and dynamically extraction of features. It dominated low-level signal-processing features in most music information retrieval research. J Symbolic has 111 implemented high-level features packaged for music representation. Other 49 features are designed and implemented to get more insights whereas previous MIR research does not consider and extract the features and many of them are completely novel in terms of feature analysis.

The key target of our research was to classify songs depending on features. In our software package, we used weka [33], which has a vast collection of machine learning algorithms for data mining tasks. The algorithms could be applied directly to a dataset as well as get command from a Java code. Weka contains the sophisticated tools for data pre-processing as well as for classification, regression, clustering, association rules, visualization and so on. Weka is a tool that can helps to develop new machine learning schemes. As Weka is free and available under the GNU General Public License as well as very portable which helps to incorporate our data and algorithm. As Weka is fully implemented using Java programming language and java virtual machine, it can run on almost any up-to-date computing platform which contains a wide-ranging collection of data preprocessing and modeling methods. Its user experience is soothing and helps the beginners to work with basic data mining tasks [36] [37]. Weka can facilitate standard data mining tasks and all of Weka's techniques are established which is based on the assumption, this depicts that the data is structured as a single flat file or relation, where every data point are described by a static number of attributes which can be usually numeric or nominal attributes, whereas some other attribute types are also supported.

IV. Result and Discussion

The experiments have been carried out on a set of 1200 songs in WAV format (44K, 16-bit), from four Bengali music genres – Band, "Adhunik", Folk and "Rabindrasangeet", as introduced earlier. From each genre of 300 songs, 50 were randomly chosen as the test set using the program. Then features were extracted from the training set of 250 songs and the test set of 50 songs. Then the four selected Classifiers - J48 Decision Tree, Support Vector Machine, Naïve Bayes, and K Nearest Neighbors were used to classify the training set and derive the decisions as depicted in the method in Figure 1. The decision was then mapped with the test set to find out the percentage accuracy.

TABLE I. Band and Rabindra Sangeet Binary Classification Without Continuity Feature

J48			**SVM**			**NB**		
93%			100%			86%		
	B	***R***		***B***	***R***		***B***	***R***
B	46	4	***B***	50	0	***B***	43	7
R	3	47	***R***	0	50	***R***	7	43

[a] Band => B, Rabindra Sangeet => R

First, we applied a binary classification scheme on the Band and Rabindra sangeet genres. As these two genres are quite distinct in styles, the binary classification yielded impressive results (Table-I). SVM fetched 100% accuracy, J48 performed quiet well with 93% accuracy. However, the accuracy of Naïve Bayes decreased with the continuity feature. This is due to the fact that Naïve Bayes holds the assumption that features are independent of each other which is not true with the continuity feature.

Next, we applied the binary classification on the “Adhunik” and Folk Music genres (Table-II). The results are quite impressive, though the classification results are not as good as the Band-Rabindra sangeet classification due to some commonalities. Band and Rabindra sangeet music are much more different categories compared to “Adhunik” and Folk music categories. As a result, in case of Band-Rabindra sangeet classification we got more that 90% accuracy whereas accuracy of “Adhunik”-Folk was comparatively less.

TABLE II. “ADHUNIK” AND FOLK BINARY CLASSIFICATION WITH CONTINUITY FEATURE

J48			**SVM**			**NB**		
85%			92%			72%		
	A	***F***		***A***	***F***		***A***	***F***
A	42	8	***A***	46	4	***A***	45	5
F	7	43	***F***	4	46	***F***	23	27

[b] Adhunik => A, Folk => F

TABLE III. “ADHUNIK” AND FOLK BINARY CLASSIFICATION WITHOUT CONTINUITY FEATURE

J48			**SVM**			**NB**		
84%			91%			71%		
	A	***F***		***A***	***F***		***A***	***F***
A	43	7	***A***	45	5	***A***	43	7
F	9	41	***F***	4	46	***F***	22	28

[c] Adhunik => A, Folk => F

TABLE IV. BAND AND FOLK BINARY CLASSIFICATION WITH CONTINUITY FEATURE

J48			**SVM**			**NB**		
93%			96%			83%		
	B	***F***		***B***	***F***		***B***	***F***
B	46	4	***B***	49	1	***B***	36	14
F	3	47	***F***	3	47	***F***	3	47

[d] Band => B, Folk => F

TABLE V. BAND AND FOLK BINARY CLASSIFICATION WITHOUT CONTINUITY FEATURE

J48			**SVM**			**NB**		
88%			95%			80%		
	B	***F***		***B***	***F***		***B***	***F***
B	41	9	***B***	49	1	***B***	35	15
F	3	47	***F***	4	46	***F***	5	45

[e] Band => B, Folk => F

TABLE VI. “ADHUNIK” AND BAND BINARY CLASSIFICATION WITH CONTINUITY FEATURE

J48			**SVM**			**NB**		
91%			96%			78%		
	A	***B***		***A***	***B***		***A***	***B***
A	46	4	***A***	49	1	***A***	35	15
B	5	45	***B***	3	47	***B***	7	43

[f] Adhunik => A, Band => B

With both J48 and SVM continuity feature gives slightly better performance than without the continuity feature (Table-II and Table-III). And again, it can be seen that with Naïve Bayes the performance is decreasing for above mentioned reason.

