Segregation of Songs and Instrumentals - a Precursor to Voice/accompaniment Separation From Songs in Noisy Scenario

Submitted: 20th October 2018; accepted: 2nd June 2020

Himadri Mukherjee, Sk Md Obaidullah, K.C. Santosh, Teresa Gonçalves, Santanu Phadikar, Kaushik Roy

DOI: 10.14313/JAMRIS/2-2020/23

Abstract:
The music industry has come a long way since its inception. Music producers have also adhered to modern technology to infuse life into their creations. Systems capable of separating sounds based on sources especially vocals from songs have always been a necessity which has gained attention from researchers as well. The challenge of vocal separation elevates even more in the case of the multi-instrument environment. It is essential for a system to be first able to detect that whether a piece of music contains vocals or not prior to attempting source separation. It is also very much challenging to perform source separation from audio which is contaminated with noise. In this paper, such a system is proposed being tested on a database of more than 99 hours of instrumentals and songs. Experiments were performed with both noise free as well as noisy audio clips. Using line spectral frequency-based features, we have obtained the highest accuracies of 99.78% and 99.34% (noise free and noisy scenario respectively) from among six different classifiers, viz. BayesNet, Support Vector Machine, Multi Layer Perceptron, LibLinear, Simple Logistic and Decision Table.

Keywords: *Background track, Vocals, Noisy audio, Line spectral frequency, Framing*

1. Introduction

Technology has had a profound impact in every sphere and the music industry has not been an exception to this. Audio engineers now have various advanced solutions to help them with music production. One of the primary requirements of musicians has always been for such a technology that can enable them to separate background tracks from vocals. This can be extremely helpful for acapella extraction for rearrangements. It can also help musicians in understanding minute technicalities of background tracks who have little audio engineering knowledge. The separation of vocals from music is itself a difficult task which elevates even more in the case songs due to presence of multiple instruments. It is also extremely difficult to separate vocals from clips which has been breathed upon by noise. A system of this sort can also help towards voice activity detection in songs as well and aid the separation of individual instruments in songs for further analysis. It is essential to be able to distinguish instrumentals from songs prior to extracting instrumental portions from the songs and perform any kind of analysis.

In this paper, such a system is proposed which tries to segregate instrumentals and songs from noisy clips using line spectral frequency (LSF)-based features. The system has been pictorially illustrated in Figure 1. It has been tested with multiple feature dimensions and various classifiers whose details are presented in the subsequent paragraphs.

In the rest of the paper, Sections 2, 3 and 4 describe the related works, datasets and proposed methodology, respectively. Section 5 highlights the details of the results while conclusion and future work are presented in Section 6.

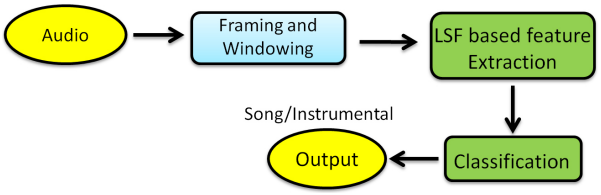

Fig. 1. Pictorial view of the System

2. Related Work

Leung *et al.* [1] used a supervised variant of independent component analysis namely ICA-FX for the task of segregating instrumentals and voices. They had also used general likelihood ratio based distance and SVM based classification; using 5 and 25 pop songs for training and testing respectively, they obtained a highest individual accuracy of 80.04%. Chanrungutai *et al.* presented a system for separating vocals from music with the aid of a non negative matrix factorization based technique. They performed pitch extraction on the separated voices; their data consisted of both real backing tracks as well as MIDI ones. A detailed account of the results is presented in [2].

Rocamora *et al.* [3] studied various audio descriptors for the task of music and voice segregation and concluded the fact that mel frequency cepstral coefficient (MFCC)-based approach is the most appropriate. They also presented a statistical classification technique with the help of a reduced descriptor set for detecting voice in songs and obtained a highest accuracy of 78.5%. Hsu *et al.* performed separation of music accompaniments and unvoiced singing voice on the MIR-1K dataset. They followed the computational auditory scene analysis framework in their experiments whose details are presented in [4].

Rafii *et al.* [5] adopted a repetitive musical structure identification based approach for segregating

voice and music; they experimented with the MIR-1K dataset and obtained a highest global normalized signal to distortion ratio of 1.11. On another work Rafii *et al.* [6] presented a system named REPET for the task of speech and music separation; they experimented with 1000 song clips and 14 songs and extended the system to aid in the pre-processing stage for detecting pitch to help in melody extraction. Liutkus *et al.* [7] further extended REPET to handle background variations as well as long excerpts in order to process full songs.

Ghosal *et al.* [8] adopted a random sample and consensus based approach for the purpose of separating songs and instrumentals; they experimented on a dataset of 300 instrumentals and songs each and obtained an accuracy of 95%. Mauch *et al.* [9] obtained an accuracy of 89.8% for the task of instrumental solo detection using a combination of four features in the thick of MFCC, pitch fluctuation, MFCC of resynthesised predominant voice and normalised amplitude of harmonic partials.

