Tree-based classification of tabla strokes

Subodh Deolekar[1,*] and Siby Abraham[2]

[1]Department of Computer Science, University of Mumbai, Mumbai 400 098, India
[2]Department of Maths and Stats, G. N. Khalsa College, University of Mumbai, Mumbai 400 019, India

This study attempts to validate the effectiveness of tree classifiers to classify tabla strokes especially the ones which overlap in nature. It uses decision tree, ID3 and random forest as classifiers. A custom made data set of 650 samples of 13 different tabla strokes were used for experimental purpose. Thirty-one different features with their mean and variances were extracted for classification. Three data sets consisting of 21,361, 18,802 and 19,543 instances respectively, were used for the purpose. Validation was done using measures like receiver operating characteristic curve and accuracy. All the classifiers showed excellent results with random forest outperforming the other two. The effectiveness of random forest in classifying strokes which overlap in nature is evaluated by comparing the known results with multi-layer perceptron.

Keywords: Classification, decision tree, random forest, tree classifiers, tabla strokes.

CLASSIFICATION is the process which assigns a specific item to one of the categories or classes specified based on its features or properties. In machine learning, classification is considered to be a task to predict the value of one or more outcomes. The real task in classification is to find a relationship among features and its associated classes. There are various categories of classification which include linear classifiers, support vector machines, quadratic classifiers, kernel estimation, decision trees and neural networks[1]. Among these, tree-based classifiers are commonly used for developing prediction algorithms for a target variable[2]. Such classifiers construct a root node with a population of branches which are comprised of internal nodes and leaf nodes.

Tree classifiers aim to partition datasets into groups of similar nature. They are said to be very effective methods of supervised learning, which lead to generate unique solutions. In cases where impurity exists in the data and where there are traces of one class overstepping into another, tree classifiers are best suited. Unlike linear models, they map nonlinear relationships quite well. Following are some of the advantages of using tree classifiers:

- Comprehensive behaviour: Tree classifiers are the best predictive models as they extensively examine each possible outcome. The partition of data is done in a much deeper way compared to other classification techniques.
- No need for tuning of parameter set: Normalization or scaling of the parameter set can be avoided. Where most classification models fail to handle nonlinearity of parameters, tree classifiers outperform with such data.
- Easy to interpret: Tree classifiers make a clear distinction with all possible solutions which are represented by different nodes. A graphical view of classification based on rules and parameter set makes decision making less ambiguous.
- Easily deal with outliers: Tree classifiers are flexible in handling data items with some missing feature values. Also, splitting of sub-trees is based on split range and not based on absolute values which depict non-sensitivity towards outliers.

Tree algorithms have been extensively used for classification of various tasks in diverse domains. This includes characterizing smoking patterns of older adults[3], analysing students' achievements in distance-learning mode to improve online teaching[4], identifying core factors of production control in agricultural investment decisions[5], understanding behavioural patterns of different kinds of astronomical objects[6], analysing financial data[7], text classification[8] and many more. In the domain of music, there are various applications of tree classifiers which consider classification of various instruments[9] and speech/music classification and segmentation[10].

This article is an attempt to measure the effectiveness of tree algorithms to classify strokes of a musical instrument called tabla. The Indian percussion instrument, which is used for solo performance as well as for an accompanied performance, often reports of cases where impurities exist in its data. In addition, one can find genuinely perceived instances where one stroke seemed to sidestep or overstep into the territory of another stroke making classification difficult. This makes classification of tabla strokes a fit case for a tree classifier. Though literature reports of a similar work of classification of tabla strokes using other classifiers[11], the present work is different in the sense that it faces the inherent constraint of classification of tabla strokes head on using tree classifiers.

Tabla instrument

Tabla plays an important role in Hindustani music which is the most popular and culturally oriented tradition of Indian music. It is used to provide rhythmic pattern in

*For correspondence. (e-mail: subodhdeolekar@gmail.com)

music compositions along with other Indian percussion instruments like Pakhwaj, Mridangam, Dholak and Dholki. Because of refined tonal quality and sophistication, tabla came to the forefront as a solo performing instrument, leaving behind all other instruments.

Tabla set, which is a combination of two drums, namely *bayaan* (left hand) and *dayaan* (right hand), as shown in Figure 1, produces strokes called *bol*s. *Bol*s, which are mnemonic syllables, can be produced using different techniques based on diverse musical traditions[12]. Table 1 gives the list of basic strokes played on *dayaan* and *bayaan* individually and collectively.

Transcription of tabla strokes is said to be a challenging task for computer scientists[13]. Unlike western rhythmic instruments (such as drums or congo), where characteristics of strokes are easily detected, tabla strokes are hard to detect. Tabla produces long and resonant *bols* which overlap with successive *bols*. Also, few *bols* which are used to represent rhythmic strokes depend on the specific context and may vary accordingly. There exist *bols*, which are different but sound similar. For example, *Ti* and *Ta* are two *bols*, which are played with subtle changes in the fingering style but are hardly identified correctly by even experienced listeners.