Next pair we compared was Band and Folk. As they share a commonality in case of rhythm and pitch it can be noted that the accuracy of classifying them is even better than Band and “Adhunik” (Table-VI and Table-VII). Here with continuity feature we get much better performance than without, e.g., 93% compared to 88% in case of J48(Table-IV). And again it can be seen that Naïve Bayes is giving worse performance (Table-V).

TABLE VII. “ADHUNIK” AND BAND BINARY CLASSIFICATION WITHOUT CONTINUITY FEATURE

J48			**SVM**			**NB**		
86%			96%			74%		
	A	***B***		***A***	***B***		***A***	***B***
A	43	7	***A***	48	2	***A***	35	15
B	7	43	***B***	2	48	***B***	11	39

[g] Adhunik => A, Band => B

“Adhunik” and Band were the next 2 candidates. In many cases band and “Adhunik” songs share commonalities which leads to comparatively poorer performance. In this pair with continuity feature J48 is giving 5% higher performance than without. SVM gives identical performance in both cases. Here again Naïve Bayes gives much poorer performance without continuity feature for the reasons explained above.

TABLE VIII. FOLK AND RABINDRA SANGEET BINARY CLASSIFICATION WITH CONTINUITY FEATURE

J48			**SVM**			**NB**		
88%			95%			72%		
	F	***R***		***F***	***R***		***F***	***R***
F	45	5	***F***	47	3	***F***	32	18
R	7	43	***R***	2	48	***R***	10	40

[h] Folk => F, Rabindra Sangeet => R

TABLE IX. FOLK AND RABINDRA SANGEET BINARY CLASSIFICATION WITHOUT CONTINUITY FEATURE

J48			**SVM**			**NB**		
88%			93%			83%		
	F	***R***		***F***	***R***		***F***	***R***
F	43	7	***F***	45	5	***F***	40	10
R	5	45	***R***	2	48	***R***	7	43

[i] Folk => F, Rabindra Sangeet => R

TABLE X. “ADHUNIK” AND RABINDRA SANGEET BINARY CLASSIFICATION WITH CONTINUITY FEATURE

J48	**SVM**	**NB**

87%			94%			74%		
	A	***R***		***A***	***R***		***A***	***R***
A	47	3	***A***	46	4	***A***	37	13
R	6	44	***R***	0	50	***R***	13	37

j. Adhunik => A, Rabindra Sangeet => R

Folk and Rabindra sangeet were tested next (Table-VIII and Table-IX). It can be seen that with J48, with or without continuity feature, the performance remains same. But with continuity feature the performance of SVM slightly increases. Also, before, the performance of Naïve Bayes decreases considerably.

TABLE XI. “ADHUNIK” AND RABINDRA SANGEET BINARY CLASSIFICATION WITHOUT CONTINUITY FEATURE

J48			**SVM**			**NB**		
87%			90%			75%		
	A	***R***		***A***	***R***		***A***	***R***
A	42	8	***A***	44	6	***A***	39	11
R	5	45	***R***	2	48	***R***	14	36

k. Adhunik => A, Rabindra Sangeet => R

TABLE XII. ALL GENRE CLASSIFICATION WITH CONTINUITY FEATURE

J48					**SVM**					**NB**				
72.5%					86%					51%				
	A	***B***	***F***	***R***		***A***	***B***	***F***	***R***		***A***	***B***	***F***	***R***
A	44	1	5	0	***A***	49	0	0	1	***A***	32	3	11	4
B	2	34	8	6	***B***	0	41	5	4	***B***	0	28	17	5
F	5	4	34	7	***F***	4	1	42	3	***F***	3	7	35	5
R	6	6	4	34	***R***	2	4	4	40	***R***	10	16	17	7

l. Adhunik => A, Band => B, Folk => F, Rabindra Sangeet => R

TABLE XIII. ALL GENRE CLASSIFICATION WITHOUT CONTINUITY FEATURE

J48					**SVM**					**NB**				
70.5%					85%					58%				
	A	***B***	***F***	***R***		***A***	***B***	***F***	***R***		***A***	***B***	***F***	***R***
A	41	1	4	4	***A***	49	0	0	1	***A***	32	1	13	14
B	1	33	10	6	***B***	0	42	5	3	***B***	0	39	9	2
F	3	8	33	6	***F***	2	2	4	5	***F***	2	12	34	2
R	6	6	4	34	***R***	3	7	2	38	***R***	5	18	16	11

m. Adhunik => A, Band => B, Folk => F, Rabindra Sangeet => R

Lastly, we tested “Adhunik” and Rabindra sangeet. In this case J48 gives identical performance with and without continuity feature. But SVM shows 4% increase in performance with continuity feature than without it. In this case, Naïve Bayes with continuity feature shows very slight 1% decrease of performance (Table-X and Table-XI).