Burute and Mane [10] used a robust principal component analysis based approach for separating background music and voice. They experimented with the MIR-1K dataset and reported results for different parameters in the thick of source to distortion ratio, source to artefact ratio, source to interference ratio and global normalised source to distortion ratio. A best global normalised source to distortion value of around 5.2 decibels was reported by them as well. Ghosal *et al.* [11] used MFCC based features for segmenting instrumentals and songs. They experimented on a database consisting of 180 songs and instrumentals each of length 40-45 seconds. The clips were monophonic in nature sampled at 22050 Hz. The dataset consisted of data from different instruments like flute, guitar, drums and piano as well as different genres like rock, classical and jazz. Among different machine learning algorithms, they obtained a highest accuracy of 93.34% using random sample and consensus classification.

Regnier and Peeters [12] attempted to detect the presence of voice in music tracks with the aid of vibrato and tremolo characteristics. They also used harmonicity based criteria to assign a clip to either of singing or non singing class. The experiment was conducted on database of 90 songs from different artists, genres, languages and tempos out of which 58 songs were used for training ad the rest for training. In the entire dataset, 50.3% were segments with vocals and the remaining were without vocals or only music. A highest F-measure value of 76.8% was obtained in their experiment. Ozerov et al. [13] applied a bayesian model adaptation-based approach for source separation over a single channel. They experimented with music and voice separation and concluded reported better results using adaptive models over non adaptive models. Hsu *et al.* [14] proposed a tandem algorithm for extraction of music pitch and separation of voice from background music. They also used a trend estimation technique to identify pitch range of singing voice and obtained average accuracy of 90%.

Tab. 1. Number of instrumental and song clips for the different datasets

Datatset	Song	Instrumental
(clip length)	(49:19:48)	(49:50:15)
D_1 (5s)	35116	35718
D_2 (10s)	17362	17771
D_3 (15s)	11431	11798
D_4 (20s)	8500	8805

Zhu *et al.* [15] proposed a multiple stage non negative matrix factorization technique for separating monoaural singing voice. They first applied the factorization technique for decomposition of spectrograms followed by application of a spectral discontinuity thresholding technique. Multitudinous experiments were performed on the MIR-1K dataset consisting of 1000 short audios and the Beach-Boys dataset consisting of 14 songs whose results are presented in [15].

3. Dataset Development

Data is an essential entity of any experiment. It is always important to ensure that the dataset contains real life characteristics as far as possible. In our experiment, audio clips of songs and instrumentals were extracted from various websites like YouTube [16]. The top three languages of India namely English, Hindi and Bangla [17] were considered in the case of songs. Songs of different genres and timelines were chosen in order to ensure that the dataset covered various qualities of songs in terms of rendering and audio engineering. The song clips had either background music or sections of instrument only parts. Different artists were chosen for collecting instrumental clips in order to get data of various types like genre, playing style, tonality and technicality. Instrumental covers of various songs, as well as original compositions using different instruments like guitar, violin and piano, along with background music constituted the instrumental part of the dataset.

This data was used to generate 4 datasets (D_1-D_4) having clips of lengths 5, 10, 15 and 20 seconds, respectively. The details of the generated datasets along with that of the original data is presented in Table 1.

In order to test the system's performance in noisy scenario, each of the datasets were subjected to 3 types of noise sources namely humming noise, highway noise and thunder noise. The signal to noise ratios ranged from -20.65 to 14.90. The amplitude based presentation for the different noise clips along with those of the instrument and song clips from the 4 datasets is presented in Figures 2- 5.

4. Proposed Method

4.1. Pre-Processing

The audio clips were split into smaller frames as the spectral characteristics tend to show a lot of deviation for longer clips. The clips were partitioned into 256 sample point wide frames with a 100 point overlap as presented in [18]. Next, the frames were sub-

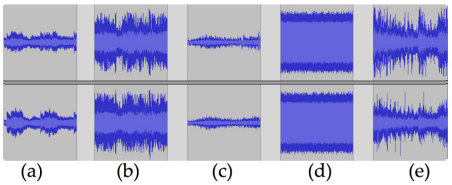

Fig. 2. Amplitude based representation for 5 second long (a) Instrumental, (b) Song, (c) Highway noise, (d) Humming noise and (e) Thunder noise clips

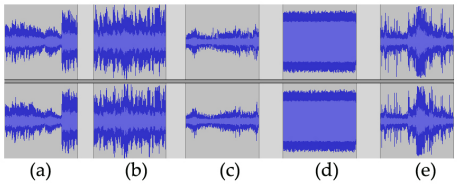

Fig. 3. Amplitude based representation for 10 second long (a) Instrumental, (b) Song, (c) Highway noise, (d) Humming noise and (e) Thunder noise clips

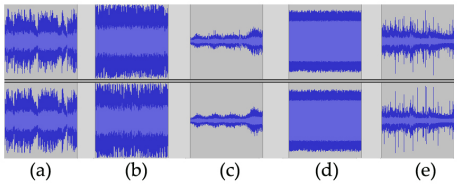

Fig. 4. Amplitude based representation for 15 second long (a) Instrumental, (b) Song, (c) Highway noise, (d) Humming noise and (e) Thunder noise clips