Figure 1. Tabla set.

Table 1. List of basic tabla strokes

Dayaan *bols*	Bayaan *bols*	Both together
Ge (गे/घे)	Na (ना)	Dha (धा)
Ka (क/की)	Tu (तू)	Dhin (धीं)
	Ti (ति)	Tin (तीं)
	Ta (ट)	
	N (न)	
	T (त/र)	
	Tra (त्र)	
	Din (दिं)	

Decision tree algorithms

Decision tree algorithms are flexible, powerful and offer high performance for prediction problems. Without any pre-defined assumption and controlled parameters, these algorithms are capable of fitting large amount of data. In principle, decision tree algorithms categorize data by considering their attribute values. Each parent node in a tree represents a test on an attribute value and child node represents corresponding classes.

Decision tree

Decision tree is a supervised learning algorithm which has pre-defined target variables. It works for both categorical and continuous input/output variables. The split of sample at root node or at the internal subsequent nodes is based on characteristics of child node. These characteristics are defined by different variables like entropy and Gini index.

In information theory, degree of disorganization in a system is called entropy. It can also be considered as a degree of randomness of elements or a measure of impurity. Mathematically, it can be calculated as

$$H(x) = -\sum p(x) \log p(x),$$

where, $p(x)$, probability of target variable x, is actually transmitted. On the other hand, Gini index performs only binary splits with categorical targets like 'success' or 'failure'. Higher value of Gini represents higher homogeneity. It is defined as

$$\text{Gini} = \sum_{i=1}^{C} (P_i)^2,$$

where C is the number of classes and P_i is the probability of ith target variable for $i \in \{1, 2, \ldots, C\}$.

ID3

Iterative Dichotomizer 3 (ID3) algorithm classifies data using attribute values. A tree consists of decision nodes and decision leafs which produce a homogeneous result. It is based on top–down greedy search to test each attribute at every node of a tree. It calculates the entropy value for each attribute[14]. It then splits the set into subsets using that attribute for which the information gain is maximum. Information gain is the entropy of the parent node minus the entropy of the child nodes and is given by

$$\text{Information gain} = H(X) - \sum_{i=1}^{C} (P_i * H(X_i)),$$

where $H(X)$ is the entropy of the parent node, $H(X_i)$ the entropy of the child node and P_i is the probability of ith

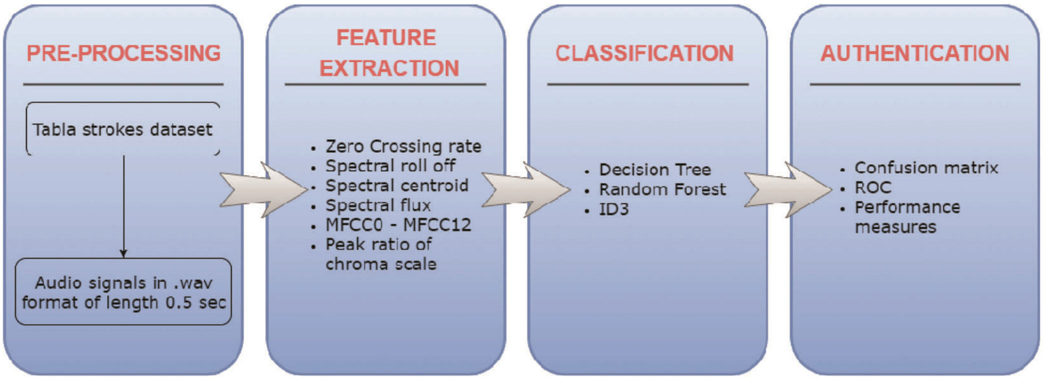

Figure 2. Overview of proposed methodology.

target variable for $i \in \{1, 2, \ldots, C\}$, C being the number of classes. In this way, ID3 tree at every stage selects the node that gives the best information gain, the one with least impurity[15].

Random forest

Random forest is a classifier which comprises a set of weak, weakly correlated and non-biased classifiers, namely the decision trees. It has been shown that random forest performs equally well or better than other methods on a diverse set of problems. It has been widely used in classification problems as diverse as bioinformatics[16], medicine[17], transportation safety[18] and customer behaviour[19].

Random forest offers a useful feature that improves our understanding of a classification problem under scrutiny. It gives an estimate of the importance of each attribute for final prediction. It is often used for analysis when both classifier and identification of important variables are goals of the study.

Random forest collects votes from different decision trees which are randomly selected from training set data and decides the final class of test data. This is helpful for finding accurate results because a single tree might lead to a noise, but a set of decision trees will reduce the noise.

Methodology

The article proposes a methodology which classifies tabla strokes, wherein there exists an overlap among different classes, making classification difficult. The methodology, the overview of which is shown in Figure 2, consists of four steps, namely pre-processing, feature extraction, classification and authentication.