Finally, we ran the classification algorithm on all the genres (Table-XII and Table-XIII). As observed J48 gives more or less good performance. It exhibits a 2% increase of performance with the continuity feature enabled. SVM yields the best performance as before, with 1% better accuracy with the continuity feature than without. Naïve Bayes’s performance falls ~7% with the continuity feature, as expected.

V. CONCLUSION

In this paper, we presented an enhanced method to automatically classify music into different genre using a machine learning approach, and the results of the application of the proposed scheme to the classification of a large set of The Bangla music. From the results it is clearly seen that the Support Vector Machine gives the best classification regarding music genre differentiation and Naïve Bayes gives the worst. We also introduced a new feature, which we call the continuity feature. We found that both J48 Decision tree and SVM give better result in most cases with the continuity feature compared to without it. Hence the continuity feature can be regarded as a viable addition to feature extraction processes.

REFERENCES

[1] H.-R. Thompson, “Bengali,” London Oriental and African Language Library, 2012.

[2] T. E. of E. Britannica, “Bengali language,” *Encyclopædia Britannica*, 28-Jul-2017. [Online]. Available: https://www.britannica.com/topic/Bengali-language. [Accessed: 11-Mar-2019].

[3] “Languages of the World,” *Ethnologue*. [Online]. Available: http://www.ethnologue.com/. [Accessed: 11-Mar-2019].

[4] S. Subhan, Heritage language maintenance among Bangladeshi immigrants in Toronto. Northern Arizona University.

[5] B. V. D. Linden, Arnold Bake: A Life with South Asian Music. Routledge, 2018.

[6] A. Halder, “The top five music streaming apps in town,” *The Daily Star*, 17-Apr-2017. [Online]. Available: http://www.thedailystar.net/bytes/the-top-five-music-streaming-apps-town-1392172. [Accessed: 11-Mar-2019].

[7] “Music of Bengal,” *Wikipedia*, 02-Mar-2019. [Online]. Available: http://en.wikipedia.org/wiki/Music_of_Bengal. [Accessed: 11-Mar-2019]..

[8] S. Dasthakur, “Rabindrasangeet and Modern Bengali Subjectivity,” Sophia Studies in Cross-cultural Philosophy of Traditions and Cultures Rabindranath Tagore in the 21st Century, pp. 203–224, 2014..

[9] D. Chathuranga and L. Jayaratne, “Automatic Music Genre Classification of Audio Signals with Machine Learning Approaches,” *GSTF Journal on Computing (JoC*), vol. 3, no. 2, 2013.

[10] C. N. S. Jr., A. L. Koerich, and C. A. A. Kaestner, “A machine learning approach to automatic music genre classification,” *Jou rnal of the Brazilian Computer Society*, vol. 14, no. 3, pp. 7–18, 2008.

[11] D. Pye, “International Conference on Acoustics, Speech, and Signal Processing,” in *ICASSP Proceedings of the Acoustics, Speech, and Signal Processing*, 2000, pp. 2437–2440.

[12] A. Rosner and B. Kostek, “Automatic music genre classification based on musical instrument track separation,” *Journal of Intelligent Information Systems*, vol. 50, no. 2, pp. 363–384, 2017.

[13] E. Guaus and P. Herrera, "A basic system for music genre classification", in *8th International Conference on Music Information Retrieval*, Viena, 2007.

[14] J. Barbedo and A. Lopes, "Automatic Genre Classification of Musical Signals", *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, 2006. Available: 10.1155/2007/64960.

[15] G. Tzanetakis and P. Cook, “MARSYAS: a framework for audio analysis,” *Organised Sound*, vol. 4, no. 3, pp. 169–175, 2000.

[16] G. M. Kalliris, A. D. Charalampos , and U. Christian, “Special Issue on Intelligent Audio Processing, Semantics, and Interaction,” *Journal of the Audio Engineering Society*, vol. 64, no. 7, pp. 464–465, 2016..

[17] S. Stober and A. Nürnberger, "Adaptive music retrieval–a state of the art", *Multimedia Tools and Applications*, vol. 65, no. 3, pp. 467-494, 2012. Available: 10.1007/s11042-012-1042-z.