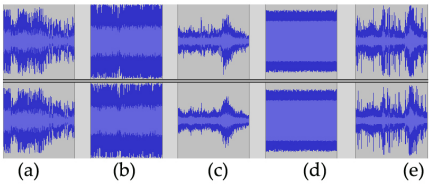

Fig. 5. Amplitude based representation for 20 second long (a) Instrumental, (b) Song, (c) Highway noise, (d) Humming noise and (e) Thunder noise clips

jected to a windowing function (Hamming Window as presented in [19]) in order to get rid of the jitters which might lead to spectral leakage at the time of frequency based analysis. The mathematical representation of hamming window $B(n)$ is given by Equation (1) where the value of r ranges between the frame boundary of size R

$$B(n) = 0.54 - 0.46\cos\left(\frac{2\pi r}{R-1}\right). \quad (1)$$

4.2. Feature Extraction

Line spectral frequency [20] is a method for representing linear predictive coefficients with better interpolation properties. Here, a signal is considered as the output of an all pole filter ($H(z)$). The inverse filter of $H(z)$ is represented by Equation 2, where r_1, r_2...r_T designate the predictive coefficients up to the order T

$$R(z) = 1 + r_1 z^{-1} + r_2 z^{-2} + r_3 z^{-3} + ... + r_T z^{-T}. \quad (2)$$

$R(z)$ is decomposed into polynomials $F(z)$ and $G(z)$ as shown in Equation 3 and Equation 4, respectively, whose zeros lie on the unit circle. They are also interlaced with each other, thus helping in computation

$$F(z) = R(z) + z^{-(T+1)} R(z^{-1}) and \quad (3)$$

$$G(z) = R(z) - z^{-(T+1)} R(z^{-1}). \quad (4)$$

Each of the datasets were used for extraction of 5, 10, 15 and 20 dimensional standard line spectral frequency features. Since the clips were of disparate lengths, a disparate number of frames were produced for the clips, producing features of disparate dimension. In order to tackle this problem, the band wise sums for the energy values were computed which were then used to compute the ratio of distribution of energy across the bands. Along with this, the band numbers were also added to the feature set graded in descending order based on total energy content.

It was experimentally found that a clip of just 2 seconds produced 880 frames; if a 5 dimensional LSF was extracted for the clip then a feature set of 4400 dimension was obtained. However with the help of the proposed technique, this dimension was brought down to just 10 (5 ratio distribution values and 5 band numbers). Thus, LSF values of 5, 10, 15 and 20 dimensions produced even dimensional feature sets of 10, 20, 30 and 40 dimensions which were independent of the length of the clips. Trends of the feature values for the songs and instrumental clips for the 40 dimensional features (best result in noise free scenario) is presented in Figure 6.

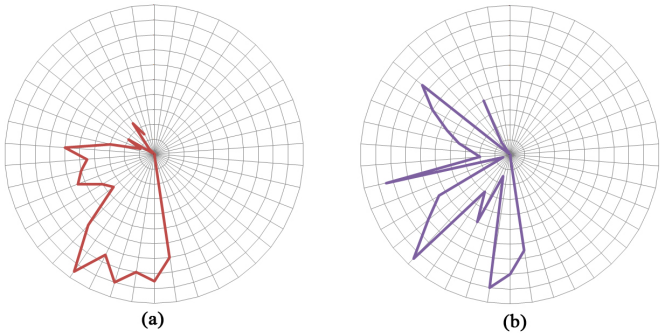

Fig. 6. 40 dimensional feature values for (a) Song, (b) Instrumental

4.3. Classification

Each feature set for each datasets was fed into different classifiers popularly used in pattern recognition problemsin the thick of BayesNet (BN) [21], Support Vector Machine (SVM) [22], LibLinear (Lib) [23], Multi

Layer Perceptron (MLP) [24], Simple Logistic (SL) [25] and Decision Table (DT) [26].

BayesNet: is a bayesian classifier that makes use of a Bayes Network for learning with the aid of different quality parameters and search algorithms. The base class provides data structures like conditional probability distributions, structure of network, etc. and different facilities which are similar to that of the Bayesian Network learning algorithm like K2.

Support Vector Machine: is a supervised learning algorithm that can be used for classification as well as regression analysis. SVM builds a bi-class model from a set of training instances and then associates each instance to either class.

LibLinear: is a functional and linear type of classifier which is suitable for either large number of instances or large feature sets. It is also suitable for regression problems.

Decision Table: is one of the simplest supervised learning algorithms; it consists of 2 parts namely, schema which defines the features to be included in the table and body which embodies the set of instances along with their feature values and class labels.

Multi Layer Perceptron: is a feed forward variant of an artificial neural network which maps an input and output set; it consists of nodes which are connected by links having weights associated to it. It is one of the most popular classifiers in pattern recognition problems.

Simple Logistic: is a classifier used to build linear logistic regression models. The classifier has a base learner associated with it along with number of iterations which aids to automatically select attributes.

The extremely popular open source classification tool named Weka [27] was used in the present experiment. We used 5 fold cross validation for all the classifiers with default parameters. The details of the obtained results are presented in the subsequent section.