Pre-processing

As most of the publicly available datasets lack authenticity, the methodology uses datasets generated exclusively for this work. Samples of tabla strokes were recorded in the required audio format. These were then clipped to specific time duration to have the same length for each stroke. Initially the duration was fixed to be 0.5 sec. Those *bols*, whose resonance does not last for such a long duration were clipped to 0.2–0.3 sec. A sample of different *bols* used in the work is shown in the waveform in Figure 3.

Feature extraction

Different sets of spectral and temporal features were considered for analysing different contents of the tabla strokes for classification purpose[20]. Features like zero-crossing, spectral centroid, spectral roll-off, spectral flux, mel-frequency cepstral coefficients (MFCC0–MFCC12), and chroma frequencies with their mean and standard deviation were extracted from the audio file of individual strokes. A short description of these features is given below:

Zero crossing rate: It represents the number of times the waveform crosses zero axis. It usually has higher values for high percussion sounds.

Spectral centroid: It is the weighted mean of the frequencies present in the music piece. The value of it changes according to the accumulation of frequencies.

Spectral roll-off: It is a measure of the shape of a signal. It represents the frequency at which high frequencies decline to zero.

Spectral flux: It is a measure of change in the power spectrum of a signal. It is used to determine the timbre of an audio data.

Mel-frequency cepstral coefficient (MFCC): MFCC represents a set of short-term power spectrum characteristics of the music piece and has been used in state-of-the-art recognition and music categorization techniques.

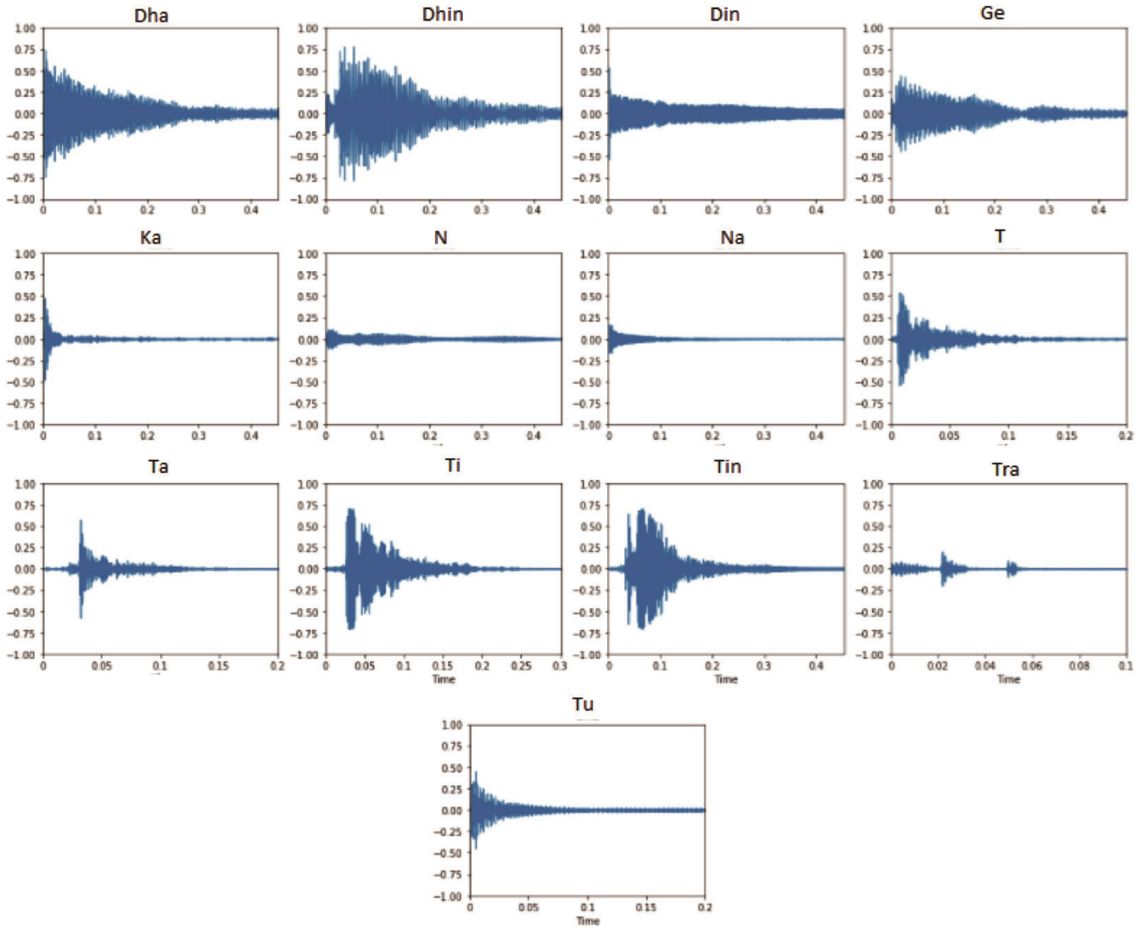

Figure 3. Sample tabla strokes.