[18] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl, “Aggregate features and ADABOOST for music classification,” *Machine Learning*, vol. 65, no. 2-3, pp. 473–484, 2006.

[19] S. Ntalampiras, “A Novel Holistic Modeling Approach for Generalized Sound Recognition*,” IEEE Signal Processing Letters*, vol. 20, no. 2, pp. 185–188, 2013.

[20] G. Guo and S. Li, “Content-based audio classification and retrieval by support vector machines,” *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 209–215, 2003.

[21] L. R. Aguiar, M. G. Y. Costa, and N. C. Silla, “Exploring Data Augmentation to Improve Music Genre Classification with ConvNets,” 2018 *International Joint Conference on Neural Networks (IJCNN)*, 2018.

[22] S. Das and A. K. Kolya, “A Theoretic Approach to Music Genre Recognition from Musical Features Using Single-Layer Feedforward Neural Network,” *Advances in Intelligent Systems and Computing Emerging Technologies in Data Mining and Information Security*, pp. 145–155, 2018.

[23] A. Rahman, “Bangla Music Genre Classification,” *Multidisciplinary Computational Intelligence Techniques*, pp. 124–138, 2012.

[24] R. Mittra and V. Varadarajan, “A technique for solving 2D method-of-moments problems involving large scatterers,” *Microwave and Optical Technology Letters*, vol. 8, no. 3, pp. 127–132, 1995.

[25] G. Tzanetakis, A. Ermolinskyi, and P. Cook, “Pitch Histograms in Audio and Symbolic Music Information Retrieval,” *Journal of New Music Research*, vol. 32, no. 2, pp. 143–152, 2003.

[26] E. D. Scheirer, “Tempo and beat analysis of acoustic musical signals,” *The Journal of the Acoustical Society of America*, vol. 103, no. 1, pp. 588–601, 1998.

[27] F. Pachet and P. Roy, “Analytical features: a knowledge-based approach to audio feature generation,” *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 1, 2009.

[28] C. Mckay and I. Fujinaga, “Improving automatic music classification performance by extracting features from different types of data,” *Proceedings of the international conference on Multimedia information retrieval - MIR 10*, 2010.

[29] Q. Liu, A. Sung, and M. Qiao, “Temporal Derivative-Based Spectrum and Mel-Cepstrum Audio Steganalysis,” *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 3, pp. 359–368, 2009.

[30] T. Ayers and B. Nolan, “Developing a Distributed Java-based Speech Recognition Engine,” The ITB Journal, vol. 5, no. 1, 2004.

[31] W. Dai and W. Ji, “A MapReduce Implementation of C4.5 Decision Tree Algorithm,” *International Journal of Database Theory and Application*, vol. 7, no. 1, pp. 49–60, 2014.

[32] T. R. Patil and S. S. Sherekar, “Performance analysis of Naive Bayes and J48 classification algorithm for data classification,” *International journal of computer science and applications*, vol. 6, no. 2, pp. 256–261, Apr. 2013.

[33] G. Holmes, A. Donkin, and I. H. Witten, *WEKA: a machine learning workbench*. Hamilton, N.Z.: Dept. of Computer Science, University of Waikato, 1994.

[34] T. Raissi, A. Tibo, and P. Bientinesi, “Extended Pipeline for Content-Based Feature Engineering in Music Genre Recognition,” 2018 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[35] B. Jamie, and U. C. E. B. Conservatoire. “Libxtract: a Lightweight Library for audio Feature Extraction.” In *ICMC*. 2007.

[36] C. McKay and I. Fujinaga. “Improving automatic music classification performance by extracting features from different types of data.” in *Multimedia Information Retrieval*, pp. 257–266, 2010.

[37] C. McKay and I. Fujinaga. Overview of Omen.” in *Multimedia Information Retrieval*, pp. 7–12, 2006.

Abhijit Bhowmik completed his B.Sc. in Computer Science & Engineering and M.Sc. in Computer Science from the American International University–Bangladesh (AIUB). Currently, he is working as a Senior Assistant Professor and Special Assistant, Office of Student Affairs (OSA) in the Department of Computer Science, AIUB. His research interests include wireless sensor networks, video on demand, software engineering, mobile & multimedia communication and data mining.

AZM Ehtesham Chowdhury completed B.Sc. in Computer Science & Engineering and M.Sc. in Computer Science from American International University-Bangladesh, Dhaka, Bangladesh. Currently, he is working as an Assistant Professor in the Department of Computer Science, AIUB. His current research interest includes Data Science, Data Mining, Software Engineering, Intelligent Systems. Computer Vision, Pattern Recognition and Human Computer Interaction