5. Result and Discussion

5.1. Noise Free Scenario

The accuracies obtained for datasets D_1–D_4 using different classifiers are presented in Table 2 to Table 5.

From Table 2 it is observed that the highest and lowest accuracies of 99.78% and 50.69% were obtained using MLP and LibLinear, respectively, which are the overall best and worst results among all the classifiers with default parameters. The same behaviour is found for D_2 dataset as can be seen in Table 3; the highest and lowest accuracies being 98.27% and 52.27%. On the other hand, for D_3 (Table 4), the highest and lowest accuracies were 99.57% and 52.84%, obtained using BayesNet and LibLinear, respectively. Highest and lowest accuracies of 99.39% and 51.77% were obtained using MLP and LibLinear, respectively for D_4 (Table 5).

Table 6 shows the highest and lowest accuracies obtained for all experiments based on the feature dimension. It can be seen from the Table that LibLinear produced the lowest accuracy for every feature dimension. It can also be observed that the top 2 results were obtained using MLP on D_1 (shortest clips in experiment) and D_4 (longest clips in experiment) which shows the effectiveness of MLP.

It is also observed from Tables 2–5 that highest accuracies of 99.57%, 99.31%, 84.86% were obtained for BayesNet, SVM and LibLinear respectively. The lowest accuracies for the same classifiers were found to be 67.01%, 59.67% and 50.69% respectively. In the case of MLP, Simple Logistic and Decision Table, highest accuracies of 99.78%, 89.91% and 97.47% respectively were obtained. The lowest accuracies for the same were found to be 69.08%, 60.24%, 71.73% respectively.

Concluding, the classifiers can be organized in the following manner based on their best performance: MLP, BayesNet, SVM, Decision Table, Simple Logistic and LibLinear. MLP is very suitable for audio signal based applications as demonstrated in [24, 19, 18] which is depicted in the obtained results as well.

The confusion matrix for the best result (D_1, MLP with 40 dimensional features) is presented in Table 7 where it can be observed that 0.19% of the song clips and 0.25% of the instrumental clips were confused with each other. One possible reason for this is that during the generation of the dataset (splitting of clips into shorter clips), some of the instrumental parts from the songs might have got entirely isolated for the 5 second clips (it was observed that various songs had instrumental sections of more than 5 seconds at a stretch) which interfered with the trained model.

Since the best result was obtained for MLP, we further experimented with it by varying the number of training iterations (ephocs); the obtained accuracies are depicted in Table 8.

From the Table it can be observed that the highest accuracy (99.81%) was obtained for 1800 iterations and that value maintained for further iterations. The confusion matrix for this experiment is presented in Table 9 where it can be seen that the number of misclassified samples for both the classes decreased with respect to the default configuration of MLP as shown in Table 7.

We had further experimented by varying the number of folds in cross validation for the same setup (best accuracy with lowest number of training iterations); the details are presented in Table 10. We had varied the number of folds for cross validation from 2-10 to observe the performance of MLP for test and training sets of different sizes. It can be seen from the Table that the best accuracy was obtained for 5 and 7 folds which further decreased on increasing the number of folds.

5.2. Noisy Scenario

The obtained accuracies for the different feature dimensions for the 4 datasets in the presence of highway noise is presented in Table 11.

It can be seen from the Table that highest accuracies of 97.77%, 97.85% and 99.16% for D1-D3 respectively were obtained for 40 dimensional features

Tab. 2. Obtained accuracies for D_1 using different classifiers and feature dimensions

Dimension	BN	MLP	DT	SL	SVM	Lib
10	83.38	86.22	86.51	84.86	84.86	84.86
20	67.01	72.83	72.92	61.71	88.15	57.37
30	69.47	72.36	71.82	61.53	72.21	50.69
40	95.30	**99.78**	94.96	62.36	73.01	54.55

Tab. 3. Obtained accuracies for D_2 using different classifiers and feature dimensions

Dimension	BN	MLP	DT	SL	SVM	Lib
10	67.38	69.08	72.06	60.24	64.57	60.24
20	75.351	92.51	94.42	64.64	96.89	62.47
30	89.62	94.96	96.62	62.16	94.98	57.03
40	98.27	98.18	96.76	73.97	61.21	52.27

Tab. 4. Obtained accuracies for D_3 using different classifiers and feature dimensions

Dimension	BN	MLP	DT	SL	SVM	Lib
10	67.53	70.26	71.73	60.61	61.89	60.62
20	78.56	99.30	99.04	61.97	99.31	59.56
30	93.27	97.44	97.36	61.49	92.55	52.84
40	99.57	99.52	94.13	89.91	59.67	72.56

Tab. 5. Obtained accuracies for D_4 using different classifiers and feature dimensions

Dimension	BN	MLP	DT	SL	SVM	Lib
10	85.25	85.26	85.42	60.77	84.19	60.63
20	83.55	96.32	96.51	69.11	96.16	66.51
30	90.97	99.39	97.47	62.62	95.53	51.77
40	90.97	92.58	90.93	70.43	64.85	61.84

Tab. 6. Highest and lowest accuracies obtained for all experiments based on feature dimension along with the classifier and dataset

Dimension	Highest	Lowest
10	86.51 (D_1, DT)	60.24 (D_2,SL; D_2,Lib)
20	99.31 (D_3, SVM)	57.37 (D_1, Lib)
30	99.39 (D_4, MLP)	50.69 (D_1, Lib)
40	99.78 (D_1, MLP)	52.27 (D_2, Lib)

Tab. 7. Confusion matrix for D_1 with 40 dimensional features using MLP showing the number of correctly and misclassified instances

	Song	Instrumental
Song	35051	65
Instrumental	89	35629

using MLP. In the case of D4, the best result was obtained for 30 dimensional features.