Altogether, 13 coefficients from MFCC0 to MFCC12 are identified for this feature.

Chroma frequencies: Chroma frequency vector discretizes the spectrum into chromatic keys and represents the presence of each key. It provides a robust way to describe a similarity measure between music pieces.

These low level musical features are helpful in the classification process.

Classification

We have used three classifiers, namely decision tree, ID3 and random forest to classify tabla strokes. The choice of tree classifiers is deliberate since it works well with missing attribute values and nonlinear working set[21]. For decision tree, the measures used for selecting input variables are entropy and Gini and split on each node is binary. ID3 classifier makes use of information gain based on the training samples and builds the tree. For random forest, the output is determined based on the majority votes of the trees. As a result, random forest uses a large number of trees collectively.

Authentication

The methodology has used measures like ROC curve, tree structure based on Gini/entropy and accuracy to authenticate the performance of classifiers. Graphical view of the decision tree makes it easy to interpret results along with other performance measures.

Experimental results

Detailed experiments were conducted using different tabla strokes. The tabla set used for recording was tuned to C# scale. The original data comprised 650 strokes which were split into 50 samples each of 13 tabla strokes recorded from different expert tabla players in order to have diversity. The recording was done using a microphone input in an environment with less noise. After feature extraction, the three datasets of moderate size consisting of 21,361, 18,802 and 19,543 instances respectively, were used for classification. For each classifier, the data were split into 70% for training and 30% for testing. The sample rate was kept as 44,100 Hz.

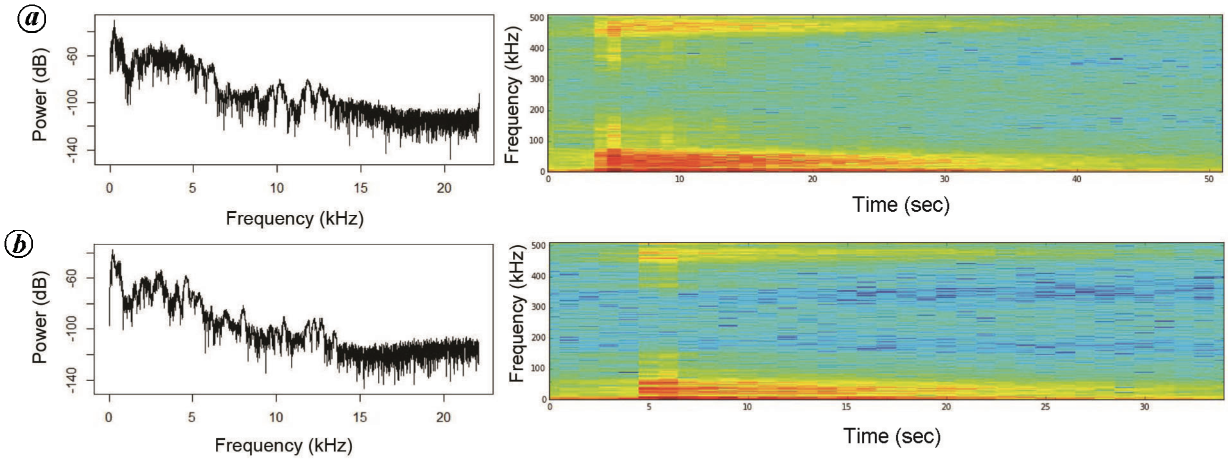

Figure 4. Waveform and spectrogram of two bols: (***a***) *ti* and (***b***) *ta* respectively.

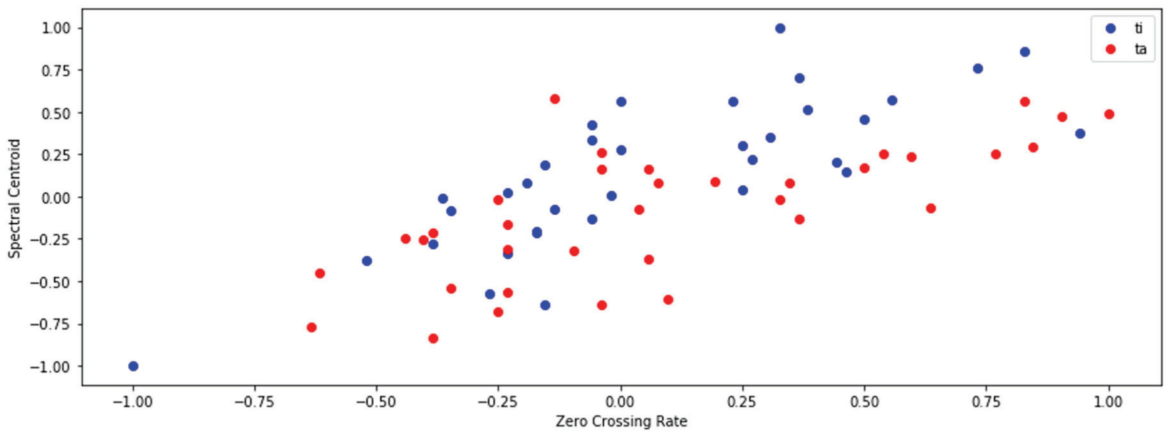

Figure 5. Spectral centroid and zero crossing rate for two bols *ti* and *ta* respectively.