The obtained accuracies for the different feature dimensions for the 4 datasets in the presence of humming noise is presented in Table 12.

It can be seen from the Table that the best results for all 4 datasets was obtained using MLP. The 40 dimensional features produced highest results (96.44% and 98.55%) for D1and D2 respectively while the 30 dimensional features produced the best result for the remaining 2 datasets(98.20% and 98.17% respectively). It can be seen that for D4, the accuracy dropped significantly for 40 dimensional features in comparison to the 30 dimensional features which points towards over fitting the neural network. In the case of D3 it can be seen that an increase of 16.21% in accuracy was obtained for the 30 dimensional features as compared to the 20 dimensional features thereby demonstrating the inability of the 20 dimensional features to handle the 15 second long noisy clips.

The obtained accuracies for the different feature dimensions for the 4 datasets in the presence of thunder noise is presented in Table 13.

It can be seen from the Table that in the case of D2, the best result was obtained for decision table with 20 dimensional features. The obtained accuracies for 10 and 30 dimensional features were quite less as compared to the 20 dimensional features as can be seen in the Table as well. In The remaining datasets produced best results with MLP and 20 dimensional features out

Tab. 8. Accuracies using MLP for D_1 with 40 dimensional features for different training iterations

Iterations	100	200	300	400	500	600	700	800
Accuracy (%)	99.66	99.72	99.72	99.79	99.78	99.78	99.78	99.78
Iterations	900	1000	1100	1200	1300	1400	1500	1600
Accuracy (%)	99.79	99.80	99.79	99.80	99.80	99.80	99.79	99.80
Iterations	1700	**1800**	1900	2000	2100	2200	2300	2400
Accuracy(%)	99.80	**99.81**	99.81	99.81	99.80	99.81	99.80	99.81

Tab. 9. Confusion matrix for D_1 with 40 dimensional features using MLP at 1800 learning iterations

	Song	Instrumental
Song	35052	64
Instrumental	74	35644

of which D2 produced the best result (99.19%) among all the noisy scenarios. The confusion matrix for the same is presented in Table 14.

Since the best result for noise free scenario using MLP was obtained using MLP with 1800 training iterations, so the same configuration was also used and an accuracy of 99.34% (highest among all noisy scenarios) was obtained whose confusion matrix is presented in Table 15.

It can be seen from the Table that though the number of misclassifications for song clips increased slightly (only 3 more instances) but the number of misclassified instrumental clips reduced by 31% in comparison to the default iteration setup of MLP.

We had further experimented with the thunder noise scenario and 20 dimensional feature set of D_2 as the best result for noisy scenario was obtained for it. We increased the power of the thunder noise signal by 2 (N1), 5 (N2), 10 (N3) and 20 (N4) dB and added it to the noise free clips of D_2 to observe the system's performance. The amplitude based representation of the noise clips along with that of the original data is presented in Figure 7.

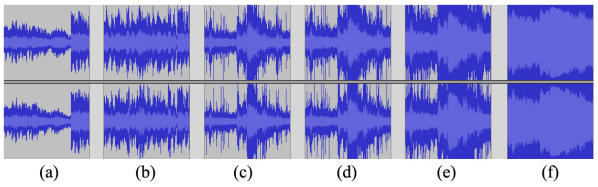

Fig. 7. Amplitude based representation for 10 second long (a) Instrumental, (b) Song, (c) N1, (d) N2, (e) N3 and (f) N4 clips

The trend of the feature values for the 2 type of clips in thunder noise scenario as well as N1, N2, N3 and N4 scenario is presented in Figure 8.

The obtained accuracies for N1-N4 is presented in Table 16.

It can be seen from the Table that the accuracy dropped slightly for N2 with respect to the original thunder noise scenario. A similar trend is also observed for N2 and N3. However, the accuracy dropped

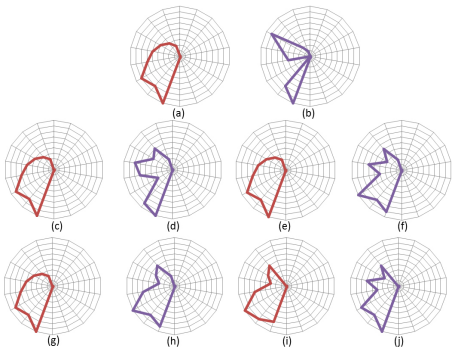

Fig. 8. 40 dimensional feature values for (a), (b) Instrumental

significantly when the noise was increased by 20 dB. The dataset was manually investigated for this scenario and it was fond that in most of the clips, the proportion of noise was extremely high in comparison to the original data and in many cases the original data was inaudible. The confusion matrices for the 4 scenarios is presented in Table 17.

It can be seen from the Table that in the case of N2 and N3, there is no major difference of the number of misclassified instances though the noise component in N3 is twice to that of N2 which points to the system's ability to handle noisy clips.

5.3. Statistical Significance Tests

Friedman test [28] on the 40 dimensional feature set of D_1 (overall highest among all scnarios) was carried out to check for statistical significance. The dataset was divided into 5 parts (N) and all the 6 classifiers (k) were involved in the test. The distribution of ranks and accuracies are presented in Table 18.

The Friedman statistic (χ_F^2) [28] was calculated with the help of Table 18 in accordance with Equation 5 where R_j corresponds to the mean rank of the j^{th} classifier.

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right]. \quad (5)$$

The critical values of χ_F^2 for the above setup was found to be 11.070 and 9.236 at significance levels of 0.05 and 0.10 respectively; we got a value of 15.54 for χ_F^2 thereby rejecting the null hypothesis.

6. Conclusion

In this paper, a system for segregating instrumentals and songs from noisy audio is presented using line

Tab. 10. Accuracies for different number of cross validation folds of cross using MLP for D_1 with 40 dimensional features and 1800 training iterations

# Folds	2	3	4	5	6	7	8	9	10
Accuracy (%)	99.36	99.75	99.69	99.81	99.71	99.81	99.78	99.75	99.73

Tab. 11. Obtained accuracies for D_1-D_4 and 10-40 dimensional features in highway noise scenario

	BN	MLP	DT	SL	SVM	Lib
D1						
10	68.80	68.89	70.28	61.84	61.66	61.51
20	67.81	76.15	77.95	63.16	62.70	59.08
30	72.06	78.21	78.46	63.36	62.27	55.58
40	97.14	97.77	96.61	69.35	56.98	56.96
D2						
10	66.91	69.77	72.05	60.75	60.52	60.11
20	73.36	97.40	92.17	64.50	63.35	61.05
30	87.09	91.37	95.33	69.73	63.37	63.37
40	97.02	97.85	93.03	87.66	95.12	77.31
D3						
10	80.30	80.89	80.63	73.86	74.58	74.09
20	83.77	91.29	92.66	70.58	72.13	64.30
30	95.49	97.51	94.32	79.96	79.78	71.45
40	97.49	99.16	96.70	77.04	83.87	65.77
D4						
10	86.55	87.05	87.30	86.96	86.96	86.96
20	80.25	94.22	89.91	65.29	64.50	62.65
30	95.62	98.15	97.83	77.99	84.95	69.37
40	90.57	94.02	94.73	67.20	58.59	50.70

Tab. 12. Obtained accuracies for D_1-D_4 and 10-40 dimensional features in humming noise scenario

	BN	MLP	DT	SL	SVM	Lib
D1						
10	67.27	68.08	70.07	63.10	62.07	62.14
20	62.96	70.07	67.16	61.28	59.41	56.20
30	77.66	85.63	92.51	65.77	65.13	55.77
40	77.22	96.94	90.56	65.28	64.82	53.50
D2						
10	64.95	66.05	67.78	60.53	61.10	60.27
20	65.74	74.12	72.31	62.14	60.59	58.28
30	97.53	97.94	97.80	97.73	97.59	93.62
40	97.78	98.55	98.08	72.75	82.16	51.72
D3						
10	87.21	88.00	87.38	69.83	68.47	69.50
20	65.22	81.99	70.70	62.21	60.35	58.68
30	97.61	98.20	97.05	87.23	81.01	83.51
40	91.71	96.92	93.49	84.38	86.45	74.34
D4						
10	65.54	68.23	68.86	60.02	60.53	60.07
20	72.25	89.49	88.96	62.58	59.96	56.98
30	98.02	98.17	98.01	95.03	98.14	90.02
40	71.74	83.32	91.10	66.92	67.50	52.09

spectral frequency based features. The clips were subjected to multifarious noise sources to test the robustness of the system. We have applied different popular classifiers on the feature sets and obtained the highest result using MLP algorithm for both noise free as well as noisy scenario.. It was also observed that LibLinear produced most of the accuracies in the lower side.

As future work we will experiment with a larger and more robust dataset and observe the performance of various other classifiers. We will also experiment with other machine learning techniques including deep learning based approaches and use different

Tab. 13. Obtained accuracies for D_1-D_4 and 10-40 dimensional features in thunder noise scenario