Pre-processing work was done with the help of Audacity[22] software. The features were extracted using MARSYAS (music analysis, retrieval and synthesis for audio signals) framework[23]. The feature values extracted were stored in .csv file for further processing. Python was used to implement tree algorithms. Intel(R) Core i5-CPU with 1.70 GHz with 4 GB RAM was used for the experimental purpose.

Feature extraction

The experiment results validated the overlapping nature of original tabla strokes. Features extracted demonstrate this clearly. For example, spectral centroid and zero crossing rate of audio samples of two bols *ti* and *ta* were considered for a representative nature. Figure 4 *a* and *b* show the power spectrum of these two sample bols and Figure 5 shows the overlap between these two features.

Similar features were extracted for others bols like *dha*, *dhin*, *tin*, *din* where we found less overlap. Figure 6 *a* and *b* shows these less overlapping classes.

Decision tree representation

The work described used tree classifiers, viz. decision tree, ID3 and random forest on the extracted features set. Figure 7 shows a sample decision tree generated using Gini index criteria and Figure 8 shows the decision tree generated from entropy criteria using the dataset 1 in both cases.

It can be observed that the decision tree splits the node 2-way. It grows depth-wise with Gini index and expands depth-wise with entropy. That means, Gini index favours the larger partition, whereas entropy favours smaller partitions with many distinct values. The values for Gini index and entropy become zero when all the observations belong to same class label.

In Figure 7 we see that the PeakRatio_Minimum_Chroma_A feature is used for the initial split with Gini index value as 0.917. When we used entropy for the base calculation, a wider range of results was observed, whereas the Gini index capped at one. In Figure 8 PeakRatio_Average_Chroma_A is chosen for the split and has entropy value as 3.585.

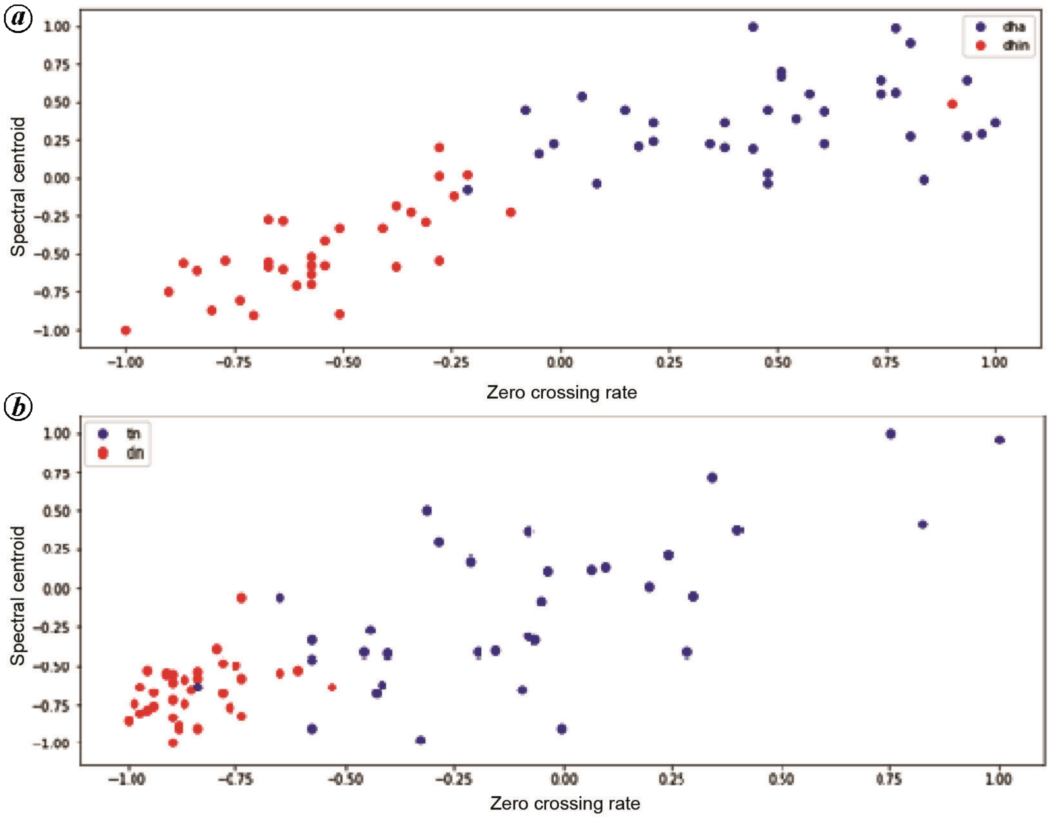

Figure 6. Spectral centroid and zero crossing rate for: (***a***) *dha* and *dhin*; (***b***) *tin* and *din* strokes.