	BN	MLP	DT	SL	SVM	Lib
D1						
10	62.37	63.42	63.34	60.99	60.84	61.17
20	68.58	86.47	83.80	61.57	58.96	58.76
30	83.86	84.73	81.25	77.48	77.08	72.91
40	59.39	73.38	69.64	58.99	54.81	50.16
D2						
10	61.15	62.11	63.96	58.97	58.96	58.66
20	91.78	99.19	99.03	68.63	69.43	66.89
30	68.30	79.86	71.23	78.79	75.27	66.90
40	59.14	87.67	87.64	58.94	52.64	49.87
D3						
10	62.00	65.99	65.62	61.15	56.96	57.86
20	88.20	92.94	90.54	62.71	63.83	58.25
30	83.59	84.36	86.94	84.27	80.89	81.67
40	69.95	93.93	98.12	75.68	71.60	54.96
D4						
10	62.21	64.66	66.22	59.71	57.51	58.34
20	95.30	98.89	96.91	84.32	82.68	83.52
30	61.87	63.07	65.50	62.34	54.71	50.56
40	81.53	92.92	84.36	65.95	64.27	55.01

Tab. 14. Confusion matrix for D2 with 20 dimensional features and thunder noise scenario using MLP with default parameters

	Song	Instrumental
Song	17250	112
Instrumental	172	17599

Tab. 15. Confusion matrix for D2 with 20 dimensional features and thunder noise scenario using MLP with 1800 training iterations

	Song	Instrumental
Song	17247	115
Instrumental	118	17653

Tab. 16. Obtained accuracies for D2 using MLP when subjected to N1-N4 sources

Noise Scenario	N1	N2	N3	N4
Accuracy	98.84	95.23	94.36	68.66

features to further minimize the errors. We also plan to pre-process the data for noise removal to make the system more robust which is critical for live audio. The system will be further extended to detect instrument sections from songs in real time to separate the vocals and instrument tracks.

AUTHORS

Himadri Mukherjee* – Dept. of Computer Science, West Bengal State University, Kolkata, India, e-mail: himadrim027@gmail.com.
Sk Md Obaidullah – Dept. of Computer Science and Engineering, Aliah University, Kolkata, India, e-mail: sk.obaidullah@gmail.com.
K.C. Santosh – Dept. of Computer Science, The University of South Dakota, SD, USA, e-mail: santosh.kc@usd.edu.
Teresa Gonçalves – Dept. of Informatics, University of Evora, Evora, Portugal, e-mail: tcg@uevora.pt.
Santanu Phadikar – Dept. of Computer Science and Engineering, Maulana Abul Kalam Azad University of Technology, Kolkata, India, e-mail: sphadikar@yahoo.com.
Kaushik Roy – Dept. of Computer Science, West Bengal State University, Kolkata, India, e-mail: kaushik.mrg@gmail.com.
*Corresponding author

REFERENCES

[1] T.-W. Leung, C.-W. Ngo, and R. Lau, "ICA-FX features for classification of singing voice and instrumental sound". In: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 2, 2004, 367–370, 10.1109/ICPR.2004.1334222, ISSN: 1051-4651.

[2] A. Chanrungutai and C. A. Ratanamahatana, "Singing voice separation for mono-channel music using Non-negative Matrix Factorization". In: *2008 International Conference on Advanced Technologies for Communications*, 2008, 243–246, 10.1109/ATC.2008.4760565.

[3] M. Rocamora and P. Herrera, "Comparing audio descriptors for singing voice detection in music audio files". In: *11th Brazilian symposium on computer music, San Pablo, Brazil*, vol. 26, 2007.

[4] Chao-Ling Hsu and J.-S. Jang, "On the Improvement of Singing Voice Separation for Mo-

Tab. 17. Obtained accuracies for D2 using MLP when subjected to N1-N4 sources

	Song	Instrumental
Song	17001	361
Instrumental	45	17726
(N1)		

	Song	Instrumental
Song	16286	1076
Instrumental	601	17170
(N2)		

	Song	Instrumental
Song	15989	1373
Instrumental	607	17164
(N3)		

	Song	Instrumental
Song	11236	6126
Instrumental	4886	12885
(N4)		

Tab. 18. Rank and accuracies for parts of D_1

Classifiers		Parts of D_1					Mean Rank
		1	2	3	4	5	
MLP	A	99.64	99.99	99.98	100.0	99.92	1.8
	R	(3.0)	(1.0)	(2.0)	(2.0)	(1.0)	
BN	A	99.9	99.72	100.0	100.0	99.46	1.7
	R	(1.5)	(2.0)	(1.0)	(2.0)	(2.0)	
SL	A	99.9	61.8	78.45	99.97	75.81	3.9
	R	(1.5)	(4.0)	(5.0)	(4.0)	(5.0)	
DT	A	95.4	92.98	99.18	97.72	96.01	3.6
	R	(4.0)	(3.0)	(3.0)	(5.0)	(3.0)	
SVM	A	76.33	55.12	96.05	89.99	76.97	4.8
	R	(5.0)	(5.0)	(4.0)	(6.0)	(4.0)	
Lib	A	55.01	50.66	50.76	100.0	62.36	5.2
	R	(6.0)	(6.0)	(6.0)	(2.0)	(6.0)	

naural Recordings Using the MIR-1K Dataset", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, 2010, 310–319, 10.1109/TASL.2009.2026503.

[5] Z. Rafii and B. Pardo, "A simple music/voice separation method based on the extraction of the repeating musical structure". In: *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Prague, Czech Republic, 2011, 221–224, 10.1109/ICASSP.2011.5946380.