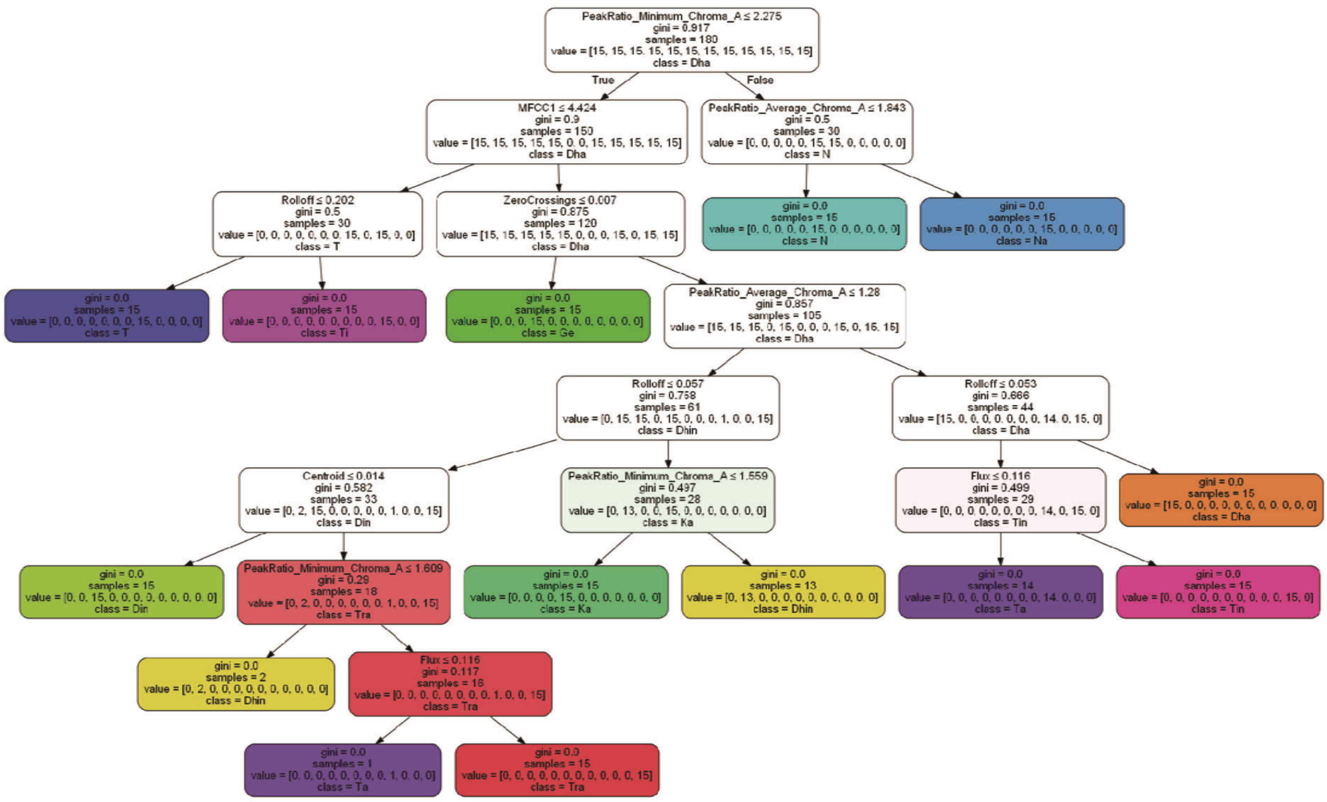

Figure 7. Decision tree predicting the tabla strokes based on Gini index value.

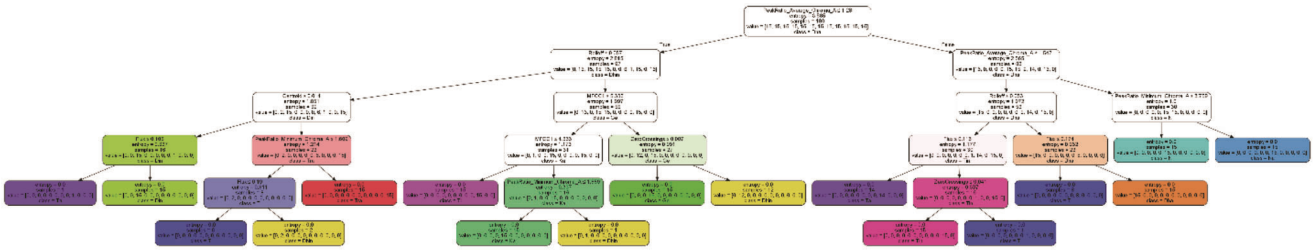

Figure 8. Decision tree predicting the tabla strokes based on entropy value.