[6] Z. Rafii and B. Pardo, "REpeating Pattern Extraction Technique (REPET): A Simple Method for Music/Voice Separation", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 1, 2013, 73–84, 10.1109/TASL.2012.2213249.

[7] A. Liutkus, Z. Rafii, R. Badeau, B. Pardo, and G. Richard, "Adaptive filtering for music/voice separation exploiting the repeating musical structure". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, 53–56, 10.1109/ICASSP.2012.6287815.

[8] A. Ghosal, R. Chakraborty, B. C. Dhara, and S. K. Saha, "Song/instrumental classification using spectrogram based contextual features". In: *Proceedings of the CUBE International Information Technology Conference*, New York, NY, USA, 2012, 21–25, 10.1145/2381716.2381722.

[9] M. Mauch, H. Fujihara, K. Yoshii, and M. Goto, "Timbre and Melody Features for the Recognition of Vocal Activity and Instrumental Solos in Polyphonic Music". In: *ISMIR*, 2011, 233–238.

[10] H. Burute and P. B. Mane, "Separation of singing voice from music accompaniment using matrix factorization method". In: *2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, 2015, 166–171, 10.1109/ICATCCT.2015.7456876.

[11] A. Ghosal, R. Chakraborty, B. C. Dhara, and S. K. Saha, "Instrumental/song classification of music signal using RANSAC". In: *2011 3rd International Conference on Electronics Computer Technology*, vol. 1, 2011, 269–272, 10.1109/ICECTECH.2011.5941603.

[12] L. Regnier and G. Peeters, "Singing voice detection in music tracks using direct voice vibrato detection". In: *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, 1685–1688, 10.1109/ICASSP.2009.4959926.

[13] A. Ozerov, P. Philippe, F. Bimbot, and R. Gribonval, "Adaptation of Bayesian Models for Single-Channel Source Separation and its Application to Voice/Music Separation in Popular Songs", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, 2007, 1564–1578, 10.1109/TASL.2007.899291.

[14] C.-L. Hsu, D. Wang, J.-S. R. Jang, and K. Hu, "A Tandem Algorithm for Singing Pitch Extraction and Voice Separation From Music Accompaniment", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 5, 2012, 1482–1491, 10.1109/TASL.2011.2182510.

[15] B. Zhu, W. Li, R. Li, and X. Xue, "Multi-Stage Non-Negative Matrix Factorization for Monaural Singing Voice Separation", *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, 2013, 2096–2107, 10.1109/TASL.2013.2266773, Conference Name: IEEE Transactions on Audio, Speech, and Language Processing.

[16] "Youtube". https://www.youtube.com/, 2020. Accessed on: 2020-09-20.

[17] "Ethnologue: Languages of the World". https://www.ethnologue.com/, 2020. Accessed on: 2020-09-20.

[18] H. Mukherjee, S. M. Obaidullah, S. Phadikar, and K. Roy, "SMIL - A Musical Instrument Identification System". In: J. K. Mandal, P. Dutta, and S. Mukhopadhyay, eds., *Computational Intelligence, Communications, and Business Analytics*, Singapore, 2017, 129–140, 10.1007/978-981-10-6427-2_11.

[19] H. Mukherjee, S. Phadikar, P. Rakshit, and K. Roy, "REARC-a Bangla Phoneme recognizer". In: *2016 International Conference on Accessibility to Digital World (ICADW)*, 2016, 177–180, 10.1109/ICADW.2016.7942537.

[20] K. K. Paliwal, "On the use of line spectral frequency parameters for speech recognition", *Digital Signal Processing*, vol. 2, no. 2, 1992, 80–87, 10.1016/1051-2004(92)90028-W.

[21] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian Network Classifiers", *Machine Learning*, vol. 29, no. 2, 1997, 131–163, 10.1023/A:1007465528199.

[22] N. Cristianini and J. Shawe-Taylor. "An Introduction to Support Vector Machines and Other Kernel-based Learning Methods", March 2000.

[23] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A Library for Large Linear Classification", *The Journal of Machine Learning Research*, vol. 9, 2008, 1871–1874.

[24] H. Mukherjee, C. Halder, S. Phadikar, and K. Roy, "READ—A Bangla Phoneme Recognition System". In: S. C. Satapathy, V. Bhateja, S. K. Udgata, and P. K. Pattnaik, eds., *Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications*, Singapore, 2017, 599–607, 10.1007/978-981-10-3153-3_59.

[25] M. Sumner, E. Frank, and M. Hall, "Speeding Up Logistic Model Tree Induction". In: A. M. Jorge, L. Torgo, P. Brazdil, R. Camacho, and J. Gama, eds., *Knowledge Discovery in Databases: PKDD 2005*, Berlin, Heidelberg, 2005, 675–683, 10.1007/11564126_72.

[26] R. Kohavi. "The power of decision tables". In: N. Lavrac and S. Wrobel, eds., *Machine Learning: ECML-95*, volume 912, 174–189. Springer, Berlin, Heidelberg, 1995.

[27] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update", *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, 2009, 10–18, 10.1145/1656274.1656278.

[28] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets", *Journal of Machine Learning Research*, vol. 7, 2006, 1–30.