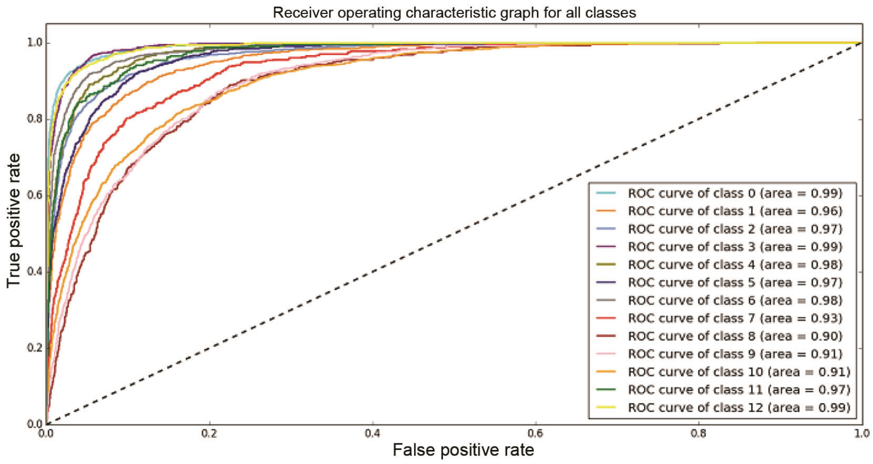

Figure 9. ROC curve for all the classes.

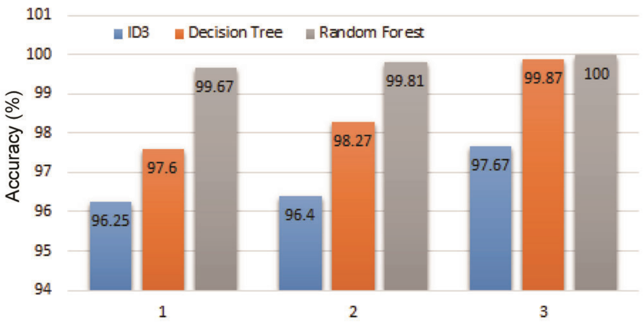

Figure 10. Accuracy comparison among ID3, decision tree and random forest algorithm.

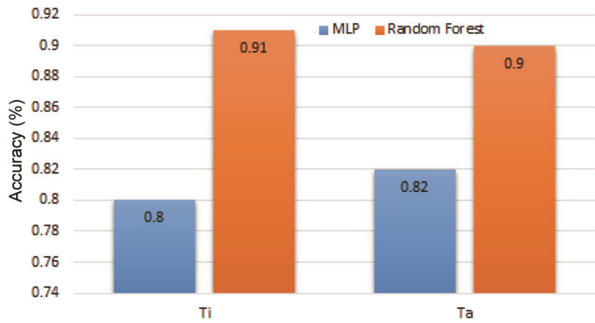

Figure 11. Comparison of accuracy of *ti* and *ta* strokes for MLP and random forest algorithm.

Classifier performance varies depending upon the characteristics of data used in the classification process. We tried to perform various empirical evaluation strategies to compare classifier performance.

ROC

Receiver operating characteristic (ROC) curves were used to evaluate the tradeoff between true and false positive rates of decision tree classifier. It is considered to be a plot of sensitivity versus specificity for all possible thresholds of different classes.

Figure 9 shows the ROC curve for the random forest classifier. It is observed that random forest performs well by classifying all the possible tabla strokes with accuracy close to 100%. It was found that with overlapping bols *ti* and *ta*, which are represented by class 8 and 9, the accuracy was more than 90%.

Accuracy measurement

Experimental results showed that random forest outperformed other classifiers in terms of accuracy with all three datasets. While random forest gave an accuracy up to 100%, ID3 (97%) and decision tree (99%) have also performed well. Figure 10 illustrates these findings.

Accuracy comparison of overlapping strokes

We tried to compare the results of random forest with earlier work with multilayer perceptron (MLP) classifier[11] by considering overlapping tabla strokes of *ti* and *ta*. It was observed that random forest exhibited a much better accuracy of around 91% while that of MLP hovered around 80–82%, showing the effectiveness of tree algorithms, especially the random forest in properly classifying overlapping class instances. Figure 11 gives an idea about the supremacy of random forest as a tree classifier for tabla strokes classification compared to the available study in literature.

All these results show that random forest algorithm is well suited to classify tabla strokes and works significantly better than the other two tree classifiers, namely decision tree and ID3. The performance of random forest in classifying overlapping classes was also showed to be better than the one offered by other classifiers.

Conclusion

The proposed work highlights the effectiveness of tree classifiers in classifying tabla strokes. It uses decision tree, ID3 and random forest as classifiers. Detailed experiments were conducted using three different data sets. Thirty-one different features of the strokes with their mean and variances were extracted. The classification results show that random forest outperforms the other two classifiers. The performance of random forest was compared with other known classifiers and results showed that random forest is better suited to classify overlapping classes. The work can be extended to classify strokes of other percussion instruments also.

1. Sagar, S. N., A comparative study of classification techniques in data mining algorithms. *Orient. J. Comput. Sci. Technol.*, 2015, **8**(1), 13–19.
2. Yan-yan, S. and Ying, L. Decision tree methods: applications for classification and prediction. *Shanghai Arch. Psychiatry*, 2015, **27**(2), 130–135.
3. Moon, S. S., Kang, S.-Y., Jitpitaklert, W. and Kim, S. B., Decision tree models for characterizing smoking patterns of older adults. *Expert Syst. Appl.*, 2012, **39**(1), 445–451.
4. Wilton, W. T., Fok, H. C., Yi, J., Li, S., Au Yeung, H. H., Ying, W. and Fang, L., Data mining application of decision trees for student profiling at the Open University of China. In IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, Beijing, 2014, pp. 732–738.
5. Lu, J., Liu, Y. and Li, X., The decision tree application in agricultural development. In *Artificial Intelligence and Computational Intelligence* (eds Deng, H. *et al.*), AICI 2011, Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, vol. 7002.
6. Franco-Arcega, A., Flores-Flores, L. G. and Gabbasov, R. F., Application of decision trees for classifying astronomical objects. In 12th Mexican International Conference on Artificial Intelligence, Mexico City, 2013, pp. 181–186.
7. Yu, G. and Wenjuan, G., Decision tree method in financial analysis of listed logistics companies. In International Conference on Intelligent Computation Technology and Automation, Changsha, 2010, pp. 1101–1106.
8. Srinivasan Ramaswamy, Multiclass text classification a decision tree based SVM Approach, CS294 Practical Machine Learning Project, Citeseer, 2006.
9. Kursa, M., Rudnicki, W., Wieczorkowska, A., Kubera, E. and Kubik-Komar, A., Musical instruments in random forest. In *Foundations of Intelligent Systems* (eds Rauch, J. *et al.*), ISMIS 2009. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, vol. 5722.
10. Lavner, Y. and Ruinskiy, D., A decision-tree-based algorithm for speech/music classification and segmentation. *EURASIP J. Audio, Speech, Music Processing*, 2009.
11. Deolekar, S. and Abraham, S., Classification of tabla strokes using neural network. In *Computational Intelligence in Data Mining – Volume 1* (eds Behera, H. and Mohapatra, D.), Advances in Intelligent Systems and Computing, Springer, New Delhi, 2016, vol. 410.
12. Courtney, D., *Learning the Tabla*, M. Bay Publications, 2001, vol. 2, ISBN 0786607815.
13. Chordia, P., Segmentation and recognition of tabla strokes. International Conference on Music Information Retrieval, 2005, pp. 107–114.
14. Navada, A., Ansari, A. N., Patil, S. and Sonkamble, B. A., Overview of use of decision tree algorithms in machine learning. In IEEE Control and System Graduate Research Colloquium (ed. Shah Alam), 2011, pp. 37–42.
15. Jin, C., De-lin, L. and Fen-Xiang, M., An improved ID3 decision tree algorithm. In the Proceedings of 4th International Conference on Computer Science and Education, 2009, pp. 127–130.
16. Qi, Y., Random forest for bioinformatics. In *Ensemble Machine Learning* (eds Zhang, C. and Ma, Y.), Springer, Boston, MA, 2012, pp. 307–323.
17. Dubrava, S. *et al.*, Using random forest models to identify correlates of a diabetic peripheral neuropathy diagnosis from electronic health record data. *Pain Med.*, 2017, **34**, 107–115.
18. Saha, D., Alluri, P. and Gan, A., A random forests approach to prioritize Highway Safety Manual (HSM) variables for data collection. *J. Adv. Transport.*, 2016, **50**, 522–540.
19. Armando Vieira, Predicting online user behaviour using deep learning algorithms. Computing Research Repository – arXiv.org, 2015, http://arxiv.org/abs/1511.06247.
20. Brent, W., Physical and perceptual aspects of percussive timbre, UCSanDiego Electronic Theses and Dissertations, 2010.
21. Chattamvelli, R., *Data Mining Methods*, Alpha Science International, Oxford, UK, 2009.
22. Audacity® software is copyright © 1999–2017 Audacity Team. website: https://audacityteam.org/. It is free software distributed under the terms of the GNU General Public License. The name Audacity® is a registered trademark of Dominic Mazzoni.
23. George Tzanetakis and Perry Cook, MARSYAS: a framework for audio analysis. *Org. Sound*, 1999, **4**(3), 169–175.

ACKNOWLEDGEMENTS. We thank Mr Arun Kundekar and his team for their help in recording tabla strokes and offering their expertise in the domain.

Received 28 December 2017; revised accepted 13 May 2018

doi: 10.18520/cs/v115/i9/1724-1731